

加速度センサプログラマー用ソフトウェア

ビュートビルダーG 説明書

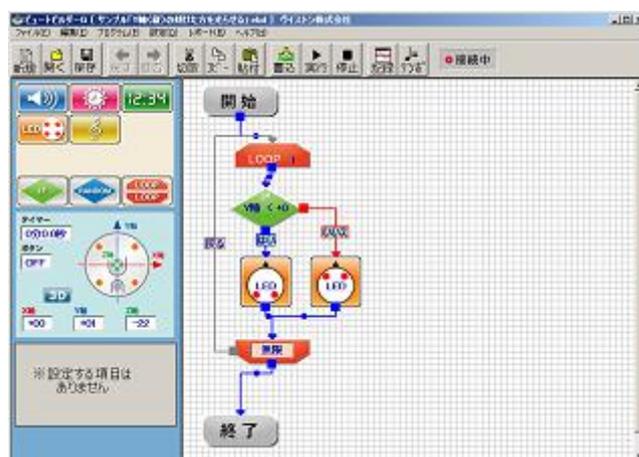
ガイストン株式会社

加速度センサプログラマーは、**加速度センサ**や**ブザー**、**LED**を搭載し、PCからそれらをプログラミングできる学習用教材です。加速度センサは、スマートフォンやゲーム機のコントローラにも使われており、重力や「振る」「傾ける」などの動きに反応します。センサの応答が速く、反応も体の動きに直結するので、センサによる「計測」を体でわかりやすく理解できます。更に、ブザーやLEDと組み合わせて動かすことで、ゲームなどを通じて楽しくプログラミングを理解できます。

加速度センサプログラマーのプログラミングには、専用ソフトウェア「ビュートビルダーG」を使います。ビュートビルダーGは、**マウスによる直観的な操作**が特徴で、誰でも簡単に**フローチャートを作る感覚**でプログラミングできます。



加速度センサプログラマー
説明では「本体」と記述します



ビュートビルダーG
説明では「ソフトウェア」と記述します

本説明書は、以下の3部構成となっています。順番に読み進めることで、スムーズに本体を使ったプログラムを習得できます。

【本説明書の構成】

① 使い始めるための準備 (4~13 ページ)

ソフトウェアのインストール方法や本体との通信など、本体に関する基本事項を説明しています。最初に必ずお読みください。

② プログラミングの学習 (14~54 ページ)

例題を使って、ソフトウェアの使い方やプログラミングの基礎を説明しています。プログラミングが一通り理解できるまでお読みください。

③ その他機能の説明 (55~71 ページ)

機能や操作別に、本ソフトウェアの使い方を説明しています。一通りプログラミングができるようになり、ソフトウェアの細かい機能を使う場合にお読みください。

目次

使い始めるための準備	4
1. 学習の前に	4
基本的な用語の確認	4
必要なものの準備	4
2. 本体の各部説明	5
各部の名称.....	5
電池の交換方法.....	6
電源の ON/OFF	7
3. ソフトウェアをインストールする	8
ソフトウェアのダウンロード.....	8
ソフトウェアのインストール.....	8
ソフトウェアの起動	10
4. 本体と通信する	10
本体と PC の接続.....	11
センサの反応の確認	12
DirectX ランタイムのインストール.....	13
プログラミングの学習	14
5. ソフトウェアの使い方を確認する	14
ソフトウェアの画面の説明.....	14
プログラムの作成手順	16
複数のブロックを使ったプログラム.....	21
LED を使ったプログラム.....	26
6. 複雑なプログラムを作成する	31
繰り返しのプログラム	31
分岐のプログラム	37
複数の分岐を使ったプログラム.....	43
7. その他のブロックの説明	48
ブザーブロック	48
ウェイトブロック	48
タイマーブロック	49
ランダムブロック	49
8. その他便利な機能	50
操作を間違えたときに元に戻す	50
加速度センサの値をグラフで見る	50
プログラムを印刷・画像で保存する	51

9. 課題の解答例	51
課題①	52
課題②	52
課題③	53
課題④	53
課題⑤	54
その他機能の説明	55
10. メニュー・ツールバー	55
メニューの項目説明	55
ショートカットキー	56
11. アクションブロックの概要	57
ブザーブロック	57
ウェイトブロック	57
タイマーブロック	58
音符ブロック	58
LEDブロック	59
演算ブロック	59
分岐ブロック	60
ランダムブロック	61
繰り返しブロック	62
12. 上級者向け機能	63
上級者向け機能の使用設定	63
メモリマップの表示	64
加速度センサの調整	65
13. レポート作成・データの記録	66
レポート作成	66
データの記録	66
14. ファームウェアの書き換え	68
ファームウェアのダウンロード	68
本体への書き込み	68
15. FAQ	70

使い始めるための準備

「使い始めるための準備」は、本体を使うための基本的な操作や説明を記述しています。最初に本体を使うときには、必ずこの内容を読んで、本体を使う準備をしましょう。

1. 学習の前に

本体を使って学習を始める前に、本説明書に登場する単語や必要な道具などについて説明します。

基本的な用語の確認

本説明書では、PCの初歩的な操作方法や用語についての解説を省きます。説明書を読む前に、次の用語や操作方法がどのようなものか、あらかじめ調べておきましょう。

本説明書で登場する、PCの基本的な操作方法や用語

1. マウスに関する用語・操作方法

- (ア) クリック、ダブルクリック、右クリック
- (イ) ドラッグ
- (ウ) マウスカーソル

2. ファイルやフォルダに関する用語・操作方法

- (ア) フォルダ
- (イ) マイコンピュータ (Windows7の場合は「コンピューター」)
- (ウ) デスクトップ
- (エ) フォルダを開く
- (オ) ファイルのコピー、フォルダのコピー

必要なものの準備

本体でプログラムを作成する場合、次の物を準備してください。

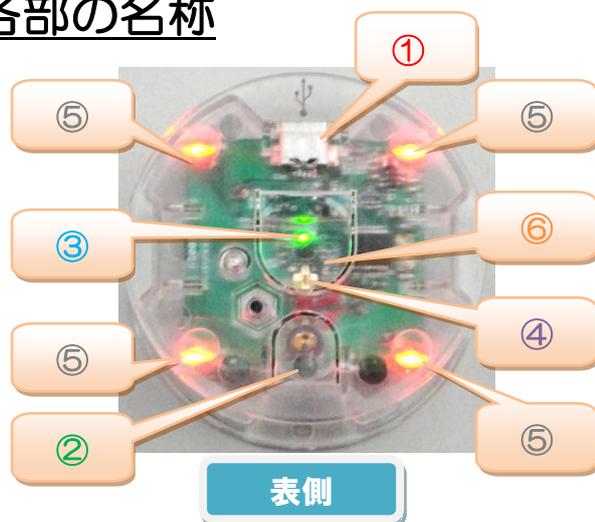
準備するもの

- | | |
|--------------------------------|-----------------------------|
| 1. 加速度センサプログラマー本体 | 1 個 |
| 2. USB ケーブル (mini-B、本体付属) | 1 本 |
| 3. 単四アルカリ乾電池 (またはニッケル水素充電池) | 1 本 |
| ※上記以外の電池は使わないでください | |
| 4. プラスドライバー#2 (電池交換用) | 1 本 |
| 5. PC (下の条件を満たすもの) | 1 台 |
| ◆OS が Windows 2000/XP/Vista/7 | ◆メモリが 128MB 以上 |
| ◆CPU が Pentium-Ⅲ以上 (1GHz 以上推奨) | ◆画面のサイズが XGA (1024x768) 以上 |
| ◆USB2.0 の接続ポートが一つ以上付いている | ◆DirectX9.0b 以降がインストールされている |

2. 本体の各部説明

本体に備わったボタンやセンサに関する説明、電池の入れ方などを説明します。

各部の名称



- ① **USB コネクタ**：付属の USB ケーブルで、PC と本体を接続します。
- ② **電源ボタン**：本体の電源の ON/OFF を行います。
- ③ **電源ランプ**（緑色）：本体の電源が ON の時に点灯します。
- ④ **操作ボタン**：本体単体で使うときに、押してプログラムを開始します。また、プログラム中でも汎用のボタンとして使えます。
- ⑤ **LED**（左上、左下、右上、右下）：プログラムから点灯・点滅を操作できます。
- ⑥ **加速度センサ**：本体の傾きや振動、本体にかかる重力を感知します。
- ⑦ **電池ホルダー**：単四乾電池を取り付ける箇所です。
- ⑧ **ブザー**：プログラムから音を鳴らせます。
- ⑨ **ねじ穴**：カバーを止めるねじの穴です。この部分のねじを外して電池を交換します。

電池の交換方法

- ① プラスドライバーを使って、ねじ穴のねじを取り出してください。



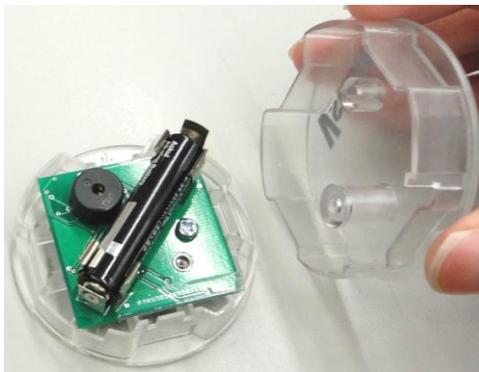
- ② カバーを開けて、基板のイラストの向きに合わせて単四乾電池を電池ホルダーに入れてください。

※電池の向きを間違えないでください



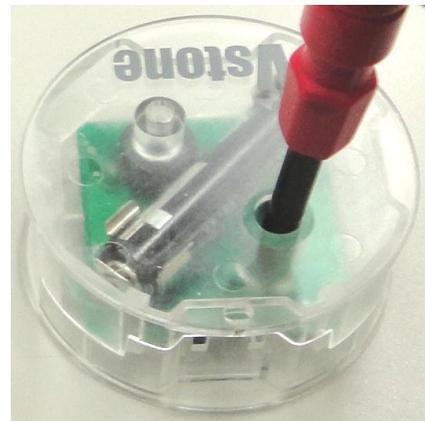
- ③ ②で外したカバーを、向きを確認してはめ直してください。

※隙間が空かないよう取り付けてください



- ④ ねじを締めてください。

※ねじが傾いた状態で無理やり締めないでください



電池は、本体単体で動かすときに使います。PC とつないで使う場合は、電池が無くても本体が動作します。

電源の ON/OFF

本体の電源ボタンで、電源の ON/OFF を行います。

【電源を ON にする】

電池を入れて電源ボタンを押します。
ブザーが短く鳴って、電源ランプが点灯します。



【電源を OFF にする】

電源ボタンを長押し（約 3 秒）します。
ブザーが鳴って LED が全て消えるまで押し続けてください。



電源を ON にしたまま放置すると電池が消耗するため、使わないときは電源を OFF にしてください。

3. ソフトウェアをインストールする

ソフトウェアのダウンロード

ソフトウェア「ビュートビルダーG」は、製品のサポートページからダウンロードする必要があります。下記の URL にアクセスし、プログラムをダウンロードしてください。

◆加速度センサプログラマー サポートページ
http://www.vstone.co.jp/products/vs_gs001/download.html
(トップページ → サポート → 加速度センサプログラマー → ダウンロード)
プログラムは 2 種類あります。次の説明を確認して、好きな方をお選びください
「**インストーラ版**」…ご家庭など個人の PC でお使いの方向けです
「**ZIP 圧縮版**」…学校などソフトのインストールが制限されている PC でお使いの方向けです

ソフトウェアのインストール

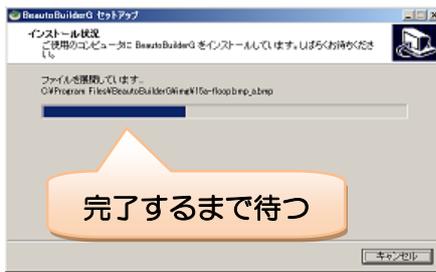
インストーラ版のインストール手順

サポートページから「BeautoBuilderG_Inst_???.exe」というファイルをダウンロードしてください(??? はバージョンを表す番号です)。ダウンロードしたらダブルクリックして実行し、下図の手順でインストールを進めてください。





「インストール」をクリック



完了するまで待つ



「完了」をクリック

ZIP 圧縮版のインストール手順

① サポートページより「BeautoBuilderG_Release?? .zip」というファイルをダウンロードしてください。

② ダウンロードしたファイルを右クリックし、開いたメニューから「すべて展開」をクリックしてください。

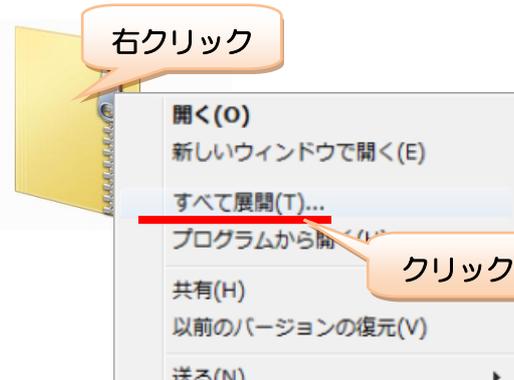


BeautoBuilderG_Release01.zip



OS によって
アイコンが異なる

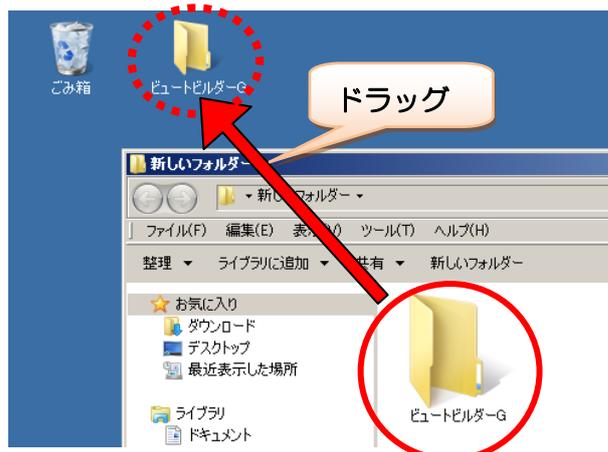
数値は
バージョンを表す



右クリック

クリック

③ すべて展開すると「ビュートビルダー-G」というフォルダができるので、デスクトップなど使いやすい場所にフォルダをドラッグして移動してください。



ドラッグ

ビュートビルダー-G

インストールが完了したら、一度ソフトウェアを起動してみましょう。選択したインストール方法に合わせて、以下の手順で起動してください。

ソフトウェアの起動

インストーラ版の起動方法

デスクトップに右図のショートカットができるので、ダブルクリックしてください。

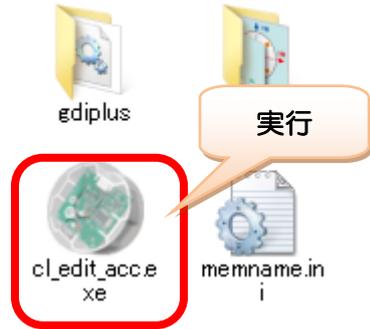


ZIP 圧縮版の起動方法

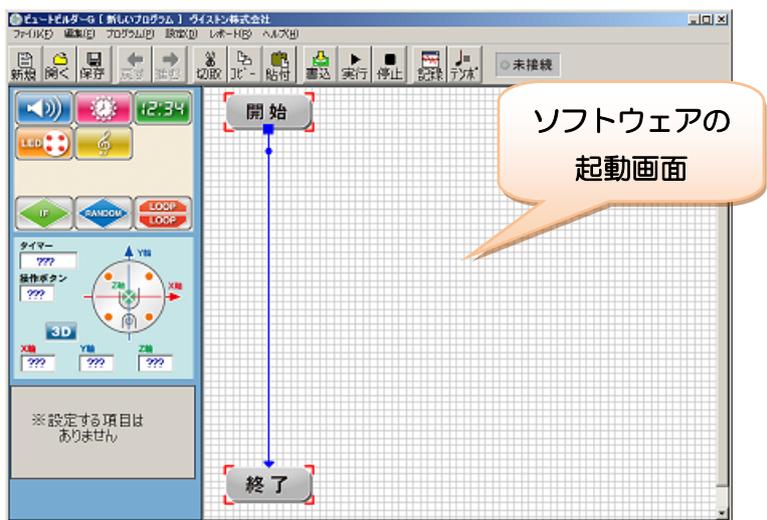
① フォルダ「ビュービルダー-G」を開いてください。



② フォルダの中から「cl_edit_acc.exe」というプログラムをダブルクリックして実行してください。



ソフトウェアが起動すると、下のような画面を表示します。



ソフトウェアが起動しない場合は、対応する DirectX ランタイムが PC にインストールされていない可能性があります。13 ページを参照に最新の DirectX ランタイムをインストールしてください。

4. 本体と通信する

ソフトウェアをインストールしたら、起動して本体と通信できるか確認してみましょう。

本体とPCの接続

本体とPCの接続には、付属のUSBケーブルを使います。次の手順で接続し、電源をONにしてソフトウェアとの通信を確認しましょう。

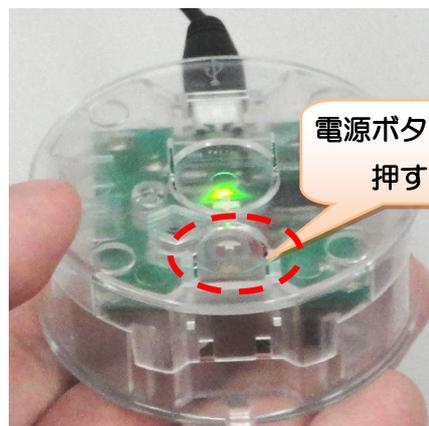
① ケーブルの口が小さい方を本体に、大きい方をPCに接続してください

② 本体の電源ボタンを押して、電源をONにしてください。

口の大きい方を
PCに接続

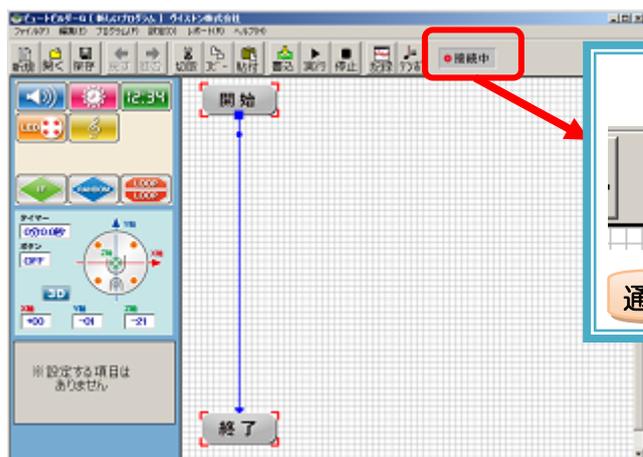


口の小さい方を
本体に接続



電源ボタンを
押す

③ ソフトウェアが自動的に通信を開始します。通信中は、画面左上の表示が「未接続」から「接続中」に変わります。



通信中は表示が切り替わります

未接続

通信していない

接続中

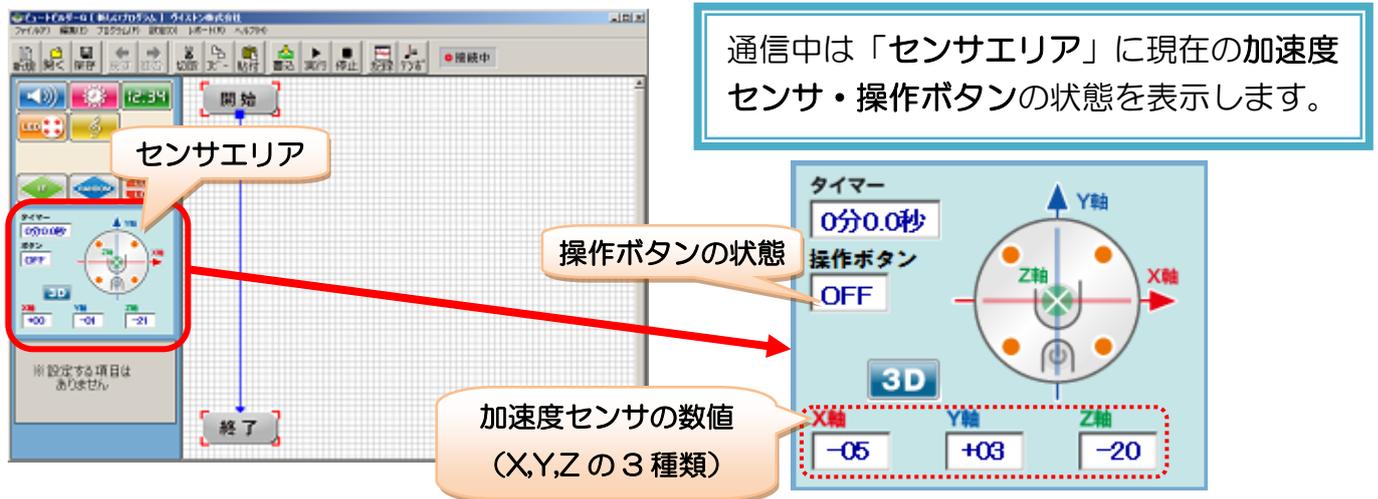
通信している

通信中に以下のようななった場合、一度本体のケーブルを抜き差しして、再接続してください

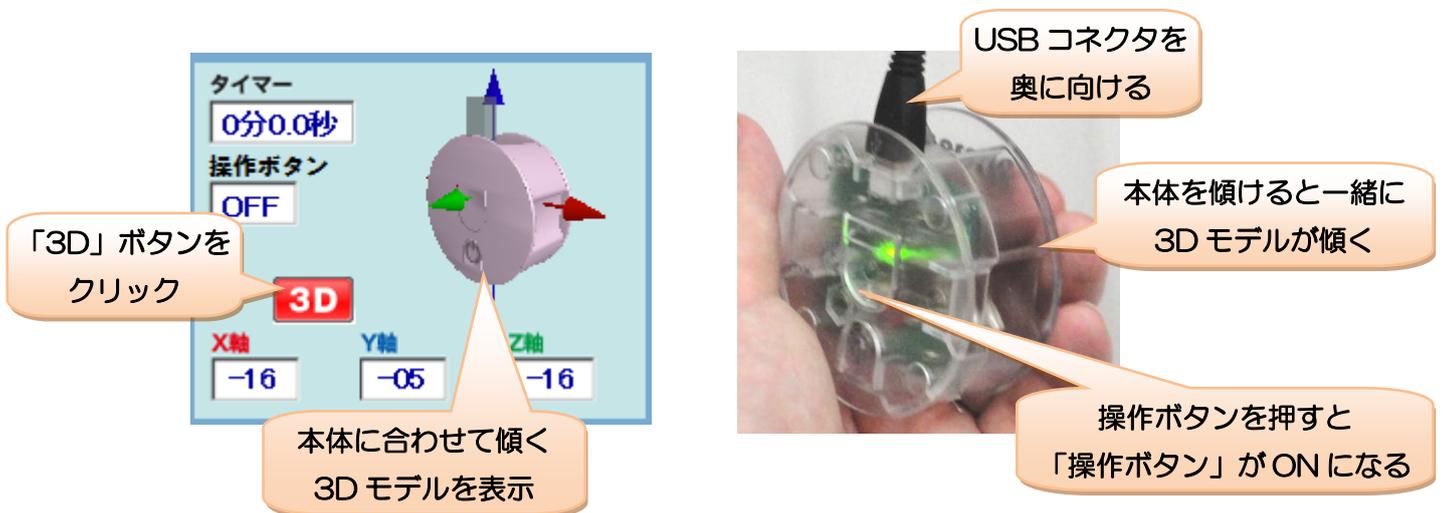
- ※ブザーが鳴りっぱなしになる
- ※センサの数値（後述）が全く変化しない

センサの反応の確認

接続できたら、センサの反応を試してみましょう。画面左側の「センサエリア」には、本体との通信中にセンサの情報を表示します。本体には、「加速度センサ」「操作ボタン」の二種類のセンサを搭載しており、ボタンを押したり本体を傾けたりすると、センサエリアの数値が変化します。



また、「3D」ボタンをクリックすると、現在の加速度センサの値に応じた本体の傾きを 3D モデルで表示します。3D モデルは、本体の USB コネクタが奥側に来る向きを基準に表示します。本体を傾けたり操作ボタンを押したりして、反応を確かめてみましょう。



※本体を周囲にぶついたり、ケーブルを無理に引っ張らないように注意してください。
※DirectX9.0b 以降が PC にインストールされていない場合、「3D」ボタンは利用できません。

