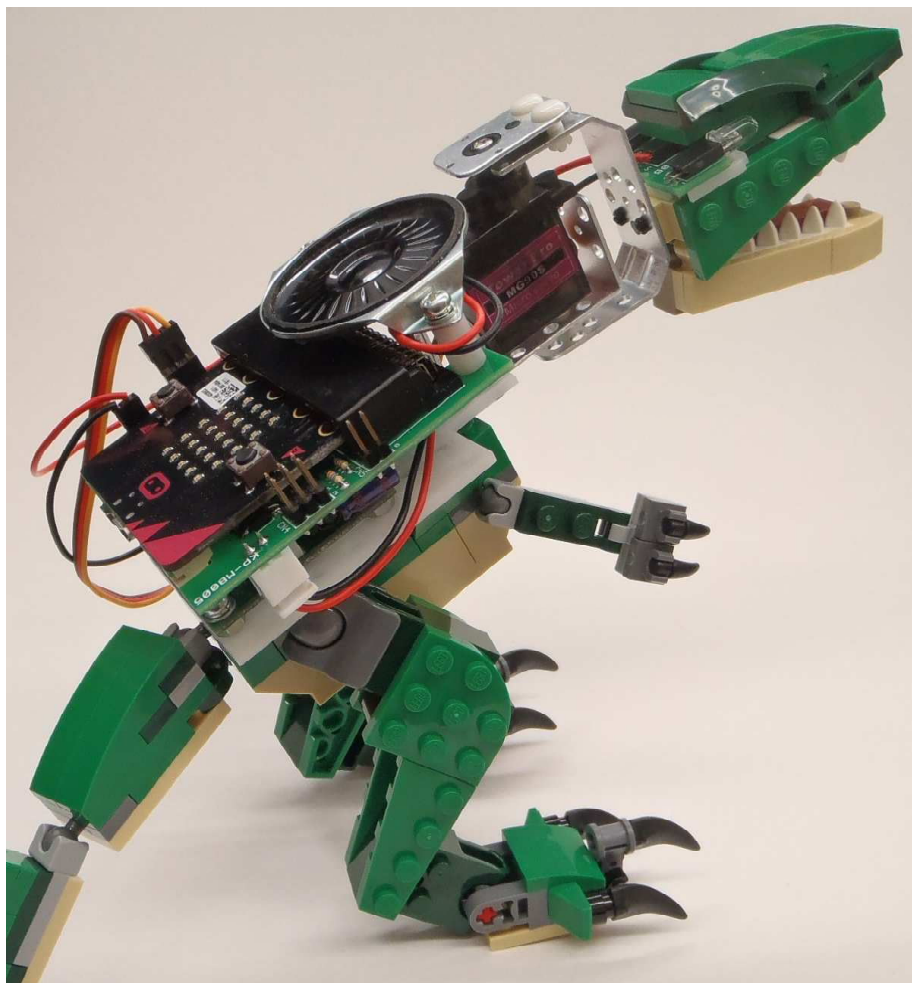


micro:bit用 サーボ&ボイスコントロール基板

micro:bit プログラムガイド

LEGO CREATOR ダイナソー 対応バージョン



※ここではパソコンを用いたプログラム作成方法のみを記載しています。
制御基板の組込、レゴブロックの組立は「組立説明書」を参照ください。

●当製品以外に必要な道具

パソコン(空きUSB端子のあるもの)

※プログラムの製作、micro:bit基板へのプログラム転送、開発に使用します。
WEBブラウザ(Chrome, Firefox など)がインストール済みで、
インターネットへの接続環境が必要です。

microUSBケーブル

※パソコンとの接続に使用します。

●当製品以外に必要なパーツ

電源 単3形電池 3本

※またはUSBコネクタを持ったモバイルバッテリー
(電圧:5V、容量1500mAh以上)

micro:bit 基板 本体

1 micro:bitの使い方

1. micro:bit基板とパソコンの接続

まず、micro:bit基板をパソコンに接続して単体の状態で試します。

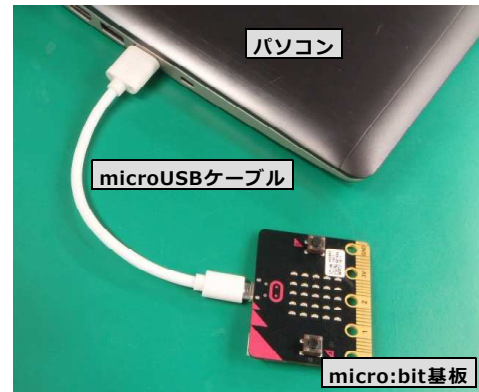
micro:bit基板はmicroUSBケーブルを使用してパソコンと接続します。空いているUSBポートにmicroUSBケーブルを差し込んで写真を参考に接続してください。

パソコンとmicro:bit基板を接続すると、1つのドライブとして認識されます。

※OSによって表示される内容が異なるので注意してください。

必要なもの

- micro:bit基板
- microUSBケーブル
- パソコン



● Windows 7 の場合

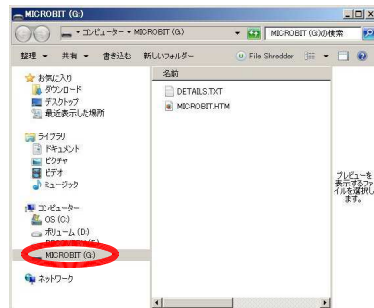
A. 「自動再生」のウィンドウが表示されます。ドライブ名「MICROBIT(?)」



