

次世代キッズ プログラミング教室 2020

第1回目：2020年11月29日（日）

micro:BITボードについての基礎、基本表示、センサーの利用

第2回目：2020年12月 6日（日）

外部出力、無線通信、おまけ

第3回目：2020年12月20日（日）

総合演習（復習）、みんなが作ったものの発表など

micro:bitを観察しよう！

【micro:bitとは】(公式HP : <https://microbit.org/ja/guide/>)

micro:bitは、2016年にイギリスの公共放送局であるBBCが中心となって開発した小型の教育用コンピューターです。

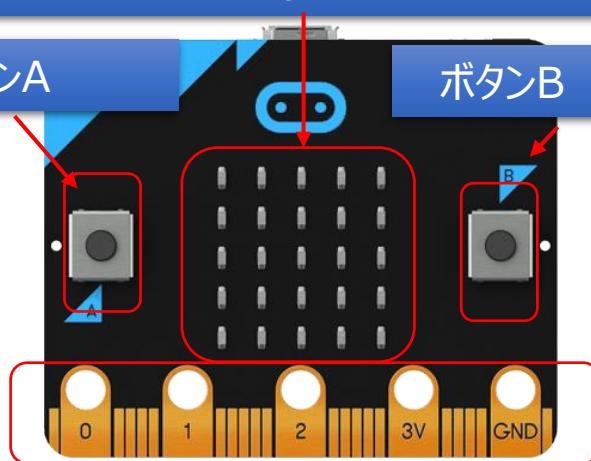
25個の小さな赤色LED・2個のスイッチ・さまざまなセンサー・無線通信機能などを搭載し、プログラムで制御することができます。

【おもて】

LED画面（25個の小さな赤色LED）・
明るさ（光の強さ）センサ

ボタンA

ボタンB



外部入出力端子
(外部機器を制御)

【うら】

無線アンテナ

マイクロプロセッサ
(温度センサ内蔵)

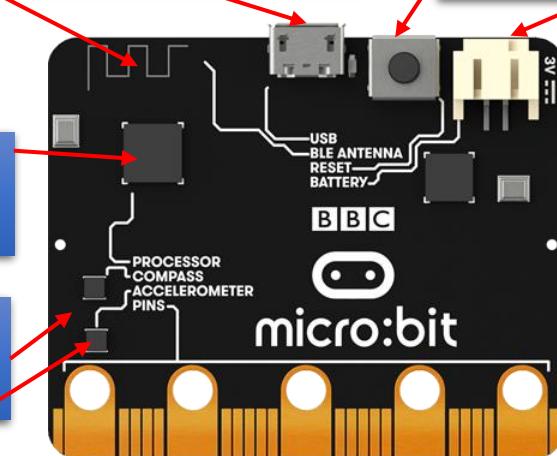
地磁気センサ
(方角がわかる)

加速度センサ
(向き・動きがわかる)