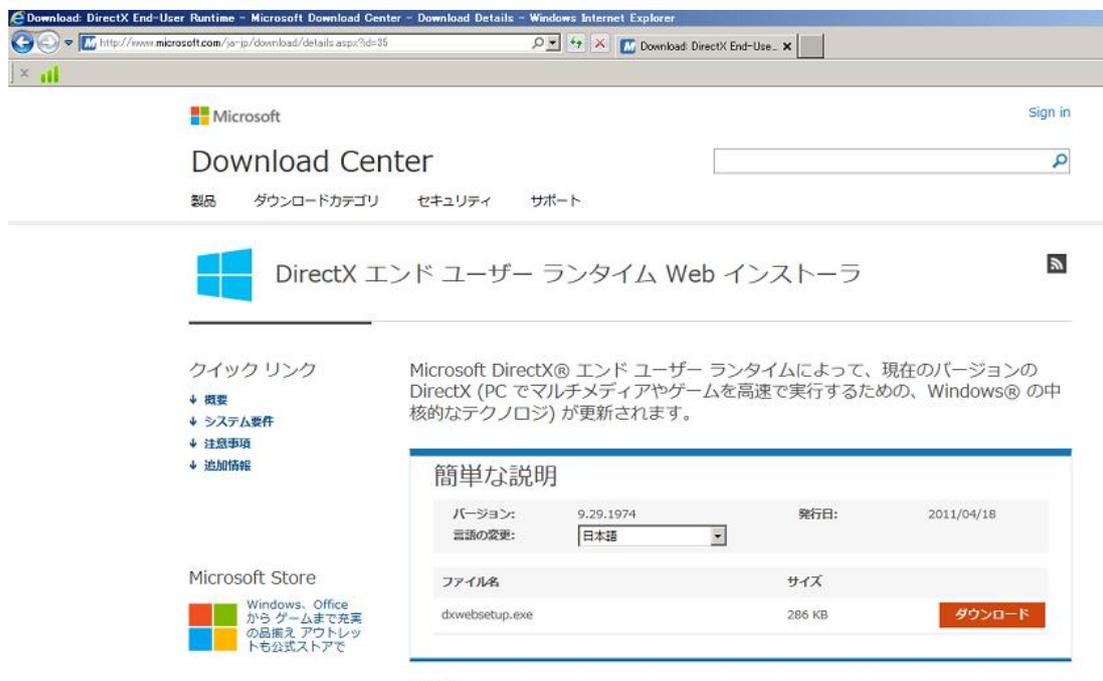
DirectX ランタイムのインストール

センサエリアの「3D」ボタンを使えない場合、次の手順で最新の DirectX ランタイムを PC にインストールしてください。

DirectX ランタイムは、Microsoft 社の web ページよりダウンロードしてインストールします。2013 年 3 月現在、最新の DirectX ランタイムは、以下の web ページで公開されています。

<http://www.microsoft.com/ja-jp/download/details.aspx?id=35>

web ページにアクセスしたら、「言語の変更」で「日本語」を選び、「ダウンロード」をクリックしてください。



The screenshot shows the Microsoft Download Center page for DirectX End-User Runtime. The page title is "DirectX エンド ユーザー ランタイム Web インストーラ". The main content area includes a "簡単な説明" (Simple explanation) section with the following details:

バージョン:	9.29.1974	発行日:	2011/04/18
言語の変更:	日本語		
ファイル名	サイズ		
dxwebsetup.exe	286 KB		ダウンロード

クリックすると「dxwebsetup.exe」というファイルをダウンロードするので、そちらを PC にインストールしてください。

※ DirectX ランタイムは、必ず最新のものをインストールしてください。古いバージョンをインストールすると、正しく動作しない場合があります。

プログラミングの学習

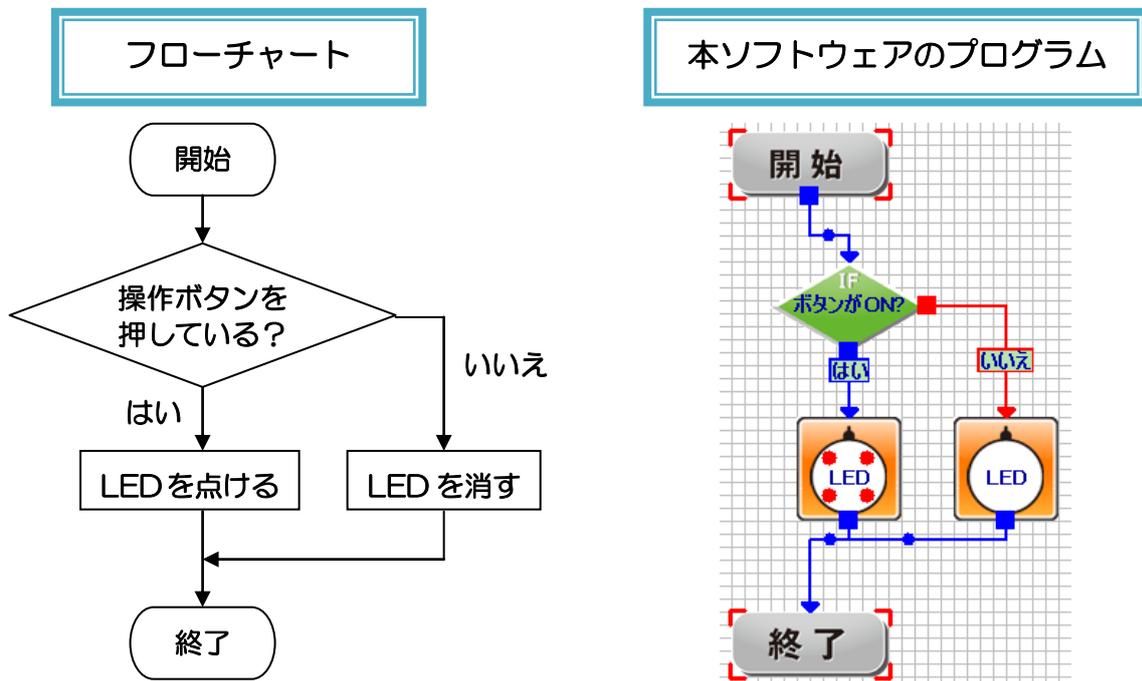
「プログラミングの学習」は、例題を参考に、実際にプログラムを作りながらソフトウェアの使い方やプログラミングを学習します。使い始めるための準備を済ませて本体を使う準備ができれば、続いてプログラミングの学習を進めましょう。

5. ソフトウェアの使い方を確認する

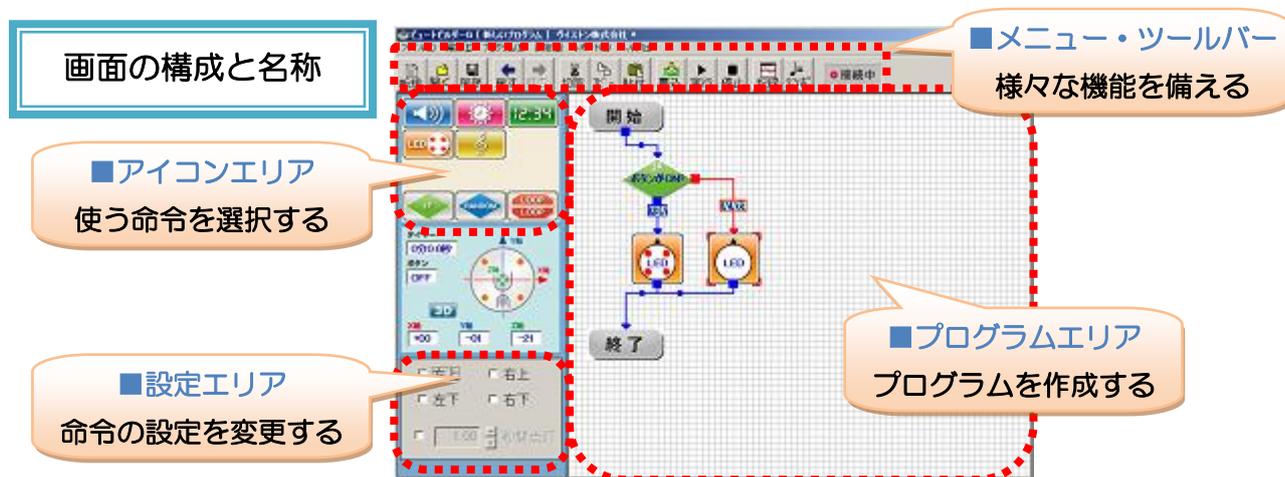
ソフトウェアをPCにインストールして、本体と通信できることを確認したら、今度はプログラムの作成に進みましょう。最初に、簡単な例題を参考にソフトウェアの操作方法を説明していくので、使い方を覚えましょう。

ソフトウェアの画面の説明

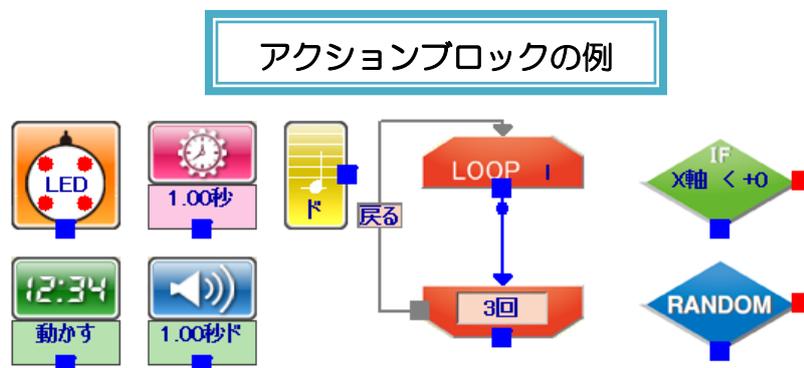
本ソフトウェアのプログラムは、「フローチャート」に似せて作られています。フローチャートはプログラムの設計図のようなもので、命令や手順を四角や丸の図形で描き、実行する順番に合わせて矢印でつないでいきます。



本ソフトウェアでは、画面右側の「プログラムエリア」でフローチャートを作りプログラミングします。また、画面左側では、「アイコンエリア」フローチャートで使用する命令などの記号を選んだり、「設定エリア」で命令の設定を変更したりします。また、上部には「メニュー」と「ツールバー」があり、プログラムの実行や保存などの操作に使います。



プログラムで使用する命令の記号を、本ソフトウェアでは「アクションブロック」または「ブロック」と呼びます。主なアクションブロックとしては下記の物が挙げられます。各ブロックの概要は、順次説明していきます。

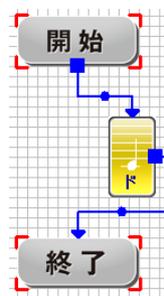


プログラムの作成手順

それでは、簡単なプログラムを作成しながら、プログラムの作成手順の基本を覚えましょう。

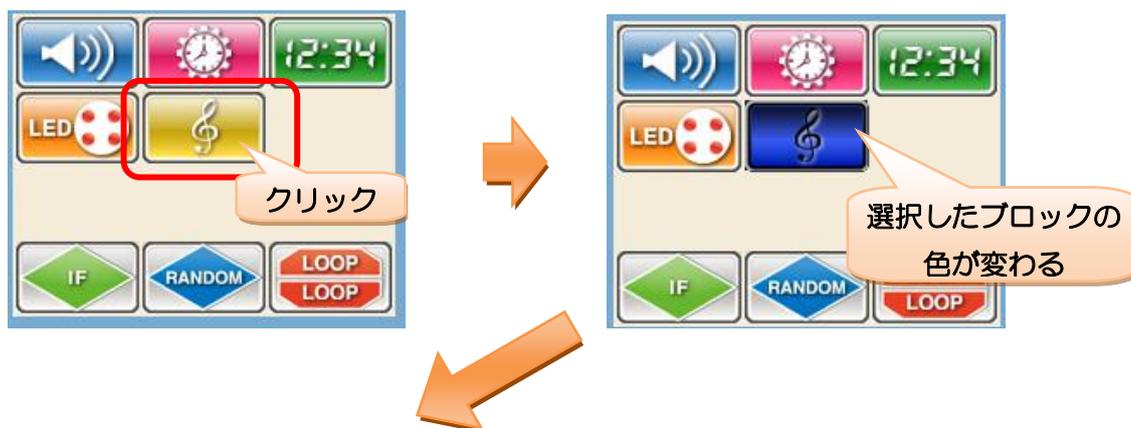
【例題 1】

ブザーから「ド」の音を鳴らす
プログラムを作りましょう。

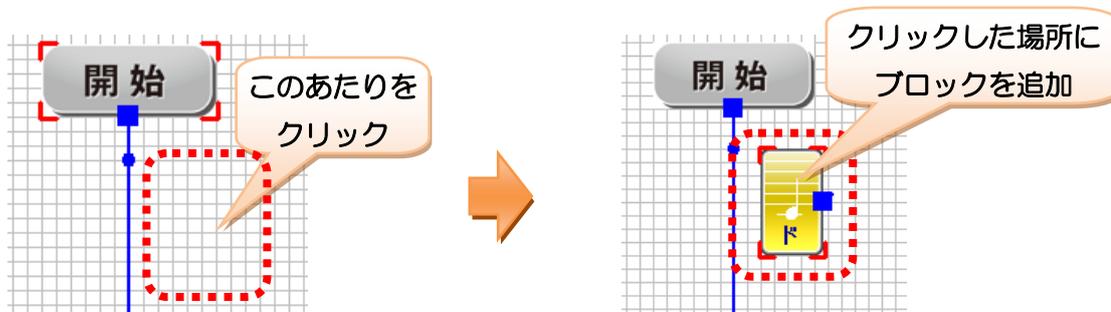


アクションブロックを選ぶ

- ① アイコンエリアから音符の命令(下図のボタン)をクリックしてください。
クリックすると、選んだブロックのボタンの色が変わります



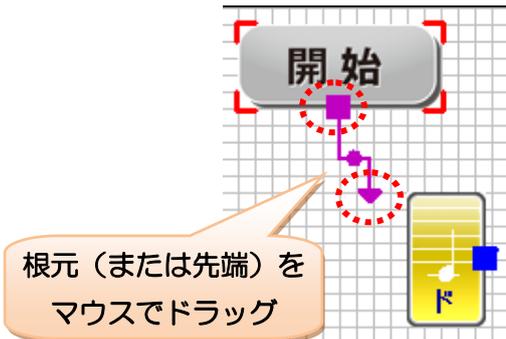
- ② プログラムエリアの「開始」の下辺りをクリックしてください。
クリックすると、その場所に選んだブロックが作成されます



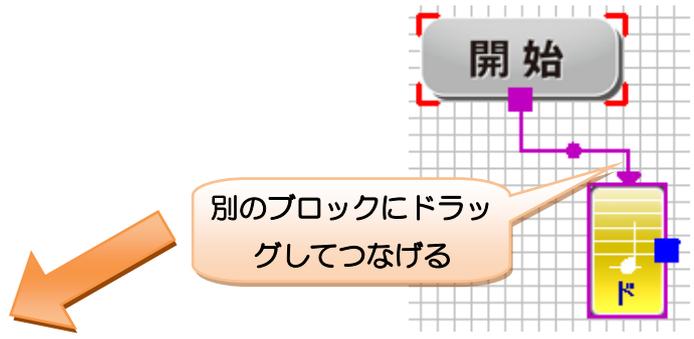
ブロックを接続する

プログラムを実行する順番でブロックを接続します。プログラムは「開始」のブロックから始まり、矢印でつながったブロックを順番に実行していきます。

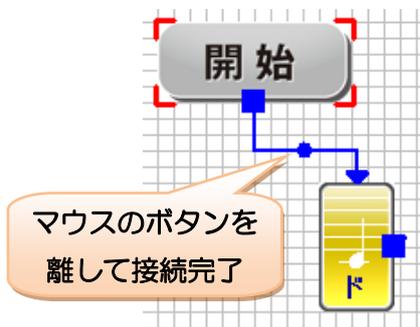
- ① 矢印の根元（または先端）をドラッグすると、矢印を動かします。



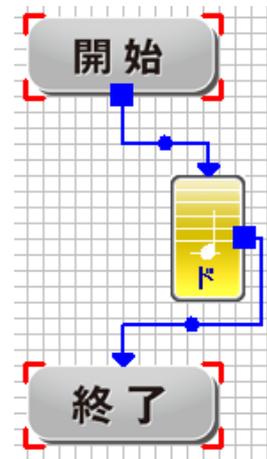
- ② ドラッグしたままマウスカursorを別のブロックに重ねると、自動的に矢印がつながります



- ③ マウスのボタンを離すと、接続が完了します。



- ④ 「開始」と「ド」、「ド」と「終了」のブロックを矢印でつなぎましょう。



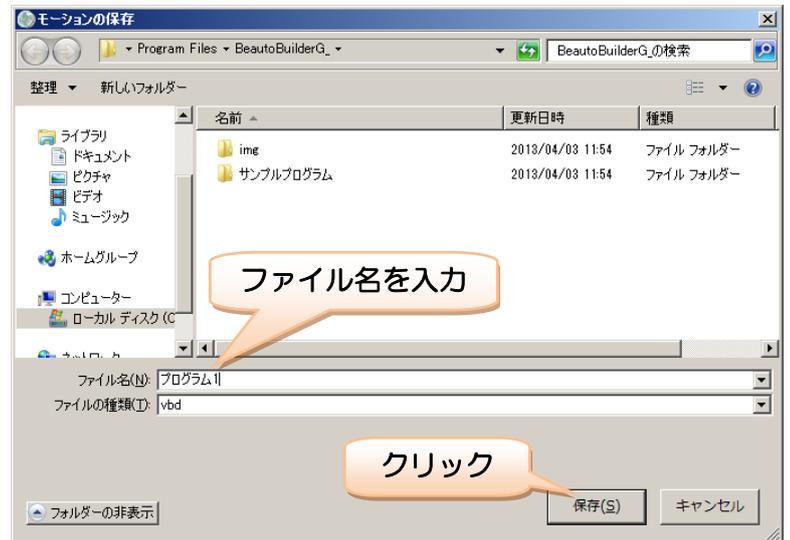
プログラムを保存する

作成したプログラムは、ファイルに保存することができます。

- ① ツールバーの「保存」ボタンをクリックします。



- ② ファイル名を入力して「保存」をクリックします。



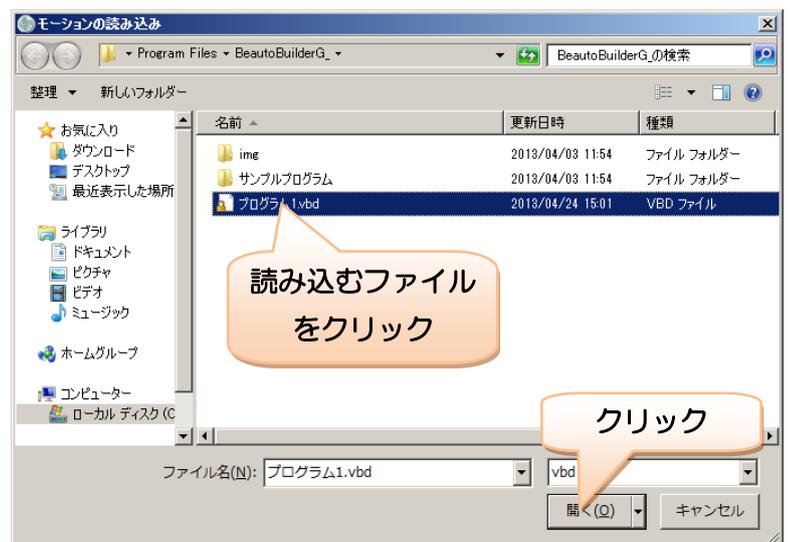
プログラムを読み込む

ファイルに保存したプログラムは、次の手順で読み込みます。

- ① ツールバーの「開く」ボタンをクリックします。



- ② 読み込むファイルをクリックして「開く」をクリックします。



「サンプルプログラム」のフォルダには、様々なサンプルプログラムを収録しています。学習の参考にしましょう。

プログラムを実行する

ここで、作成したプログラムを実行してみましょう。

プログラムの実行方法には、次の二通りがあります。

テスト実行

PC と通信しながら実行して、画面に現在実行しているブロックを表示します。フローチャートの流れを確認しながら実行できるので、プログラムの間違いを探しやすくなります。

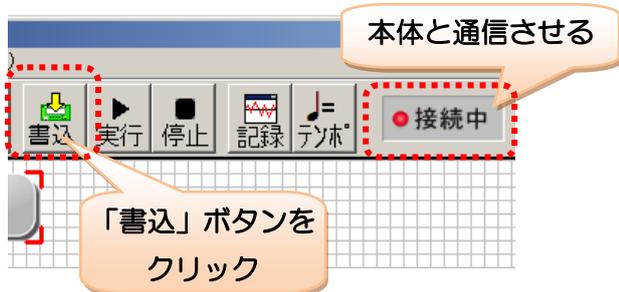
単体で実行

本体にプログラムを記憶して、本体だけで実行します。

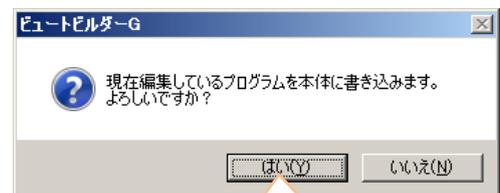
※どちらの方法でも、プログラムを本体に記憶させます。また、本体は一度に 1 種類のプログラムしか記憶できません

単体で実行

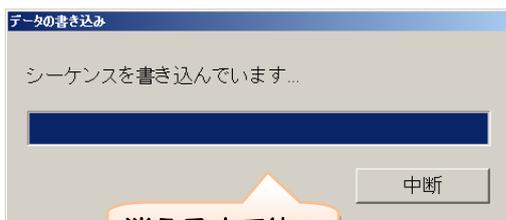
① PC と本体を通信させ、「書込」ボタンをクリックしてください



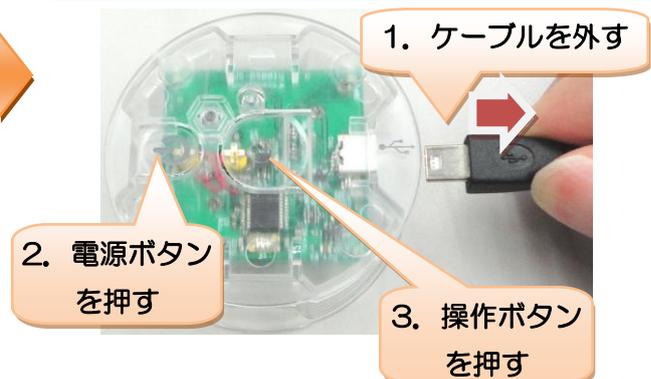
② 確認メッセージが表示されるので、「はい」をクリックしてください



③ 書き込み中の画面が表示されるので、自動的に消えるまで待ってください

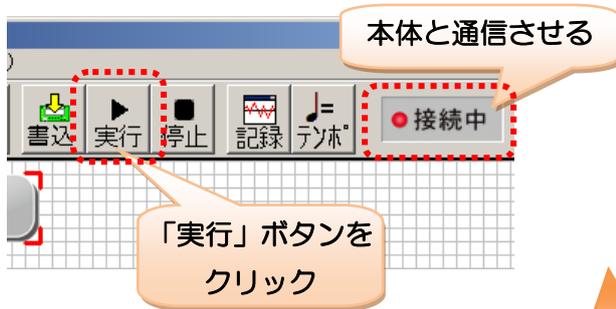


④ 本体のケーブルを外して、電源ボタン → 操作ボタンの順番で押してください

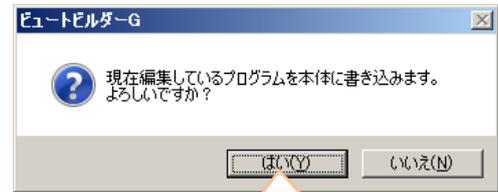


テスト実行

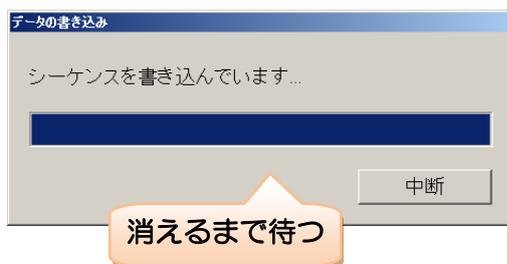
① PC と本体を通信させ、「実行」ボタンをクリックしてください



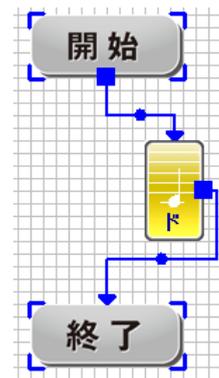
② 確認メッセージが表示されるので、「はい」をクリックしてください



③ 書き込み中の画面が表示されるので、自動的に消えるまで待ってください



④ 自動的にプログラムを開始します。実行中のブロックは青の枠線で囲って表示します。



プログラムの書き込み・実行が正しく行われない場合

テスト実行で、命令は実行しているのにブロックが青い枠で囲まれない

テスト実行中は、本体との通信処理に時間がかかるため、プログラム冒頭で青い枠の表示が飛ばされたり、分岐ブロックなど実行時間が短い命令に枠が表示されない場合があります。

