プログラミング・ガイド
(MR-9122,MR-9123) 2022年10月14日版

ようこそ、ロボットプログラミングの世界へ

MR-9122

MR-9123

みなさんの周りにはどんな働くロボットがありますか？
ロボット掃除機や自動運転のクルマ、プロ棋士と対決するロボットなど、今や生活の様々な場面で、ロボットの活躍を見ることができます。このようなロボットたちは、**「プログラム」**によって人が操縦することなく自律的に動き、活躍しているのです。
プログラミング･フォロは、ロボット･プログラミングにはじめてふれる人のためのロボットです。プログラミング･フォロをプログラムで動かしながら、プログラムの基本をまなんでいきましょう。

●このガイドでは、
プログラミング・フォロ for CHROME(MR-9122) および、
プログラミング・フォロ スピーク for CHROME (MR-9123)の
プログラミング方法について解説しています。
●本書の中では、**プログラミング･フォロ**のことを
「フォロ」または「ロボット」と記していきます。

1.プログラミング・ポータル

プログラミング・ポータルにアクセスする

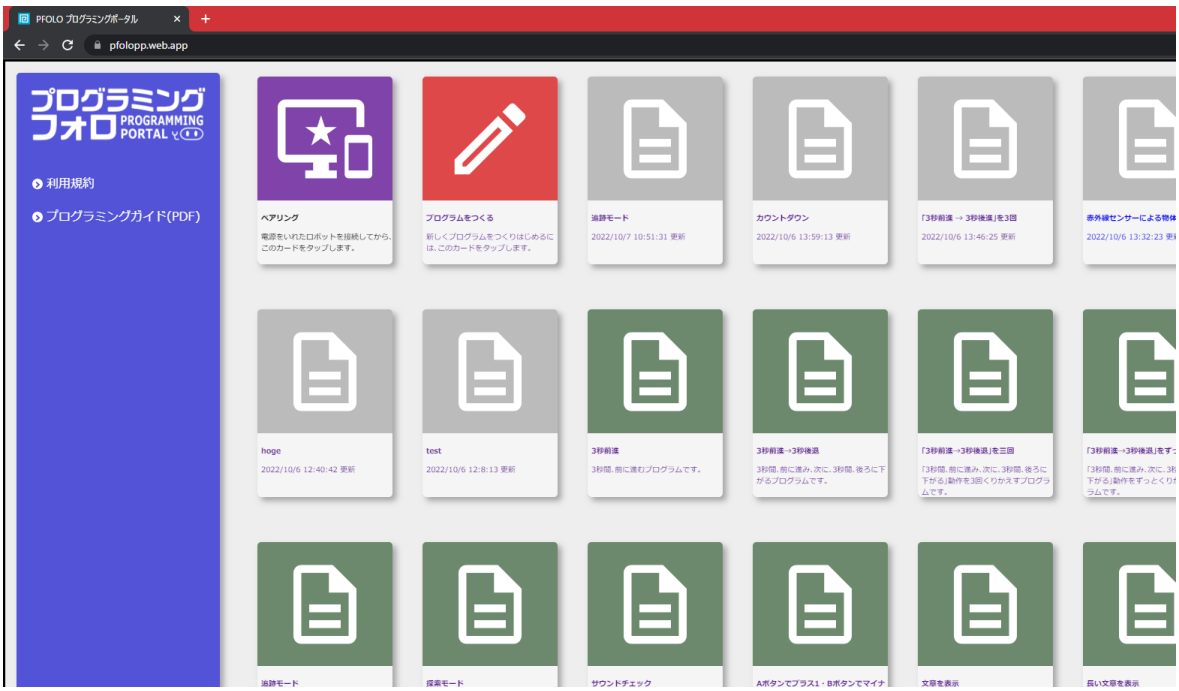
ブラウザーには Google Chrome または Microsoft Edge を使います。

1 ブラウザーのアドレスバーに下記URLを入力してサイトに移動します。

pfolopp.web.app

※完全なURLは「https://pfolopp.web.app」ですが、多くの場合、上記のように省略された形で入力できます。

2 「プログラミング・ポータル」のページが開きます。



プログラミング・ポータル 画面の説明

プログラミング・ポータル 画面

①「ペアリング」カード

ロボットを**ペアリング**(P4)します。

②「プログラムをつくる」カード

あたらしくプログラムをつくるときにタップします。

③「プログラムカード」

作成ずみのプログラムがカードとして表示されます。

プログラムを削除する

削除したいプログラムのカード上で長押しするとメッセージが表示されるので、

OKを押します。

※マウスで操作しているときは、かわりに右クリックしてください。

※最初から収録されているプログラムを消すことはできません。

3 ペアリングする

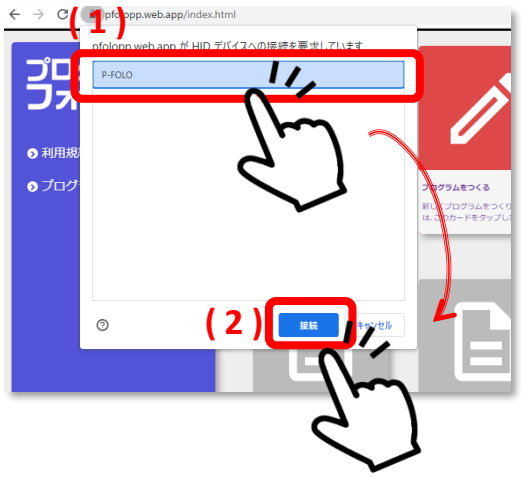
ロボットを始めてパソコンにつなぐときは**「ペアリング」**が必要です。
ペアリングを行っていない状態では、**ロボットとWebアプリとの通信ができません**。
ペアリングは**最初に一回だけ**で、次に使うときは必要ありません。