リセットボタン

プログラム書き込み、
電源用USB

電源コネクタ

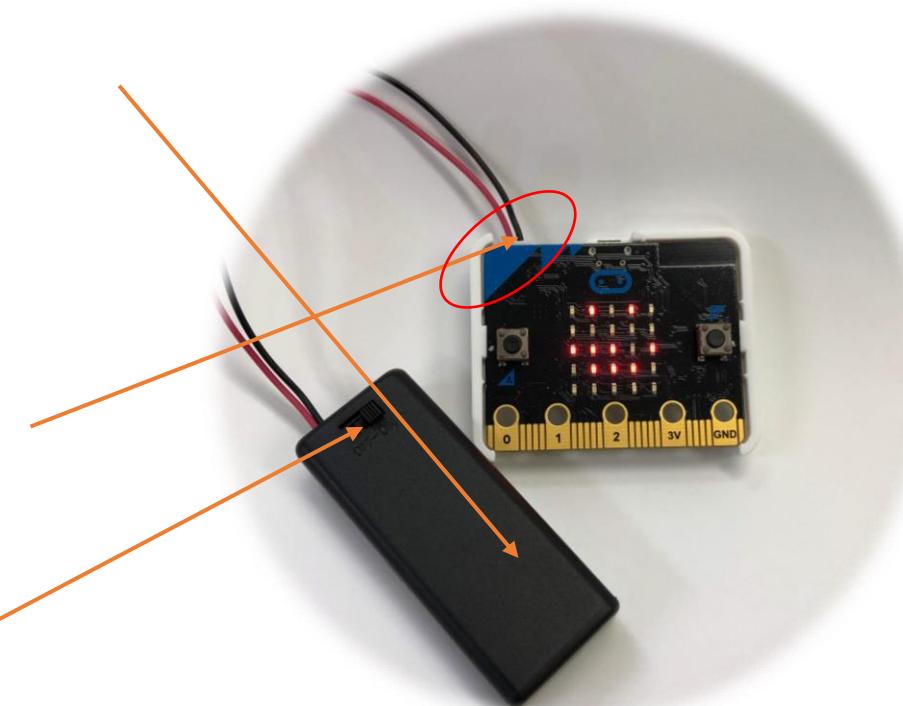


micro:bitキットをセットアップしよう！

①電池ボックスに電池を入れよう

②電源コネクタにつなごう

③電池ボックスをON



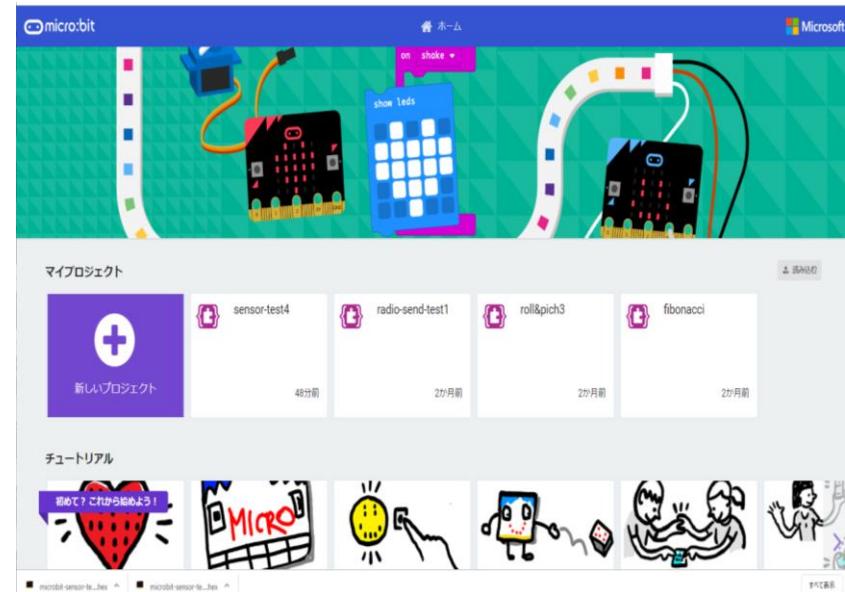
micro:bitは動きだしたかな？

それでは、プログラミングを始めよう！

【準備：MakeCodeエディタを立ち上げます】

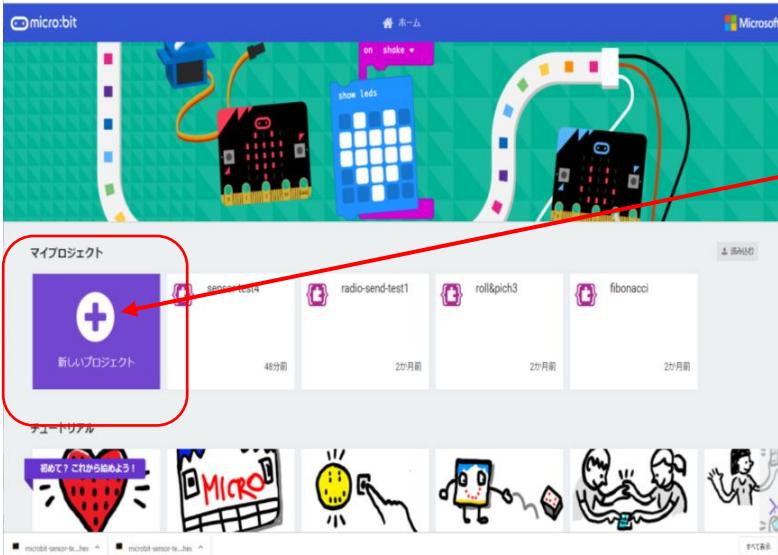
【パソコン（WindowsやMac）の場合】

- ①インターネットに接続していることを確認します。
- ②ブラウザ（Chrome やSafariなど）を立上げます。
- ③`https://makecode.microbit.org/`にアクセスします。



【Windows 10 バージョン 10240.0 以降のパソコンをお持ちなら】

- ①Microsoftから無料プログラムソフト「**MakeCode for micro:bit**」をダウンロードしインストールしておくと便利です。



【ブロック・プログラミング】

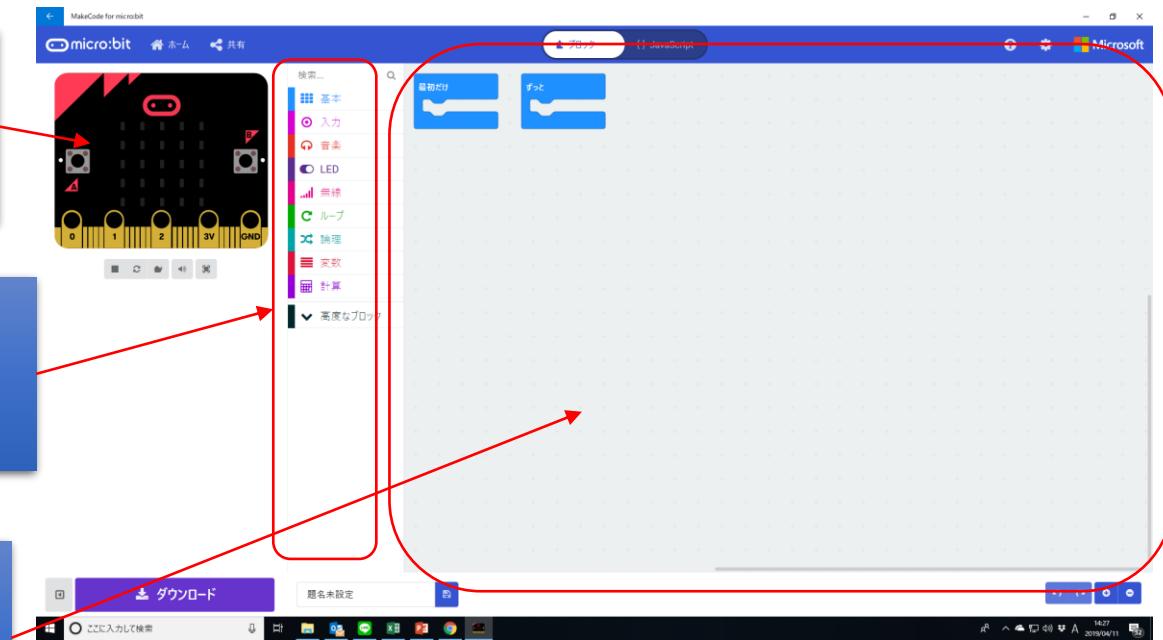
- ①ここをクリックし「新しいプロジェクト」を作ります
- ②ツールボックスからプログラムブロックを選んで、ワークスペースにドラッグし、ならべて、動作させたいプログラムを作っていきます



micro:bitシミュレータ(エミュレータ)
(micro:bitの動作が確認できる)

ツールボックス
(いろいろな命令ブロックメニューがそろっている)

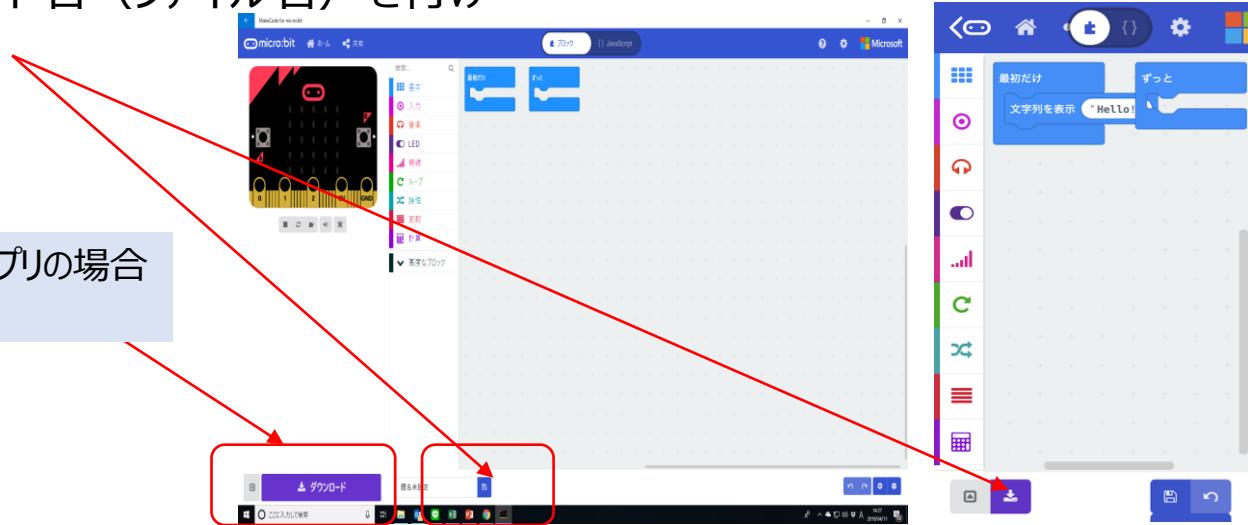
プログラミングエリア(ワークスペース)
(命令ブロックをならべてプログラムを組むところ)