書き込みに時間がかかり（10秒以上）、書き込みに失敗してしまう

プログラムの書き込み中に通信が不安定になったため、書き込みに失敗しています。書き込み時間がかなり長い場合、「中断」をクリックして書き込みを中断し、本体からケーブルを抜き差しして接続し直してから、プログラムを書き直してください。

書き込み中にケーブルを抜いたら本体と通信できなくなった

PC が本体を認識できない状態になっています。ケーブルを外してから電池を抜き差しし、一度本体を再起動させてください。

複数のブロックを使ったプログラム

今度は、簡単なメロディを鳴らすプログラムを参考に、「複数のブロックを使う」「ブロックの設定を変更する」について覚えましょう。

プログラムの作成手順

① プログラムに命令を追加する

アイコンエリア（画面左上）から命令を選んでクリックし、プログラムエリア（画面右）をクリックしてブロックを追加する

② ブロックに矢印を接続して実行順序を決める

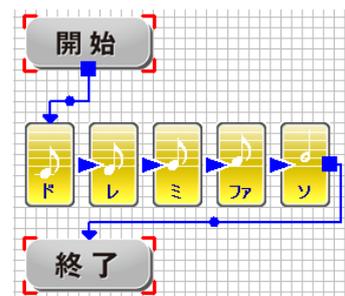
ブロックの矢印の根元（または先端）をクリックし、つなぎたいブロックまでドラッグする。

③ 作成したプログラムを書き込んで実行する

PC と本体を接続し、「書込」または「実行」ボタンをクリックして書き込む。

【例題 2】

ブザーから「ドレミファソ」のメロディを鳴らすプログラムを作りましょう。
ただし、「ドレミファ」は 8 分音符、「ソ」は 2 分音符の長さにしましょう。



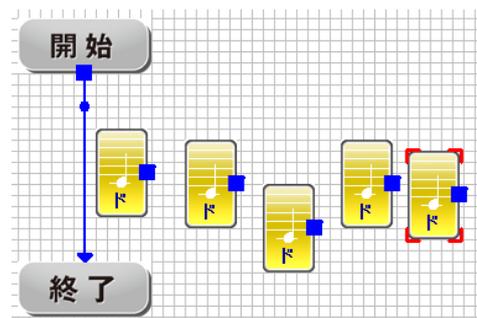
複数のブロックを並べる

まず、必要なアクションブロックをプログラムエリアに追加します、先ほどと同じように「アイコンエリアで音符の命令をクリック」→「プログラムエリアをクリック」の操作で、プログラムエリアに音符のブロックを合計 5 個作りましょう。

新しくプログラムを作り始める場合は、「新規」のボタンをクリックしましょう。

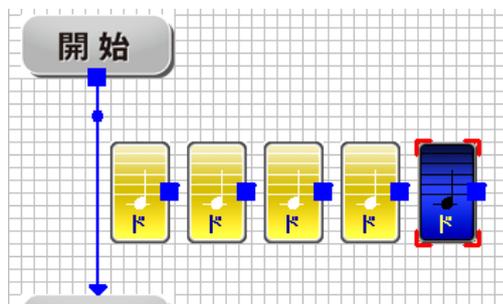


プログラムエリアに音符のブロックを 5 個作りましょう。

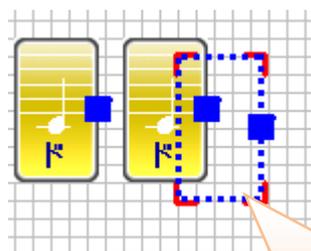


ブロックを追加したら、プログラムを見やすくするため、ブロックをマウスでドラッグしてきれいに並べましょう。

ブロックは、ドラッグして位置を動かします。
音符のブロックを横一列に並べましょう。

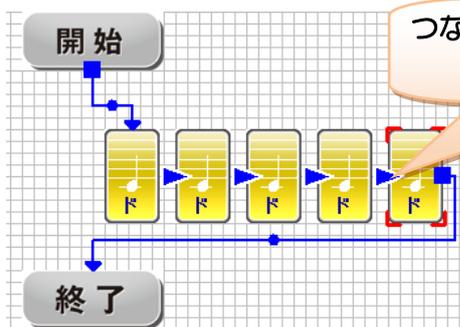


ブロック同士を重ねて置くことは
できません



ブロックを重ねようと
すると、青い点線で表示

ブロックを整列したら、下図のように順番に矢印を接続しましょう。ここまでの操作は、先ほどの例題と同じです。

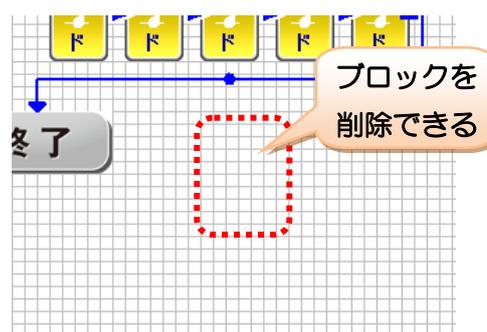
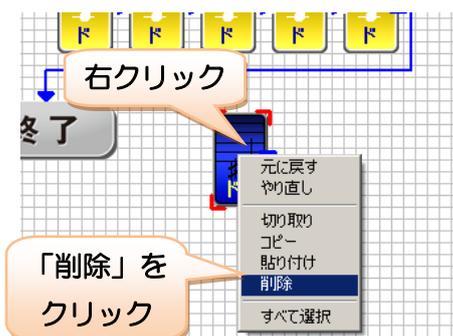


アクションブロックは常に整列して見やすいプログラムを作りましょう。プログラムが見つづらいと、正しく動かない場合に原因を調べにくくなります。

ブロックの削除と複数のブロックの選択

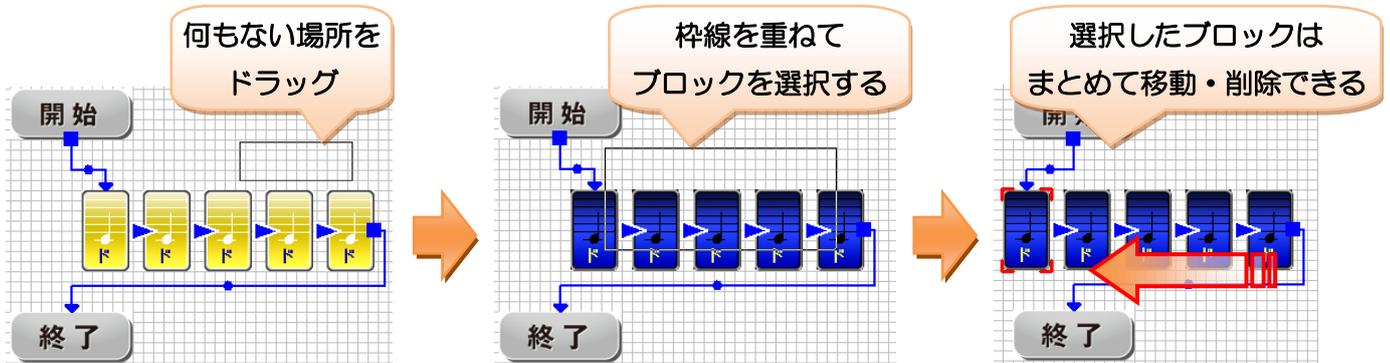
もし不要なアクションブロックが出てきたら削除しましょう。

ブロックを削除する場合は、右クリックしてメニューを開き、「削除」を選択します。
もしくは、ブロックをクリックして、キーボードの Delete (Del) を押しても削除できます



ブロックの削除や移動は、「選択」の機能を使って一度に行うことができます。

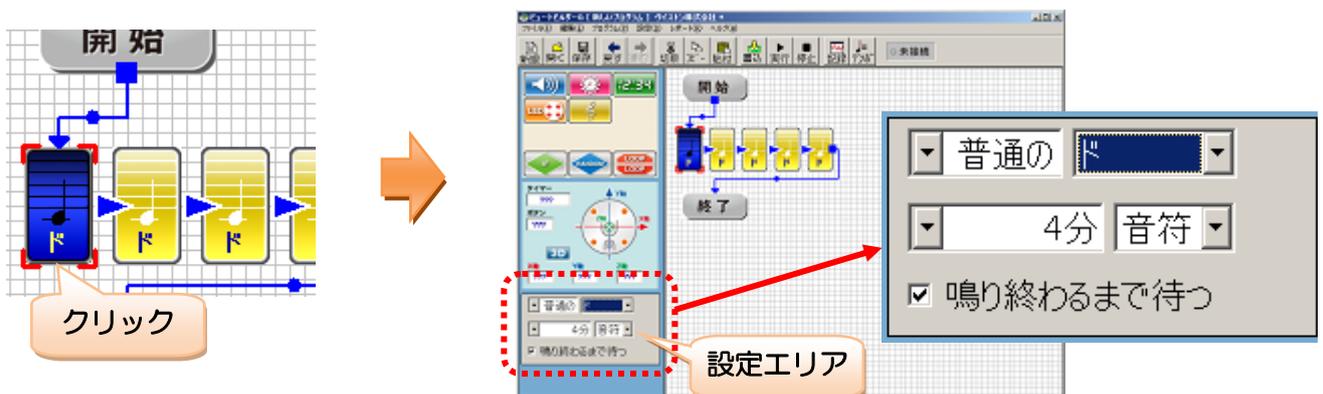
プログラムエリアの何も無いところをドラッグすると枠線を表示します。
枠線とブロックを重ねるとブロックの色が変わり「選択状態」になります。
選択状態のブロックは、まとめて移動したり削除したりできます。



ブロックの設定を変更する

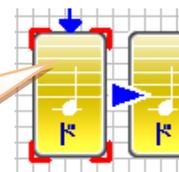
最後に、ブロックの設定を変更しましょう。設定の変更は、ブロックをクリックして画面左の「設定エリア」から行います。

ブロックをクリックすると、「設定エリア」にクリックしたブロックの現在の設定を表示します。設定エリアの項目を変更すると、ブロックの設定が変わります。

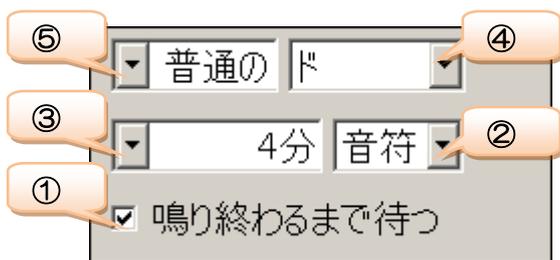


設定エリアで変更できるブロックは、常にプログラムエリアで赤い枠に囲まれて表示されます。

四隅に赤い枠を表示



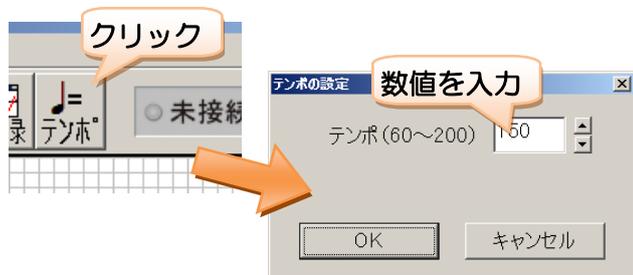
設定エリアの内容は、ブロックの種類に応じて変わります。



【音符のブロックの設定】

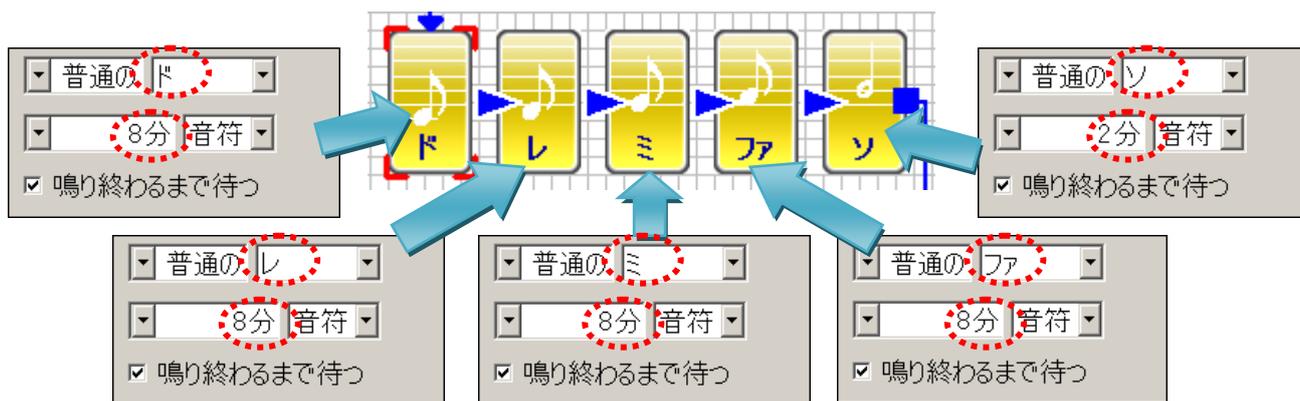
- ① 音の高さ：低い/普通/高い の3種類
- ② 音程：ド～シ の12種類
- ③ 音長：32分音符～全音符 の11種類
- ④ 種類：音符/休符 の2種類
- ⑤ 「ブザーが鳴り終わるまで待つか」の選択

「鳴り終わるまで待つ」の設定は、ここでは必ずチェックを入れてください。この設定の説明は、次のLEDを使ったプログラムで説明します。



音符ブロックで作成した音楽は、「テンポ」の設定で速さを変えることができます。テンポは、画面上の「テンポ」ボタンをクリックし、別画面を開いて設定できます。

プログラムエリアのブロックを、それぞれ下図の通りに設定変更しましょう。



設定を変更したらプログラムを実行してみましょう。

◆正しく動作しない場合

音の長さがおかしい…「音長」の設定が間違っていないか確認しましょう

音程がおかしい…「音程」の設定が間違っていないか確認しましょう

音が鳴らない…矢印をつなぎ忘れていないか確認しましょう。

※設定を変更するときは、必ず最初に変更するブロックをクリックして、別のブロックを間違えて変更しないようにしましょう

ここまで作成したプログラムは、全て「開始」から「終了」まで、矢印の通りに順番で実行します。プログラムは、途中で戻ったり、実行することに順番が変わったりしません。このような一本道のプログラムの構造を「順次」と言います。

【順次のプログラム】
 矢印の通りに決まった順番で実行。
 途中で戻ったり分岐したりしない。

非常に単純な仕組みなのでわかりやすいが、
 センサに反応するなど、複雑なプログラムは
 作ることができない

```

            graph TD
            A([開始]) --> B[処理 1]
            B --> C[処理 2]
            C --> D[処理 3]
            D --> E([終了])
            
```

順次プログラムの練習課題

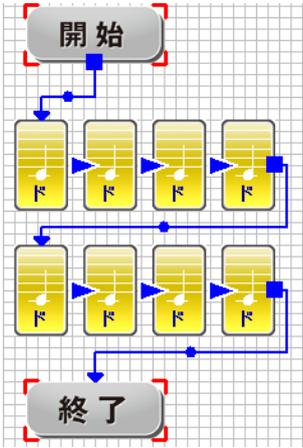
【課題①】
 「ドレミファソラシド」の音を、次の条件で鳴らすプログラムを作ってみましょう。

- ① 一つの音は 8 分音符の長さにする
- ② 最後の「ド」の音は 1 オクターブ高くする
- ③ テンポを 170 にする

【ヒント①】
 音符のブロックを 8 個作り、
 矢印をつなげます。

【ヒント②】
 オクターブは「音の高さ」
 で変更できます。

【ヒント③】
 テンポの設定は「テンポ」
 ボタンをクリックします。



音の高さ

▼ 普通の ド ▼

▼ 4分 音符 ▼

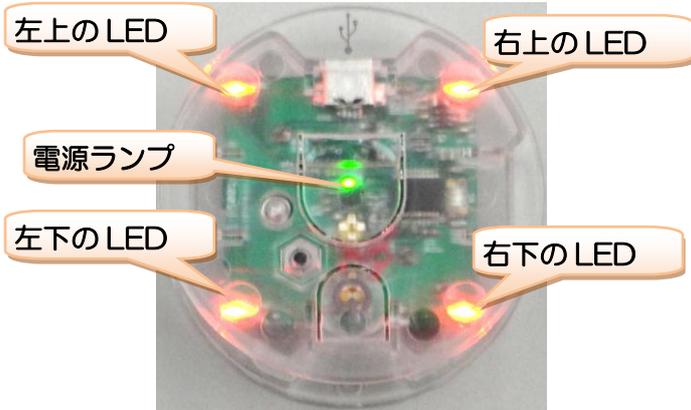
鳴り終わるまで待つ

テンポボタン

止 記録 テンポ 未接続

LEDを使ったプログラム

本体には、プログラムから点滅させられるLEDが4個ついています。向きと名前を覚えましょう。

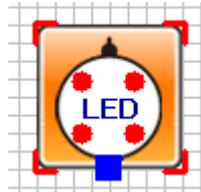


赤いLEDはプログラムから点滅させられます。
緑のLEDは「電源ランプ」で、電源がONの時に点灯します（プログラムからは点滅できません）

LEDの点滅は、「LED」のブロックを使います。



LEDのブロック

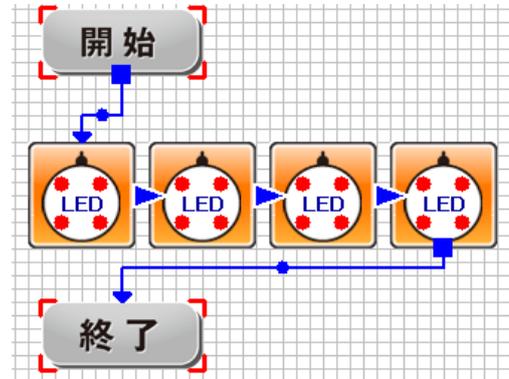
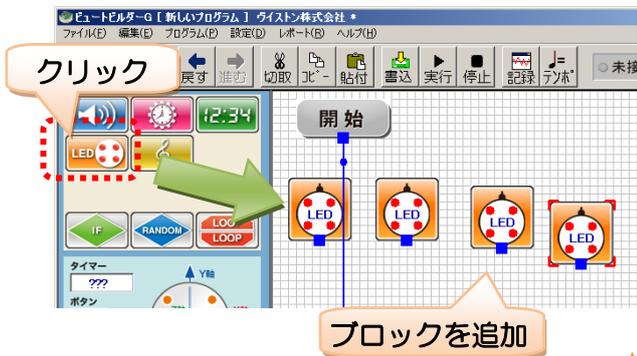


それでは、続いてプログラムからLEDを点滅させてみましょう。

【例題3】
LEDを時計回りで順番に光らせるプログラムを作しましょう

① プログラムエリアにLEDのブロックを4個追加しましょう。

② 矢印を順番に接続し、ブロックを整列しましょう。

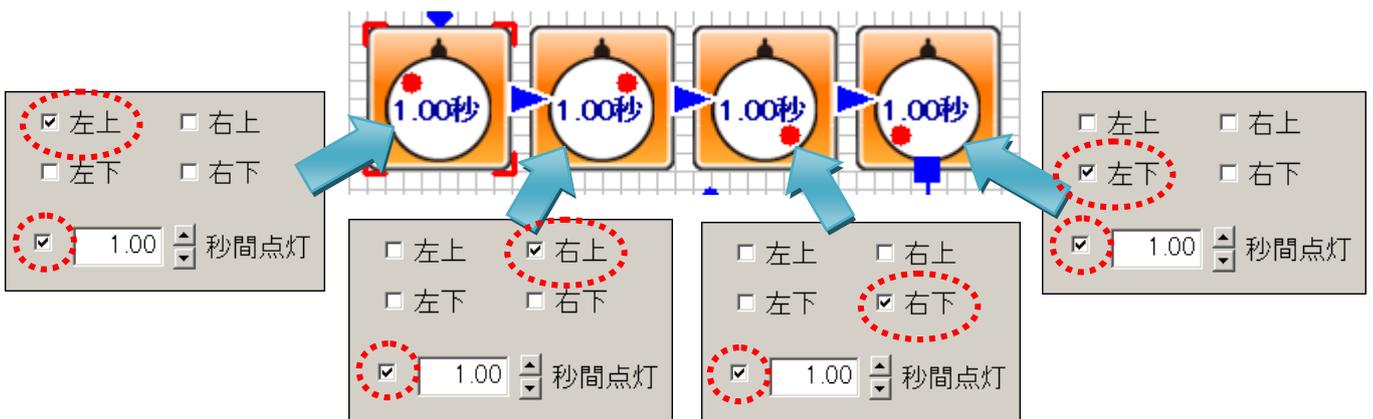


③ 設定エリアでブロックの設定を変更しましょう。
ここでは、「時間の設定」にチェックを入れてください。

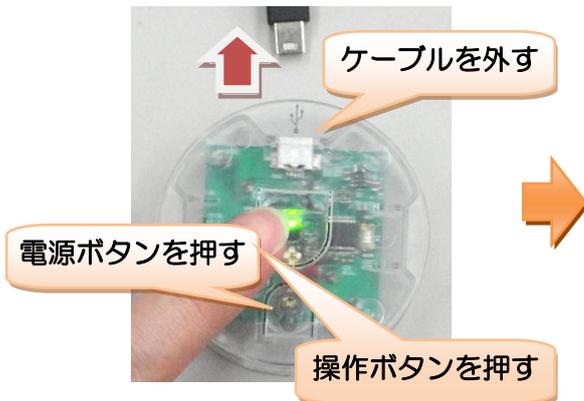


【LEDのブロックの設定】

① LEDの点滅：点ける・消すをチェックで選択
② 時間の設定：チェックすると、指定の時間だけLEDを点けます。時間が過ぎたらLEDを消して次の命令に進みます



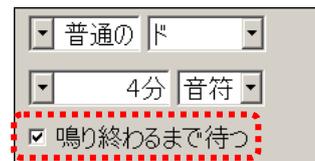
- ④ 設定を変更したら、プログラムを書き込んで実行してみましょう。



時間に関する設定の説明

音符・LEDのブロックには、それぞれ「鳴り終わるまで待つ」「??秒間点灯」という設定がありました。これらはいずれも「指定の時間だけ待ってから次の命令に進む」という設定です。例題を参考に、この設定を変えるとプログラムがどのように変わるか確認してみましょう。

「鳴り終わるまで待つ」「??秒間点灯」の設定の特徴
 チェックする…指定の時間だけ待ってから次の命令に進む
 ⇒ブザーやLEDが消えてから次の命令に進む



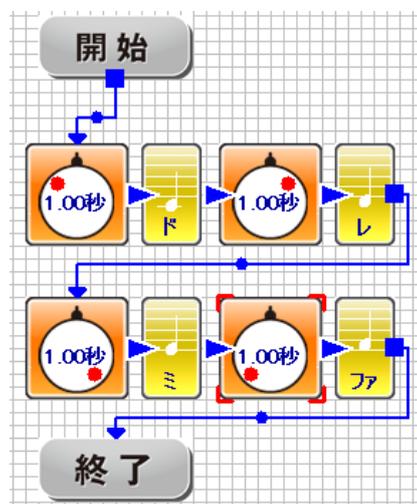
チェックしない…音を鳴らしたら、すぐ次の命令に進む(ブザー)
 点滅を変更したら、すぐ次の命令に進む(LED)
 ⇒ブザーやLEDを点けた・鳴らしたまま、別の命令ができる



【例題 4】

LEDとブザーを交互に実行するプログラムを作りましょう。

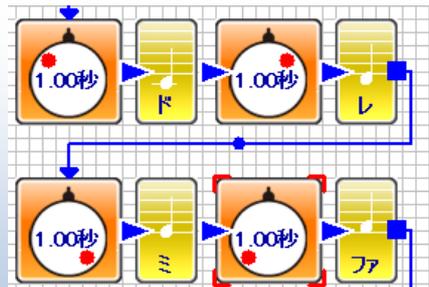
- ① 交互に4回ずつ実行する
- ② LEDは時計回りの順
- ③ ブザーはドレミファの順