① **ロボットの電源スイッチをONにする**

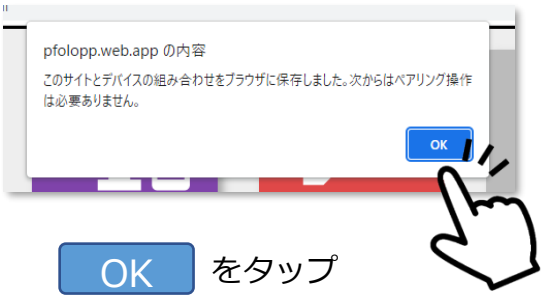
② **ロボットとパソコンをUSBケーブルで接続する**

③ **「ペアリング」をタップ**

④ (1)「P-FOLO」をタップして選択
(2) 接続 を押す



⑤



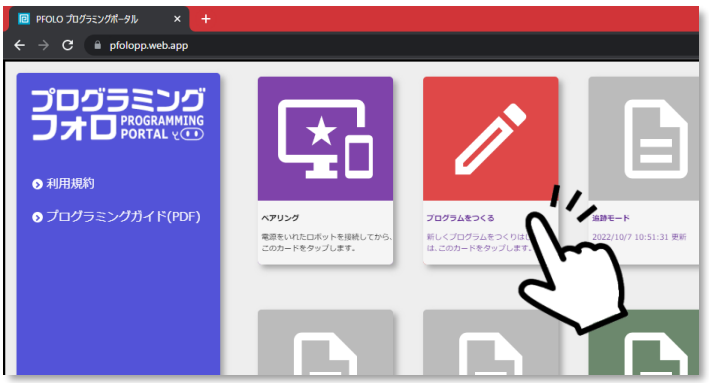
OK をタップ

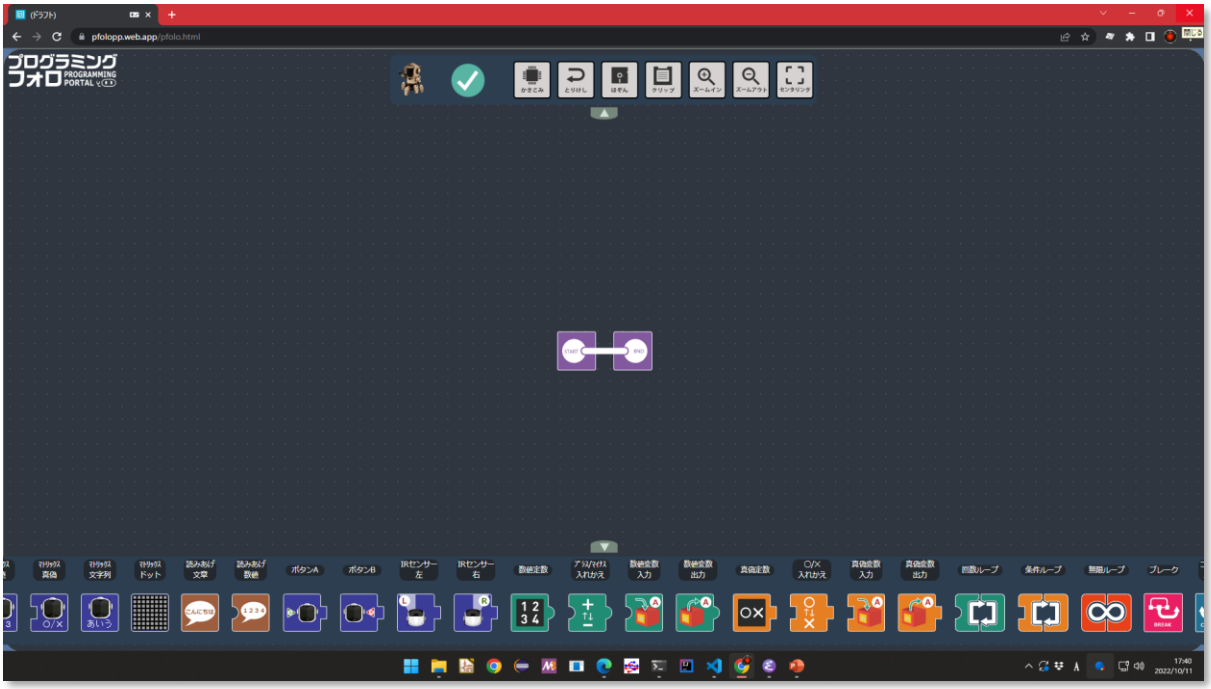
⑥ **ペアリング終了!**

ペアリングは一度だけ行えばよく、
次回以降は、ペアリングは必要ありません。

4 プログラム編集画面を開く

プログラムを新しく作るときは、「プログラミングポータル」画面の、「プログラムをつくる」のカードをタップします。





プログラム編集画面が開きます。

プログラム編集画面

プログラム編集画面

①ワークスペース

この場所にアイコンをならべてプログラムを作成します。

※「スタート」と「エンド」のアイコンは最初から配置されています。

②ドック

プログラムで使用する命令アイコンが並んでいます。

これらのアイコンを、ワークスペースの「スタート」/「エンド」の間に

ならべていくことでプログラムを作成します。

③スイッチボード

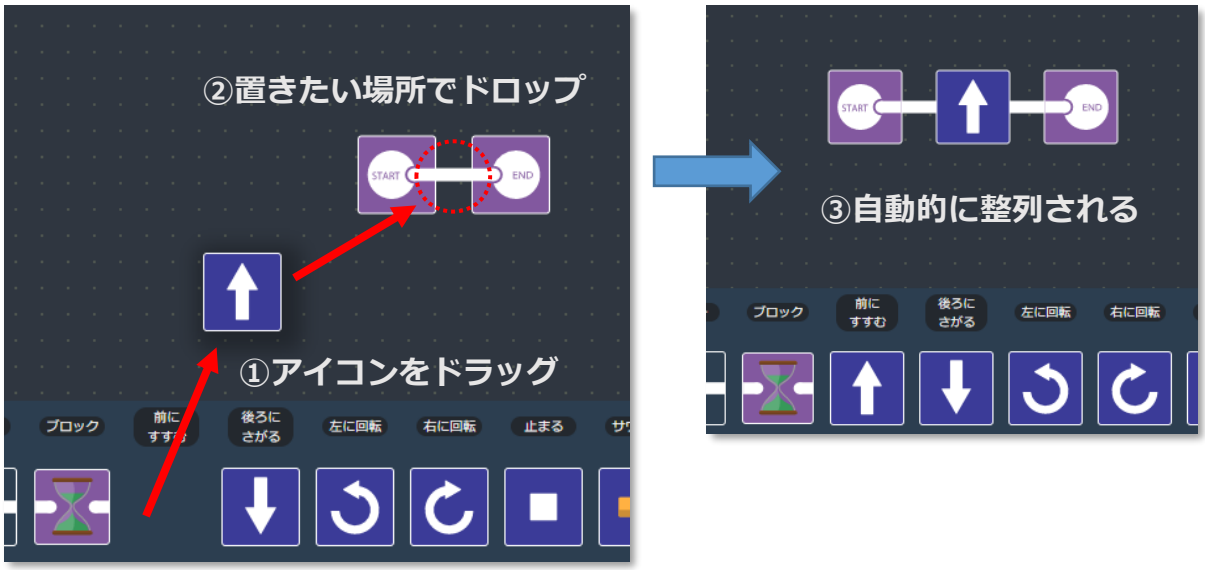
プログラムの書込みなどの操作をするためのボタンが並んでいます。
ロボットの接続状態もここに表示されます。

④プログラミング・ポータルへ

このロゴをタップすると、
「プログラミング・ポータル」ページに移動します。

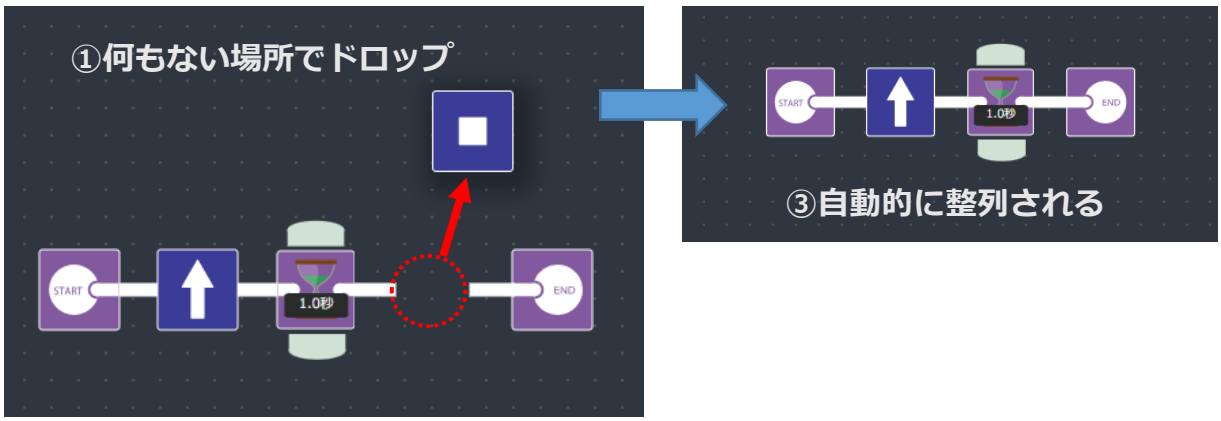
アイコンの操作

①アイコンを置く



②アイコンをならびかえる

③アイコンを消す



スイッチボードの説明

USB チェック

プログラミングモジュールと正しく接続されているときに緑色のチェックマークが表示されます

かきこみ

プログラムをプログラミングモジュールに書き込みます

とりけし

プログラム作成操作を１つ前の状態に戻します

ほぞん

作成したプログラムを保存します

クリップ

現在プログラムを表示しているブラウザー画面のURLアドレスを、クリップボードに保存します。このURLをメールやメモ帳に貼り付けて他の人に伝えることで、別のパソコンのブラウザーで同じ画面が開きます。

ズーム

ワークスペース表示の拡大、縮小をします

センタリング

ワークスペースをセンタリングします

アイコンの説明　：　ドックのアイコン

ドックのアイコンの詳しい説明は、巻末の**「アイコンの説明」**で説明しています。次ページ以降の「やってみよう」でプログラミングに慣れて、オリジナルのプログラムを作るときに活用してください。

２．ロボットを動かしてみよう

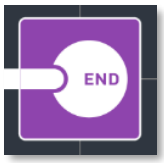
この章で使うアイコン

スタート

プログラムの開始地点を表します。

前にすすむ

ロボットが前へ向かって進みます。



エンド

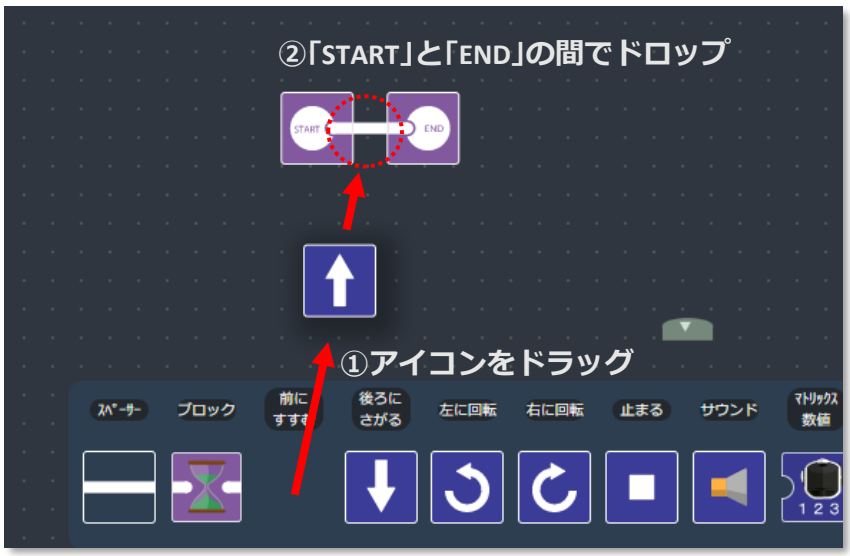
プログラムの終了地点を表します。

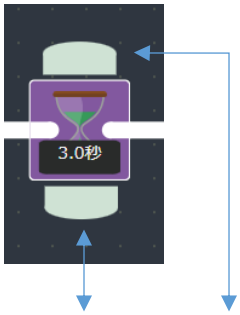
ブロック

設定した時間だけ、プログラムが次のアイコンに進むのをブロックします。

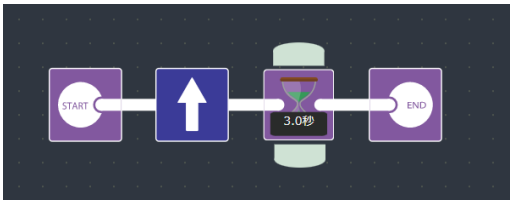
「３秒間、前にすすむ」プログラムをつくる

「前にすすむ」アイコンと、「ブロック」アイコンを画面下のドックからドラッグして、スタートとエンドの間にドロップしましょう(アイコンの操作についてはP7参照)。





「ブロック」アイコンは、上下のボタンで値を増やしたり減らしたりできる。ここでは「3.0秒」に設定しよう。



「３秒間、前にすすむ」プログラム完成!

プログラムを書き込もう

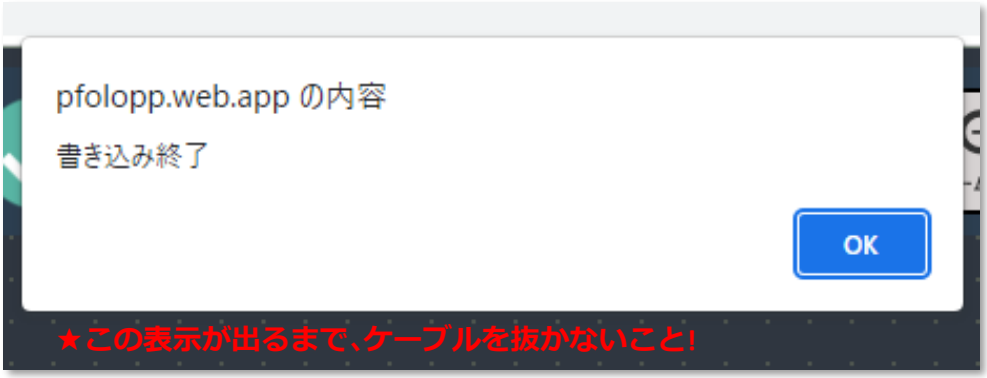
プログラムができたら、ロボットに転送しましょう。

①ロボットの電源スイッチを「ON」にします。
緑のインジケータランプが点灯します。

②付属のUSBケーブルをつかって、
パソコンとロボットを接続します。
オレンジのインジケータランプが点灯します。
コネクターをよく見て、まっすぐ差し込みましょう。ムリな方向に差し込もうとすると、コネクタが破損します。

③「かきこみ」ボタンをタップしてプログラムを書き込みます。

④画面に書き込み終了と表示されれば完了です。



動作チェックをしよう

書き込んだプログラムを実行してみましょう。

写真のように、**フェイスカバーを軽く押さえながら**、USBケーブルを引き抜きます

フェイスカバーを押さえずに、ケーブルだけひっぱると、マイコンボードごと外れてしまうので注意!

⇩

ファンクションボタンAと
ファンクションボタンBを**同時に押す**と、
青のインジケータランプが点滅したあと、プログラムを実行します

⇩

3秒間、前に進む!