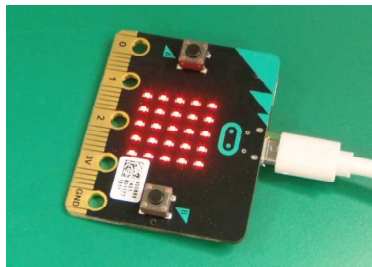
B. 全般オプション

「フォルダを開いてファイルを表示」を選択

C. エクスプローラーが起動して「MICROBIT」という名前のドライブがあれば接続成功です。



D. プログラムが入っていると起動してmicro:bit基板上のLEDが点灯する場合があります。

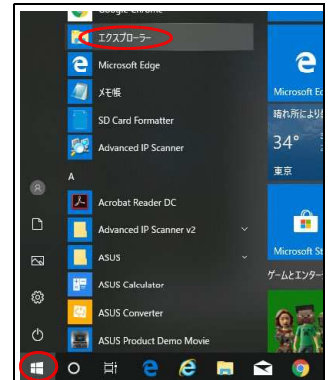


● Windows 10 の場合

A. 右角に数秒間だけ表示が出ます。「MICROBIT(?)」

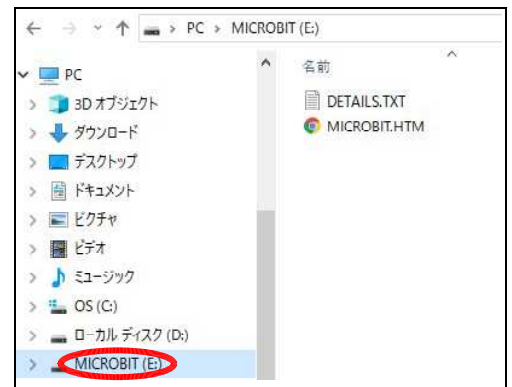


B. 左下角の[スタート]ボタンをクリックしてスタートメニューを表示し「エクスプローラー」を起動してください。



スタートボタン→

C. エクスプローラーが起動して「MICROBIT」という名前のドライブがあれば接続成功です。



D. プログラムが入っていると起動してmicro:bit基板上のLEDが点灯する場合があります。

micro:bit基板の使い方の詳細については公式ページ

「 <http://microbit.org/ja/guide/quick/> 」でご確認ください。

1 micro:bitの使い方

2. 開発ツールの起動

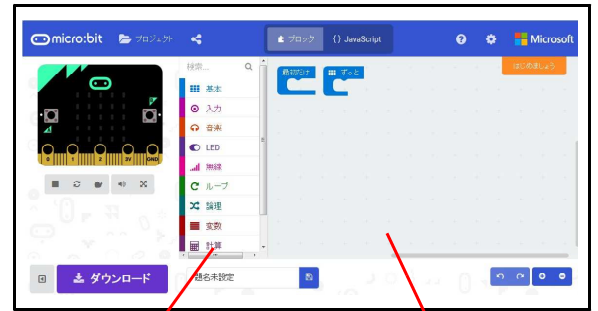
micro:bitのプログラム開発はWEBブラウザを使用して行います。プログラムのインストール作業は必要ありません。

WEBブラウザで下記のURLにアクセスしてください。

<https://makecode.microbit.org/#>

※インターネットへの接続が必要です。

しばらくすると、micro:bitの「コードエディタ」が表示されます。



ツールボックス
(使うブロックを選ぶ)

プログラミングエリア
(ブロックを並べる)

[コードエディタ]

3. プログラムの作成

プログラムの作成はブロックをツールボックスから選んで、つなげ合わせることで行います。まずは「最初だけ」の中にツールボックスからブロックを選んで下図のように並べた後「値」の入力を行ってください。



この部分をクリックして
時間を設定する「値」を入力します。
単位はミリ秒です。

500ミリ秒は0.5秒です。



[基本]をクリックすると右に選択できるブロックの候補が表示されます。

基本 - アイコンを表示

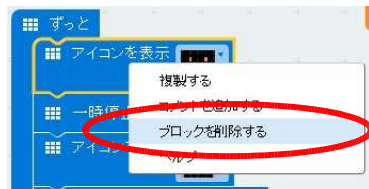
基本 - 一時停止(ミリ秒)