プログラムを作成したら、次の4通りに設定を変更して、動作を確認しましょう。

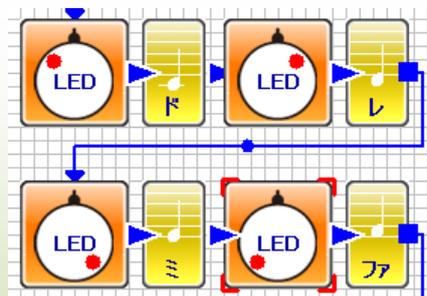
① 全ての「鳴り終わるまで待つ」
「??秒間点灯」をチェックする

- ◆LEDが点滅→ブザーが鳴るの順番で実行される
- ◆ブザーが鳴っている間はLEDが光らない



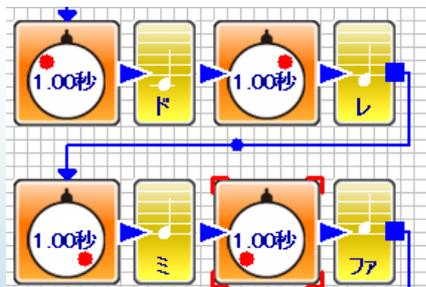
② 全てのブザーの「鳴り終わるまで待つ」をチェックする
全てのLEDの「??秒間点灯」をチェックしない

- ◆ブザーが途切れず連続で鳴る
- ◆ブザーの音程と一緒にLEDの点滅が変わる



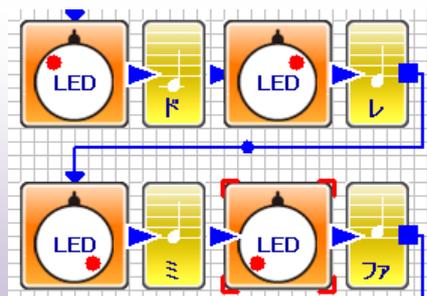
③ 全てのブザーの「鳴り終わるまで待つ」をチェックしない
全てのLEDの「??秒間点灯」をチェックする

- ◆LEDが一旦消えず、連続で時計回りに動く
- ◆LEDの点滅と一緒にブザーの音程が変わる



④ 全ての「鳴り終わるまで待つ」「??秒間点灯」
をチェックしない

- ◆プログラムが瞬時に終了する

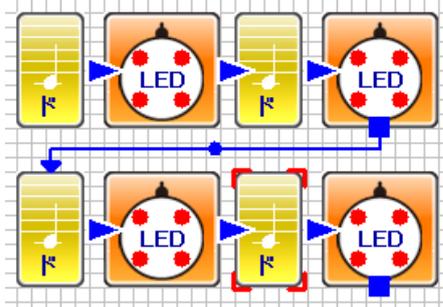


「鳴り終わるまで待つ」「??秒間点灯」を全てチェックしない場合、どのブロックでも待ち時間が無いため、それぞれのブロックが瞬時に実行され、すぐにプログラムが終了します。これらの設定は使いどころがやや難しいですが、後で説明する「分岐」のところでも重要になります。

LED とブザーを使ったプログラムの練習課題

【課題②】
 LED を次の条件で反時計回りに点滅させるプログラムを作ってみましょう。
 ① 左上の LED から始める
 ② LED は 1.5 秒ごとに切り替わる
 ③ LED が切り替わるごとに、ブザーから「ソラシド」の順番で音を鳴らす

【ヒント①】
 音符→LED の順番で交互にブロックをつなげます



【ヒント②】
 「鳴り終わるまで待つ」はチェックせず、「??秒間点灯」はチェックします

普通 鳴り終わるまで待つ

チェックしない

左上 右上
 左下 右下
 1.00 秒間点灯

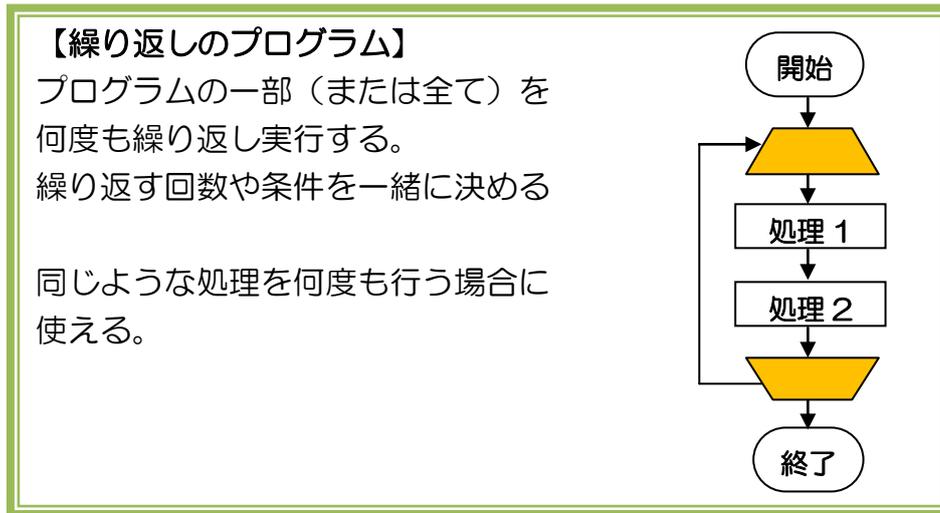
チェックする

6. 複雑なプログラムを作成する

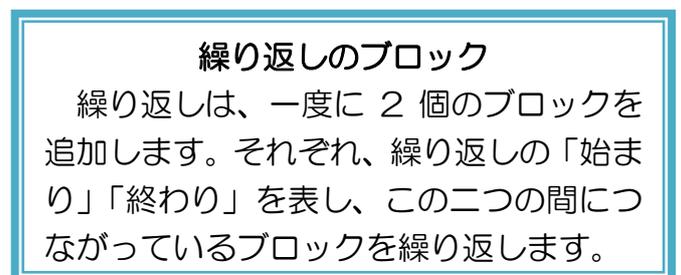
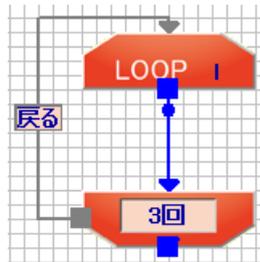
これまで説明した「順次」のプログラムは仕組みが単純なので簡単に理解できたと思います。今度は、「繰り返し」「分岐」の二つのプログラム構造を学習し、もう少し複雑なプログラムを作ってみましょう。

繰り返しのプログラム

繰り返しのプログラムは、名前の通りプログラムの一部（または全て）を繰り返し実行する構造です。繰り返しには、繰り返す回数や条件が必ず存在します。



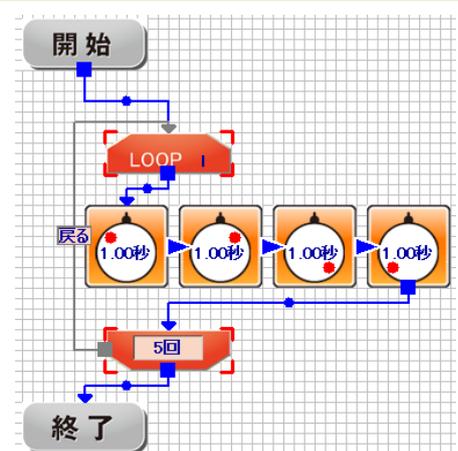
繰り返しを行うには、以下の「繰り返し」のブロックを使用します。



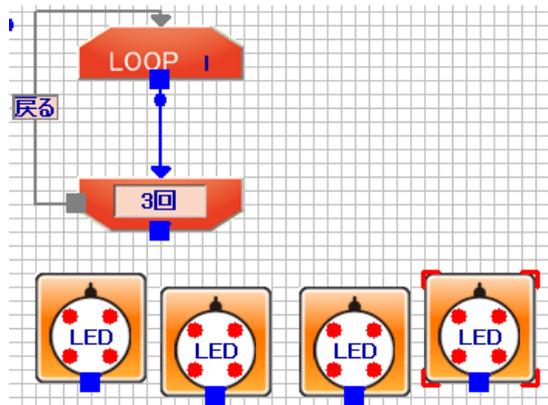
【例題 5】

LED を時計回りに 5 周点滅するプログラムを作りましょう。

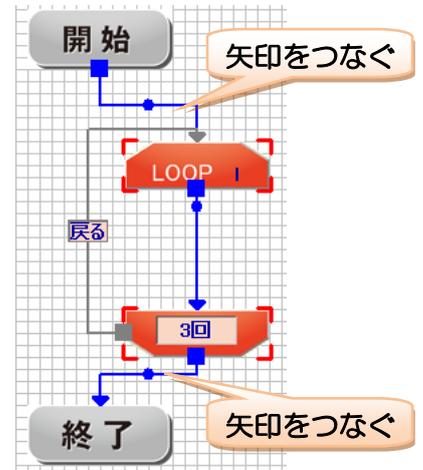
5 周程度なら、繰り返しを使わなくても、LED ブロックを $4 \times 5 = 20$ 個並べればプログラムを作れます。しかし、これが 100 回など非常にたくさんの場合になると、繰り返しを使わなければプログラムがとても大変になります。



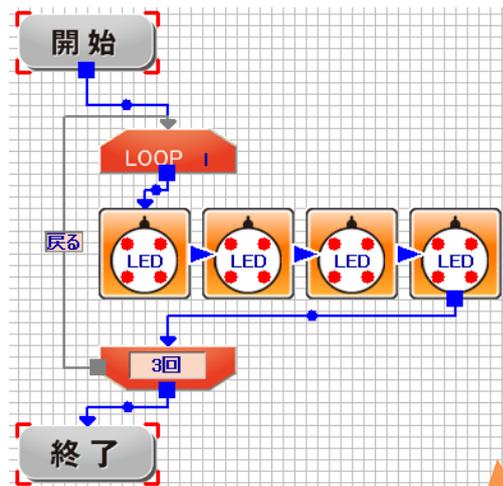
① 繰り返し1セット、LED5個のブロックをそれぞれ追加しましょう。



② 繰り返しの始まりと「開始」、繰り返しの終わりと「終了」を矢印でつなぎましょう



③ 繰り返しの始まりと終わりの間に、LEDブロック4個をつなぎましょう



④ 繰り返しのブロックの設定を「5回繰り返す」にしましょう。

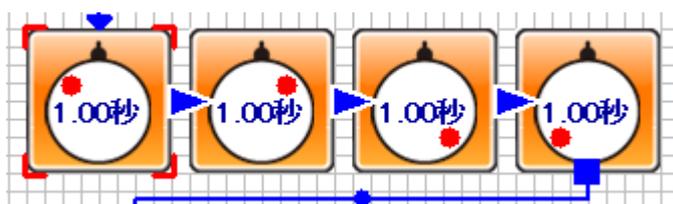
① 5 回くりかえす

② ずっとくり返す(無限)

【繰り返しのブロックの設定】

- ① ??回繰り返す：繰り返す回数を設定
- ② ずっと繰り返す：繰り返し続ける

⑤ LEDのブロックを時計回りに点滅するように設定しましょう。



点滅の切り替えを設定する

左上 右上

左下 右下

1.00 秒間点灯

チェックする

プログラムを入力できたら、書き込み・実行して動作確認しましょう。

複数の繰り返しを組み合わせたプログラム

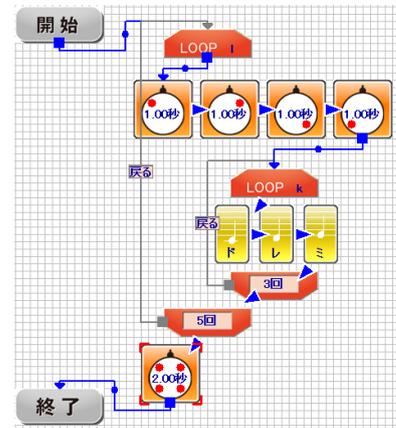
繰り返しは、一つのプログラムで複数使うことができます。また、繰り返しの中に別の繰り返しを入れることもできます。

【例題 6】

次の条件を満たすプログラムを作りましょう。

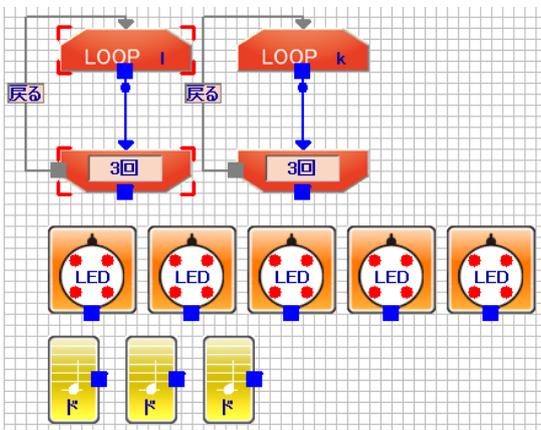
- ① LED を時計回りに 5 周点滅する
- ② LED が 1 周点滅するごとに、ブザーから「ドレミ」の音を 3 回鳴らす
- ③ プログラムの最後に、4 つの LED を 2 秒間点灯する

繰り返しは二つ使います。どのブロックがどの繰り返しの中に入るか、よく考えてプログラムしましょう。



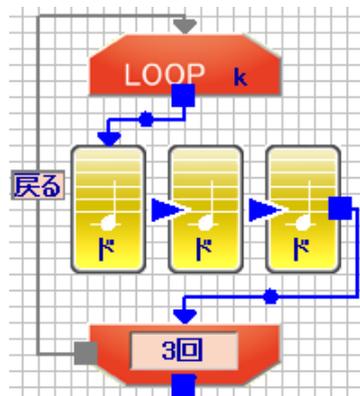
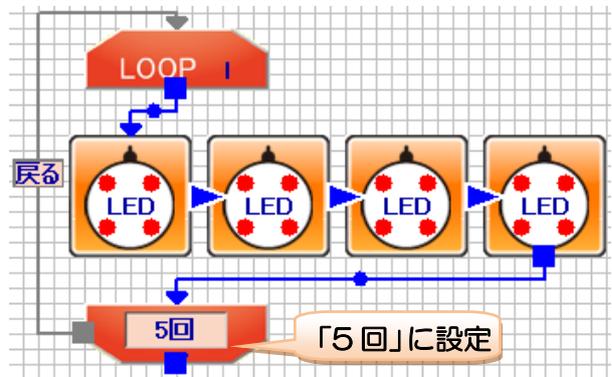
① 必要なブロックを追加しましょう。

LED : 5 個 音符 : 3 個
繰り返し : 2 セット

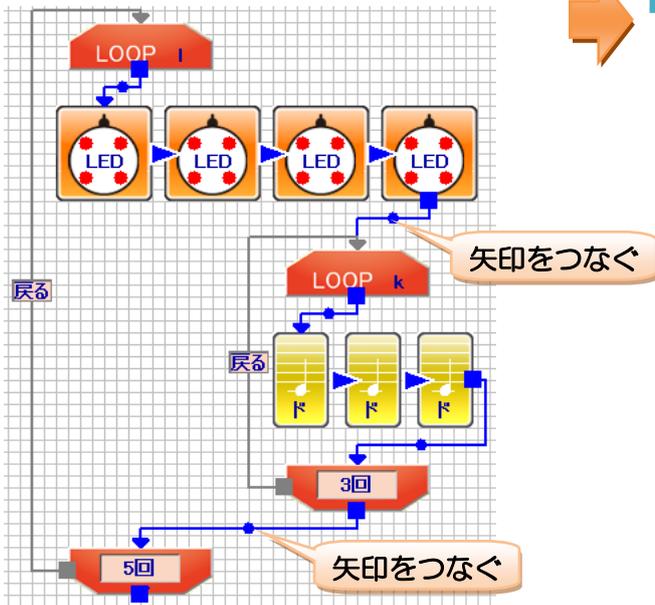


② 繰り返しのセットを作りましょう。

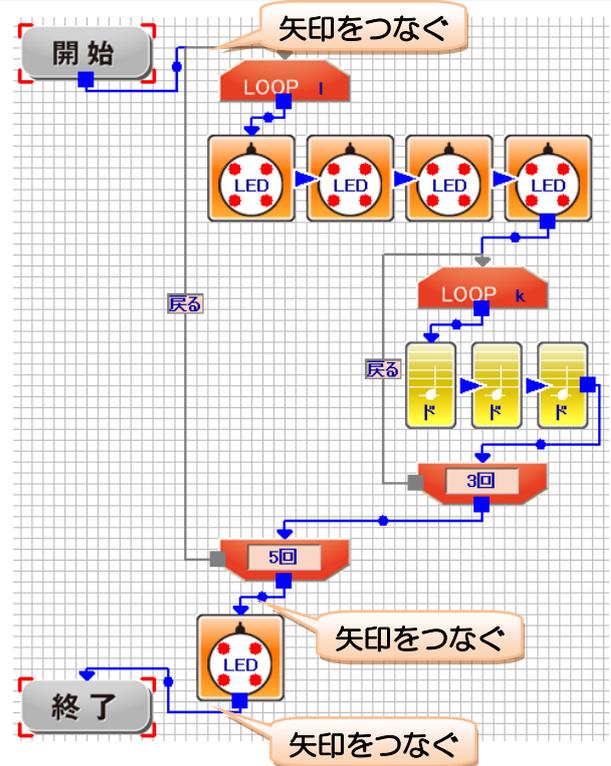
1. LED 4 個を含む繰り返し (5 回)
2. 音符 3 個を含む繰り返し (3 回)



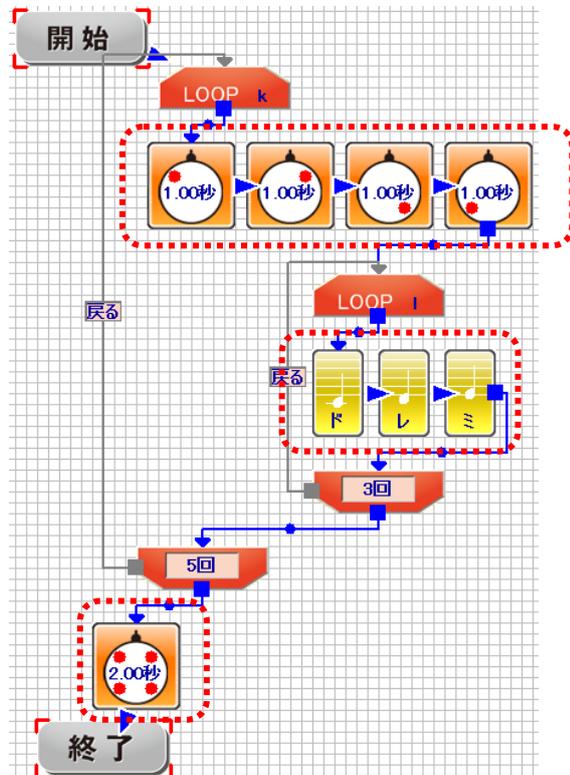
③ LED の繰り返しの中に音符の繰り返しをつなぎましょう。



④ 繰り返しの外に LED ブロックを 1 個 つなぎ、最初と最後に開始・終了をつなぎましょう



⑤ LED と音符のブロックの設定をそれぞれ変更しましょう。

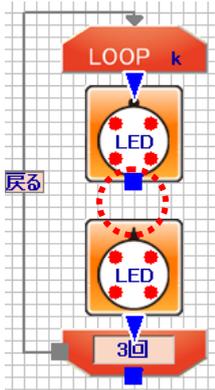


3 回繰り返す音符のブロックは、5 回の繰り返しの中にもまれるので、合計で $5 \times 3 = 15$ 回繰り返します。また、最後の LED 点灯 (2 秒) は、どの繰り返しにも含まれないので、一回しか実行しません。

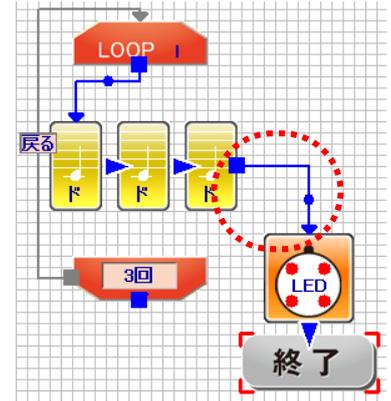
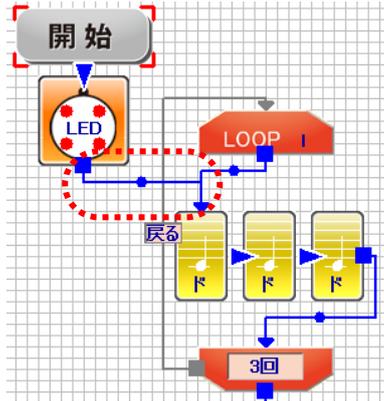
繰り返しの注意点

繰り返しは便利な命令ですが、矢印をつなぎ間違えると正しく動きません。次のようなつなぎ方をしないように注意しましょう。

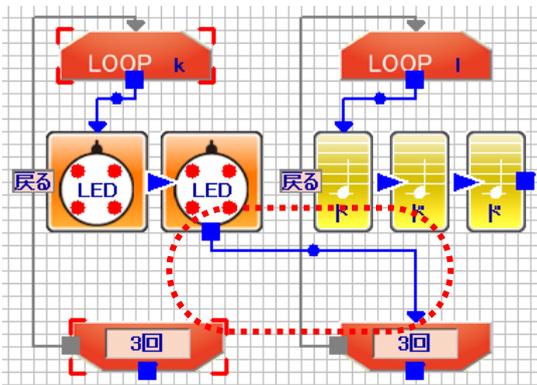
途中で矢印が途切れていると、そこで終わってしまいます



始まりを通らずに繰り返しの中につないだり、終わりを通らずに外の命令につなぐと、繰り返し回数などがおかしくなります



別の繰り返しの始まり・終わりにつなぐと、動作がおかしくなります



繰り返しを使ったプログラムの練習課題

繰り返しを使ったプログラムの課題にチャレンジしてみましょう。

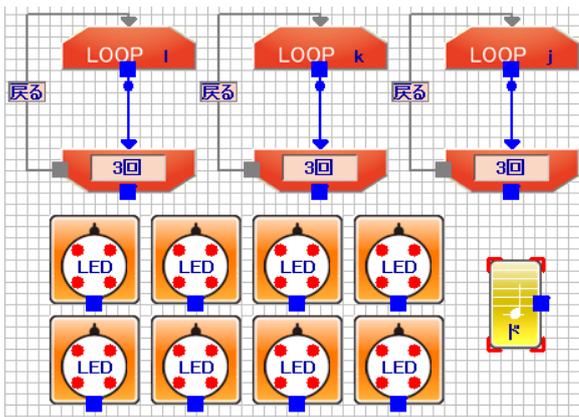
【課題③】

LED を、時計回り・反時計回りで順番に点灯するプログラムを作りましょう。

- ① 時計回りは 2 回、反時計回りは 3 回繰り返す
- ② 時計回り・反時計回りのセットを 5 回繰り返す
- ③ 最後に「ド」の音を 1 回鳴らしてプログラムを終了する