プログラム実行中は、青のインジケータランプが点灯します

⇩

スタンバイモードにもどる

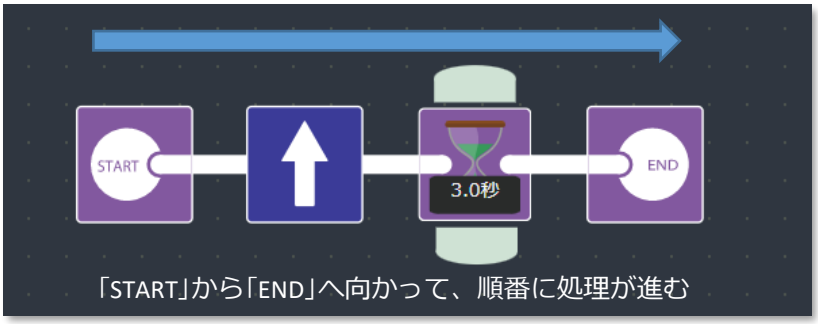
プログラムが終了すると
緑のインジケータランプが点灯します
緑のランプは、**ロボットがプログラムの実行待ちの状態**であることを表します
この状態で、ファンクションボタンAとファンクションボタンBを同時押しすると、再びプログラムを実行させることができます。

3．いろんな動きをやってみよう

プログラムの基本・順次処理

前の章でつくったプログラムをおさらいしましょう。

①「START」から始まり、②前に向かって進み、③3秒間ブロック、そして④終了、というプログラムでした。



プログラムとは、かんたんにいえば、ロボットにしてほしいことを順番にならべたものなのです。そしてロボットは、プログラムにかかれた命令を、順番どおりに実行していきます。このことを「順次処理」といい、プログラムの一番基本となる考えかたです。
この章では、他のアイコンもつかってみながら、「順次処理」の考えかたになれていきましょう。

この章で使うアイコン

前にすすむ

ロボットが
前へ向かって進みます。

後ろにさがる

ロボットが
うしろへ向かって進みます。

左に回転

ロボットがその場で
左へ回転します。

右に回転

ロボットがその場で
右へ回転します。

サウンド

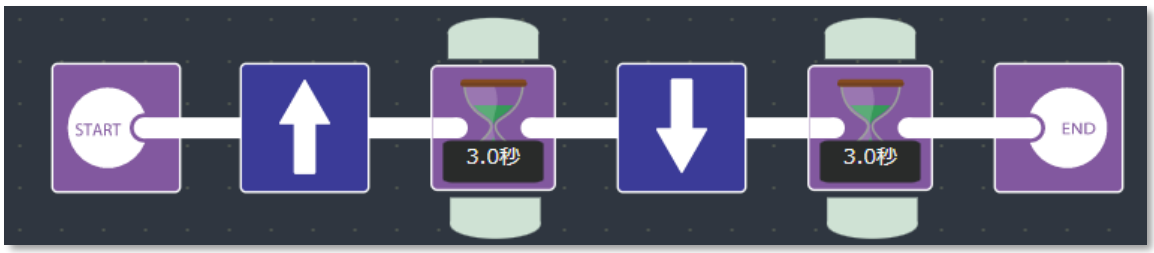
効果音を再生します。
効果音の種類は、パラメータボタンで切り替えることができます。

ブロック

設定した時間だけ、プログラムが次のアイコンに進むのをブロックします。

3秒間前に進む → 3秒間後ろに進む

下図のような「3秒間前進→3秒間後進」するプログラムを作ってみましょう。



<3秒間前進→3秒間後進>

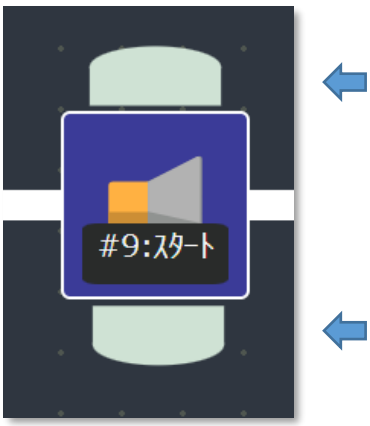
動作チェックをしよう

START → 前進(3秒間) → 後進(3秒間) → END

という順番で命令が実行されていくことを確認しましょう。

やってみよう

「左に回転」アイコンと「右に回転」アイコンも使ってみましょう。
どれくらい時間､回転させるかは､「左に回転」アイコン /「右に回転」アイコンの後ろにおく「ブロック」アイコンに設定する秒数によってコントロールします。
「サウンド」アイコンは、ロボットに効果音を再生させる命令です。
上下のボタンで、効果音の種類を切り替えられますから、いろいろな音を鳴らしてみてください。

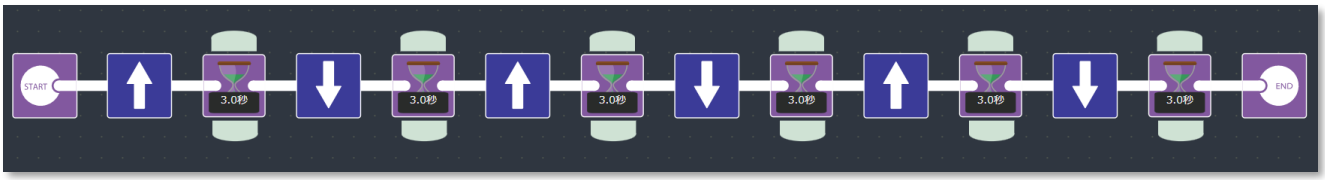


上下のボタンで、効果音の種類を切り替えられる

4．同じ動きを繰り返そう

同じ動きのアイコンをたくさんならべる？

前章でつくった「3秒間前進→3秒間後進」を、3回繰り返すプログラムを作ろうとしたとき、どうすれば実現できるでしょうか？



かんたんに考えるなら、「3秒間前進→3秒間後進」という動作を3回ならべることです。でもこのプログラム、前進と後進を繰り返すだけの単純な動きなのにとても長くて読みにくいですね。それに「繰り返す回数を100回にしてください」なんて言われたらもう大変ですね。

この章ではこういった単純な動作を繰り返し行うときに便利な「繰り返し処理」について学んでいきます。

この章で使うアイコン

回数ループ

囲まれた部分のプログラムを指定した回数だけ繰り返します。
繰り返す回数は、前におくアイコンによって設定します。

数値定数

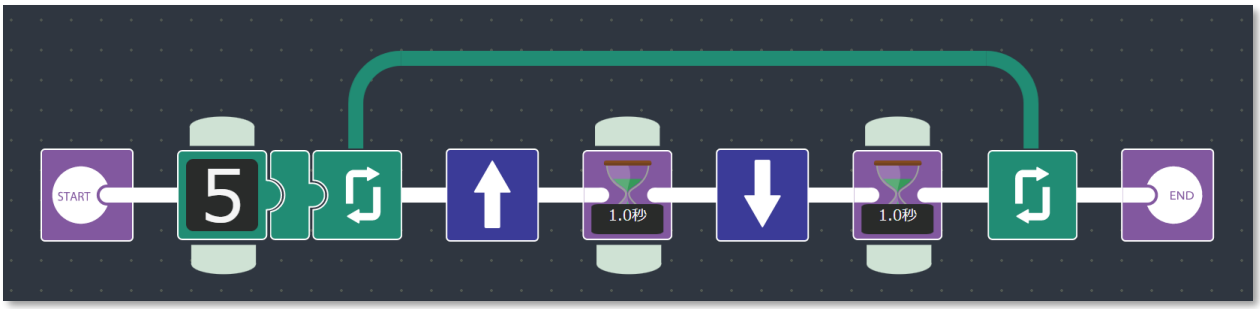
回数ループなどに使う数値を指定します。
上下のボタンによって数値を増減できます。

無限ループ

囲まれた部分のプログラムを無限に繰り返します。

「1秒間前進 → 1秒間後進」を5回繰り返す

「1秒間前進→1秒間後進」を5回繰り返すプログラムをつくります。
繰り返す動作を「回数ループ」の中に置き、繰り返す回数を「回数ループ」の前に置く「数値定数」アイコンで設定します。
「数値定数」アイコンは、上下のボタンで値を増減できます。ここでは5回繰り返したいので「5」としています。

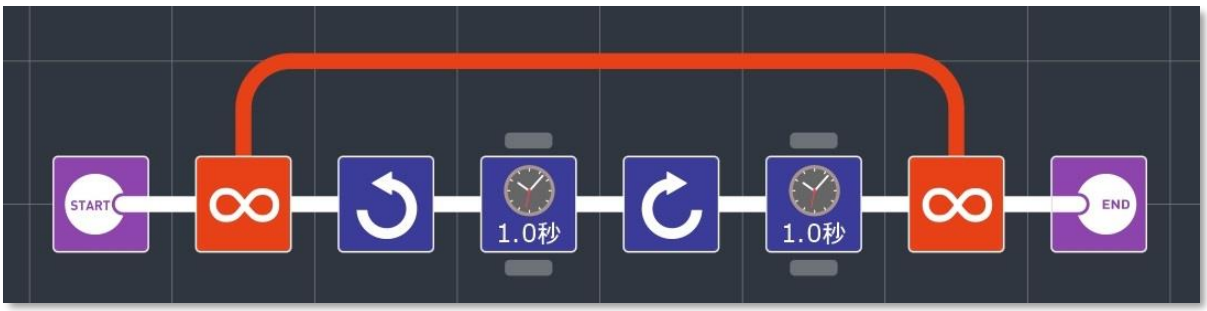


動作チェックをしよう

「前進(1秒間)→後進(1秒間)」を5回繰り返していますか？

やってみよう

回数を指定せず、無限にくりかえすループが「無限ループ」です。
左回転と右回転を、それぞれ1秒ずつ、ずっと繰り返すプログラムを作ってみましょう。
※「無限ループ」を使ったプログラムは「END」アイコンに到達しないため、電源スイッチをオフにするまで動き続けます。



5. 赤外線センサーを使ってみよう

条件分岐 ～場合によって処理を変える

ここまで、「順次処理」、「繰り返し処理」という、プログラミングの基本的な考えかたをみてきました。この章では、プログラミングの基本的な考えかたの3つめ、「条件分岐」を学習しましょう。

「条件分岐」とは、「場合」によって処理を変えること。フォロは赤外線センサーで、顔の前にモノがあるかどうかを見ることができます。そして、顔の前にモノが「ある**場合**」と、「ない**場合**」とで、ちがう動きをさせることができるのです。