を選びます。

▼をクリックすると「値」(ここでは絵柄)を選択することができます。表示したい絵柄を選んでください。

ブロックのくぼみの部分をあわせると自動的にブロック同士が結合されます。

ブロックを削除したいときはそのブロックを選択して右クリックメニューを出して「ブロックを削除する」を選択します。

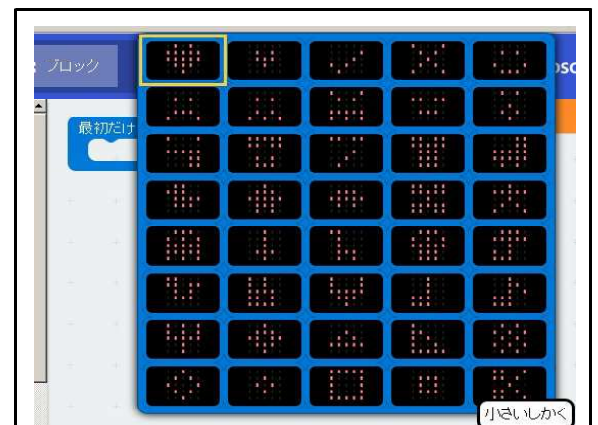


プログラムについて

このプログラムはmicro:bit基板上的LED(5×5)の表示を指定した時間経過ごとに絵柄を変えるプログラムです。

好みの「絵柄」、「時間」に設定してください。

「最初だけ」の中なので電源を入れた後(リセット後)1回のみ実行されます。表示は最後にセットした絵柄のままになります。



1 micro:bitの使い方

4. プログラムの保存と書き込み

①まず作ったプログラムに名前をつけてください。
コードエディタの下部の中央付近の枠の中にプログラムの名前を記入します。例「test」

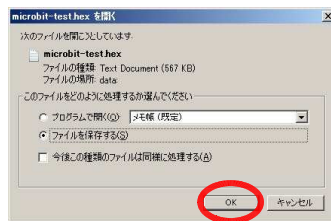
②名前を記入後、「ダウンロード」ボタンをクリックします。

保存が完了するとブラウザが「Chrome」の場合はウィンドウの下部にファイル名が表示されます。

その他ブラウザの場合は「microbit-test.hex を開く」のウィンドウが表示される場合がありますがその場合は「OK」をクリックしてください。

「test」と入力した場合は
ファイル名は「microbit-test.hex」となります。

保存場所は「ダウンロード」フォルダです。



同一のファイル名があった場合はファイル名の後に(数字)が付加されます。

例 -> microbit-test(1).hex



②ダウンロードをクリックすると「ダウンロード」フォルダに保存されます。

①ここにプログラムの名前を入力します



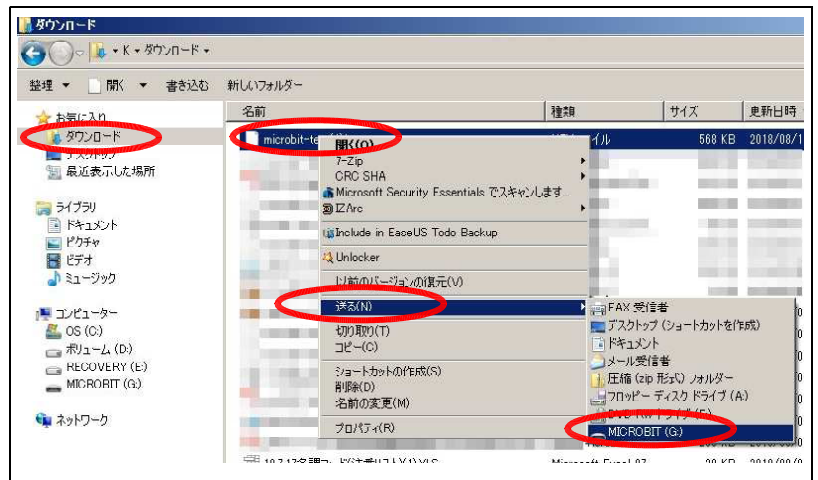
ブラウザが「Chrome」の場合は保存されると下部にファイル名が表示されます。

③エクスプローラなどで「ダウンロード」フォルダを開いてください。

micro:bit基板が、パソコンと接続されているか確認してください。(接続していない場合はここで接続してください)

④保存したファイル(micro:bit基板に書き込みするファイル)を選択して、マウスの右クリックメニューを開きます。

送る(N)を選択->右に出るメニューの中から「MICROBIT(?)」を左クリックで選択します。



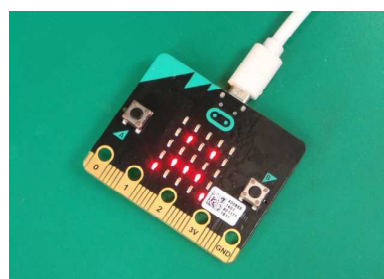
「コピー中」と表示されますので、しばらく待ちます。

書き込みが完了すると、自動的にリセットされ、プログラムが動作します。

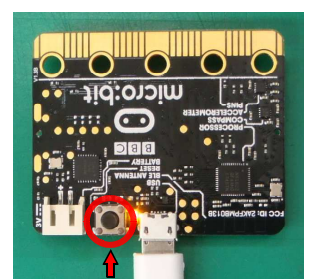
※自動リセット後はパソコンとも再接続されません。

写真のように設定した柄が表示されていればプログラム成功です。

※「リセット」ボタンを押すと、プログラムが再起動し絵柄が切り替わります。



【プログラム実行画面】

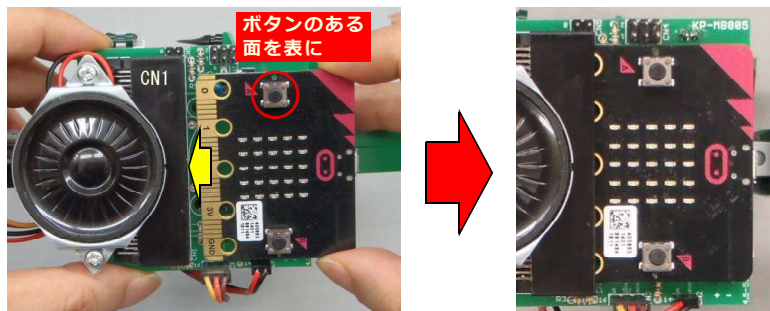


【リセットボタン】

2 プログラミングの準備

1. micro:bit基板を制御基板に取り付けます。

micro:bit基板は制御基板のCN1(黒いコネクタ)に差し込みます。micro:bit基板の穴のあいている側をコネクタに向け、ボタンが上面にある方向でまっすぐに差し込んでください。制御基板の右端とmicro:bit基板の右端が同じ位置になるまで押し込みます。