【プログラムを保存します】

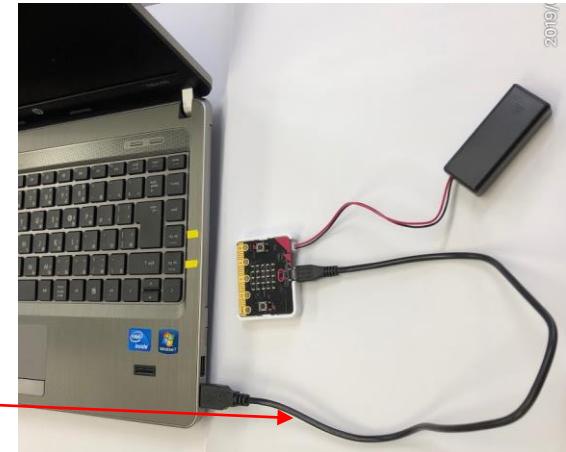
③できたプログラムはプロジェクト名（ファイル名）を付けて、ファイルに保存します

○Windows10のMakeCodeアプリの場合
はダウンロードをクリック



【micro:bitとパソコンを接続しプログラムを micro:bitにダウンロード（書込む）準備をします】

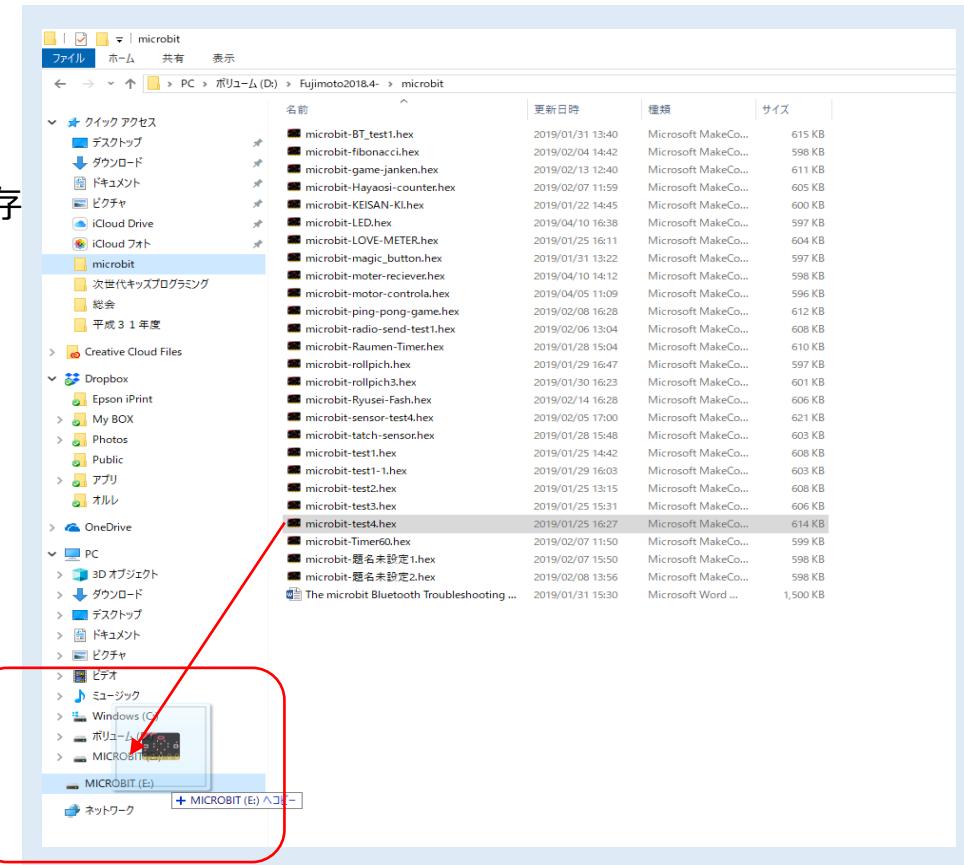
④プログラムが書けたら、USBケーブルでパソコンと micro:BITを接続する



【micro:bitに書き込み、うまく動くかチェックします】

⑤micro:BITにプログラムを書き込む

○Windowsはエクスプローラ、MacはFinderを使い、保存したプログラムファイルをmicro:BITに
ドラッグ・ドロップ



【micro:bitの動作をたしかめて、思い通りに動かなければ
②にもどってプログラムをチェック】

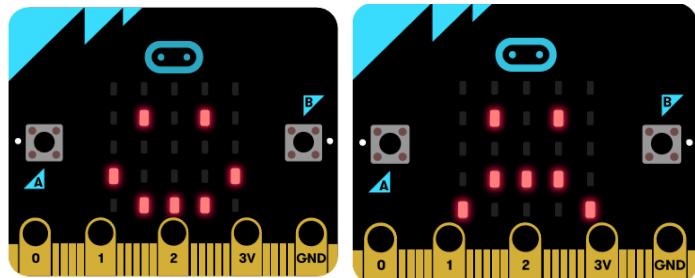
プログラミングを始めてみよう

目標

まずは、プログラムを作りmicro:bitを動かしてみよう

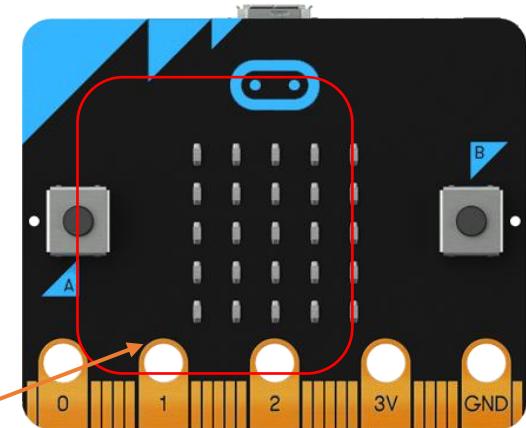
やりたいこと

用意されている顔文字（アイコン）を交互にLED画面に表示させる



「やりたいこと」を分解してみる

LED画面



②アイコンを表示させる

一番目のアイコンを表示

二番目のアイコンを表示

①繰り返して

プログラミング・ブロックで考えると

分解した図をプログラミング・ブロックで考えてみると、「ずっと」と「アイコンを表示」ブロックを組合せて使うことになるよ



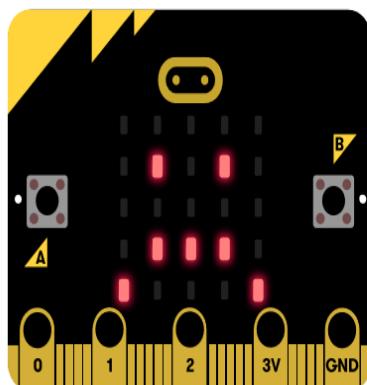
同じことをずっと繰り返すブロックを使って、処理を繰り返させる



「アイコンを表示」ブロックを使って顔文字を表示させる

それでは、早速エディタでプログラミングしてみよう！！

- ①「makecode.microbit.org」へアクセスする。またはアプリを立ち上げる
- ②「新しいプロジェクト」をクリック
- ③エディタ上でブロックプログラミング！ 出来たら名前を付けて保存
- ④できたプログラムファイルをmicro:bitにダウンロード（書込む）