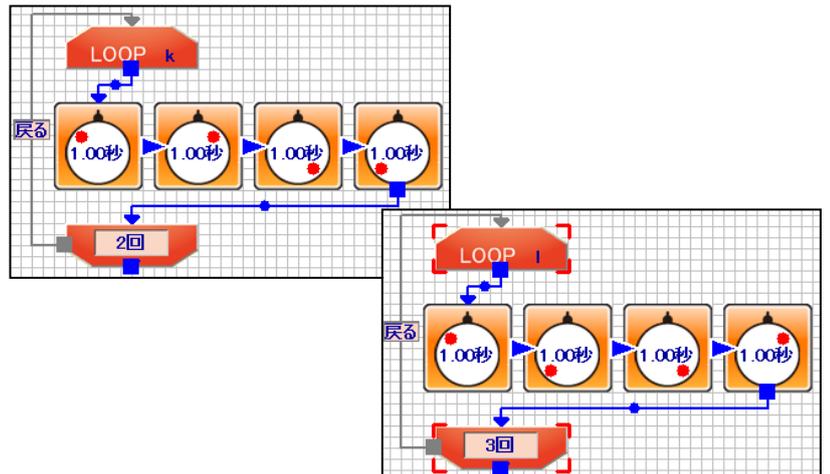
【ヒント①】

繰り返し：3 セット、LED：8 個、
音符：1 個を追加します。



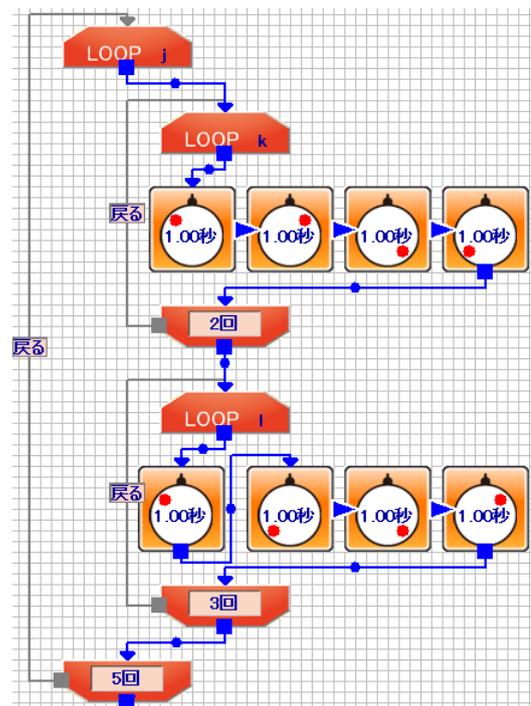
【ヒント②】

時計回り (2 回繰り返し)、反時計回り
(3 回繰り返し) を作ります。



【ヒント③】

時計回り・反時計回りの繰り返しをつなぎ、
それを更に別の繰り返し (5 回) の中につな
ぎます。



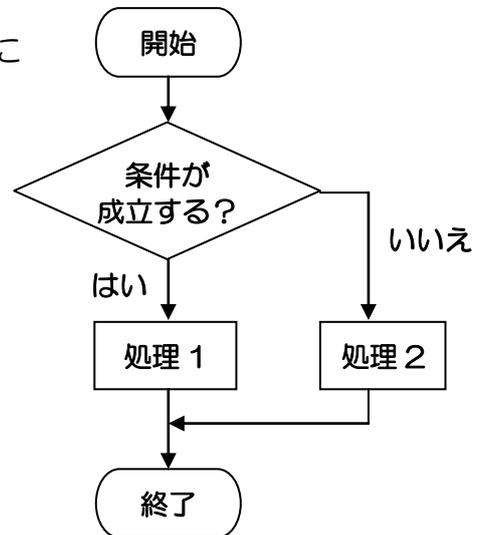
分岐のプログラム

分岐のプログラムは、決められた条件によって、プログラムを実行する道がいくつかに分かれる（分岐）構造です。

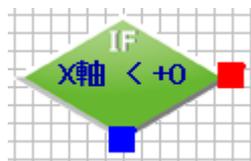
【分岐のプログラム】

あらかじめ条件を決めて、プログラム実行中にその条件が成立するかどうかで、進む道が二つに分かれる。

便利な命令だが、設定する条件や「はい」「いいえ」の矢印のつなぎ方を間違えると、逆の方向に進んでしまったり、どちらか一方にしか進まなくなってしまうので、よく考えてプログラムを作る必要がある。



分岐のプログラムを作成するには、以下の「分岐」のブロックを使用します。



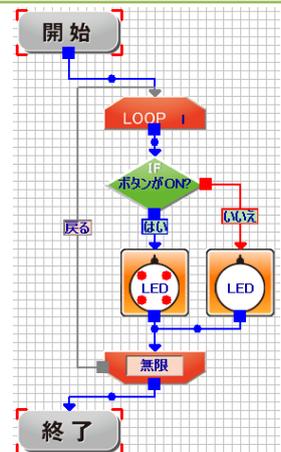
分岐のブロック

分岐のブロックには、青と赤二つの矢印があり、条件に応じてどちらかに進みます。条件には、操作ボタン、加速度センサ、タイマーなどを使います。

【例題 7】

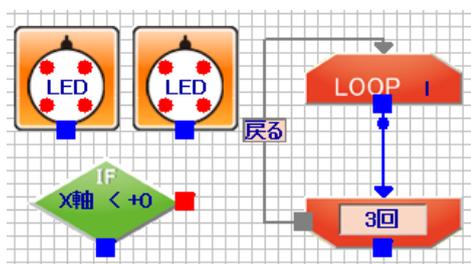
操作ボタンを押したときだけ LED が光るプログラムを作しましょう

分岐の条件は「操作ボタンが ON であるか」です。また、繰り返しを使ってすぐにプログラムが終了しないようにしています（「ずっと繰り返す」にしないと、すぐにプログラムが終了します）。

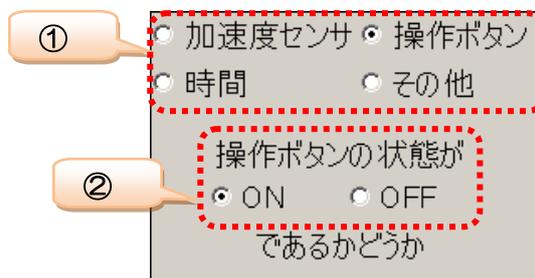


① 必要なブロックをプログラムに追加
しましょう。

繰り返し：1 セット LED：2 個
分岐：1 個



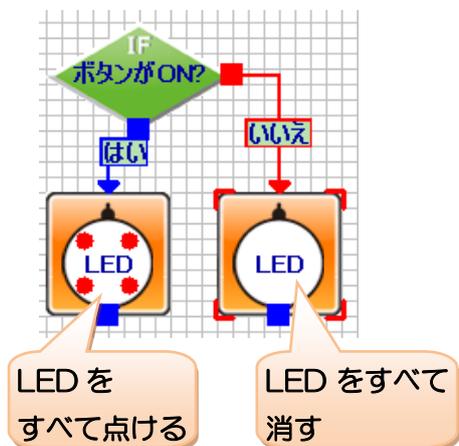
② 分岐ブロックの設定を変更しましょう。設定は、
条件に何をを使うかによって大きく変わります
(ここでは操作ボタンを使う場合の説明をします)



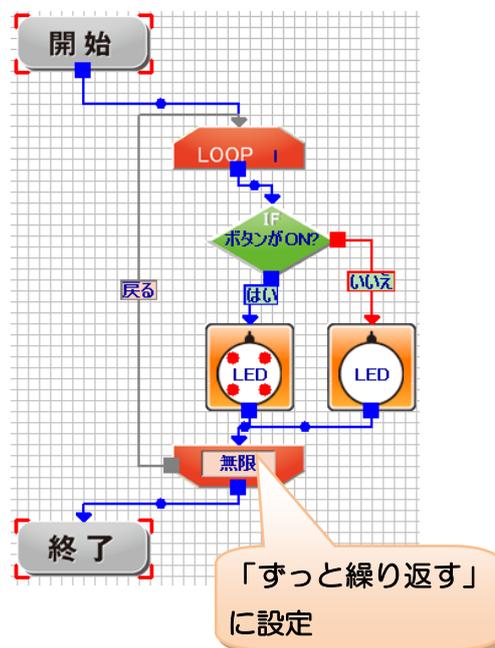
【分岐のブロックの設定 (操作ボタンの場合)】

- ① 条件に何をを使うかの選択
- ② 操作ボタンが「ON/OFF であるか」の設定

③ 分岐と LED のブロックを接続して、
LED の設定を変更しましょう。LED
は時間を指定しないようにします。



④ 分岐と LED を繰り返しの中につなげて、繰り返しを「ずっと繰り返す(無限)」
に設定しましょう。最後に開始・終了とつなげましょう。



プログラムを作成したら、書き込んで実行してみましょう。

◆正しく動作しない場合

すぐにプログラムが終わる…「ずっと繰り返す」の設定をし忘れていないか確認しましょう。

LED が光りっぱなし/消えっぱなし…分岐の条件が間違っていないか確認しましょう。

操作ボタンを押すと LED が消える (動作が逆) …分岐の条件が間違っていないか、または矢印の
つなぎ方が逆になっていないか確認しましょう。

分岐の命令と直接関係ありませんが、例題 7 では繰り返しを使ったプログラムでよく使われる二つのポイントが含まれています。それぞれ「繰り返しの利用」「時間の設定」です。

繰り返しの利用

分岐ブロックの使い方は、右図①画像の形でも正しいですが、プログラムで分岐を使うときは、繰り返しを使って何度も分岐を実行する場面が多くなります。

◆例えば…

操作ボタンを押したら反応

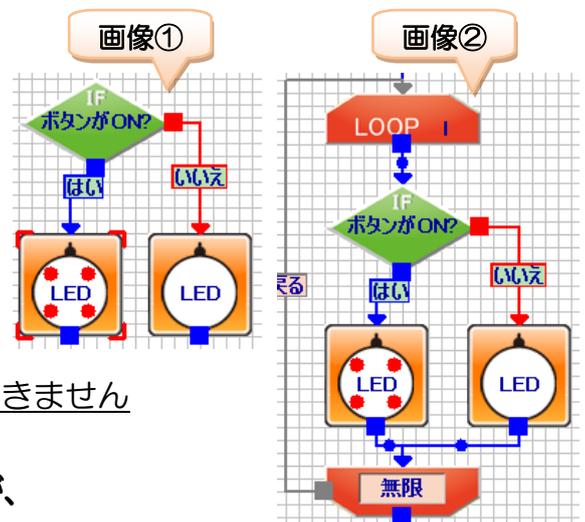
前後左右に傾けたら反応

1分経ったら反応

…などは、プログラムを繰り返し続けないと反応できません

(瞬時にプログラムが終了)

そのため、右図②画像のように、ずっと繰り返す構造が、分岐を使ったプログラムの基本形になります。



時間の設定

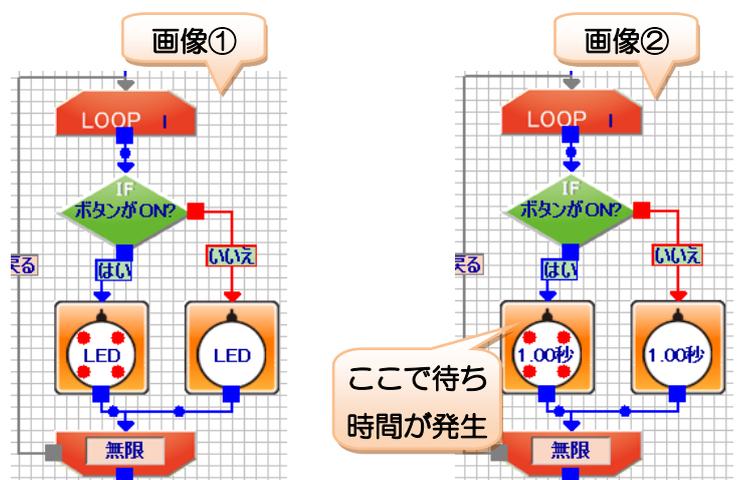
分岐を使ったプログラムは、操作ボタンやセンサに素早く反応させたい場面が多いですが、その場合「鳴り終わるまで待つ」「??秒間点灯」などを使わないよう設定します。

下図①画像と②画像の違いは、LED ブロックに時間を設定しているかどうかですが、1 回の繰り返しかかる時間が大きく違います。

◆①画像のプログラム…1 回の繰り返しが一瞬で終わる = 反応が速い

◆②画像のプログラム…1 回繰り返すのに 1.00 秒かかる = 反応が遅い

②画像の場合、毎回 LED ブロックで 1 秒間待ち時間が発生します。当然、待っている間は分岐ブロックを実行できないので、その間操作ボタンに反応があっても、見逃してしまいます。



これらのポイントは分岐を使うときの基本形ですが、当然これらを使わない分岐型プログラムも存在します。大事なのは「なぜこれらが必要か」を理解して、場面ごとに使い分けることです。

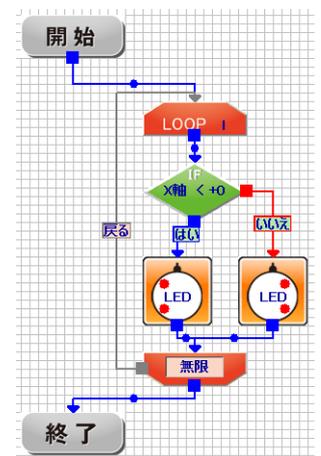
加速度センサを使った分岐プログラム

次は、加速度センサを使った分岐プログラムを作ってみましょう。

【例題 8】

本体を左右に傾けると、傾いた方のLEDが光るプログラムを作りましょう。

操作ボタンは、常に「ON/OFF」のどちらかの状態ですが、加速度センサの値は-32～+31の間で数値が変化します。分岐ブロックの条件の設定が操作ボタンより少し複雑になるので、「右に傾けたとき」「左に傾けたとき」の状態とLEDの光り方を良く考えてプログラムしましょう。

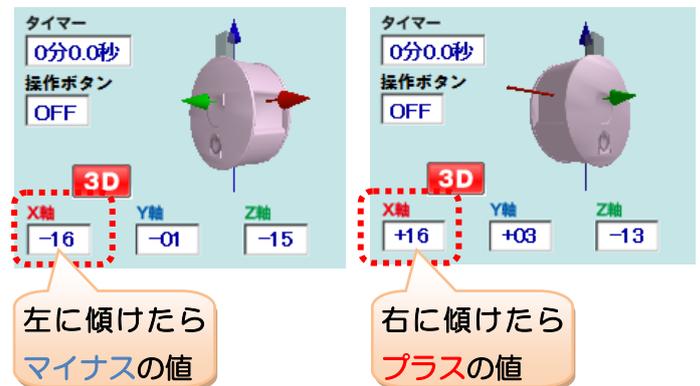


プログラムを始める前に、まずは本体を左右に傾けたときに、どのセンサがどう変化するかを調べましょう。

左右の傾きは、加速度センサのX軸で調べます。PCと本体をつないで、センサエリアから、左右に傾けたときの値をメモしましょう。

左に傾けた場合【マイナスの値=0より小さい】

右に傾けた場合【プラスの値=0より大きい】

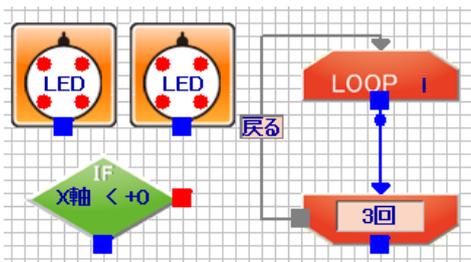


分岐の命令を使う場合は、このように「分岐させたいときに、どこがどう変化するか」を最初に調べる必要があります。この変化は、頭で予想するだけでなく、実際に本体を使って分岐させる時と同じ状態を再現して調べましょう。

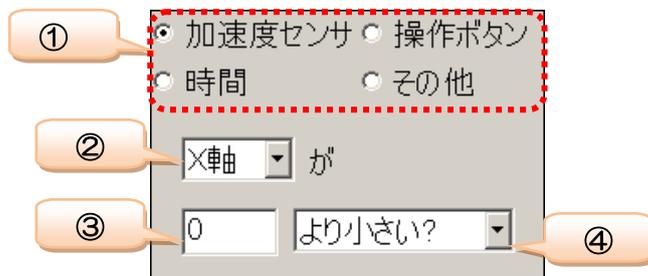
分岐させる時の本体の変化を調べたら、今度はそれを使ってプログラムを作りましょう。

① 必要なブロックをプログラムに追加しましょう。

繰り返し：1セット LED：2個
分岐：1個



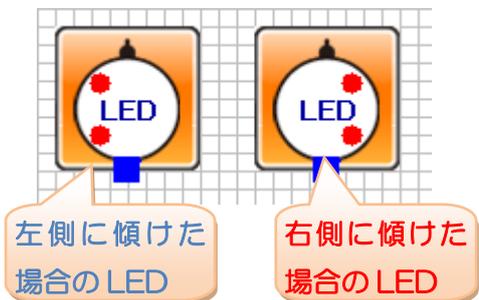
② 分岐ブロックの設定を変更しましょう。ここでは「加速度センサ」を使う設定をします。



【分岐のブロックの設定（加速度センサの場合）】

- ① 条件に何をを使うかの選択
- ② 加速度センサの軸：X/Y/Zの三つから選択
- ③ 比較する数値：センサと比べる数値を設定
- ④ 比較する条件：センサの値が、③の値と比べて **小さいか/同じか/大きい**かの条件を設定

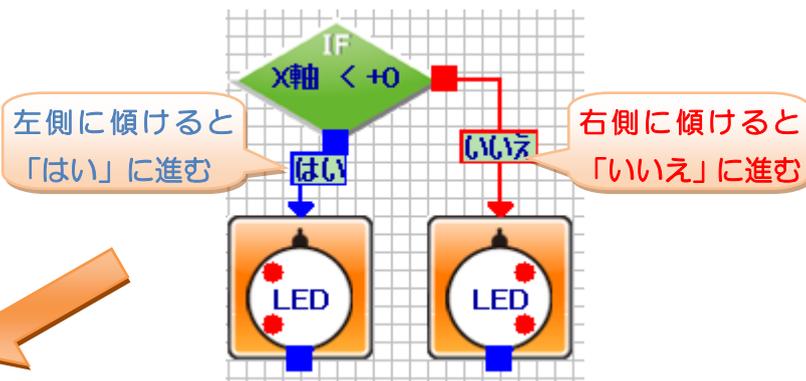
③ LEDの設定を変更しましょう。LEDは時間を指定しないようにします。



④ **傾けた方のLEDが点く**よう、「はい」「いいえ」の矢印を正しくつなぎましょう。

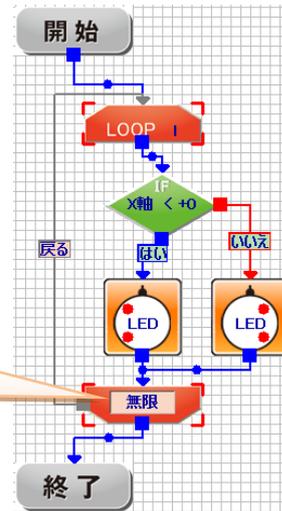
【条件：X軸が0より小さいか?】

- ◆左に傾けた場合のLEDは...
X軸が0より小さくなる = 「はい」につなぐ
- ◆右に傾けた場合のLEDは...
X軸が0より大きくなる = 「いいえ」につなぐ



⑤ 分岐とLEDを繰り返しの中につなげて、繰り返しを「ずっと繰り返す(無限)」に設定しましょう。最後に開始・終了とつなげましょう。

「ずっと繰り返す」に設定



正しく動作するプログラムができれば、続けて次の例題も試みましょう。

【例題 9】

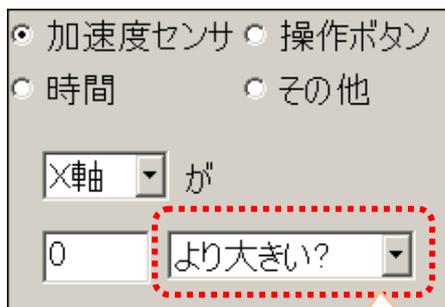
例題 8 のプログラムを元に、今度は「傾けた方と反対側の LED が光る」プログラムを作りましょう。

分岐の動きを逆にしたプログラムを作成しましょう。分岐の動きを逆にする場合は、次の 3 通りの方法があります。全て試してみましょう。

1. 分岐ブロックの条件を逆にする（「X 軸が 0 より大きいか？」に変える）
2. 矢印のつながぎ方を逆にする（「はい」と「いいえ」の矢印を逆にする）
3. 「はい」と「いいえ」につないだ命令の設定を逆にする（LED ブロックの光り方を変える）

【1. 分岐の条件を逆にする方法】

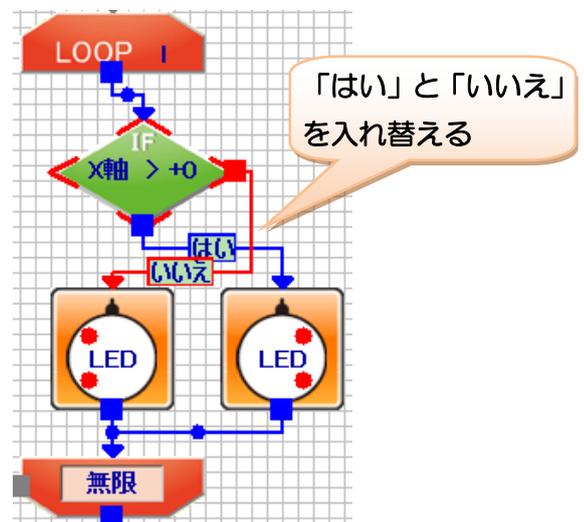
分岐ブロックの設定で、比較する条件を「より小さい」から「より大きい」に変更します。



条件を「より大きい」に変更する

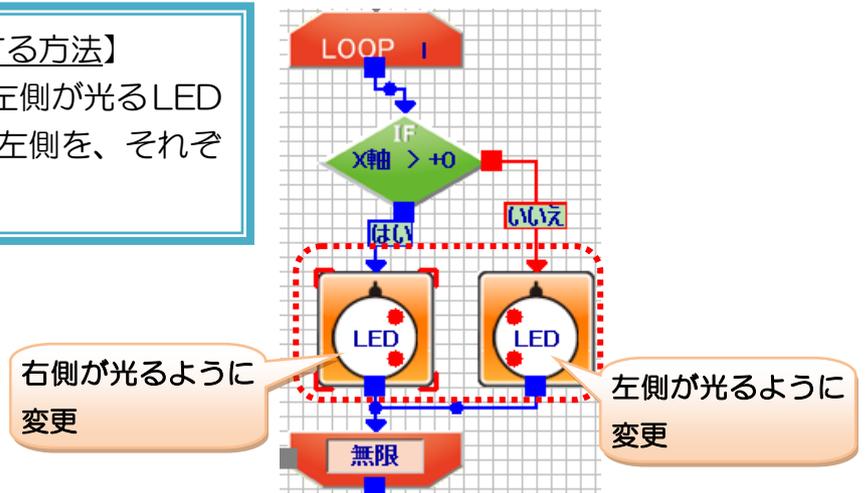
【2. 矢印のつながぎ方を逆にする方法】

「はい」と「いいえ」の矢印を、入れ替えてつなぎます。



【3. 命令の設定を逆にする方法】

2つのLEDブロックの設定を、左側が光るLEDは右側を、右側が光るLEDは左側を、それぞれ光らせるように変更します。



分岐の命令を正しく動かすためには、「条件」「矢印のつなぎ方」「命令の設定」の三つが正しくなる組み合わせを考える必要があります。

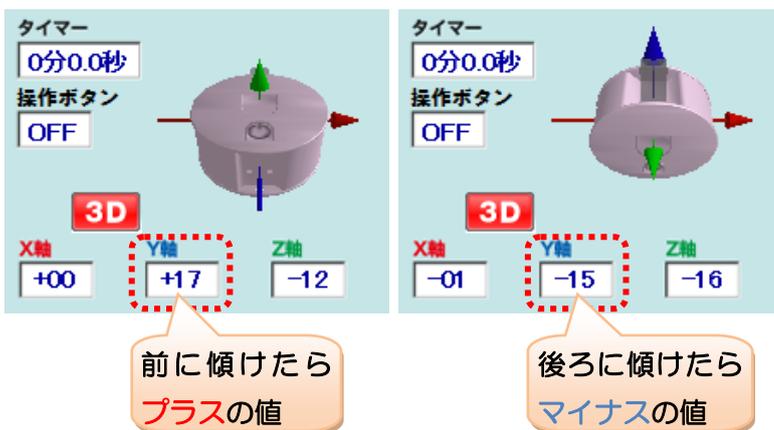
分岐プログラムの課題

【課題④】

本体を前後に傾けたときに、傾けた方のLEDが光るプログラムを作りましょう。

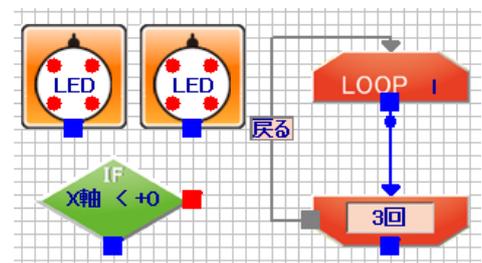
【ヒント①】

前後の傾きは、Y軸を調べます。



【ヒント②】

分岐：1個、LED：2個
繰り返し：1セットを追加します。



例題8を応用してプログラミングしましょう。プログラムが正しく動かない場合は、「条件」「矢印のつなぎ方」「命令の設定」の三つが正しいか見直しましょう。

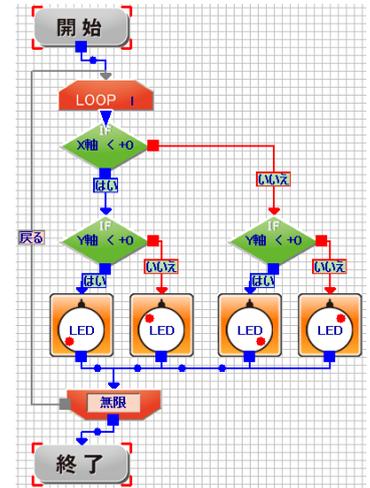
複数の分岐を使ったプログラム

分岐命令は扱いが難しいですが、使いこなせばとても複雑なプログラムを作ることができます。更に、複数の分岐を組み合わせれば、2通り以上の分岐を作ることにも可能です。

【例題 10】

本体を前後左右に傾けて、傾けている方のLEDを1個だけ光らせるプログラムを作りましょう。

最終的に4個のLEDのどれか一つを光らせるので、分岐の結果が4通りに分かれるプログラムにします。分岐ブロックを連続でつないで、分岐先を増やします。

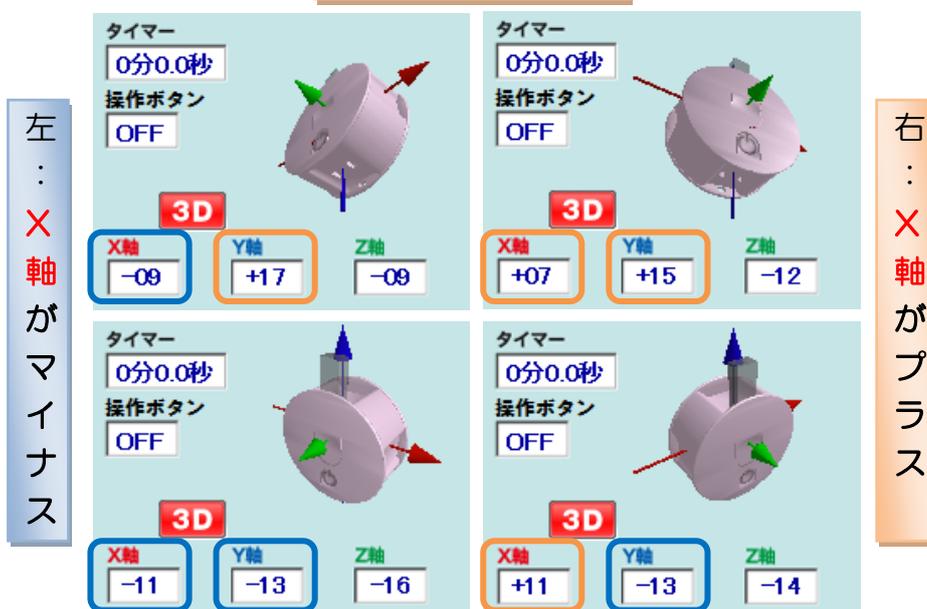


まずは、前・後・左・右に本体を傾けたときの、センサの変化を調べましょう。X軸（左右）とY軸（前後）の2方向について、プラス・マイナスの組み合わせで4通りに分けられます。