2. 各デバイスの端子の割り当て番号を確認します。 プログラムを作る際に、どのポート番号に何が接続されているか分かっていることが必要です。

プログラム前に確認しておいてください。

■ サーボモーター出力端子

頭部のサーボモーターに接続されています。
micro:bit基板のポート番号は「P16」です。
高度なブロック>入出力端子>

サーボ 出力する 端子 P16 (出力のみ) 角度 90

で制御します。
回転範囲は中央を中心として
最大±約70~90度です。

※中央位置、動作範囲はサーボモーターの
個体差があります。

■ LED出力端子

頭部のLED基板に接続されています。
micro:bit基板のポート番号は「P14」です。
高度なブロック>入出力端子>

デジタルで出力する 端子 P14 値 0

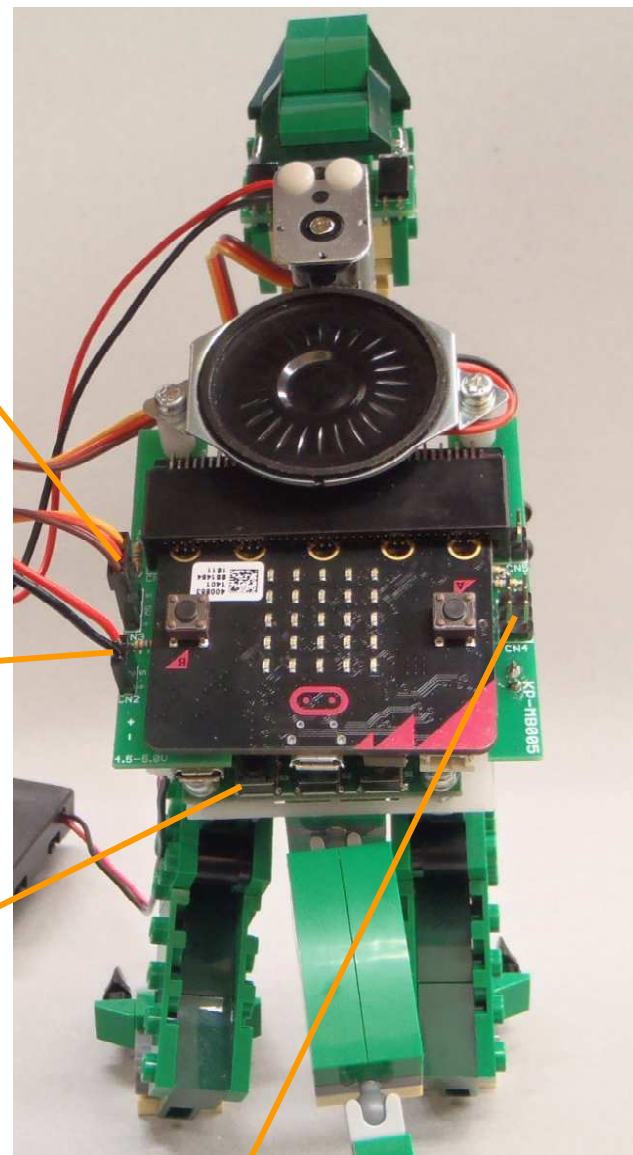
で制御します。
値が「0」で消灯、「1」で点灯です。

■ 録音再生ボード V2

基板間のコネクタでmicro:bit基板に直接接続されています。
micro:bit基板のポート番号は「P8」です。
高度なブロック>入出力端子>

デジタルで出力する 端子 P8 値 1

で制御します。
立ち上がりパルスが発生させて再生タイミングを制御します。
(値を「1」→100ms待ち→「0」に戻す)



■ 汎用入出力端子

「1」にmicro:bit基板のポート番号「P1」が接続されています。
「2」にmicro:bit基板のポート番号「P2」が接続されています。
スイッチ、センサー、サーボモーターの追加など機能を拡張したいときに使用します。

※サーボモーターは同時に3個動作させることが可能です。
※中央の端子は電源電圧(4.5~5.0v)がそのまま出力されます。

3

プログラミング基礎編 LED(目)の点滅

1. LED(目)を点滅させるプログラムを記載します。
下図のようにブロックを並べてください。



クリックして
値を書き換え
「1」で点灯



「入力」をクリックすると選択画面が表示されます。ここでは「ボタン(A)がおされたとき」を選択してください。



「ループ」をクリックすると選択画面が表示されます。ここでは「くりかえし(4回)」を選択してください。



「端子[P0▼]」の箇所は、クリックすると選択画面に切り替わります。ここでは「P14」を選択してください。



サーボ・デジタル関連は、まず「高度なブロック」をクリックします。下に現れる「入出力端子」をクリックすると選択画面が表示されます。ここでは「デジタルで出力する」を選択してください。