ここで動きをチェック



題名未設定



ここに入力して検索



9:25
2019/04/26

わくわくプログラミング①

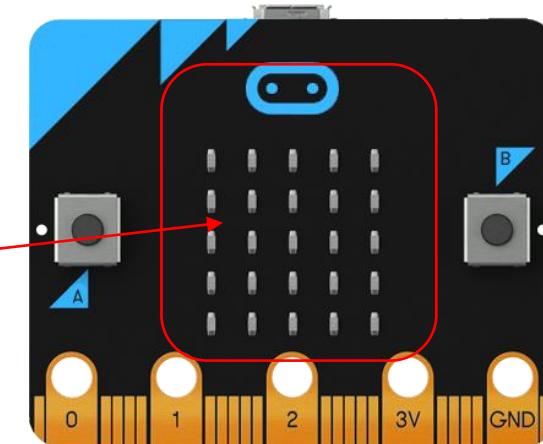
目標 自分のネームプレート（名札）を作つてみよう

やりたいこと

micro:bitのLED画面に、電光掲示板のように繰り返して自分の名前をカタカナで表示させる

「やりたいこと」を分解してみる（タロウの場合）

LED画面



②自分の名前をカタカナで表示

一文字目（タ）を表示させる

二文字目（ロ）を表示させる

三文字目（ウ）を表示させる

①繰り返して

プログラミング・ブロックで考えると

分解した図をプログラミング・ブロックで考えてみると、「ずっと」と「LED画面に表示」を使うことになるよ



→ 同じことをずっと繰り返すブロックを使って、繰り返させる



→ LED画面に表示させるブロックを使って、白いところを点灯させる

それでは、エディタ画面でプログラミングしてみよう！！

- ①「makecode.microbit.org」へアクセス、またはアプリを立ち上げる
- ②「新しいプロジェクト」をクリック
- ③エディタ上でブロックプログラミング！ 出来たら名前を付けて保存
- ④できたプログラムファイルをmicro:bitにダウンロード（書込む）

← MakeCode for micro:bit

micro:bit ホーム 共有

ブロック JavaScript

?

Microsoft

検索...

最初だけ

ずっと

LED画面に表示

LED画面に表示

LED画面に表示

画面ウィンドウの表示例

ダウロード

題名未設定

ここに入力して検索

10:21 2019/04/12

13

チャレンジ① !!!

「基本」メニューの中の「アイコンを表示」ブロックを使って、名前の前後に好きなアイコンを表示させてみよう

チャレンジ② !!!

「基本」メニューの中の「文字列を表示」ブロックを使って、名前をローマ字で表示させて、表示の出方の違いを見てみよう
？文字列に日本語を入れたらどうなる？

チャレンジ③！！

Aボタンを押したときだけ、名前を表示するように変更してみよう

→ 「ずっと」ブロックの代わりに、「**入力**」メニューにある「**ボタンAが押されたとき**」ブロックを使うよ



わくわくプログラミング②

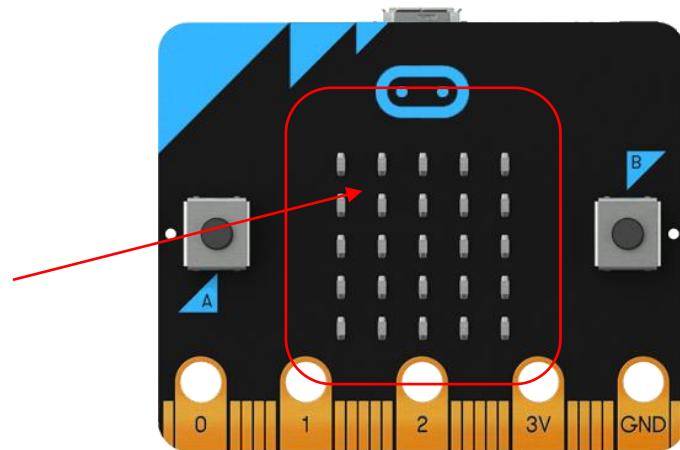
目標 部屋などの明るさを測ってみよう

やりたいこと

micro:bitのLED画面センサがとらえる明るさを、LED画面に表示させる

「やりたいこと」を分解してみる

LED画面
明るさセンサ



①繰り返して

②明るさをLEDに表示させる

明るさの測定（入力）

LED画面に明るさを表示

プログラミング・ブロックで考えると

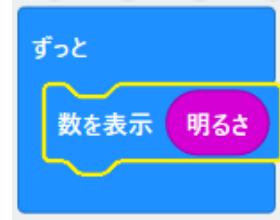
分解した図をプログラミング・ブロックで考えてみると、「ずっと」と「明るさ」「数を表示」を使うことになるよ



→ 同じことをずっと繰り返すブロックを使って、繰り返させる



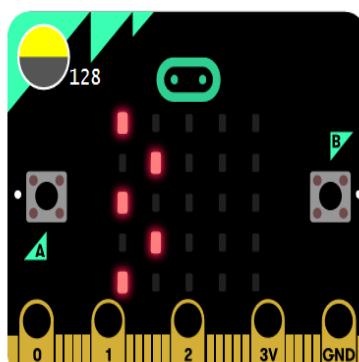
→ 明るさの測定数字が入っているブロックを使って、明るさを測定する
(「入力」メニューにあるけど「変数」あつかいになるから楕円のブロックになっている)



→ LED画面に数字を表示させる

それでは、エディタ画面でプログラミングしてみよう！！

- ①「makecode.microbit.org」へアクセス、またはアプリを立ち上げる
- ②「新しいプロジェクト」をクリック
- ③エディタ上でブロックプログラミング！ 出来たら名前を付けて保存
- ④できたプログラムファイルをmicro:bitにダウンロード（書き込む）



画面ウィンドウの表示例

ダウンロード

題名未設定



チャレンジ①！！

いろいろな場所で明るさを測ってみよう。どんな数字になるかを観察しよう。

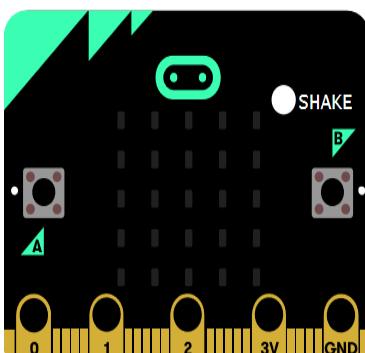
チャレンジ②！！

micro:bitには「温度センサ」や「方角センサ（コンパス）」「加速度センサー」なども入っています。明るさの代わりに温度や方角を測定して、どんな数字になるかを観察しよう。

チャレンジ③！！

「加速度センサー」を使って何か表示させてみよう





検索...