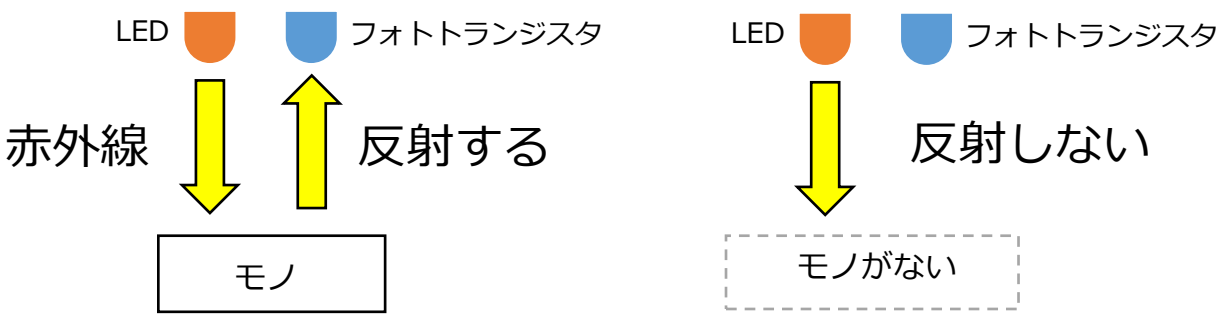
赤外線センサーのしくみ

フォロの顔を前からみると、2つのくぼみがあることがわかります。

このくぼみをのぞき込むと、「小さなまるいもの」が2つついているでしょう。

このうち、黒いほうを「フォトトランジスタ」、うすいむらさきいろのほうを「赤外線LED」といいます。赤外線LEDは、「赤外線」という人間の目には見えない光を出す部品です。「フォトトランジスタ」は光があたっているときとあたっていないときとで電気の流れかたが大きく変わる性質を利用して、光があたっているかどうかを調べるために使われます。

下の図をみてください。フォロの前にモノがあるときは、LEDから出た赤外線がモノにはねかえってフォトトランジスタにあたります。一方、モノがないときは、赤外線がはねかえってこないので、フォトトランジスタには赤外線はあたりません。



フォトトランジスタは、赤外線があたっているときとあたっていないときで電気の流れかたが大きく変化するので、フォロは前にモノがあるかどうかを電気の変化として知ることができるのです。

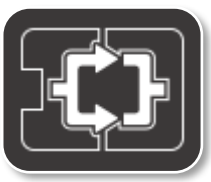
この章で使うアイコン

IRセンサー右

フォロの顔の右側についた赤外線センサーの状態を調べます。赤外線センサーの前にモノがある場合は「○」を、ない場合は「×」を、次のアイコンにわたします。

IRセンサー左

フォロの顔の左側についた赤外線センサーの状態を調べます。

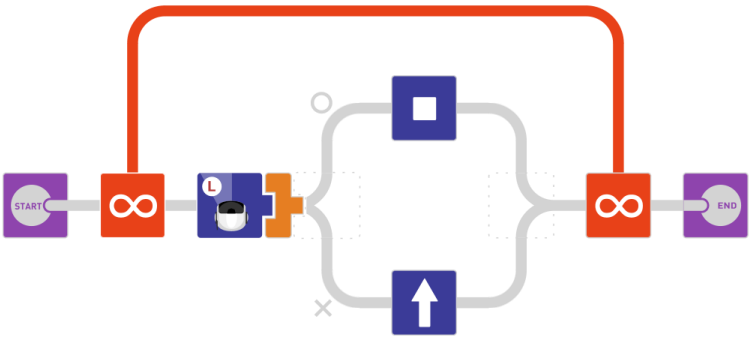


条件分岐

直前においたアイコンからうけとった「○」または「×」によって、処理を分岐させます。

IRセンサーで処理をわける①

「条件分岐」を使って、左前方にモノがないときは前進し、左前方にモノがあるときは停止するプログラムを作りましょう。

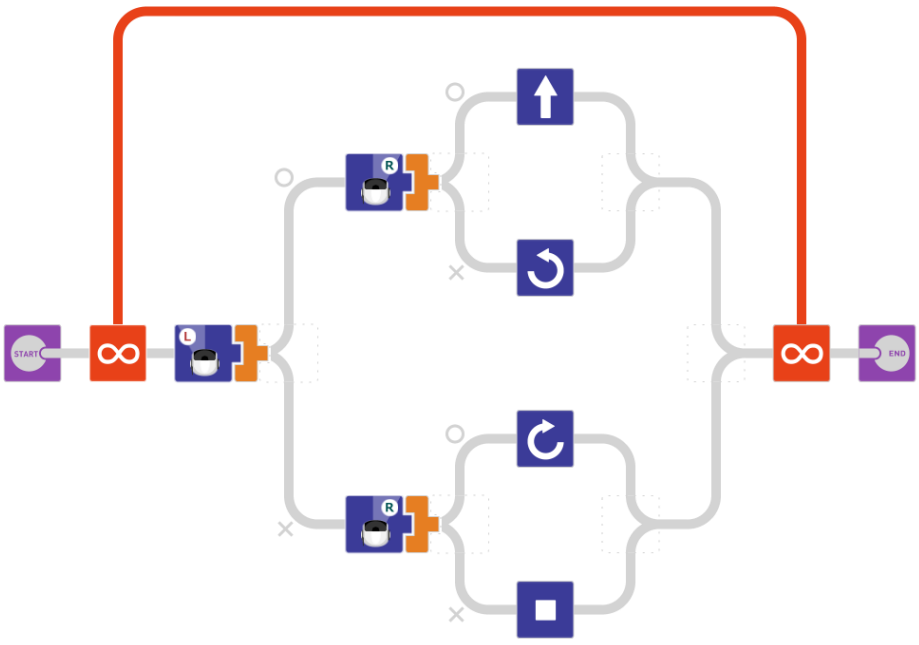


<左前方にモノがない場合は前進、ある場合は停止>

このプログラムの内容をくわしくみてみましょう。
まず、START直後からEND直前まで、プログラムの全部が無限ループで囲まれていますから、囲まれた部分の処理をずっとくり返すことになります。
まず「IRセンサー左」アイコンで、フォロの顔の左側についた赤外線センサーの状態が調べられます。このとき、赤外線センサーの前にモノがあるときは「○」が、ないときは「×」が次のアイコンにわたされます。
次に置かれた「条件分岐」アイコンが、うけとった「○」または「×」に応じて処理をわけています。「○」をうけとったときは、○のほうの道すじに進み、「止まる」アイコンが実行されます。「×」をうけとったときは、×のほうの道すじに進み、「前にすすむ」アイコンが実行されます。

IRセンサーで処理をわける②

5-1のプログラムは、「条件分岐」によって、「①左前方にモノがない場合」、「②左前方にモノがある場合」の2つの場合について、処理をわけていました。
「条件分岐」で処理をわけたあとに、さらに「条件分岐」をおくことにより、より多くの場合について、処理をわけることができます。



<追跡モード>

このように「条件分岐」のあとに、さらに「条件分岐」で処理をわけることを**「入れ子（ネスト）」**といいます。入れ子を使うことで条件分岐を3通り以上にすることができます。

6. ドットマトリックスLEDを光らせよう

「ドットマトリックスLED」とは

「ドットマトリックスLED」とは、LEDをタテヨコ･規則的にたくさん並べたもの。フォロには、8行 x 8列･合計64個のLEDからなるドットマトリックスLEDが搭載されています。たくさん並んだLEDのひとつひとつを「ドット」と呼び、どのドットを光らせるかをプログラムでコントロールすることにより、図形や文字など、いろいろな情報を表示させることができます。

この章で使うアイコン

マトリックス ドット

どのドットを光らせるかを直接指定します。横方向に並べて配置すると連結して、最大8行x80列のマトリックスを表現できます。

マトリックス 数値

直前においたアイコンからうけとった数値を表示します。

マトリックス 真偽

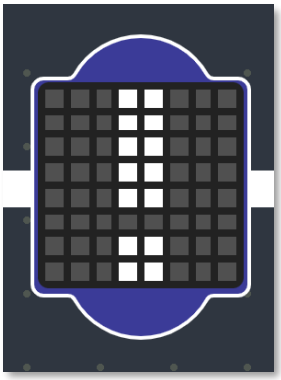
直前においたアイコンからうけとった「○」または「×」を表示します。

マトリックス 文字列

プログラムの中に配置すると、文字列の入力をうながし、入力された文字列を「マトリックス ドット」アイコンの並びに展開します。

ドット単位で光りかたをコントロール

「マトリックス ドット」アイコンは、ドットマトリックスLEDのうち、どのLEDを光らせるかを直接指定することができます。



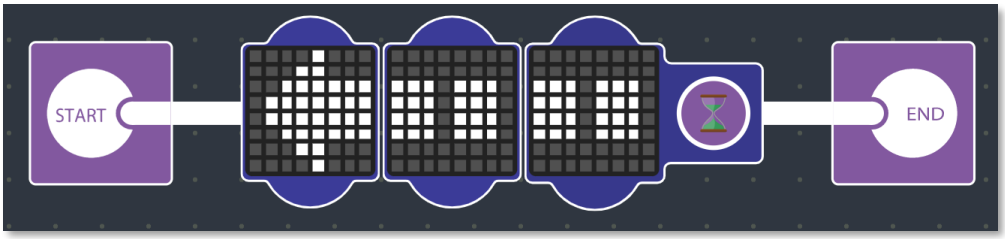
8行 x 8列に並んだ小さい正方形がドットを表しており、タップするごとに、「光らせるか」・「光らせないか」を切り替えることができます。

左のアイコンは、ドットを点灯させて「！」を描いてみたところ。

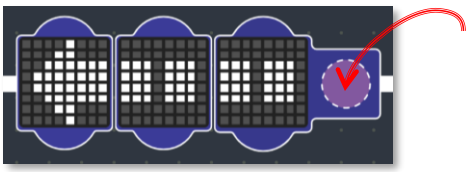
ドットが小さくてタップしづらい、というときは、「スイッチボード」の「ズームインボタン」で大きく表示してみましょう。

「マトリックス ドット」アイコンは、横方向に並べて配置するとくっついて、ひとつの横長のアイコンとしてあつかわれるようになります。

次のプログラムは、アイコンを3つ使って表現したやじるしをスクロール表示させる例です。アイコンを3つ並べることにより、8行 X 24列のマトリックスをつくっています。このようにしてつくった横長のマトリックスは、フォロに搭載された8行 x 8列のマトリックスLEDでは一度に表示しきれませんから、向かって左向きにスクロール表示されます。



「マトリックス ドット」アイコンを2個以上ならべたときに、最後のアイコンに表示されるマークは、スクロール表示が一巡するまで、次のアイコンに進むのをブロックすることを示しています。
このマークはタップすることで、ブロックする/しないを切り替えられます。