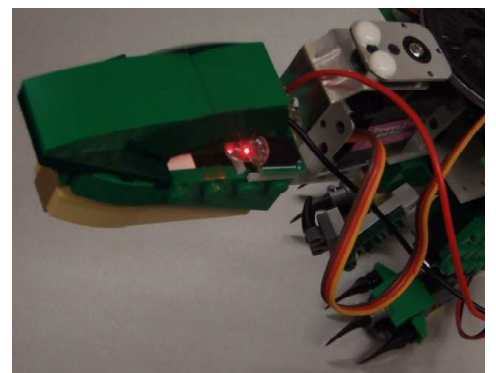
■プログラムについて

ボタン「A」が押されたときに動くプログラムをつくります。

「くりかえし」は、この緑の枠で囲われた「行」を指定回数で繰り返します。

端子「P14」はLED基板が接続された端子番号です。
この値を「0」「1」にすることによって点灯/消灯を制御します。
0：消灯、1：点灯

「一時停止」は時間を空けるタイマーです。
ここでは点滅が見えるように、0.2秒にしています。
「消灯」-0.2秒-「点灯」-0.2秒を4回(1.6秒)間繰り返します。



[LED(目)の点滅]

■プログラムの書き込み

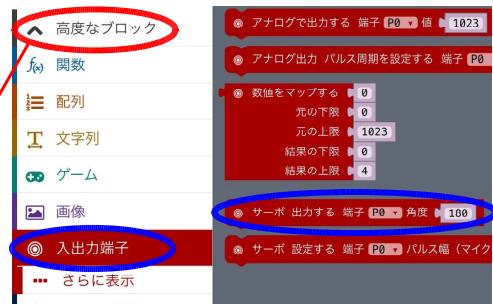
プログラムの完成後、前章の記載手順でプログラムを保存しパソコンとmicroUSBケーブルでmicro:bit基板に接続し書き込みをしてください。

写真のように、LED(目)が点滅すればプログラムの実行成功です。
くりかえし回数や一時停止時間を変えて点滅が変化するか、いろいろ試してみてください。

4

プログラミング基礎編 サーボモーターを動かす

- LED(目)の点滅プログラムの下にサーボモーターの動作プログラムを追記します。
下図のようにブロックを並べてください。



LED(目)の点滅プログラムの下に、「サーボ 出力する」「一時停止」ブロックを4行追加します。

値はそれぞれ「120」「40」に書き換えてください。

「一時停止」はサーボモーターの移動時間の待ち時間です。

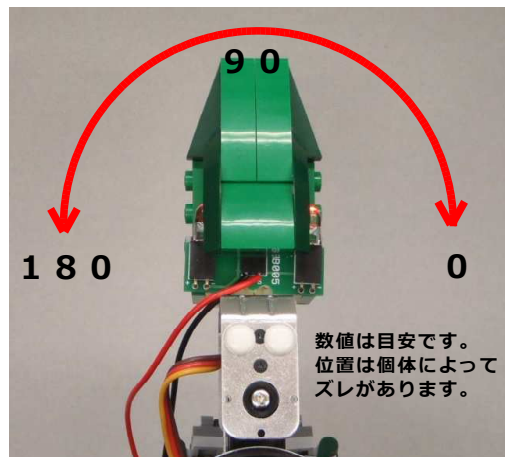
2. サーボモーターの動作範囲について

付属サーボモーターは数値「0」～「180」までを指定可能ですが個体差があります。

「0～20」「160～180」の範囲では個体によっては端に行き当たる場合があります。

端に当たると過負荷の状態になり「ジー」と音が鳴り続け電流がたくさん流れて発熱します。長時間そのままにすると破損することもありますので注意してください。

安全のために「20～160」くらいの範囲で動作させることを推奨します。



3. プログラムの書き込みと実行

プログラムの完成後、先ほどと同じ手順で書き込みしてください。

書き込み完了後、「A」ボタンを押すとLEDが点滅し終了後にサーボモーターが「120」の位置に移動して、すぐに「40」の位置まで移動して止まります。

「A」ボタンを押した後は危険なので顔を近づけたり、不安定な場所に本体を置かないように注意してください。

写真の方向に頭が回転していれば成功です。



[電源スイッチ ON]

サーボモーターの動作には、PW1またはPW2の端子からの電源供給が必要です。電池ボックスの場合はスイッチが「ON」になっているかご確認ください。

注意：「サーボ 出力する」の角度は指定は厳格な角度数値ではありません。接続するサーボモーターによっても変化します。目安程度に考えてください。「1」は正確な1°ではありません。

5 プログラミング基礎編 サーボモーターをゆっくり動かす

1. サーボ出力するブロックではサーボモーターの回転速度は変えられません。「くりかえし」と「変数」を使ってサーボモーターの速度を見かけ上、変化させる(間欠動作させて目標の位置に達するまでの時間を長くする)プログラムを作ります。前章のプログラムの下に下図のようにブロックを追加してください。



■プログラムについて

前章ではサーボモーターの角度の数値を直接入力していましたが、ここでは「変数」(変化する値)を使用します。

まずはサーボモーターの移動開始点をセットします。この開始点(40)と変数値(40)を同じ値にします。

「くりかえし」の中に変数を1だけ増やすが入っているのでループを回るごとに変数の値(角度)が「1」ずつ増えます。

値が増える時間を変化させることで結果として目標の角度になる時間が変化します。

図の場合は「0.025秒」ごとに「1」を増やすとなっているので「100回」繰り返すと、 $0.025秒 \times 100回 = 2.5秒$ です。つまり角度が「40」→「140」(100増やした値)になるまで2.5秒かかることとなります。