基本

入力

音楽

LED

無線

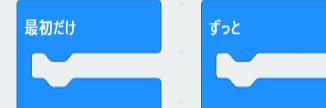
ループ

論理

変数

計算

▼ 高度なブロック



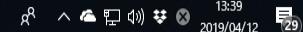
加速度センサーを使ったプログラム例

ダウンロード

題名未設定



ここに入力して検索

13:39
2019/04/12

わくわくプログラミング③

外部機器の制御

目標

明るさ検知懐中電灯を作ろう

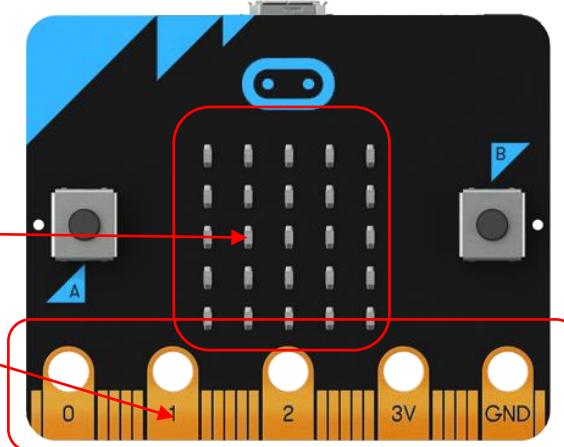
やりたいこと

micro:bitの明るさセンサを利用し、あるレベルに暗くなったら外部出力端子を使って接続したLEDライトを点灯させる

「やりたいこと」を分解してみる

論理分岐や外部入出力など少し高度になるよ

明るさセンサ
外部出力端子



①繰り返して

②明るさを測り評価する

明るさの測定（入力）

明るさはレベルより
暗い

いいえ

はい

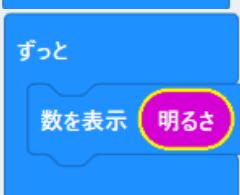
③外部のLEDを点灯させる

端子 0 をON（出力）

プログラミング・ブロックで考えると



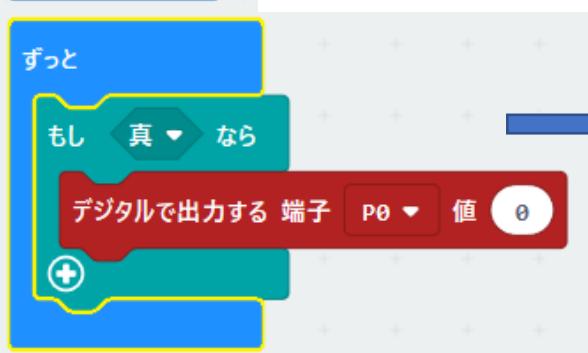
同じことをずっと繰り返すブロックを使って、繰り返させる



明るさの測定数字が入っているブロックを使って、明るさを測定する
(「入力」だけど「変数」になるから楕円のブロックになっている)

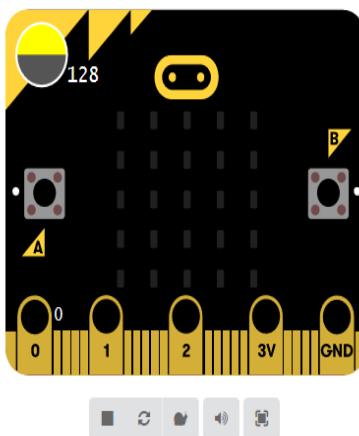


「論理式」メニューの「もし～なら」ブロックで判定、分岐させる



「高度なブロック」にある「入出力」メニューの「デジタルで出力する」で、外部LEDを点灯させる

それでは、エディタ画面でプログラミングしてみよう！！



画面ウィンドウの表示例

ダウンロード

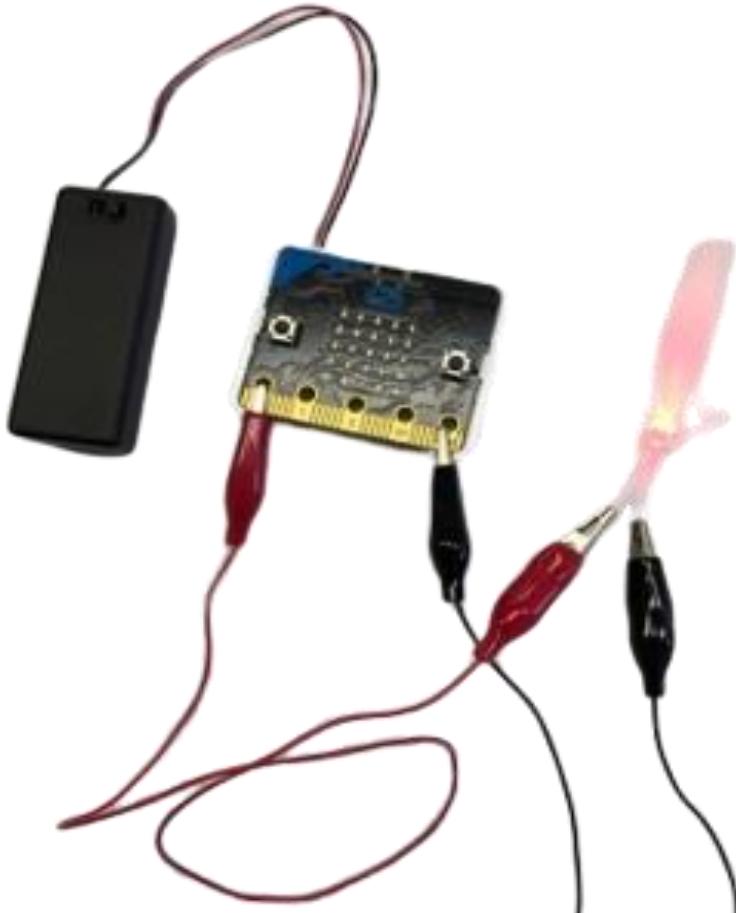
題名未設定



ここに入力して検索



それでは、micro:bitとLEDをミノムシクリップでつないでみよう



① ミノムシクリップ（ワニぐち）で、0番端子とLEDの長い足、GND端子とLEDの短い足をつなぐ

② micro:bitの電源スイッチをON

手でmicro:bitをおおうとLED点灯

LEDの足を逆につなぐと点灯するかな？
どうなるか試してみよう

チャレンジ①！！

光らせる強さを変えて調光ランプにしてみよう（やり方は自由に）

チャレンジ②！！

スピーカーをつないでメロディーを鳴らしてみよう。

チャレンジ③！！

ずっと

もし ボタン A が押されている なら
デジタルで出力する 端子 P0 値 1
でなければもし ボタン B が押されている なら (−)
変数 brightness を 2 だけ増やす
アナログで出力する 端子 P0 値 brightness を 0 以上 1023 以下の範囲に制限
でなければ (−)
デジタルで出力する 端子 P0 値 0
変数 brightness を 0 にする