【前後左右の傾きに対する加速度センサの反応】

- ◆左前… X軸が**マイナス**、Y軸が**プラス**
- ◆右前… X軸が**プラス**、Y軸が**プラス**
- ◆左後… X軸が**マイナス**、Y軸が**マイナス**
- ◆右後… X軸が**プラス**、Y軸が**マイナス**

前：Y軸がプラス

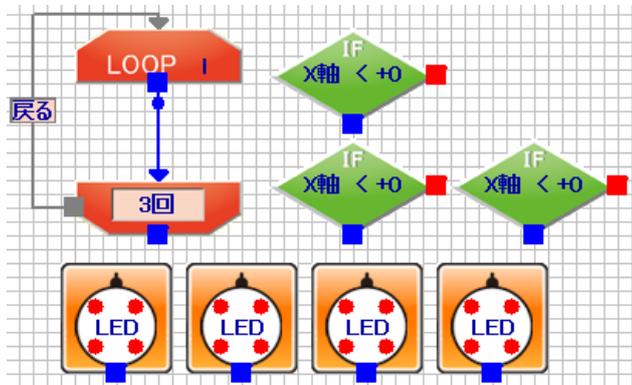


後：Y軸がマイナス

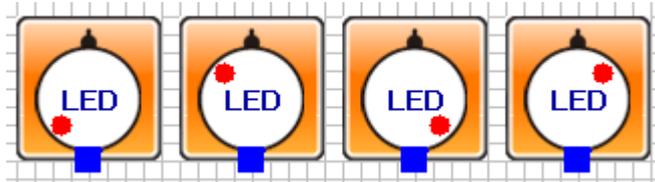
① 必要なブロックをプログラムに追加
しましょう。

繰り返し：1セット LED：4個

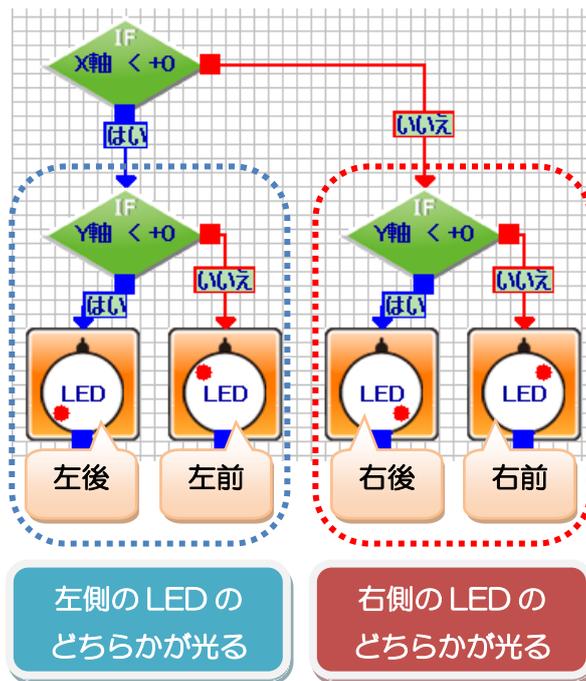
分岐：3個



② LEDブロックの設定を変更しましょう。
LEDは時間を指定しないようにします。



③ 条件を決める前に、4通りの分岐先を作ります。
一つの分岐ブロックの矢印に、それぞれ別の分岐
ブロックをつなげましょう。続いて、更にその
分岐ブロックに、1個ずつLEDブロックを右
図の通りにつなげましょう



このつなぎ方の場合、次のようなプログラムになることがわかります

◆最初の分岐ブロックは...

「はい」に進む…左側の前後が光る

「いいえ」に進む…右側の前後が光る

◆残り2個の分岐ブロックは...

「はい」に進む…後側の左右が光る

「いいえ」に進む…前側の左右が光る

つまり、正しく動かすためには、最初の分岐で左右（X軸）を、次の分岐で前後（Y軸）を区別するようにします。

④ 最初の分岐ブロックの設定をします。

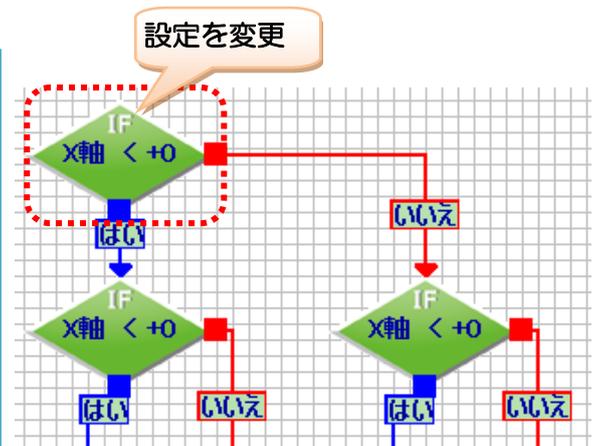
◆現在の矢印のつなぎ方

「はい」→左側が光る 「いいえ」→右側が光る

◆X軸の反応

左に傾く→マイナス 右に傾く→プラス

ここから、「傾けた方を光らせる」ためには…
「X軸が0よりも小さいか」という条件になります。



● 加速度センサ ● 操作ボタン
● 時間 ● その他

X軸 が
0 より小さい?



⑤ 残り2つの分岐ブロックの設定をします。

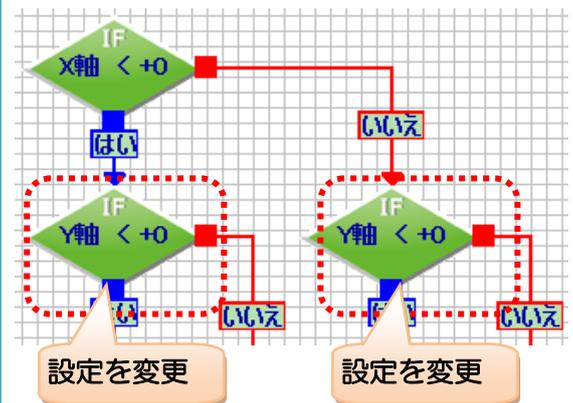
◆現在の矢印のつなぎ方

「はい」→後側が光る 「いいえ」→前側が光る

◆Y軸の反応

後に傾く→マイナス 前に傾く→プラス

ここから、「傾けた方を光らせる」ためには…
「Y軸が0よりも小さいか」という条件になります。

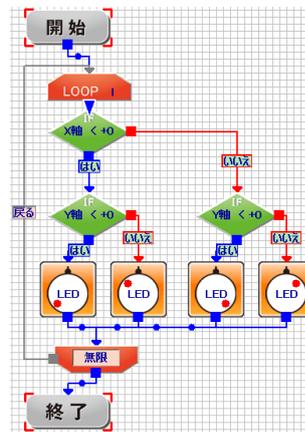


● 加速度センサ ● 操作ボタン
● 時間 ● その他

Y軸 が
0 より小さい?



⑥ 最後に、繰り返しを「ずっと繰り返す」にして接続し、更に開始・終了をつなぎましょう。



複数の分岐を組み合わせたプログラムの練習課題

【課題⑤】

例題 10 のプログラムを改造して、本体が表向きの際は傾けた方の LED を光らせて、本体を裏返すと全ての LED が消えるプログラムを作りましょう。

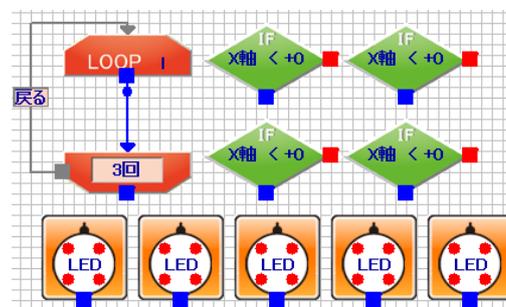
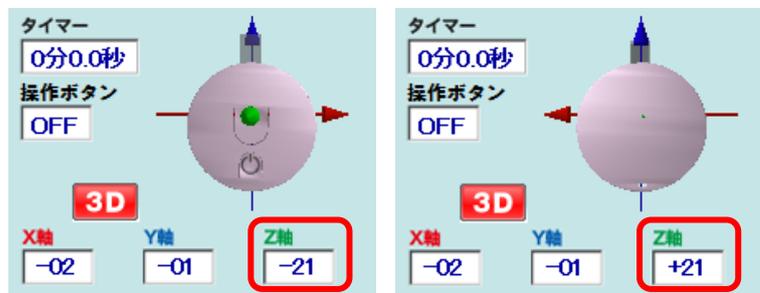
まず、分岐が何通り必要かよく考えましょう。使うセンサは X/Y/Z 軸の 3 つに増えますが、裏向きは前後左右に関係なく LED をすべて消すので、「表向きで前・後・左・右」の 4 通りに「裏向き」を加えた 5 通りの分岐になります。

【ヒント①】

表裏の区別は、Z 軸を調べます。

【ヒント②】

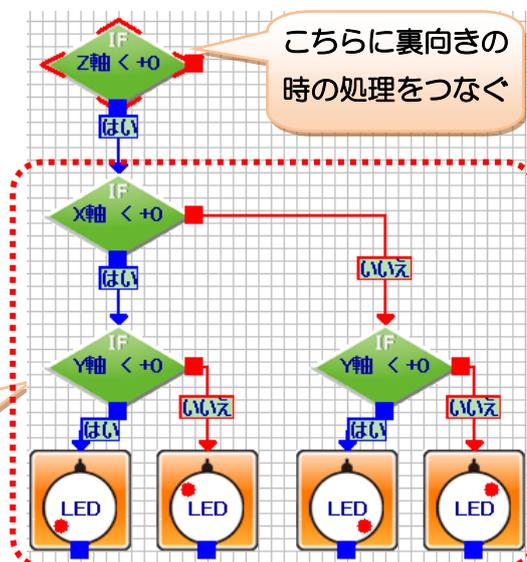
分岐：4 個、LED：5 個
繰り返し：1 セット を追加します。



【ヒント③】

最初に Z 軸で分岐させます。Z 軸が 0 より小さい場合は「前後左右」の 4 通り、Z 軸が 0 より大きい場合は「裏向き」の 1 通りの処理に分岐させます。

表向きの時
(例題 10 と同じ)



7. その他のブロックの説明

これまで説明しなかったブロックについて、以下の4種類をそれぞれ簡単に説明します。



ブザーブロック

ブザーブロックは音符ブロックのようにブザーを鳴らす命令ですが、音符ブロックと違って「鳴らす時間」を秒で設定します。また、音程は1オクターブしか使えません。

① 1.00 秒間鳴らす
② 鳴り終わるまで待つ
③ 音程 ド

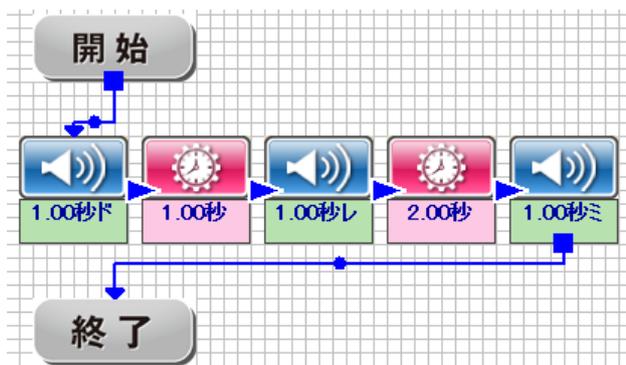
【ブザーのブロックの設定】
① 時間設定：鳴らす時間を設定
② 「ブザーが鳴り終わるまで待つか」の選択
③ 音程：ド～シの12種類

ウェイトブロック

ウェイトブロックは、「設定した時間だけ何もせずに待つ」命令です。ブザーやLEDなどの状態は継続するので、単純に時間待ちが必要な場面で使用します。

① 1.00 秒間ウェイト

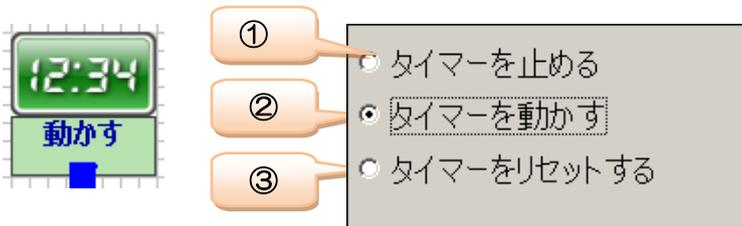
【ウェイトのブロックの設定】
① 時間設定：待つ時間を設定



【ブザーとウェイトを使ったサンプル】
ブザーをドレミの順で1秒鳴らし、間に1秒・2秒の時間待ちをします。

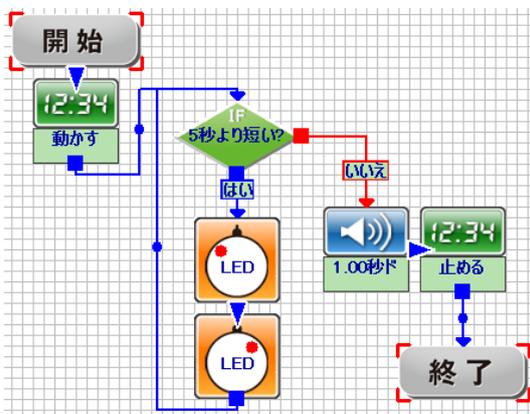
タイマーブロック

タイマーブロックは、本体に備わった「タイマー」の機能を使う命令です。この命令は分岐ブロックと合わせて使います。ラメンタイマー（3分タイマー）などのプログラムを作る場合に使います。



【タイマーのブロックの設定】

- ① タイマーの動きを止める設定
- ② タイマーを動かし始める設定
- ③ タイマーのカウントを0に戻す設定



【タイマーを使ったサンプル】
最初にタイマーを動かし、一定時間経ったらブザーを鳴らしてタイマーを止めます。

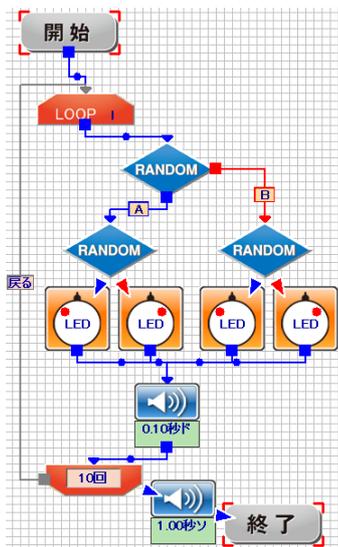
ランダムブロック

ランダムブロックは分岐ブロックと似た命令ですが、条件を決めることができません。代わりに、分岐のどちらに進むかは、本体がランダムで決めます。ルーレットゲームや占い等のプログラムを作る場合に使います。



※設定する項目はありません

【ランダムのブロックの設定】
※設定項目は特にありません



【ランダムを使ったサンプル】
ルーレットのようにランダムでLEDを光らせます。

8. その他便利な機能

操作を間違えたときに元に戻す

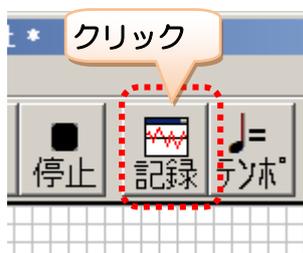
プログラム作成中に、操作や設定を間違えた場合、「戻す」ボタンをクリックすると、操作を戻して、プログラムをひとつ前の状態にすることができます。また、操作を戻しすぎた場合「進む」ボタンをクリックすると、操作を一つ進めることができます。これらの機能を使うと、プログラムのやり直しが簡単にできます。



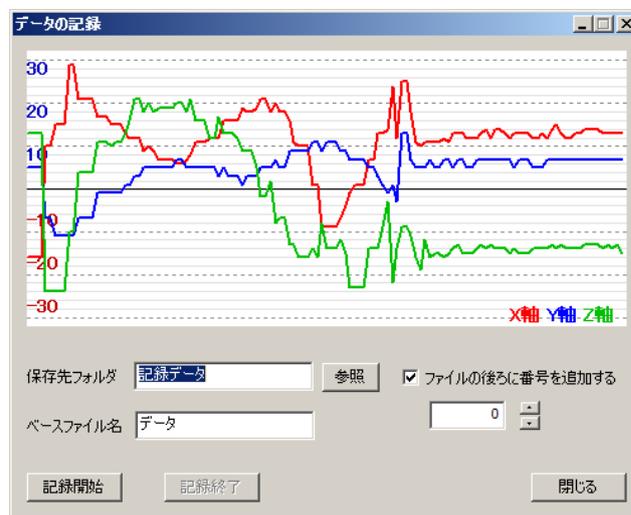
戻る…プログラムをひとつ前の状態に戻す
進む…戻した操作をもう一度進める

加速度センサの値をグラフで見る

本体とつないで「記録」ボタンをクリックすると、現在の加速度センサの値をグラフにした画面を開きます。また、この画面より、センサの値を csv 形式で PC に記録することができます。

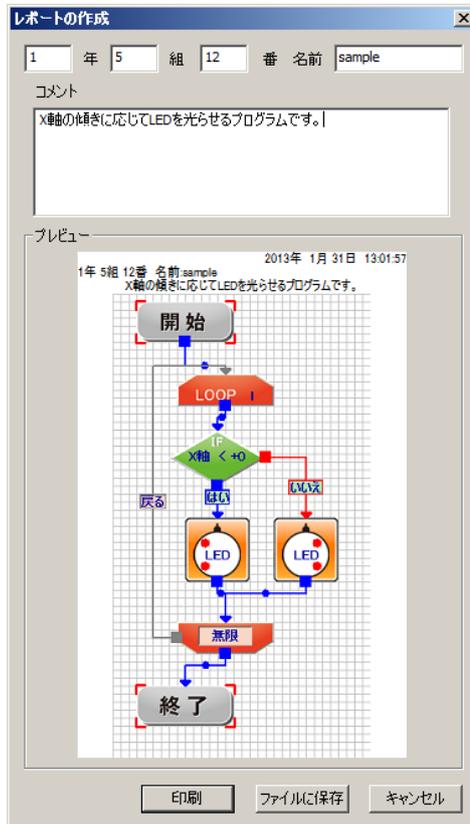


加速度センサの数値をグラフで表示。
また、保存先のフォルダ・ファイル名を指定して、センサの値を csv 形式のファイルに保存できます。



プログラムを印刷・画像で保存する

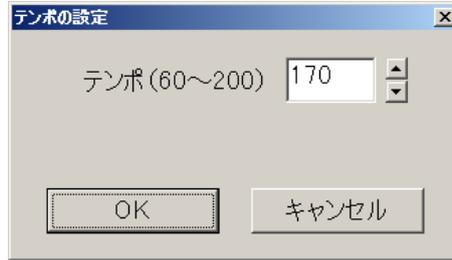
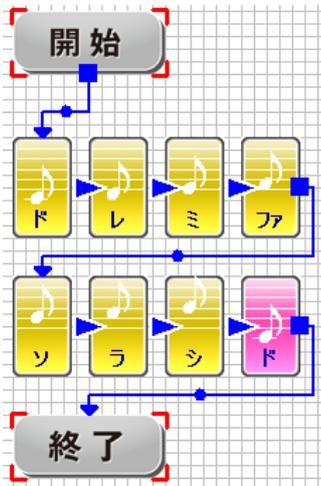
メニューの「レポート」>「レポートの作成...」をクリックすると、現在作成しているプログラムの印刷や画像データ（JPEG 形式）での保存ができます。この機能には、学年や名前、コメントなどを追加できるので、授業の宿題提出や、作成したプログラムにコメントを付けて参考資料にするなどの使い方ができます。



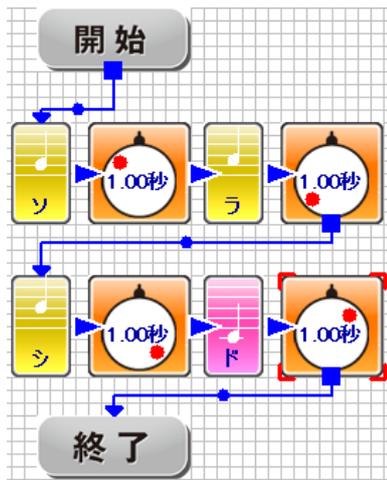
以上の内容で、プログラミングの学習は完了です。ここまでの内容を理解できれば、複雑なプログラムも作成できると思います。ソフトウェアを更に使いこなす場合、次からの「その他機能の説明」で、全ての機能を説明しているので、参照してください。