ブロックの「角度」の中には「変数」のブロックを入れることができます。



[変数関連のブロック]

変数: 固定的な値ではなく、いろいろな値を代入することのできる箱のようなものです。時間や条件によって「値」を変化させたい場合は「変数」を使用します。複数の変数を使うときは、それぞれ分かりやすいように名前をつけます。

2. プログラムの書き込みと実行

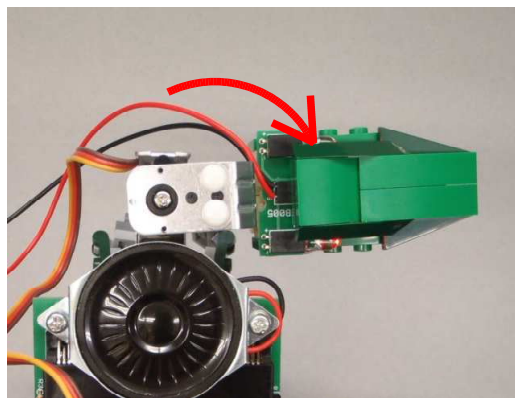
先ほどと同じ手順で書き込みしてください。

書き込み完了後、「A」ボタンを押すとLED点滅終了後に、サーボモーターが回転します。

「A」ボタンを押した後は危険なので顔を近づけたり、不安定場所に本体を置かないように注意してください。今度は、ゆっくりとサーボモーターが回転し最後に通常速度で中央に戻れば成功です。



「変数を()だけ増やす」のブロックの中の数値はマイナスも入れることができます。この場合は90から「-1」ずつ引かれて、70回繰り返し、最後は20になります。プログラムを実行すると、この場合は逆向きにゆっくり回転します。



6

プログラミング基礎編 音を鳴らす

1. 録音再生ボードの制御を行うプログラムを追加します。下図のようにブロックを並べてください。音がすぐ鳴り始めるように「ボタンAが押されたとき」のすぐ下に3行追加します。



■プログラムについて
録音再生ボードに接続された端子(P8)にパルス信号を送ります。

プログラムにより録音再生ボードの再生ボタンを押した状態と同じ状態に電気的に行います。

一旦、端子(P8)を「1」にします。
0.1秒(100ミリ秒)待つ
端子(P8)を「0」に戻します。

※長い時間「1」のままにすると録音再生ボードのモードが切り替わり「再生繰り返しモード」になりますので注意してください。

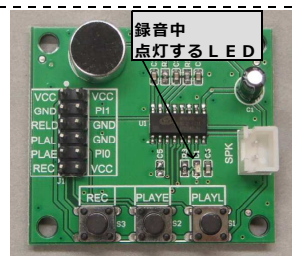
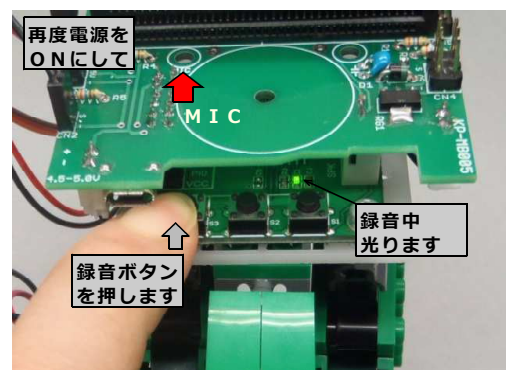
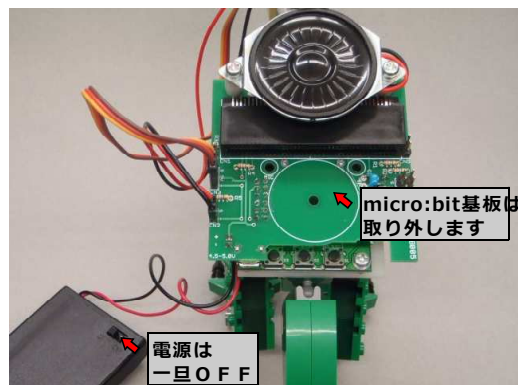
・デジタル出力端子は「P8」を使います。

2. 録音再生ボードに音声を録音します。
下記手順で行ってください。

- ① 電源をOFFにしてmicro:bit基板を取り外します。
- ② 音源(音の鳴るもの)を用意してください。
- ③ 音源のスピーカを基板の「MIC」と記載のある穴位置に近づけてください。
- ④ 録音再生基板の
左端のボタン(S3)を押すと録音が始まります。
録音再生ボードのD1が点灯します。
ボタンを離すと録音を停止します。(D1消灯)
押したままでも録音時間の限界(約40秒)が来ると自動的に停止します。

制御基板の電源をONにして、
音源の再生に合わせて、録音ボタンを押して
離してください。

録音された音の確認は中央の
再生ボタン(S2)を押すことで確認できます。



- ※録音は何度でもできます。最大で約40秒行えます。
- ※制御基板の電源がONのときは一瞬でも録音ボタンに触れると上書きされますので録音後は触らないように注意してください。
- ※micro:bit基板へは「PLAYE」のみ接続されています。

各再生ボタンの仕様は下記です。
「PLAYE(再生E)：1回だけ再生」
※長押しで「再生繰り返しモード」になります。解除は「PLAYL」を押してください。
「PLAYL(再生L)：押している間再生」