チャレンジ①のプログラム例

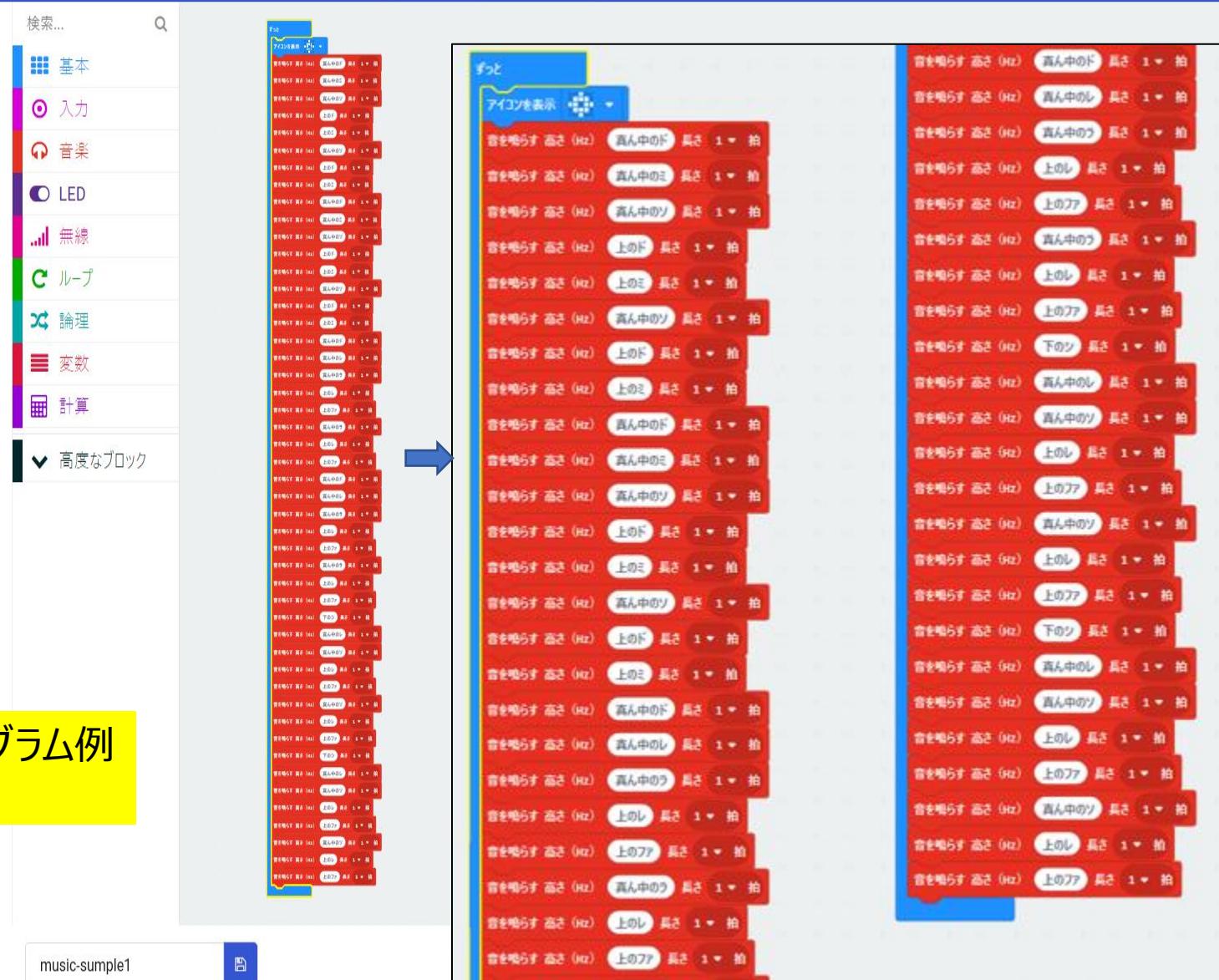
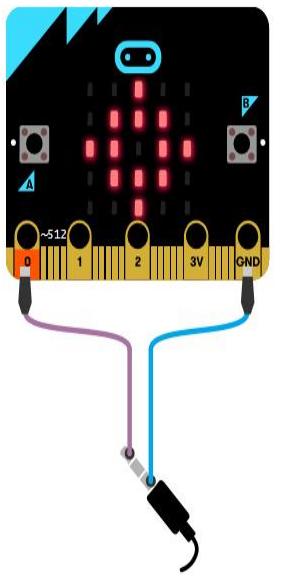
最初だけ

変数 おと を 0 にする
音を鳴らす端子を P0 にする

チャレンジ②のプログラム例

ずっと

もし おと < 1000 なら
音を鳴らす 高さ (Hz) おと 長さ 1 拍
変数 おと を 10 だけ増やす



チャレンジ③のプログラム例 メロディーを作る

わくわくプログラミング④ 無線に挑戦

目標

最後に2台のmicro:bitで無線通信じゃんけんゲームを作ろう

やりたいこと

2台のmicro:bitのボタンBを同時に押して無線通信でじゃんけんをし、勝ち、負け、あいこの判定をする。ボタンAで勝ち数を表示させる

無線通信の手始めに

①ボタンを押したら相手のmicro:bitに自分の名前文字や数字を表示させてみよう

②数字や文字でなく、自分のアイコンを表示させるにはどうすればいいのかな？

たくさんの論理分岐や無線通信などを使ってだいぶ高度になるよ

プログラミング・ブロックで考えると



①無線通信をするmicro:bitのグループを作る（最初に1回だけ）

②無線でデータを数字や文字を送信する
ここでは、「ボタンが押されたとき」イベントブロックを使っている

③「無線で受信したときの」イベントブロックを使って、受信した数字や文字を表示させる

それでは、エディタ画面でプログラミングしてみよう！！

最初だけ

無線サンプルプログラム 1

無線のグループを設定

0

ボタン A ▾ が押されたとき

無線で文字列を送信 " TARO "