9. 課題の解答例

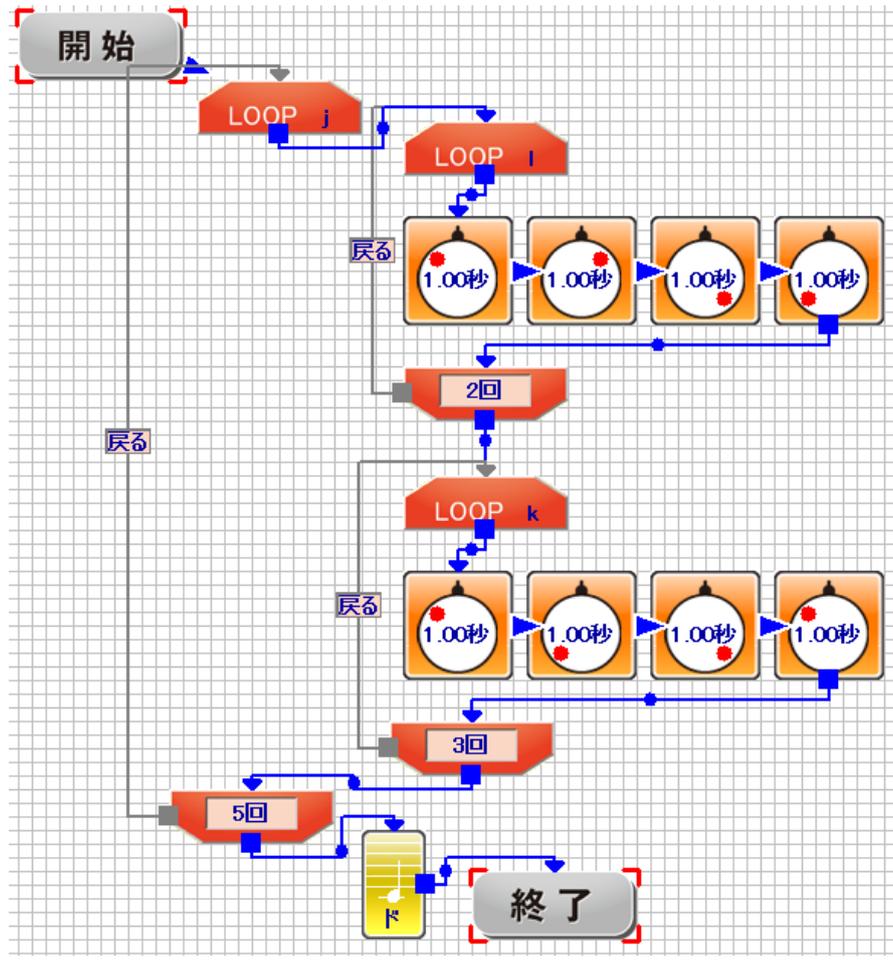
課題①



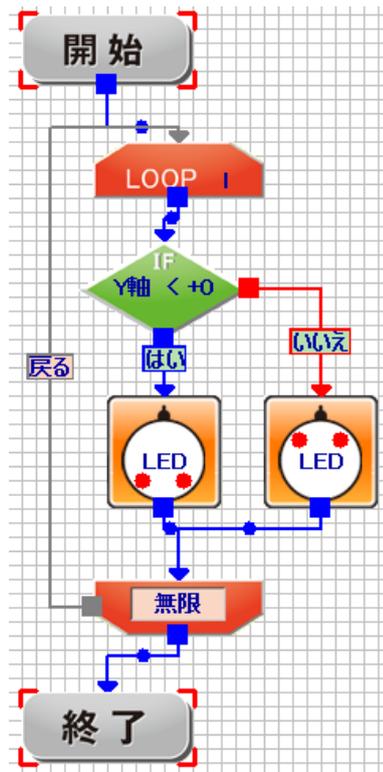
課題②



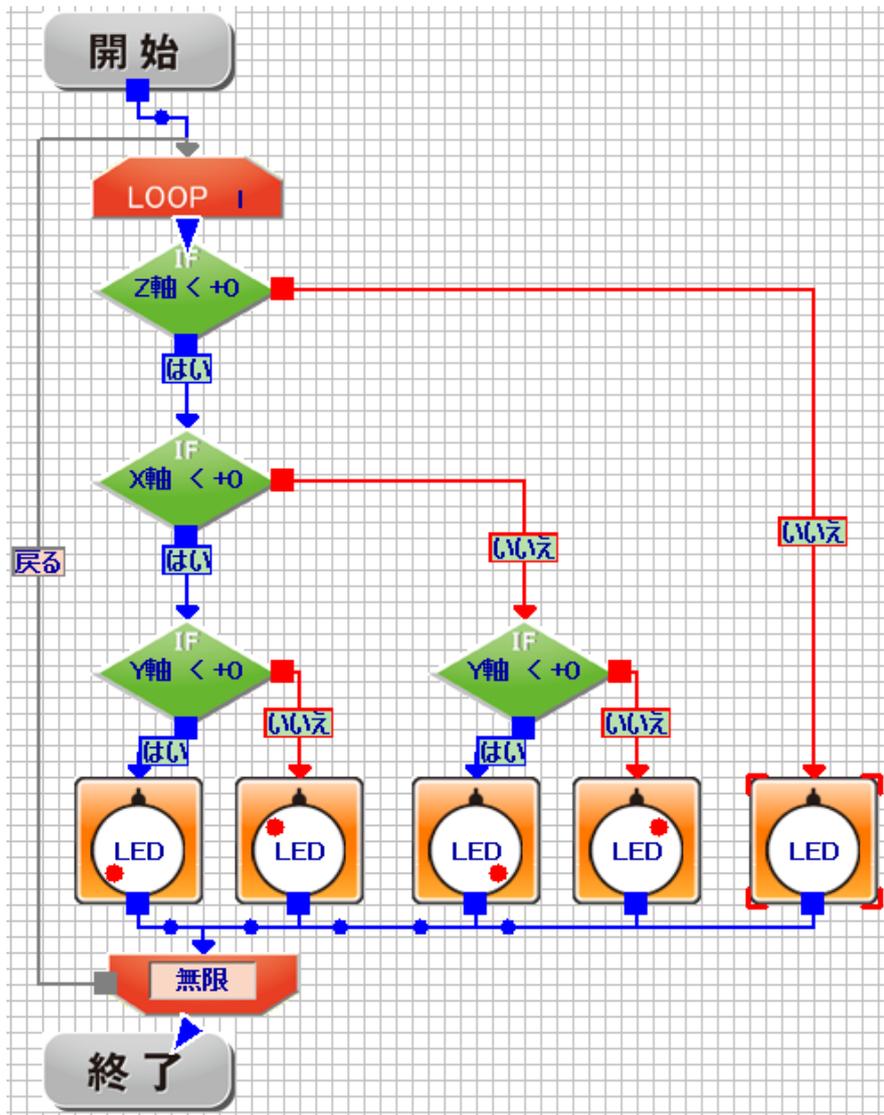
課題③



課題④



課題⑤



その他機能の説明

「その他機能の説明」は、ソフトウェアの使い方を、主に機能別に説明しています。使い始めるための準備及びプログラミングの学習がまだ済んでいない場合は、先にそちらを一度読んでください。

10. メニュー・ツールバー

ソフトウェア上部には、メニュー及びツールバーが備わっており、クリックすることで様々な操作を行います。それぞれの項目の概要は次の通りです。なお、メニューとツールバーは一部機能が重複します。機能が重複するものは、メニューの項目名の後にカッコでボタンのアイコンを併記します。



メニューの項目説明

・ ファイル

- 新規作成 (📄) 現在のプログラムを破棄して新しく作り始めます。
- 開く... (📁) ファイルに保存しているプログラムを開きます。
- 上書き保存 (💾) 現在のプログラムをファイルに上書き保存します
- 名前をつけて保存... 現在のプログラムを、ファイル名を指定して保存します。
- 終了 ソフトウェアを終了します。

・ 編集

- 元に戻す (⏪) プログラムの状態を一つ前に戻します
- やり直し (⏩) 「元に戻す」で戻したプログラムを、再び一つ操作を進めます
- 切り取り (✂) 選択したブロックを切り取り（コピーして削除）します。
- コピー (📄) 選択したブロックをコピーします。
- 貼り付け (📄) コピーしたブロックを貼り付けます
- 削除 選択したブロックを削除します
- すべて選択 プログラムエリアのすべてのブロックを選択します。

• プログラム

- ▶ プログラムの実行 (▶) 現在のプログラムをテスト実行します
- ▶ プログラムの停止 (■) テスト実行中のプログラムを停止します
- ▶ プログラムの書き込み (↓) 現在のプログラムを本体に書き込みます
- ▶ プログラムの読み込み 本体に記録されたプログラムを読み込みます

※ 「プログラムの読み込み」は、ブロックの位置を元通りに読み込むことはできません。また、ブロックの種類によっては別の命令に置き換わる場合があります。

• 設定

- ▶ テンポの設定... (♩) 音符ブロック用のテンポの設定を開きます
- ▶ 上級者向け機能の使用設定... 上級者向け機能の使用に関する設定を開きます
- ▶ メモリマップの表示 メモリマップウィンドウを開きます
- ▶ 加速度センサの調整... 加速度センサの調整画面を開きます。

• レポート

- ▶ レポートの作成... レポートの作成画面を開きます。
- ▶ データの記録... (📊) 加速度センサのグラフ表示・データ記録を行う画面を開きます。

• ヘルプ

- ▶ バージョン情報 ソフトウェアのバージョン情報を表示します
- ▶ ファームウェア書き換え 本体のファームウェアを書き換えます。

ショートカットキー

一部の機能には、以下のショートカットキーが割り当てられています。コントロールキー (CTRL) を押しながら特定のキーを入力して実行することができます。

- CTRL + Q…「終了」
- CTRL + C…「コピー」
- CTRL + X…「切り取り」
- CTRL + V…「貼り付け」
- CTRL + A…「すべて選択」
- CTRL + Z…「元に戻す」
- CTRL + Y…「やり直し」
- Delete…「削除」

ショートカットキーは、マウスポインタがプログラムエリア上にある時だけ有効になります。

11. アクションブロックの概要

ソフトウェアで使用できるアクションブロックの概要を種類別に説明します。また、各ブロックを使ったサンプルプログラムファイルを紹介しているので、そちらも合わせてご参照ください。

ブザーブロック

本体のブザーを鳴らす命令です。音符ブロックと違って「鳴らす時間」を秒で設定し、ブザーを信号・合図として使うためのブロックです。また、音程は1オクターブしか使えません。

 鳴り終わるまで待つ」、「音程 ド」があります。吹き出しで各項目が説明されています。" data-bbox="101 285 550 395"/>

① 1.00 秒間鳴らす

② 鳴り終わるまで待つ

③ 音程 ド

【ブザーのブロックの設定】

- ① 時間設定：鳴らす時間を設定
- ② 「ブザーが鳴り終わるまで待つか」の選択
- ③ 音程：ド～シの12種類

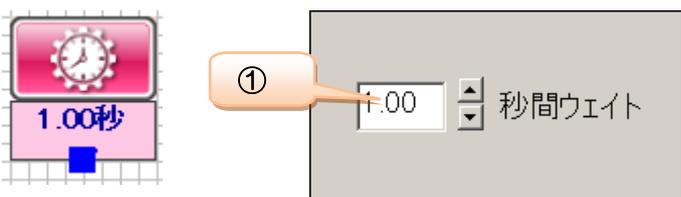
◆参考プログラム

1. サンプル「ブザー」.vbd

2. サンプル「～まで待つ」.vbd

ウェイトブロック

「設定した時間だけ何もせずに待つ」命令です。ブザーやLEDなどの状態は継続するので、単純に時間待ちが必要な場面で使用します。



① 1.00 秒間ウェイト

【ウェイトのブロックの設定】

- ① 時間設定：待つ時間を設定

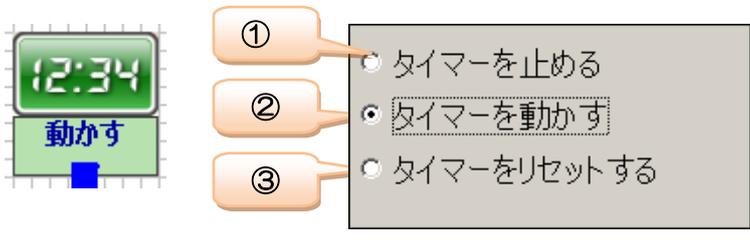
◆参考プログラム

1. サンプル「ブザー」.vbd

2. サンプル「～まで待つ」.vbd

タイマーブロック

本体に備わった「タイマー」の機能を使う命令です。この命令は分岐ブロックと合わせて使います。ラーメンタイマー（3分タイマー）などのプログラムを作る場合に使います。



① タイマーを止める
② タイマーを動かす
③ タイマーをリセットする

【タイマーのブロックの設定】

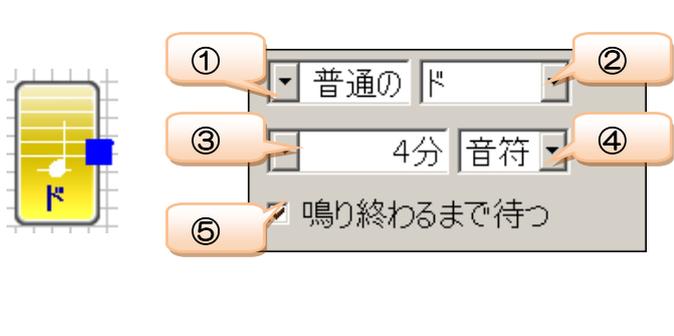
- ① タイマーの動きを止める設定
- ② タイマーを動かし始める設定
- ③ タイマーのカウントを0に戻す設定

◆参考プログラム

- 1. サンプル「タイマーを使った分岐」.vbd
- 2. サンプル「タイマーを使った分岐 2」.vbd
- 3. サンプル「ラーメンタイマー」.vbd

音符ブロック

本体のブザーから音楽を鳴らすための命令です。鳴らす時間を音符と同じ種類から選択でき、音の高さも3オクターブから選択できます。



① 音の高さ：低い/普通/高い
② 音程：ド～シ
③ 音長：32分音符～全音符
④ 種類：音符/休符
⑤ 「ブザーが鳴り終わるまで待つ」

【音符のブロックの設定】

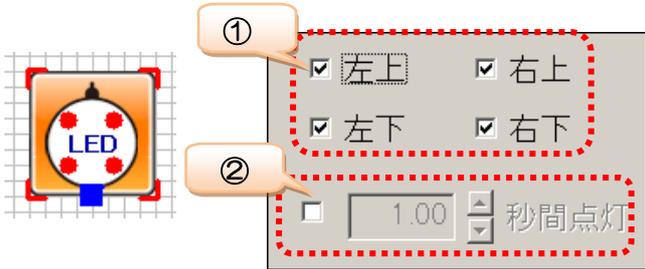
- ① 音の高さ：低い/普通/高い の3種類
- ② 音程：ド～シ の12種類
- ③ 音長：32分音符～全音符 の11種類
- ④ 種類：音符/休符 の2種類
- ⑤ 「ブザーが鳴り終わるまで待つ」の選択

◆参考プログラム

- 1. サンプル「ドの音を鳴らす」.vbd
- 2. サンプル「かえるの歌」.vbd
- 3. ブザーサンプル「オクターブ」.vbd
- 4. 音楽サンプルシンプル「Happy Birth Day」.vbd

LED ブロック

本体に 4 つ備わった赤色 LED を光らせる命令です。



【LED のブロックの設定】

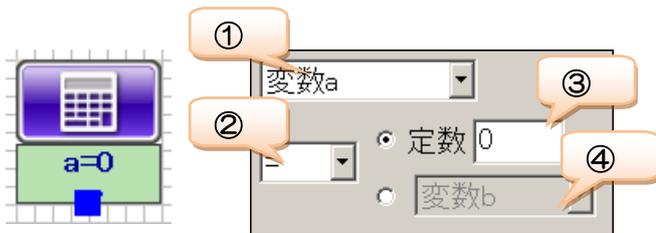
- ① LED の点滅：点ける・消すをチェックで選択
- ② 時間の設定：チェックすると、指定の時間だけ LED を点けます。時間が過ぎたら LED を消して次の命令に進みます

◆参考プログラム

- 1. LED サンプル.vbd
- 2. サンプル「～～まで待つ」.vbd

演算ブロック

本体のメモリマップに対して、値の代入や四則演算を行う命令です。本体に備わったデバイスを直接制御したり、センサの平均値を求めるなど、複雑なプログラムを作ることができます。このブロックを使用するには、上級者向け機能の使用設定の変更が必要です。



【演算ブロックの設定】

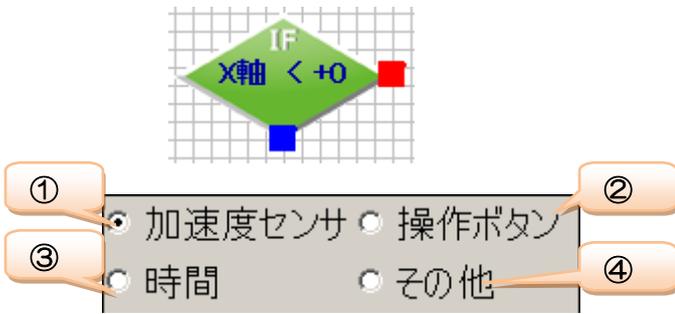
- ① 結果を代入する変数を設定
- ② 演算方法を選択
- ③ 演算に使う定数を設定
- ④ 演算に使う変数を設定

◆参考プログラム

- 1. 音楽サンプル「Happy Birth Day」.vbd

分岐ブロック

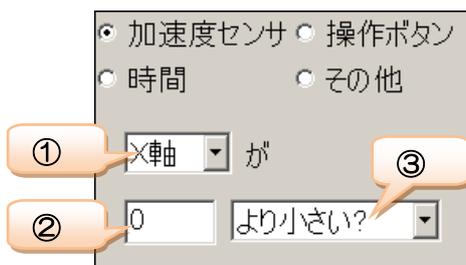
センサや操作ボタン・タイマーなどを使い、「操作ボタンが ON か?」「センサの値が 0 以上か?」など、任意の条件を設定して、条件が成立するか否かでプログラムが進む方向を分岐させる命令です。センサや操作ボタンなどの状態に応じて本体の動きを変化させることができます。



【分岐ブロックの設定（上段）】
分岐ブロックの設定エリアは、上下に大きく分かります。上段は、条件に使うセンサなどの種類を選択します。

- ① 加速度センサ
- ② 操作ボタン
- ③ タイマー
- ④ ユーザ変数など

設定エリアは、上段の選択項目ごとに大きく異なるので、それぞれ説明します。



【分岐ブロックの設定（加速度センサ）】

- ① 加速度センサの軸：X/Y/Z の三つから選択
- ② 比較する数値：センサと比べる数値を設定
- ③ 比較する条件：センサの値が、②の値と比べて **小さいか/同じか/大きい**かの条件を設定

◆参考プログラム

- 1. サンプル「X 軸(横)の傾けた方を光らせる」.vbd
- 2. サンプル「Y 軸(縦)の傾けた方を光らせる」.vbd
- 3. サンプル「裏返すと LED が消える」.vbd
- 4. サンプル「傾けた方向の LED が光る」.vbd

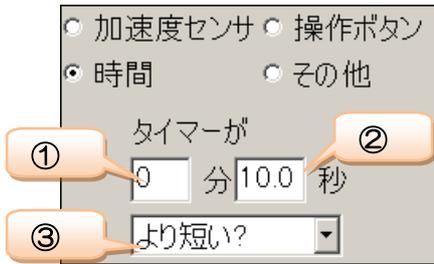


【分岐ブロックの設定（操作ボタン）】

- ① 操作ボタンが「ON/OFF であるか」の設定

◆参考プログラム

- 1. サンプル「操作ボタンを使った分岐」.vbd

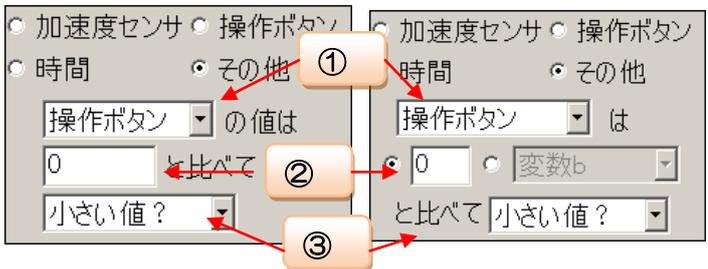


【分岐ブロックの設定（時間）】

- ① 比較する時間の「分」の設定
- ② 比較する時間の「秒」の設定
- ③ 設定した時間との比較方法。短いか/同じか/長いかの条件を設定

◆参考プログラム

1. サンプル「タイマーを使った分岐」.vbd
2. サンプル「タイマーを使った分岐 2」.vbd
3. サンプル「ラーメンタイマー」.vbd



【分岐ブロックの設定（その他）】

- ① 比較対象の変数の設定
- ② ①と比較する定数を設定。演算ブロックを使える場合、ここに変数を使うことが可能
- ③ 比較する条件を設定。小さいか/同じか/大きいかの中から選択

演算ブロックを使う設定をしている場合、「その他」の設定内容が少し変わります。

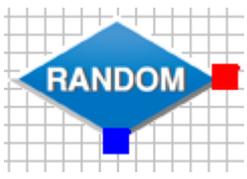
【左】演算ブロックを使う場合
【右】演算ブロックを使わない場合

◆参考プログラム

1. 音楽サンプル「Happy Birth Day」.vbd

ランダムブロック

分岐ブロックと似た命令ですが、条件を決めることができません。代わりに、分岐のどちらに進むかは、本体がランダムで決めます。ルーレットゲームや占い等のプログラムを作る場合に使います。



※設定する項目はありません

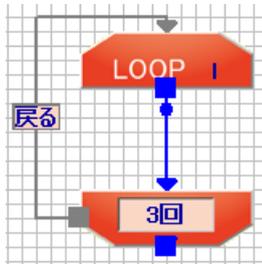
【ランダムのブロックの設定】
※設定項目は特にありません

◆参考プログラム

1. サンプル「ランダム」.vbd
2. サンプル「分岐が 3 つのランダム」.vbd

繰り返しブロック

プログラムの一部を指定の回数だけ繰り返し実行する命令です。

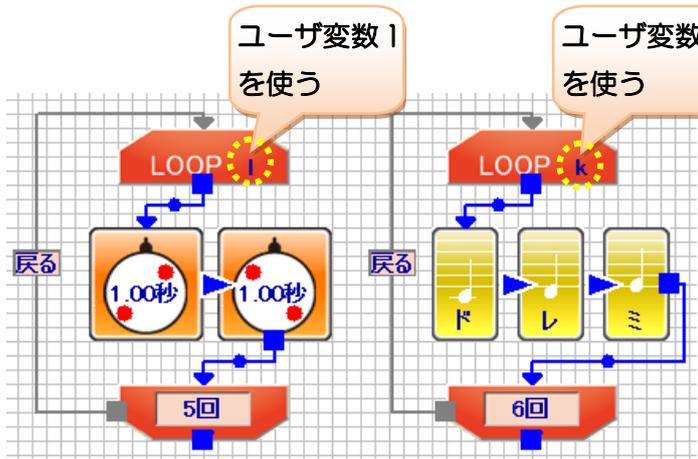


① 5 回くりかえす
② ずっとくり返す(無限)

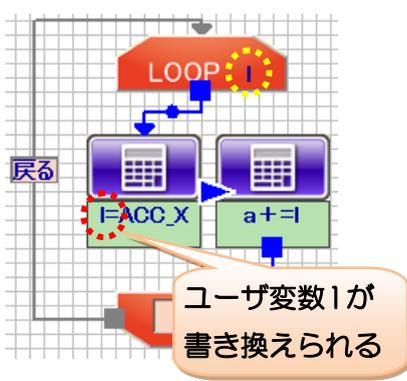
【繰り返しのブロックの設定】
① ??回繰り返す：繰り返す回数を設定
② ずっと繰り返す：繰り返し続ける

- ◆参考プログラム
- 1. サンプル「繰り返し」.vbd
 - 2. サンプル「繰り返しの中に繰り返し」.vbd
 - 3. 音楽サンプル「さくら」.vbd

なお、この命令は繰り返しの回数を知るため、繰り返し中にユーザ変数を使います。ユーザ変数は、自動的に最後の方から順番に割り当てられます。



【繰り返しブロックのユーザ変数】
繰り返しブロックは、繰り返し回数のカウント用にユーザ変数を使います。繰り返しの開始側ブロックには、割り当てられたユーザ変数が表示されます。このユーザ変数は、変数 l、変数 k、変数 j...のように、末尾から自動的に割り振られます。



ユーザ変数が書き換えられる可能性がある場合、警告を表示

変数 l が、繰り返し命令で使用する変数と重複しています。書き込みを中断しますか？

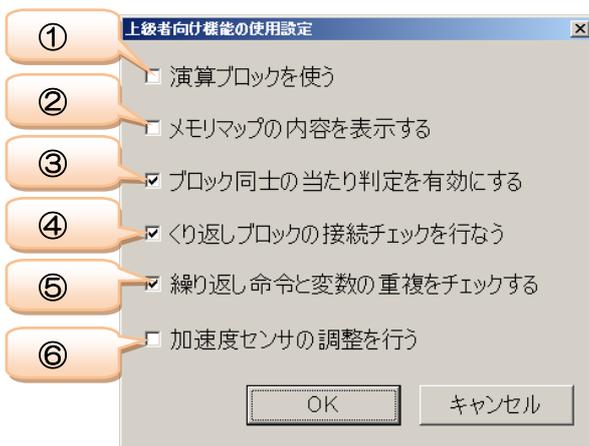
(はい) (いいえ)