タップするごとに、「ブロックする/しない」を切り替える。
「ブロックしない」場合、スクロール表示が一巡するのを待たずに、次のアイコンに処理が進む。

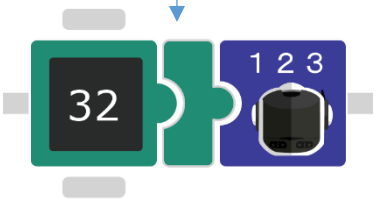
「数値」を表示させる

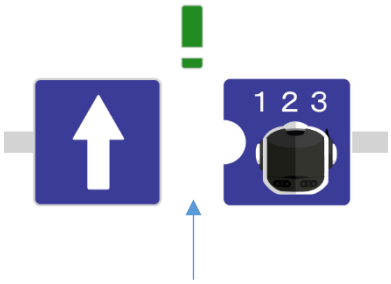
「マトリックス 数値」アイコンは、直前においたアイコンからうけとった数値を表示します。数値をうけとるのですから、直前においたアイコンは「数値をわたす」ものでなくてはいけません。フォロのプログラミングにつかえるアイコンのうち、「数値をわたす」ものは、「数値定数」、「数値変数出力」などがありますが、アイコンのかたちで見分けることができます。

「丸いでっぱり」は、次のアイコンに「数値」をわたすことを表す

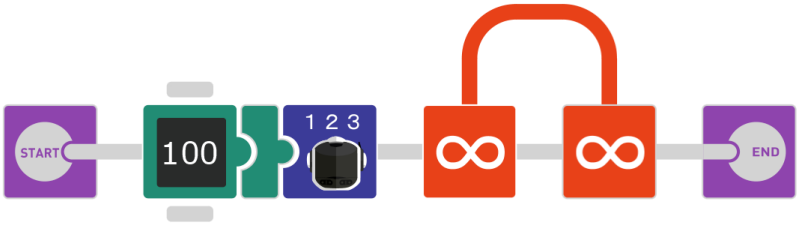
「丸いへこみ」は、直前のアイコンから「数値」をうけとることを表す

正しい組み合わせ
「数値定数」アイコンが「32」という数値を次にわたし、「マトリックス 数値」アイコンがそれをうけとっている





正しくない組み合わせ
「マトリックス 数値」アイコンに数値がわたされていない。そのため、「まちがい」をあらわす「！」が表示されている。

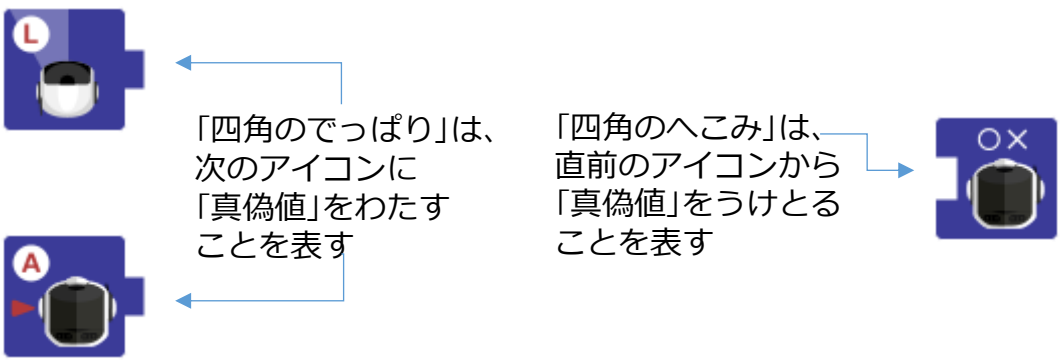


<数値100を表示するプログラム>

「真偽値」を表示させる

「マトリックス 真偽」アイコンは、直前においたアイコンからうけとった「○」または「×」を表示します。フォロのプログラミングにつかえるアイコンのなかには、命令の結果を「○」か「×」かのどちらかで、次のアイコンにわたすものがあります。たとえば「IRセンサー」アイコンは、前にモノがあるかどうかをしらべ、その結果を「○」か「×」かで次のアイコンにわたします(ある場合は「○」、ない場合は「×」)。「ファンクションボタン」アイコンは、ボタンが押されているかどうかをしらべ、その結果を「○」か「×」かで次のアイコンにわたします(押されている場合は「○」、押されていない場合は「×」)。このように「イエス」か「ノー」か、「○」か「×」かで表されるようなことがらのことを、すこしむずかしいことばでは「真偽値(しんぎち)」といいます。

「マトリックス 真偽」アイコンは「真偽値」をうけとって表示するアイコンですから、その直前には真偽値をわたすアイコンを置かなくてはいけません。このあたりのルールは、先ほど説明した「マトリックス 数値」アイコンと同じです。

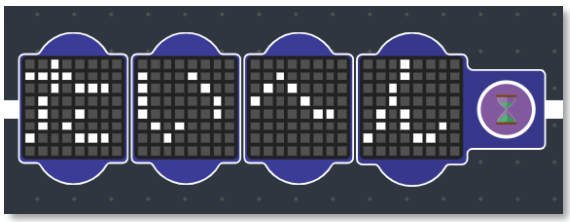


つぎのプログラムは、左の赤外線センサーの状態をしらべ、その結果を表示するという処理をずっとくり返すプログラムです。左の赤外線センサーの前になにもないときは「×」が、手を近づけるなどして、左の赤外線センサーの前にモノがある状態になると「○」が表示されます。

<左前方にモノがあるかないかを表示>

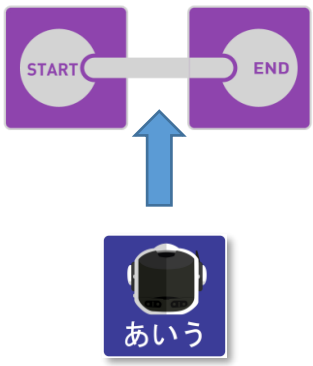
「文字列」を表示させる

この章のはじめのほうで説明した「マトリックス ドット」アイコンは、ドット単位で光かたを指定できるアイコンでした。これを使えば、文字や文章を表示させることもできそうです。

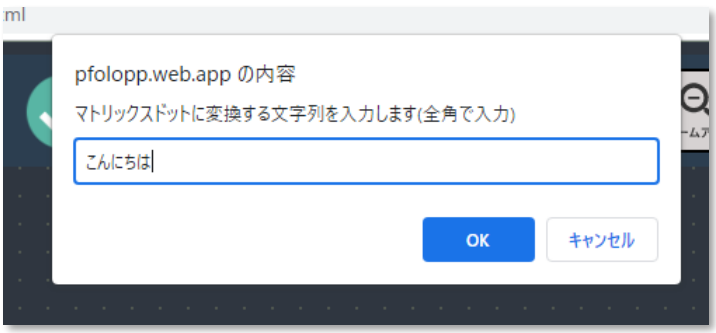


しかし、ドット単位で文字を書いていくのは、とくに文章を表示させたいときはたいへんですね。

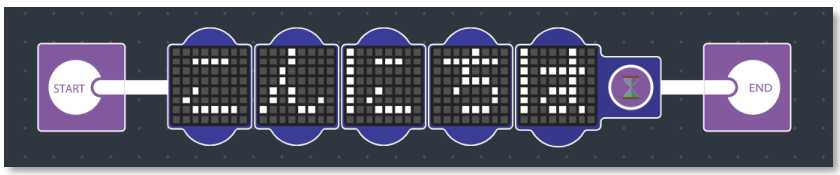
そこで、文字や文字の並び(文字列といいます)を、「マトリックス ドット」アイコンの並びに変換してくれるアイコンが用意されています。
ここでは、その「マトリックス 文字列」アイコンの使いかたを説明します。



文字列をおきたい場所に、「マトリックス 文字列」アイコンをドロップします。



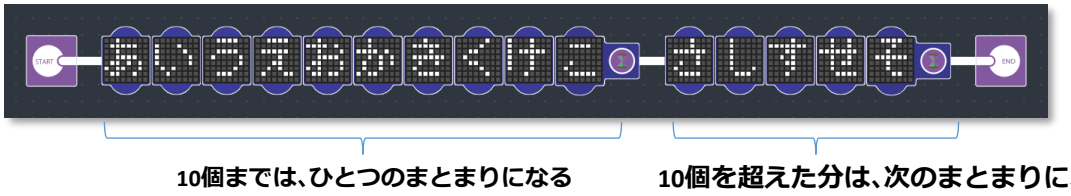
ダイアログがでてくるので、表示させたい文字列を入力して「OK」を押します。



入力した文字列が、「マトリックス ドット」アイコンの並びに変換されます。

長い「文字列」を表示させる

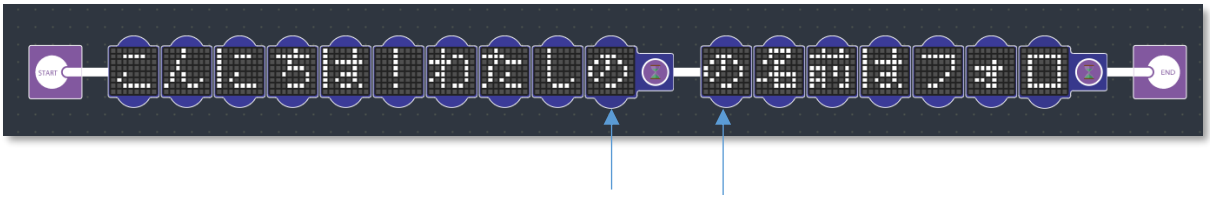
先に述べたように、「マトリックス ドット」アイコンは、横方向に並べて配置するとくっついて、ひとつの横長のアイコンとしてあつかわれるようになります。
しかし、くっつくアイコンの数には上限があり、10個を超えたまとまりになることはありません。



10個までは、ひとつのまとまりになる　　**10個を超えた分は、次のまとまりになる**

このプログラム例では、最初のまとまりが「ブロックする」になっているため、最初のまとまりのスクロール表示が一巡するのを待ってから、つぎのまとまりのスクロール表示が始まります。

次のプログラム例では、最初のまとまりと、続くまとまりの境目に当たる文字を、あえて重複させて置いています。



「の」を、あえて重複させている

このようにすると、最初のまとまりの最後の文字「の」が表示された直後、次のまとまりの最初の文字「の」が同じ位置に表示されることになるため、スクロール表示の継ぎ目がまったくわからなくなります。
このテクニックを使えば、10文字を超える文章でも、まるでひとつのまとまりであるかのように、スクロール表示させることができますよ。

7. プログラムでおしゃべりしよう

この章で解説する音声合成機能は、
プログラミング・フォロ スピーク for CHROME (MR-9123)のみに搭載されており、
プログラミング・フォロ for CHROME(MR-9122)では動作しません。

「スピーチ・シンセサイザー」とは

「スピーチ・シンセサイザー」とは、人間の音声を人工的に作り出す機械装置のことです。
プログラミング・フォロ スピーク(MR-9123)には、文章および数値から音声を合成する機能が搭載されています。
この機能は、単に録音された音声を再生するものとは違い、プログラムから操作することで、無限の表現が可能です。
それでは、さっそく「音声合成」プログラミングにチャレンジしてみましょう。