3. プログラムの書き込みと実行

録音完了後、micro:bit基板を元に戻して、先ほどと同じ手順で書き込みしてください。
書き込み完了後、「A」ボタンを押して「音」が鳴って先ほどと同じように動けば成功です。

※録音再生ボードの動作には、PW1またはPW2からの電源供給が必要です。
電池ボックスの場合はスイッチが「ON」になっているかご確認ください。



注意：電池が消耗している場合、サーボモーターと同時に使用すると電圧低下(サーボモーターの消費電力が大きいため)によりリセット(音が鳴らない。途中で止まる)される場合があります。

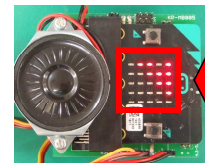
7

プログラミング応用編 明るさセンサーを使う

1. 「プログラミング応用編」では少し難しい上級者向けのプログラムを紹介します。前章までの内容を理解してから進めてください。前章では「ボタンAが押されたとき」を検知して音を鳴らしましたが次は「明るさ」センサーを使い暗くなったら音が鳴るようにします。下図のように「ずっと」の下にブロックを並べてください。



[条件式]



「明るさ」センサーはLEDと兼用になっています。LEDが光っていないのとき「センサー」になります。



■プログラムについて

ボタンAが押されたときのように「明るさ」の値が変化したことを、きっかけにプログラムをスタートさせるブロックはありませんので、この場合は「ずっと」を使って「明るさ」の値を常時見続けるようにプログラムを作ります。

次に明るさが変わったときにプログラムの流れが変わるように「もし～なら」のブロックを使って値の判断をします。

このプログラムでは明るさが「40」より小さくなったとき(暗くなったとき)「音を鳴らすプログラム」を実行します。(条件式が成立している場合に「もし」の中のブロックのプログラムが実行されます)条件が成立しない場合は上に戻って「数を表示」ブロックを実行します。「ずっと」ブロックの中なのでこの動作を繰り返し続けます。

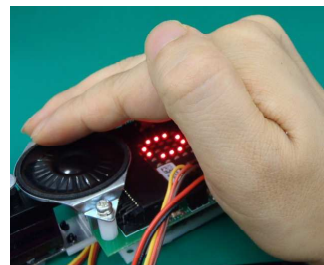
「数を表示」ブロックを使って「明るさ」センサーがどの程度の数値になっているかを表示させています。センサー兼用なので表示中は検知できません。センサーの検知はLEDが表示されていないときに行われています。LED未点灯のときに手でおおってチェックします。

2. プログラムの書き込みと実行

先ほどと同じ手順で書き込みしてください。

書き込み完了後、写真のように基板上のLED部分を手で、おおうと明るさセンサーの値を低くできます。

(写真では[0])暗くして音が鳴ればプログラムの実行成功です。



センサーの数値は「0」～「255」です。暗いとき「0」(約20ルクス以下) 明るいとき「255」(約10000ルクス)

明るさの参考値
蛍光灯照明の室内の明るさ (約500ルクス)
屋外、曇天午前中の明るさ (約25000ルクス)

要調整!

LEDで表示される数値を確認して、環境に応じて数値を調整してください。明るい場所の場合は遮っても隙間から光が入ってなかなか暗くなりません。

手で、おおっているときの数値を見てその値より少し大きめにセットしてください。例えば、おおったときの表示値「60」の場合は、セット値は「70」にします。

8 プログラミング応用編 LED(目)をゆっくり消灯

1. 「アナログで出力する」ブロックを使うことで、LED(目)で消費する電力を制御することができます。「デジタルで出力する」場合は、「1」(ON)か「0」(OFF)しかありませんでしたが、このブロックを使うことでLED(目)の明るさを調整することができます。前章で並べたブロック「ずっと」の中の一番下(音を鳴らすプログラムの下)につなげて下図のようにブロックを並べてください。

追加分

一旦完全点灯

少しずつ数を減らして暗くする

▼部分をクリックして「変数」を選択する。
変数は下記手順で追加。

「値」に「明るさ」を入れる

●変数の追加手順

「変数を追加する」を選択 → 「明るさ」と入力してOKをクリック → 「明るさ」追加完了

■プログラムについて

時間とともに変化する値が必要なのでここでも「変数」を使用します。ここでは分かりやすいように「明るさ」という名前の変数を追加して使用します。

変数(明るさ)の最初の値は完全点灯に近い「1020」をセットします。くりかえしを使用して、0.01秒(10ミリ秒)ごとに値を10ずつ減らしていきます。102回繰り返し後に値は「0」になります。

「アナログで出力する」ブロックの値に変数(明るさ)をセットしているので値に応じてLED(目)は暗くなります。

このプログラムでは、約1秒かけて(10ミリ秒で102回)ON状態からOFF状態になります。

2. プログラムの書き込みと実行

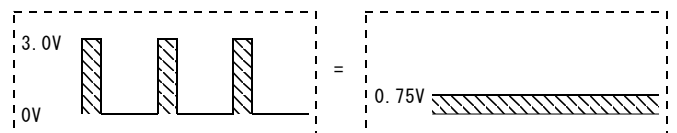
先ほどと同じ手順で書き込みしてください。

書き込み完了後、手でおおって明るさセンサーを暗くすると「もし」の中のプログラムが実行されます。最初に音がなり、LED(目)が1秒間点灯、その後、少しずつ消灯すれば成功です。