繰り返しに割り当てられたユーザ変数が、別の場所で演算ブロックから書き換えられるようになっている場合、プログラムの書き込み・実行時に警告を表示します。

12. 上級者向け機能

ソフトウェアには、上級者のために、より複雑なプログラムを作成できる機能・設定が備わっています。ただし、これらの機能は扱いが難しいため、最初は使用できないようになっています。次に説明する「上級者向け機能の使用設定」を変更すると、機能を使用できます。

上級者向け機能の使用設定



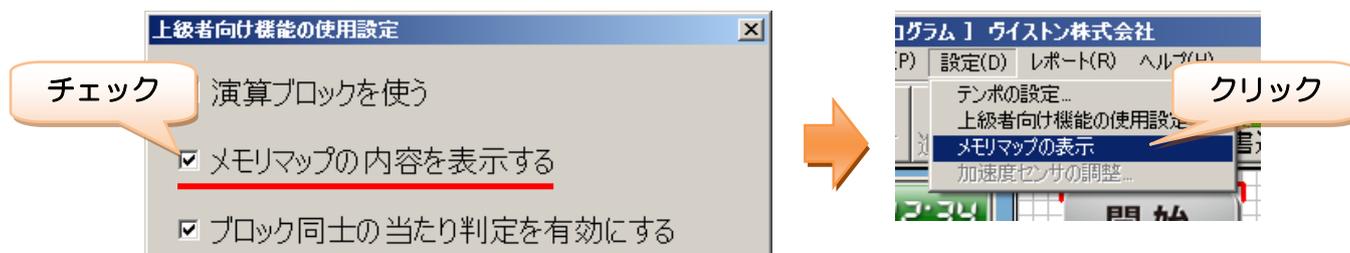
【上級者向け機能の使用設定の画面】
各項目はマウスでクリックでき、チェックの付いた項目が有効になります。プログラムのインストール直後は、左図の設定になっています。

【上級者向け機能の使用設定の項目概要】

- ① 演算ブロックを使用するかを選択します。
 - 「**アクションブロックの概要**」の演算ブロックの説明をご参照ください
- ② メモリマップウィンドウを表示するかを選択します。
 - 「**メモリマップの表示**」の項目をご参照ください
- ③ ブロック同士が重ならないようにするかを選択します。
 - アクションブロック同士を重ねて置くようになります。たくさんブロックを使うプログラムで便利です。
- ④ 繰り返しブロックを使うとき、矢印の接続のチェックを行うかを選択します
 - 繰り返しの矢印のつなぎ方が間違っても、書き込みの時に警告を出さなくなります。繰り返しを理解された方のための機能です。
- ⑤ 繰り返して使われるユーザー変数が、他の命令で使われているかチェックを行うことを選択します
 - 繰り返し命令は、繰り返し回数のカウント用にユーザー変数を一つ使います。この時使われるユーザー変数が演算ブロックで書き換えられた場合、繰り返しが正しく行われません。このようなケースを防ぐため、繰り返して使われるユーザー変数が演算ブロックで使われているときに、警告を出すようになります。
- ⑥ 加速度センサの調整を行うかを選択します。
 - 「**加速度センサの調整**」の項目をご参照ください

メモリマップの表示

「メモリマップ」とは、本体のセンサや操作ボタンの値や、その他設定値など、本体内部に記録されたあらゆる数値をまとめたものを表します。また、メモリマップの数値を羅列したものを「メモリマップウィンドウ」と言います。メモリマップウィンドウは、上級者向け機能の使用設定を変更し、メニューの「設定」→「メモリマップの表示」をクリックすると開きます。



説明	数値
操作ボタン	0
加速度X軸	-1
加速度Y軸	-1
加速度Z軸	-22
LED	16
タイマーON/OFF	0
タイマーL	0
タイマーH	0
ブザーピッチL	26
ブザーピッチH	71
ブザータイマーL	0
ブザータイマーH	0
加速度X軸オフセット	0
加速度Y軸オフセット	0
加速度Z軸オフセット	0
変数a	0
変数b	0
変数c	0
変数d	0
変数e	0
変数f	0
変数g	0
変数h	0
変数i	0
変数j	0
変数k	0
変数l	0

【メモリマップウィンドウ】

◆「説明」の項目…

数値の概要の説明です。各項目の概要は以下の通りです。

1. 操作ボタン・加速度センサ X/Y/Z 軸…センサや操作ボタンの状態
2. LED…LED の状態。特定のビットと LED の ON/OFF が連動
3. タイマーON/OFF・L・H…タイマーの状態
4. ブザーピッチ L/H・ブザータイマーL/H…ブザーの音程と音長
5. 加速度センサ X/Y/Z 軸オフセット…加速度センサの調整の値
6. 変数 a~l…ユーザ変数。演算ブロックなどで自由に読み書き可

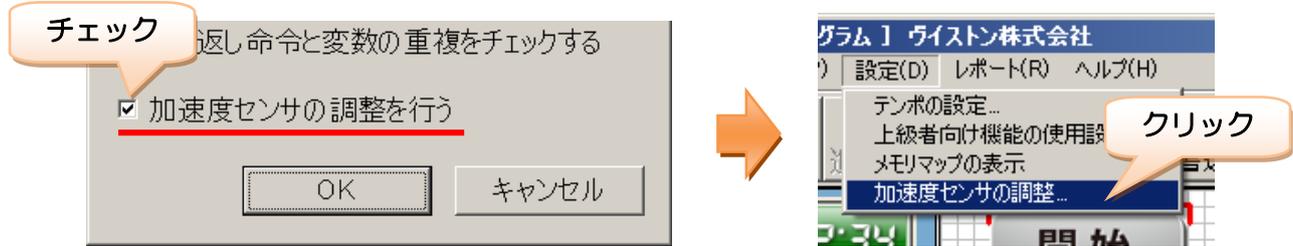
◆「数値」の項目…

本体内のメモリマップの現在値です。本体と通信している場合、逐次更新されます。

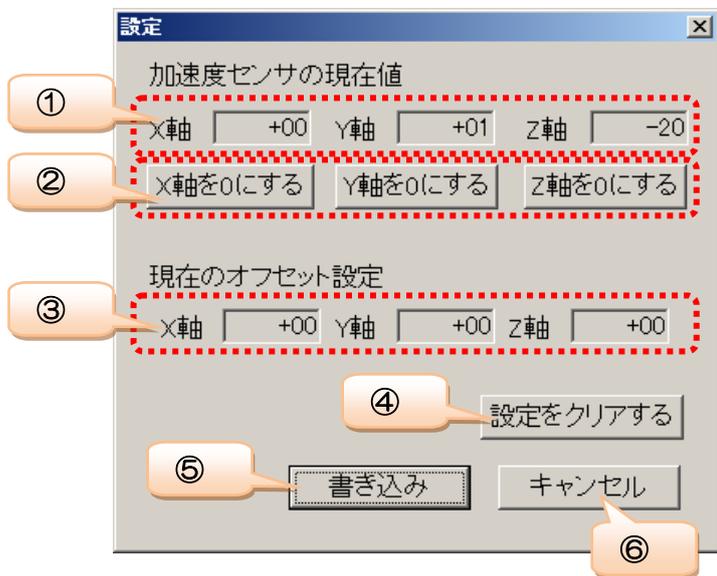
数値は全て-128~+127の範囲（符号付 1byte）です。また、L/Hが付く項目は、本来 2byte の数値が 1byte ずつ分かれて表示されたものです。

加速度センサの調整

加速度センサの値にずれがある場合などに、「**加速度センサの調整**」機能で補正することができます。加速度センサの調整を行う場合は、上級者向け機能の使用設定を変更し、メニューから「設定」→「加速度センサの調整…」をクリックします。



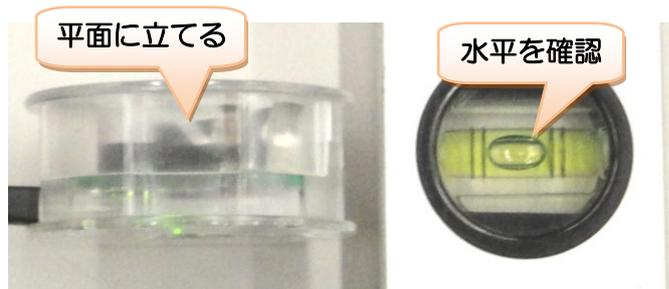
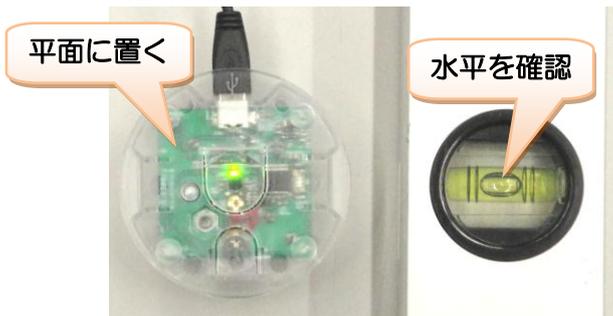
なお、加速度センサの調整を行う場合は、必ず本体と PC を接続してください。



- 【加速度センサの調整の画面】**
- ① 現在の加速度センサの値を表示します
 - ② 現在のセンサの数値を、X/Y/Z 軸の原点(±0 の位置) に設定します
 - ③ 現在の X/Y/Z 軸の調整の値です
 - ④ 「現在のオフセット」を全て 0 に戻し、設定を初期化します
 - ⑤ 調整結果を本体に書き込んで記憶させます
 - ⑥ 画面を閉じます

【X 軸・Y 軸を調整する場合】
 本体を机などの平面に置いて、水平器でなるべく平面を水平に合わせます。その状態で、「**X 軸を 0 にする**」「**Y 軸を 0 にする**」をそれぞれクリックします。

【Z 軸を調整する場合】
 本体を机などの平面に立てて、転がらないようにします。その状態で、X/Y 軸と同じく平面の水平を出して、「**Z 軸を 0 にする**」をクリックします。



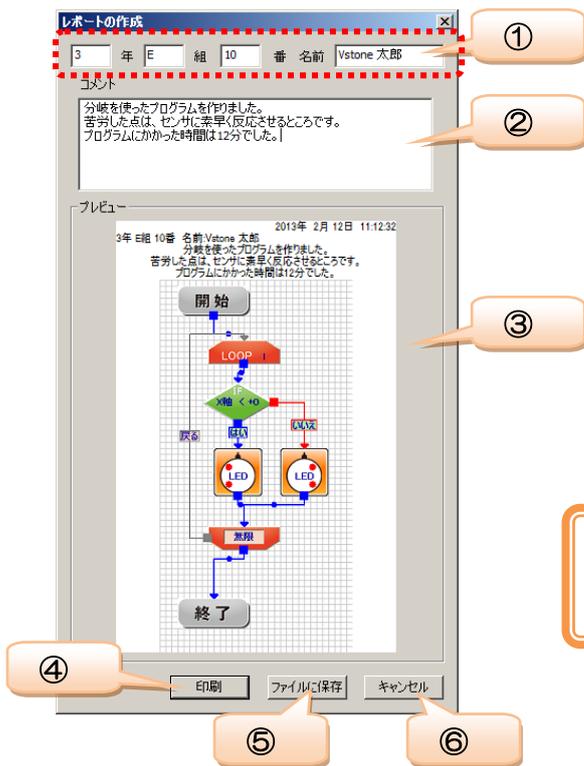
本体は教材用途のため、センサの数値を厳密に補正することはできません。また、正しく調整を行わないと、センサの値が大きくずれる場合があります。

13. レポート作成・データの記録

レポート作成

レポートの作成は、プログラムエリアの内容に、学年・組・名前やコメントを付けて、1枚の紙に印刷できる機能です。また、印刷内容を画像ファイルとしても保存できます。この機能は、宿題・課題の提出や、プログラムのサンプルとして生徒にプリントを配布するといったことに利用できます。

この機能では、大きなプログラムの場合、自動的に1枚の紙に収まるようにブロックが縮小されます。



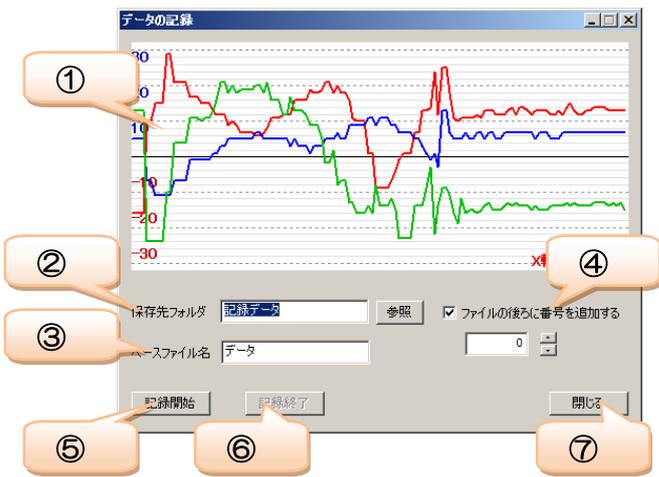
【レポート作成の画面】

- ① 学年・組・番号・名前を記入します。
- ② プログラムに関するコメントを記入します。
- ③ 印刷される内容を表示します。右上に作成日時が入ります。
- ④ 内容をプリンターで印刷します。
- ⑤ 内容を JPEG 形式の画像で保存します。
- ⑥ 画面を閉じます。

プリンターに関する設定は、お使いの PC に合わせて別途行ってください。

データの記録

データの記録は、加速度センサの値をグラフで表示したり、CSV 形式のファイルに長時間記録したりする機能です。



【レポート作成の画面】

- ① センサ値のグラフ表示です
- ② ファイルを保存するフォルダを設定します
- ③ ファイル名を設定します
- ④ ファイル名の後に番号を付けるか、また、付ける場合、その番号を設定します
- ⑤ PC への記録を開始します
- ⑥ データの記録を終了します
- ⑦ 画面を閉じます

画面下側は、センサ情報を記録するフォルダ・ファイル名を設定します。ファイル名は次のように決められ、連番で保存することが可能です。

以下の場合、“記録データ” フォルダに、次の名前で保存します。

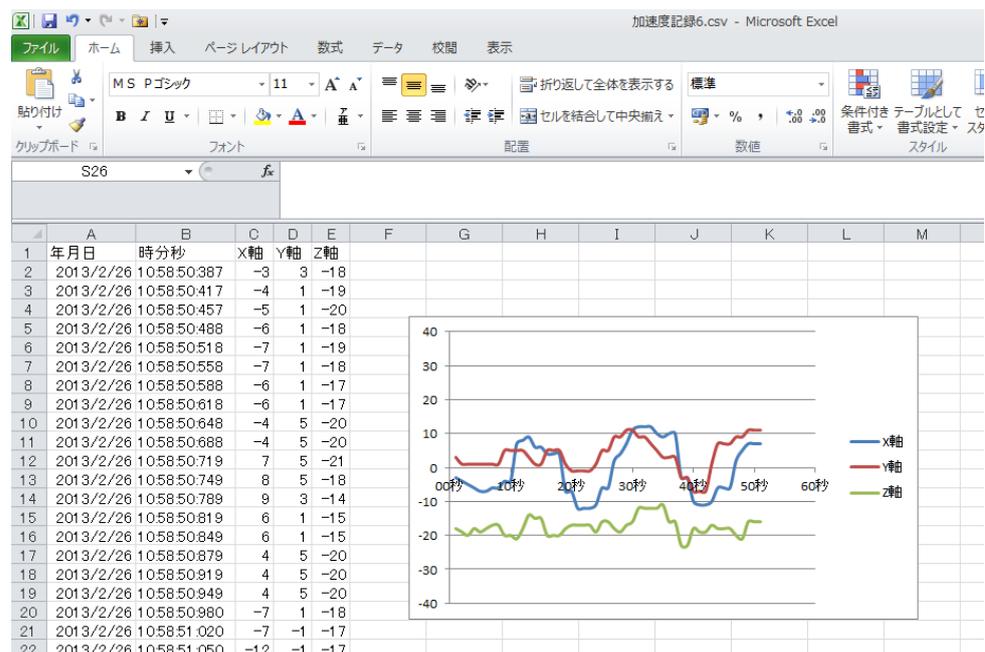
“加速度記録” + “5” + “.csv” = “加速度記録 5.csv”

一度記録終了すると、番号が自動的に加算され、次に記録するファイル名が“加速度記録 6.csv”になります。

CSV 形式で保存したファイルは Microsoft Excel などので開けるので、記録したデータの分析なども可能です。



Csv ファイルは Excel で開ける



データ記録は、例えば「歩く」「起き上がる」など、人の動きを記録して個人差や傾向を求めるといった自由研究に応用できます。なお、記録中に激しく動かしてケーブルが無理に引っ張られたりしないように注意してください。

14. ファームウェアの書き換え

製品の web ページには、「ファームウェア」を公開しています。ファームウェアは本体の制御用プログラムで、本体に機能が追加されたり、バグが修正された場合、新しいものが公開されます。ダウンロードしたファームウェアを本体に書き込むと、それらの追加機能や問題修正を行うことができます。

ファームウェアのダウンロード

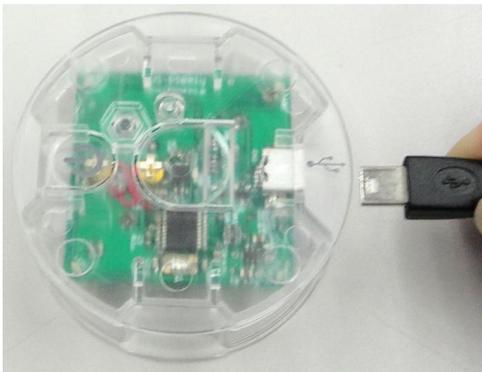
ファームウェアは、製品 web ページの「ダウンロード」で公開しています。ファイルの拡張子は「hex」です。

http://www.vstone.co.jp/products/vs_gs001/download.html

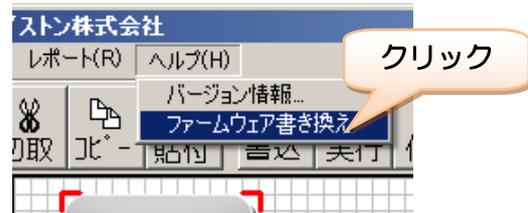
本体への書き込み

ダウンロードしたファームウェアは、次の手順で本体に書き込みます。

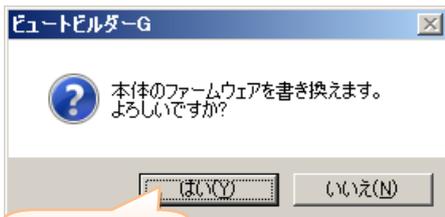
① 本体を PC と接続しましょう



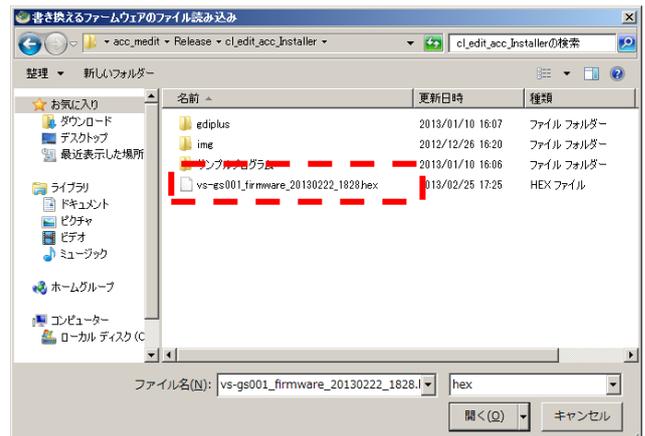
② メニューより「ヘルプ」→「ファームウェア書き換え」をクリックしましょう。



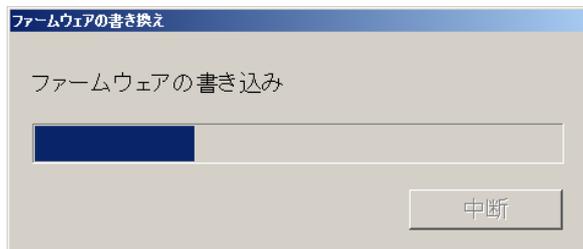
③ 書き込みの確認メッセージを表示したら「はい」をクリックしましょう。



④ Web ページからダウンロードしたファームのファイルを開きましょう。



⑤ 書き込みが完了し、画面が消えるまで待ちましょう。



⑥ 次のメッセージを表示したら、書き込み完了です。OK をクリックしてください



ファームウェアの書き換え中は、ケーブルを外したり本体を動かしたりしないでください。

15. FAQ

ソフトウェアや本体に関する問題・不明点が発生したら、下記事項を一度お確かめください。

Q.ケーブルを正しく接続して電源ボタンを押しても、本体と通信できません。

A.本ソフトウェアを同時に複数起動したり、加速度センサゲームを起動していると、通信処理が競合して正しく通信できない場合があります。他にこれらのソフトを実行していないかご確認ください。

A.お使いのPCによっては、USB 機器の接続監視ソフトの影響で通信できない場合があります。以下の手順でPCの設定を変更してご確認ください。

【USB 接続監視ソフトの終了手順】

◆Windows XP の場合

- ① OS の「スタート」ボタンをクリックし、スタートメニューの「ファイル名を指定して実行」をクリックします。
- ② 表示された入力欄に「msconfig」と入力し、「OK」をクリックします。
- ③ 開いた画面より「全般」→「スタートアップオプションを選択」を選択します。
- ④ 「スタートアップの項目を読み込む」のチェックを外し、「OK」をクリックします。
- ⑤ パソコンを再起動します。

以上で操作は完了です。

※ 上記作業を行った場合、一時的に常駐ソフトが起動しなくなります。元に戻すには上記手順の③の操作の際、「全般」→「通常スタートアップ」を選択し、「OK」をクリックしてください。Windows を再起動すると、元の状態に戻ります。

◆Windows Vista/7 の場合

- ① OS の「スタート」ボタンをクリックし、スタートメニューの「プログラムとファイルの検索」をクリックします。
- ② クリックした入力欄に「msconfig」と入力し、Enter キーを押します。
- ③ 「システム構成」の画面を開くので、「スタートアップ」タブをクリックします。
- ④ 停止させたいソフトの「チェックボックス」をクリックし、チェックを外して「OK」をクリックします。
- ⑤ 「再起動」をクリックし、パソコンを再起動します。

以上で操作は完了です

※ 終了したソフトウェアは、Windows を再起動すると起動します。ソフトウェアの機能を停止、または無効にした場合は、ソフトウェアによって再起動後も機能を停止したままのものが残ります。その場合は、必要に応じて機能を元に戻してください。

※ NEC「ワンタッチスタートボタン」が有効の場合、接続がうまくいかないということがありました。また、タスクトレイに常駐しているボイスレコーダーのユーティリティを終了すると改善されたという報告もありました。

Q.PC と本体の通信ができますが、本体単体で動作しません。

A.電池が正しい向きに入っているか、また、電池の種類や残り電力に問題ないかご確認ください

A.PC と通信中にプログラムを実行することができません。USB ケーブルを抜いて実行できるかご確認ください。

Q.本体を落としたりぶついたりしてケースが割れてしまいました

A.本体の破損については、有償での修理対応となります。説明書末尾の宛先までメールでご連絡ください。

Q.加速度センサの値が極端にずれています

A.加速度センサの調整の設定が大きすぎる可能性があります。一度「加速度センサの設定」より「設定をクリア」を行い、全ての設定値を0に戻してください。

■オプションパーツ、関連商品のご購入は・・・

No.1 の品揃え！ 各種オプションパーツ、ロボット関連製品のご購入はコチラ

<http://www.vstone.co.jp/robotshop/>

楽天・Amazon・Yahoo の各 Web 店舗、または東京、福岡の各ロボットセンター店頭でもロボット関連商品をお買い求めいただけます。

ロボットセンター東京秋葉原店 (東京支店)

〒101-0021

東京都千代田区外神田 1-9-9 内田ビル 4F

ロボットセンターロボスクエア店(福岡支店)

〒814-0001

福岡市早良区百道浜 2-3-2

TNC 放送会館 2F ロボスクエア内

商品に関するお問い合わせ

商品の技術的なご質問は、問題・症状・ご使用の環境などを記載の上メールにてお問い合わせください。

E-mail: infodesk@vstone.co.jp

受付時間 : 10:00~17:00 (土日祝日は除く)

ヴイストーン株式会社

〒555-0012 大阪市西淀川区御幣島 2-15-28

TEL: 06-4808-8701 FAX: 06-4808-8702

Vstone[®]
www.vstone.co.jp