この章で使うアイコン

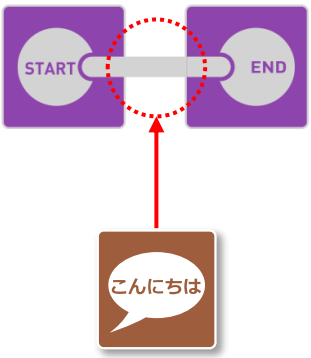
読みあげ 文章

プログラムの中に配置すると、文章の入力をうながし、
入力された文章を読みあげます。

読みあげ 数値

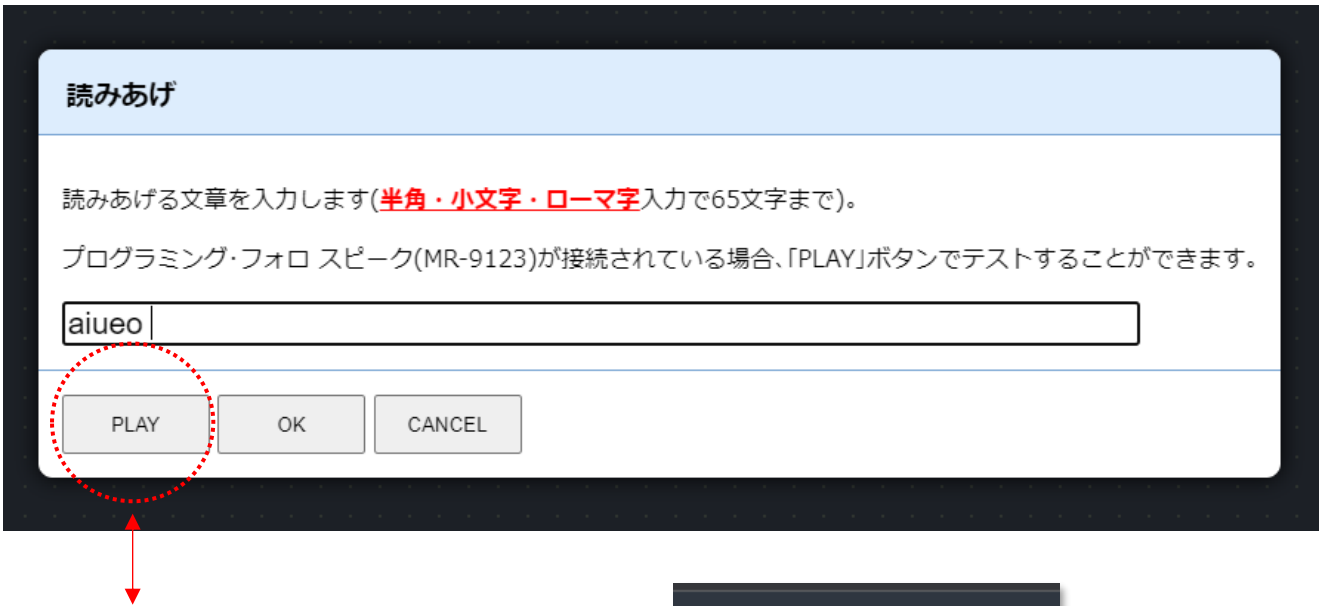
直前においたアイコンからうけとった数値を読みあげます。

文章をよみあげる

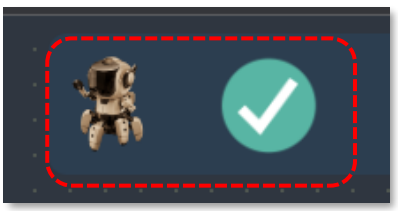


読みあげ 文章をおきたい場所に、アイコンをドロップします。

読みあげる文章を入力するダイアログが表示されます。



MR-9123をUSBケーブルで接続している場合は､｢PLAY｣ボタンで入力中の文章をテスト発声することができます。



◆文章を入力するときのポイント◆

①半角 ・ 小文字 ・ ローマ字 で入力する

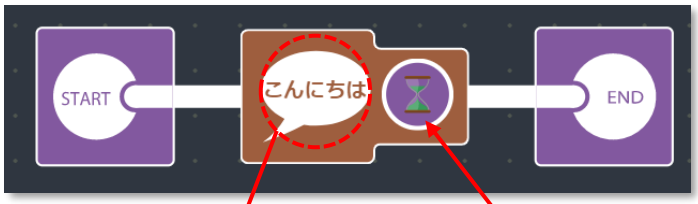
②アルファベット(小文字)以外は入力しない

(例外あり。次ページ｢文章を読みあげる～より自然に～｣をご参照ください。)

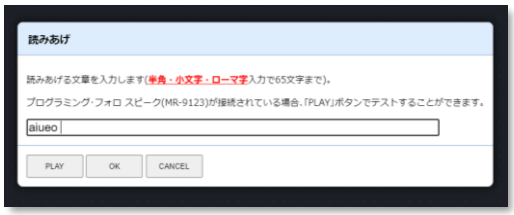
③ロボットを接続して､テスト発声で入力を確かめることをおすすめ

読みあげる文章が入力できたら､｢OK｣を押してダイアログを閉じます。

入力した文章を読み上げるプログラムの完成



タップすると、
文章を入力するダイアログを
再度呼び出せます。



タップして、
「ブロックする/しない」を切り替えます。

●「ブロックする」とき
入力した文章の読みあげが終わるまで、
次のアイコンに進みません。

●「ブロックしない」とき
入力した文章の読みあげの終了を待たず、
次のアイコンに処理が進みます。

文章をよみあげる～より自然に～

より自然に聞こえるように、文章を入力する方法を説明します。

ポイント① 発音するとおりに入力

文字で書きあらわされた文章は、実際に発声する音と異なる場合がありますが、
この場合は、実際に発声する音で入力します。

例:

「金曜日(きんようび)」→ 実際には「きんよーび」と発声 →「kinnyo-bi」と入力

「高速道路(こうそくどうろ)」→ 実際には「こーそくどーろ」と発声
→「ko-sokudo-ro」と入力

「こんにちは」→ 実際には「こんにちわ」と発声 →「konnnichiwa」と入力

「僕はロボット」→ 実際には「ぼくわろぼっと」と発声
→「bokuwarobotto」と入力

ポイント② アクセントをつける

単語や文節のなかで、強く発音する箇所(アクセント)がある場合は、
その音のあとに「’ (アポストロフィー・Shift + 7)」を置きます。

例:

「金曜日(きんようび)」→「きんよーび」(「よ-」を強く発音)

→「kinnyo-’bi」と入力

「山口県(やまぐちけん)」→「やまぐちけん」(「ぐ」を強く発音)

→「yamagu’chikenn」と入力

ポイント③ アクセント句を分ける

読みあげたい文章のなかで、強く発音する箇所(アクセント)が複数ある場合は、文節を「 (スペース)」または「;(セミコロン)」で区切り、それぞれの文節にたいしてアクセント記号(前述のアポストロフィー)を置きましょう。
また、アクセントのあるなしにかかわらず、文節をアクセント句として分割したほうが自然に聞こえる場合があります。
「 (スペース)」はアクセント句を分けるとともに、ポーズ(無音の時間)を発生させます。一方、「;(セミコロン)」を使った場合には、ポーズは発生しません。

例:

「福岡県太宰府市(ふくおかけんだざいふし)」→「ふくおかけんだざいふし」
(「お」と「ふ」を強く発音)

→「fukuo’kakenndazaifu’shi」アクセントをつけただけだと不自然

→「fukuo’kakenn;dazaifu’shi」アクセント句を分けると自然

ポイント④ 無声化

音の中には、文章中の位置によって、「母音の無声化」がおこるものがあります。
わかりやすい例としては、文末の「です」、「ます」の「す」。
このような音は、音の前に「_(アンダーバー)」をつけるとより自然に聞こえます。

例:

「終了します(しゅうりょうします)」→「しゅ-りょ-しますす」(末尾の「す」が無声化)

→「syu-ryo-;sima_su」アンダーバーを前につけることで母音の無声化

その他のテクニック

文末のひとつの音を「;(セミコロン)」で分け、さらに「'(アポストロフィー)」でアクセントをつけると、疑問文のような上がり調子にすることができます。

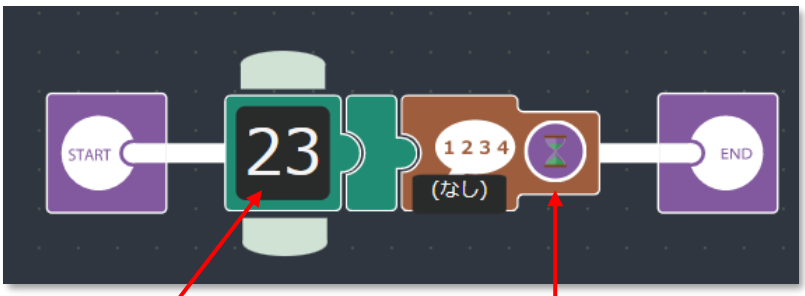
例:

「終了しますか?」→「しゅーりょーしますか↑」(上がり調子にしたい)
→「syu-ryo-;sima_su;ka'」と入力

数値を読みあげる

「読みあげ 数値」アイコンを使うと、直前においたアイコンからうけとった数値を読み上げることができます。
さらに、助数詞をつけた数値読みあげにも対応しています。

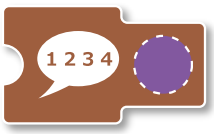
次の例は、直前においた「数値定数」アイコンによって与えられた数値を読み上げるプログラムです。