「アナログで出力する」ブロックの値には「0~1023」の値を入れることができます。

「アナログ」となっていますが、出力は高速のパルスの繰り返し(パルス幅変調[PWM])となっています。(ON:3.0V,OFF:0V)

値が"255"のときは1/4の時間ON、3/4時間OFFとなりこの端子に0.75Vが出力されている場合と同じ電力(斜線部分の面積は同じ)になります。



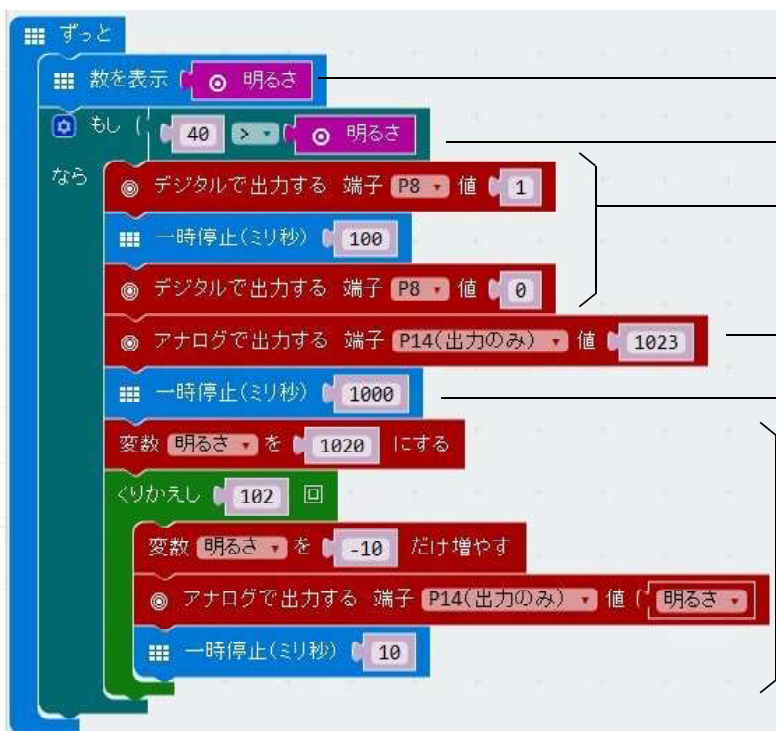
また、値が"511"のときは1/2の時間ON、1/2時間OFFとなりこの端子に1.5Vが出力されている場合と同じ電力になります。

プログラム全体 (最初だけをのぞく)



基礎編の動作 (第3章～第6章)

- ・ ボタンAのチェック
ボタンAが押されたとき
・ 音を鳴らす操作です。
- ・ 「くりかえし」を使ってLED(目)を点滅4回
- ・ 「0」でLED消灯 0.2秒の間消灯
- ・ 「1」でLED点灯 0.2秒の間点灯
- ・ サーボモーターを「120」の位置まで移動
移動時間(0.3秒)待つ
- ・ サーボモーターを「40」の位置まで移動
移動時間(0.3秒)待つ
- ・ 「変数」を使ってサーボモーターをゆっくり動かす。初期値として「40」をセット
- ・ 値を「1」ずつ増やすを100回繰り返す
間欠動作で「140」の位置まで
ゆっくり回す。(40+100=140)
- ・ 繰り返し周期の時間をセット(回る時間の調整)
- ・ サーボモーターを中央位置に移動する
- ・ LED(目)を消灯する
- ・ サーボモーターの移動時間待ち+待機時間



応用編の動作 (第7章～第8章)

- ・ 明るさセンサーの値を表示
- ・ 値が「40」より小さいかチェック
以下は値が「40」より小さいときの処理
・ 音を鳴らす操作です。
- ・ LED(目)を完全点灯します。
(値は0～1023まで)
- ・ 1秒間点灯
- ・ 1秒間かけて
ゆっくりLED(目)を消灯します。
変数名に「明るさ」を使用します。
- ・ 完全消灯後、
明るさセンサー値の表示に戻ります。



micro:bitについては初心者向けの書籍が出版されています。
詳しい使い方などは書籍をご参照ください。

「手づくり工作をうごかそう! micro:bitプログラミング」¥1800
翔泳社 ISBN: 9784798154640 【対象読者】小学校5年生以上



micro:bit基板につきまして

micro:bit基板は、2018年7月の時点で販売されているファームウェアバージョンにて動作確認をしています。将来ファームウェアのバージョン変更があり機能が変更される場合や「KP-MB005」では対応できない場合があることを予めご了承ください。

第2版 2018年 8月 31日

製作：共立電子産業株式会社
共立プロダクツ事業所

〒556-0004
大阪市浪速区日本橋西2-5-1
TEL (06) 6644 - 4447 (代)
FAX (06) 6644 - 4448
URL <http://kyohritsu.com>
EMail wonderkit@kyohritsu.com

●ご使用上の注意

・サーボモーターを動作させるときは顔を近づけないでください。想定外の思わぬ動作でケガをすることがあり危険です。また不安定な場所で動作させないようにしてください。落下、転倒などで破損することがあります。

・長時間使用しないときは電池ボックスから電池を抜いて保管してください。電池を入れたままにしていると液漏れ等で破損することがあります。