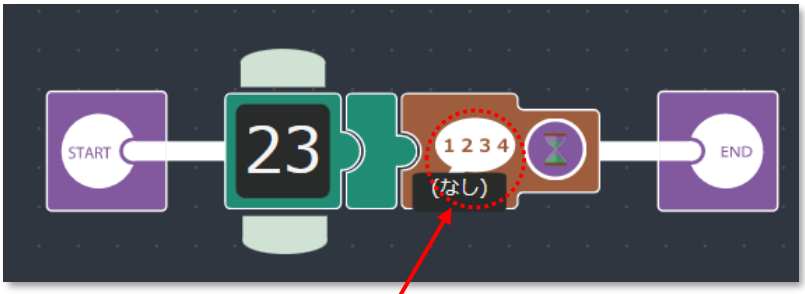
直前においたアイコンから数値を受け取ります。

タップして、「ブロックする/しない」を切り替えられます。

●「ブロックする」とき
数値の読みあげが終わるまで、次のアイコンに進みません。

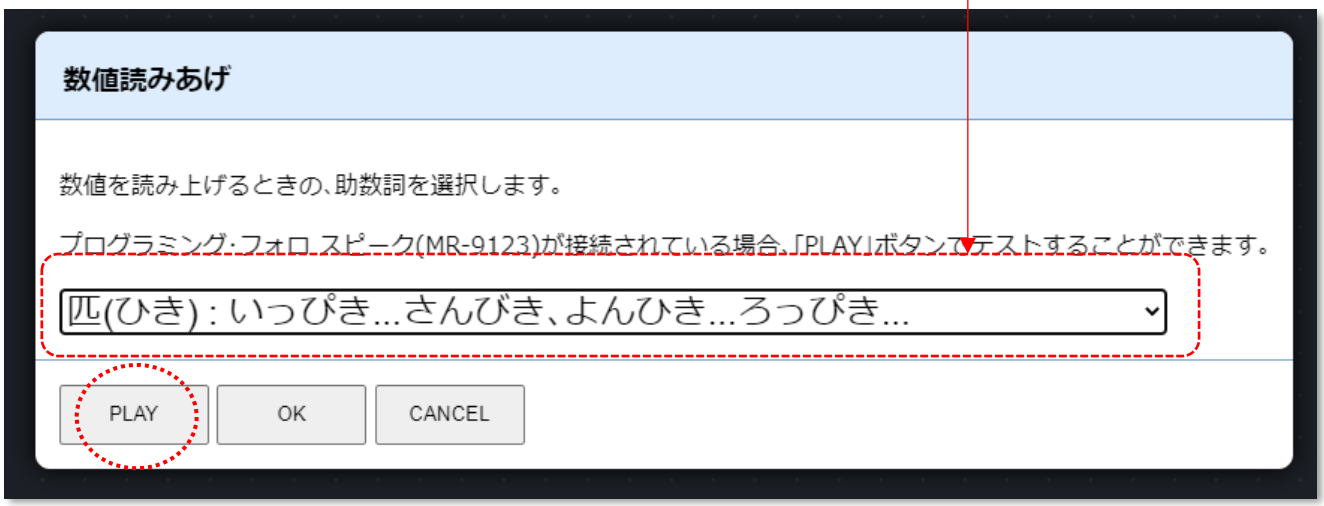


●「ブロックしない」とき
数値の読みあげの終了を待たず、次のアイコンに処理が進みます。

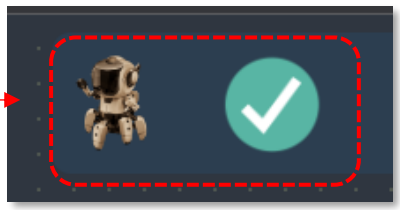


タップすると、
助数詞を選択するダイアログが開きます。

数値の読みあげに使う助数詞を選択します。

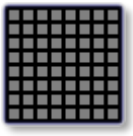


MR-9123をUSBケーブルで接続している場合は、「PLAY」ボタンで助数詞つきの数値読みあげをテスト発声することができます。



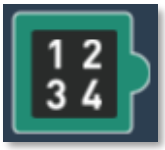
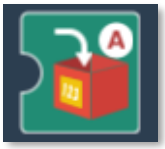
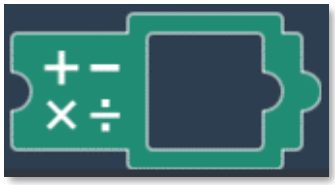
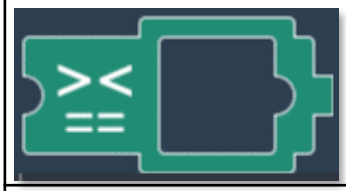
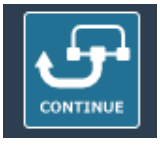
数値の読みあげに使う助数詞を選択したら、
「OK」ボタンを押してダイアログを閉じます。

8．アイコンの説明

アイコン	アイコンの名前・説明
	<前にすすむ> ロボットが前の方向へ進みます。
	<後ろにさがる> ロボットが後ろの方向へ進みます。
	<左に回転> ロボットがその場で左方向に回転します。
	<右に回転> ロボットがその場で右方向に回転します。
	<止まる> ロボットの前進、後進、左回転、右回転の動きを停止させます。
	<マトリックス 数値> 直前においたアイコンからうけとった数値を表示します。 表示できるのは-32768 ～ 32767の範囲の整数です。 小数がわたされると小数部を切り捨てます。
	<マトリックス 真偽> 直前においたアイコンから受け取った真偽値を表示します。
	**<マトリックス ドット>** ドットマトリックスLEDの点灯/消灯状態を、ドット単位で指定します。 複数個並べておくと、連結されたひとつのマトリックスとして扱われ、 最大8行 X 80列のマトリックスを表現できます。
	<マトリックス 文字列> プログラム中に置くと、文字列の入力を促すダイアログを表示し、入力された文字列の各文字をドットで表現した「マトリックス ドット」アイコンの並びに変換します。JIS第1、2水準範囲の文字を変換できます。 それ以外の文字が入力された場合は、「空白」に変換します。

アイコン	アイコンの名前・説明
	<サウンド> 効果音を鳴らします。フォロには10種類の効果音が収録されており、どの効果音を鳴らすかはパラメーターボタンを押すことで設定できます。
	<ブロック> パラメーターボタンによって設定した時間、プログラムの処理が次のアイコンに進むのをブロックします。 設定できる時間は最小0秒で、0.1秒ずつ増やすことができ最大で10秒です。初期値は1秒です。
	<IRセンサー (左 および 右)> 赤外線センサーの状態を読み取り、真偽値で次のアイコンに渡します。 赤外線センサーが物体を感知しているとき(赤外線センサーの前にモノがあるとき)は「○」を、感知していないとき(モノがないとき)は「×」をわたします。
	<ボタン (A および B)> ファンクションボタンの状態を読み取り、真偽値で次のアイコンに渡します。 ファンクションボタンが押されているときは「○」を、押されていないときは「×」をわたします。
	<条件分岐> 真偽値を受け取り、「○」を受け取った場合は上側の道すじへ、「×」を受け取った場合は下側の道すじへ処理を進めます。
	<スペーサー> プログラムを読みやすくするなどの目的で、アイコンとアイコンの間にスペースを挿入するためのアイコンです。プログラムの動作に与える影響はありません。

アイコン	アイコンの名前・説明
	<無限ループ> [無限ループ]内の処理を実行します。ループの最後に到達した場合は、[無限ループ]の先頭に戻ります。
	<条件ループ> 真偽値を受け取り、[○]の場合は、[条件ループ]内の処理を実行しループの最後に到達すると、[条件ループ]の先頭に戻りもう一度真偽値を受け取ります。[×]の場合は、[条件ループ]末尾の次のアイコンに移動します。
	<真偽定数> パラメータボタンで設定した真偽値[○]または[×]を右のアイコンに渡します。
	<真偽変数 入力> 真偽値を受け取り、パラメータボタンで指定した真偽変数(A, B, C, D, E)に格納します。
	<真偽変数 出力> パラメータボタンで指定した真偽変数(A, B, C, D, E)から真偽値を取り出して、右のアイコンに渡します。なお各真偽変数の初期値は[×]です。
	<○/×入れかえ> 真偽値を受け取り、その値を反転させた真偽値を右のアイコンに渡します。[○]を受け取ると[×]を渡し、[×]を受け取ると[○]を渡します。
	<論理演算> [論理演算]アイコンの左側と内側の真偽値で論理演算を行った結果を、真偽値で右のアイコンに渡します。論理演算の種類(「AND(論理積)」,「OR(論理和)」,「XOR(排他的論理和)」)は、パラメータボタンで選択します。
	<スタート> このアイコンからプログラムが始まります。
	<エンド> 全ての動作を停止し、プログラムを終了します。

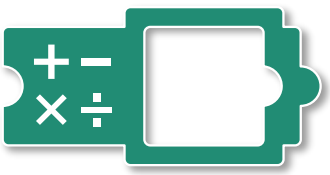
アイコン	アイコンの名前・説明
	<回数ループ> 数値を受け取り、その回数、[回数ループ]内の処理を実行します。
	**<数値定数>** 設定した数値を右のアイコンに渡します。数値は、上下に表示されるボタンをタップして増減させるか、アイコンをタップして表示されるダイアログから直接入力できます。 設定できる数値は最小-100、最大100の範囲内の整数です。
	**<数値変数 入力>** 数値を受け取り、パラメータボタンで指定した数値変数(A, B, C, D, E)に格納します。格納できる数値の範囲は、 -32768 ～ 32767の整数となります。
	<数値変数 出力> パラメータボタンで指定した数値変数(A, B, C, D, E)から数値を取り出して、右のアイコンに渡します。なお各数値変数の初期値は0です。
	**<数値演算>** [数値演算]アイコンの左側と内側で数値をそれぞれ受け取り、左側と内側の数値で算術演算を行った結果を、数値で右のアイコンに渡します。演算の種類(足し算、引き算、掛け算、割り算)は、パラメータボタンで選択します。
	<プラス/マイナス入れかえ> 数値を受け取り、その値の正負(プラスマイナス)を反転させた数値を右のアイコンに渡します。
	**<数値比較>** [数値比較]アイコンの左側と内側の数値で数値比較を行った結果を、真偽値で右のアイコンに渡します。比較の種類(大なり(>), 小なり(<), 等価(=))は、パラメータボタンで選択します。
	<ブレーク> ループ内部において使います。このアイコンに到達すると、それ以降の処理を行わず、最も内側のループの末尾に移動します。
	**<コンティニュー>** ループ内部において使います。このアイコンに到達すると、それ以降の処理を行わず、最も内側のループの先頭(※)に移動します。 ※条件ループの内部に置いた場合には、当該条件ループの条件式の先頭に移動

アイコン	アイコンの名前・説明
	＜読みあげ 文章＞ プログラムの中に配置すると、文章の入力をうながし、入力された文章を読みあげます。 **MR-9123専用**
	＜読みあげ 数値＞ 直前のアイコンから受け取った数値を読みあげます。 また、助数詞つきで読みあげることもできます。 **MR-9123専用**

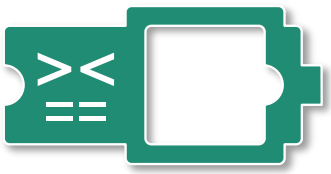
演算系アイコンの補足説明

ここでは、使いかたの理解がやや難しいと思われる、
演算系のアイコンについて、補足説明します。

論理演算

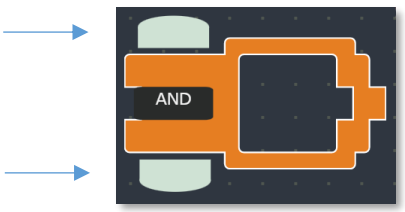


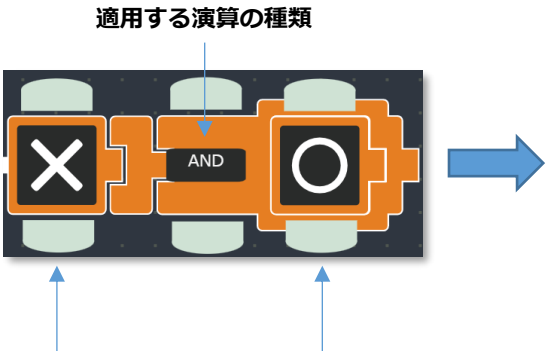
数値演算



数値比較

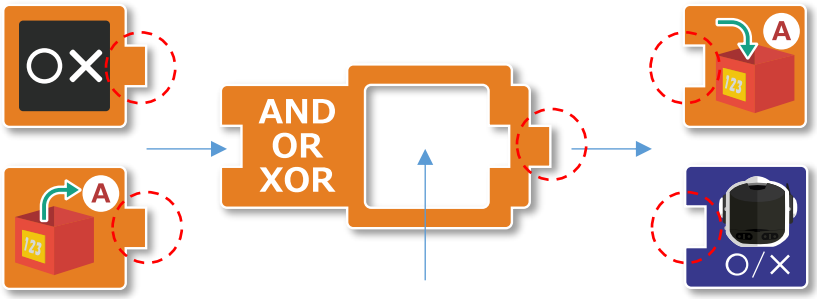
演算系のアイコンがワークスペースに置かれたときには、上下のボタンで演算の種類を切り替えられる。





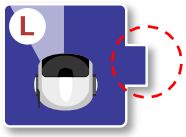
<論理演算>

「論理演算」アイコンは、2つの数値を受け取って、
「AND(論理積)」、「OR(論理和)」、「XOR(排他的論理和)」を行い、
結果を真偽値で次のアイコンに渡します。



◆演算結果を真偽値として次のアイコンに渡すため、真偽値を受け取るアイコンを配置しなくてはならない。

◆2つの真偽値を演算するので、真偽値を渡すアイコンを配置しなくてはならない。

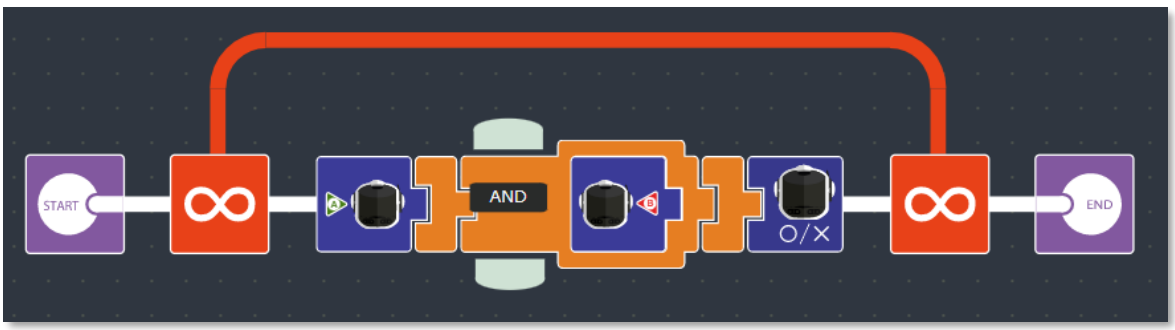


AND(論理積)は、演算対象の2つの真偽値が、共に「真(○)」であるときだけ、「真(○)」となり、その他の組み合わせはすべて「偽(×)」となる演算です。

OR(論理和)は、演算対象の2つの真偽値のうち、1つでも「真(○)」があれば「真(○)」、両方「偽(×)」のとき「偽(×)」となる演算です。

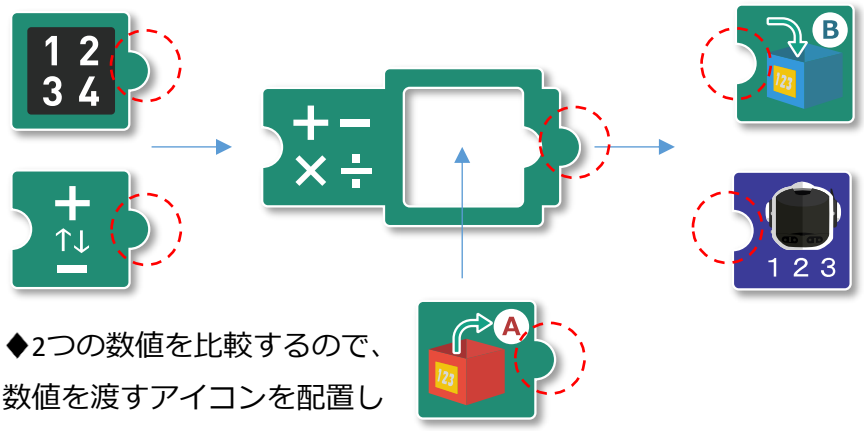
XOR(排他的論理和)は、演算対象の2つの真偽値のうち、1つだけが「真(○)」のとき「真(○)」、その他の組み合わせはすべて「偽(×)」となる演算です。

次のプログラムは、「ボタンA」アイコンと「ボタンB」アイコンの状態を論理演算して、その結果をマトリックスLEDに表示します。
「ボタン」アイコンは、ボタンが押されているとき「真(○)」、押されていないとき「偽(×)」を次に渡します。
A/Bボタンの押し方の組み合わせで、演算の結果がどうなるか、各演算で試してみてください.



<数値演算>

「数値演算」アイコンは、2つの数値を受け取って、
「足し算」、「引き算」、「かけ算」、「わり算」 を行い、演算結果の数値を次のアイコンに渡します。



◆演算結果を数値として次のアイコンに渡すため、数値を受け取るアイコンを配置しなくてはならない。

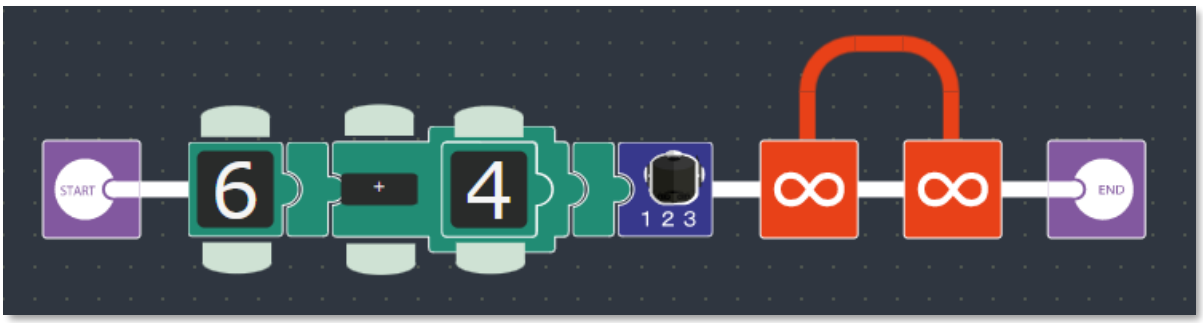
◆2つの数値を比較するので、数値を渡すアイコンを配置しなくてはならない。

足し算、引き算、掛け算、割り算については、一般的な理解と同じですが、
プログラミング・フォロの処理能力の関係で、次のような制限があります。

演算の結果は、-32768 ～ 32767 の範囲の整数となる

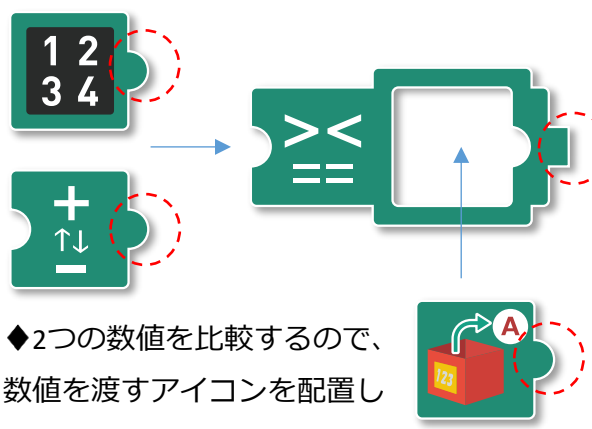
このため、計算結果が小数となる場合は、小数点以下は切り捨てられます。
また、演算の結果が-32768より小さくなる場合は下限の「-32768」となります。
演算の結果が32767より大きくなる場合は上限の「32767」となります。

次のプログラムは、2つの「数値定数」アイコンの値を計算して、その結果をマトリックスLEDに表示します。
「数値定数」アイコンの値や、演算の種類を変えてみて、結果の表示がどうなるかを確かめてみてください。



<数値比較>

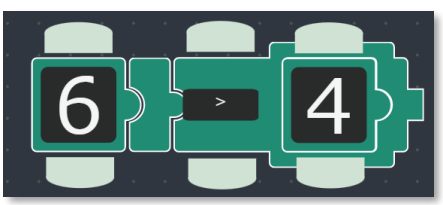
「数値比較」アイコンは、2つの数値を受け取って、
不等号/等号「> (大なり)」、「<(小なり)」、「=(イコール)」でつなぎ、
その式が正しいか/正しくないかを真偽値として次のアイコンに渡します。



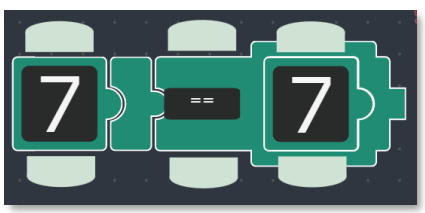
◆2つの数値を比較するので、
数値を渡すアイコンを配置し
なくてはならない。

◆式の真偽を次のアイ
コンに渡すため、
真偽値を受け取るアイ
コンを配置しなくては
ならない。

「数値比較」アイコンは、接続したアイコンと選択した演算種類から構成される式が
「正しい(○)/正しくない(×)」を真偽値として次に渡すことに注意してください。
例えば、次の式は「正しい」ので、「真(○)」が次のアイコンに渡されます。



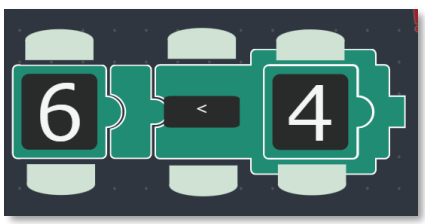
6 > 4 → 正しい



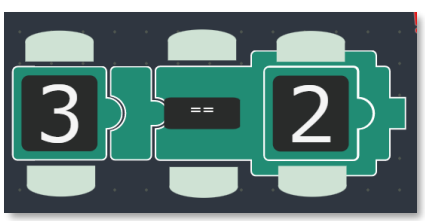
7=7 → 正しい

**「真(○)」が
次のアイコン
に渡される。**

一方、次の式は「正しくない」ので、「偽(×)」が次のアイコンに渡されます。



6 < 4 → 正しくない



3=2 → 正しくない

**「偽(×)」が
次のアイコン
に渡される。**