receivedString

文字列を受信する無線ブロック専用の変数

receivedNumber

数字を受信する無線ブロック専用の変数

無線で受信したとき receivedString

文字列を表示 receivedString

数を表示 受信したパケットの 信号強度 ▾

最初だけ

無線のグループを設定

1

無線サンプルプログラム2

ボタン A ▼ が押されたとき

無線で数値を送信

0

ボタン B ▼ が押されたとき

無線で数値を送信

1

無線で受信したとき receivedNumber

もし receivedNumber = 0 なら

アイコンを表示



でなければもし receivedNumber = 1 なら

アイコンを表示



①

ゲーム作成にチャレンジ！！

①サイコロ対戦ゲームを作つてみよう

②「ジャンケン」ゲームを作つてみよう

最初だけ

無線のグループを設定

1

ボタン A ▾ が押されたとき

変数 saikoro ▾ を 1 から 6 までの乱数 にする

数を表示 saikoro ▾

無線で数値を送信 saikoro ▾

一時停止 (ミリ秒) 100 ▾

無線サンプルプログラム3 (サイコロ対戦ゲーム)

無線で受信したとき receivedNumber

もし receivedNumber < ▾ saikoro ▾ なら

アイコンを表示



でなければもし receivedNumber = ▾ saikoro ▾ なら -

アイコンを表示



でなければ

-

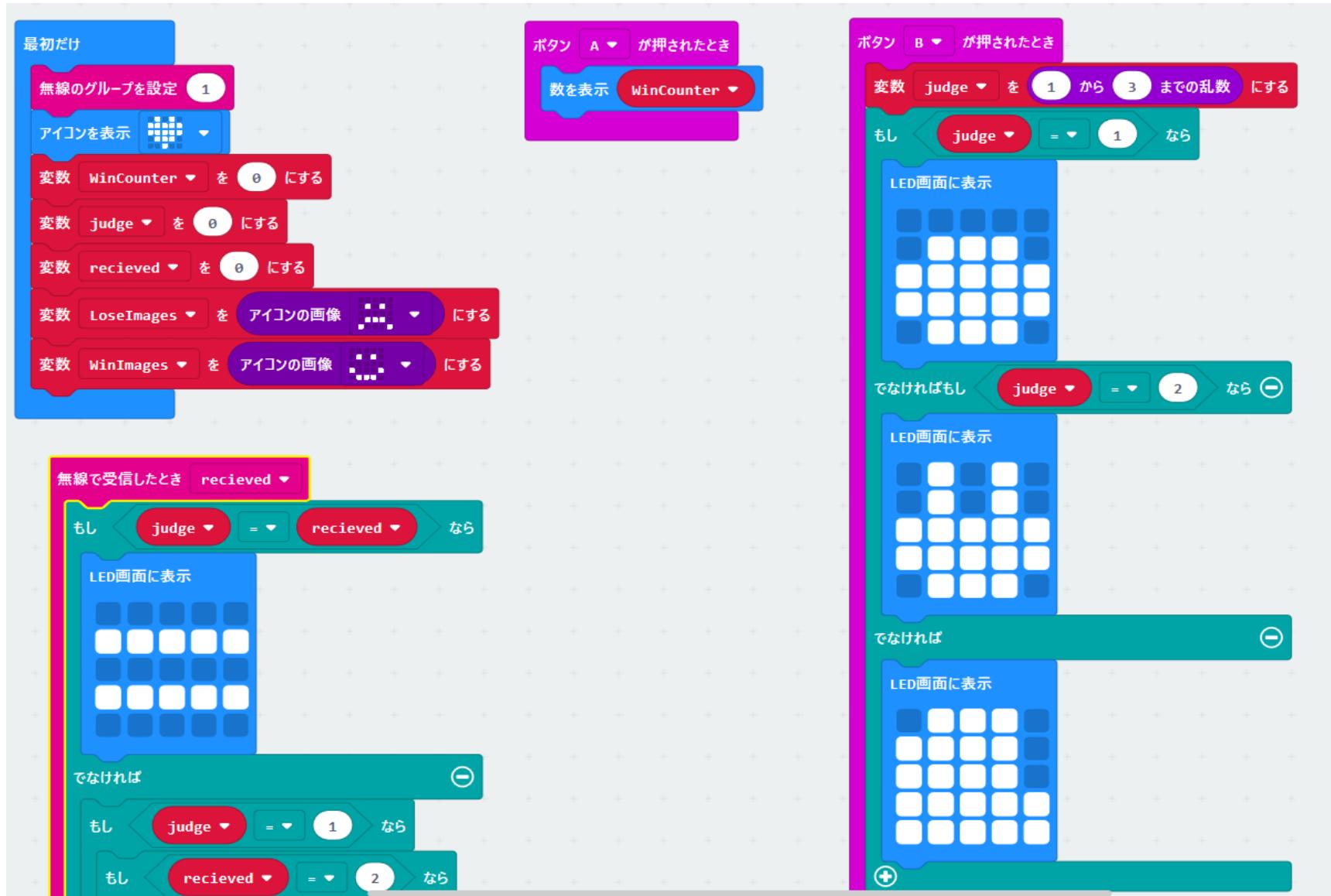
アイコンを表示

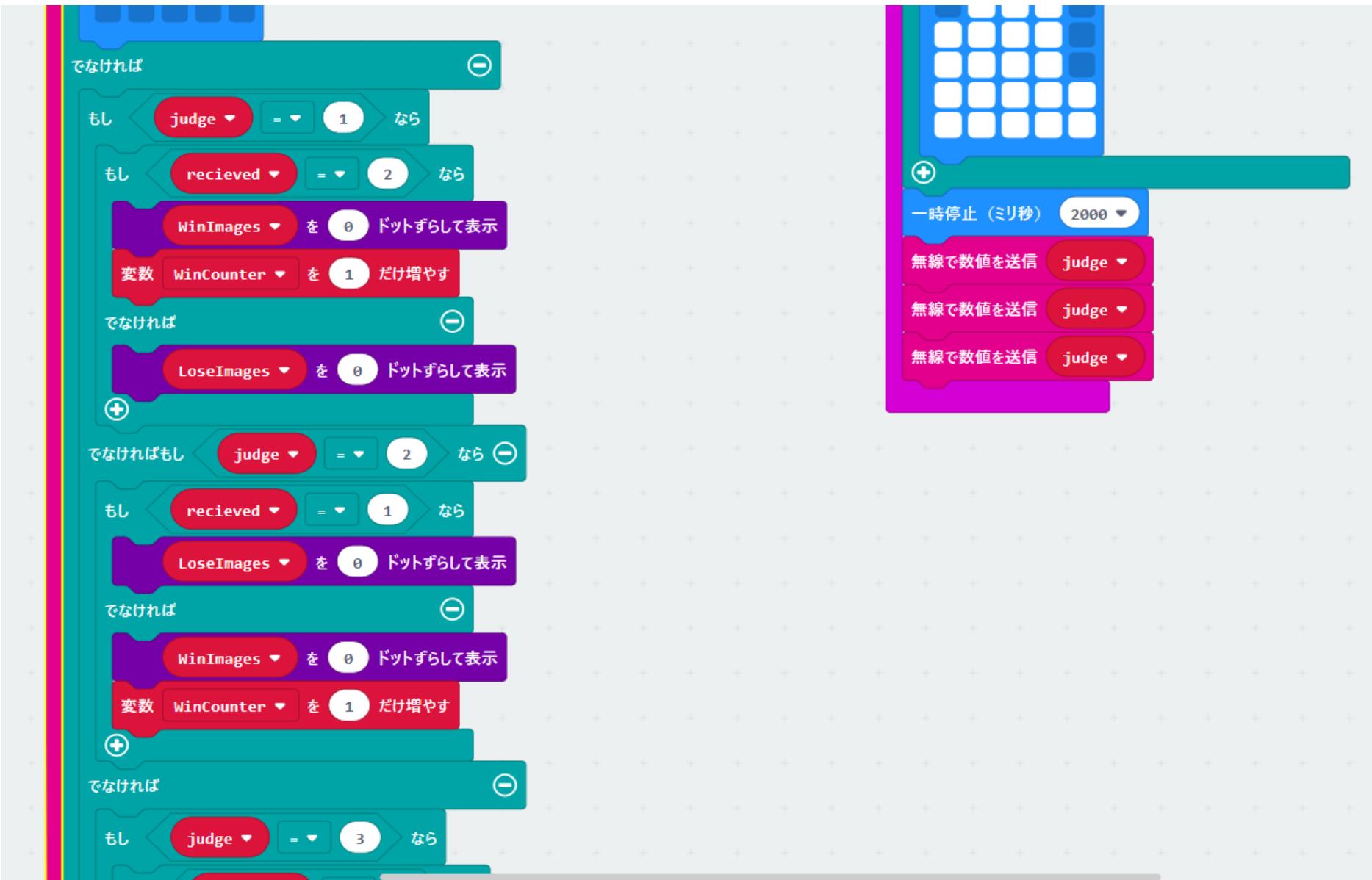


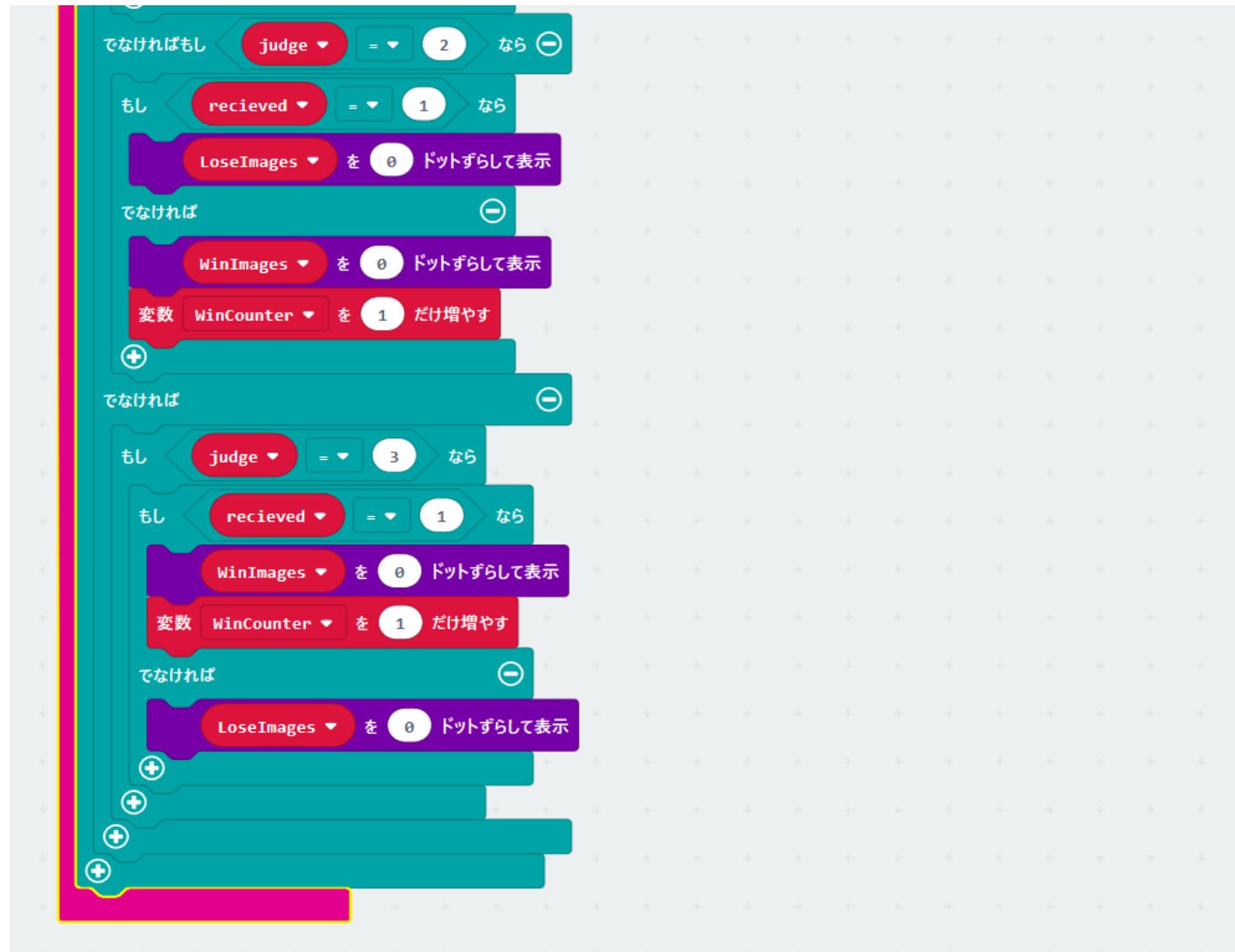
+

無線サンプルプログラム4 (ジャンケン ゲーム)

https://makecode.microbit.org/_KeeFpjTU1eVX







サポートしてくれた鞍手高校のお兄さんたち から、プログラム作品のプレゼント！！

(PIN-PON GAME)のプレゼント

https://makecode.microbit.org/_eH6Fe5eou21q

○makeCodeエディタのホーム画面右 読み込む からURLを読み込んでみてください

最後に、自分でいろいろ調べ、何か作って、 第3回目でみんなに見せてみよう！！

(プログラミングの参考ホームページ)

micro:BITのオリジナルホームページ <https://microbit.org/ja/>
サヌキテックネット <https://sanuki-tech.net/micro-bit/>

(ネット通販で関連部品を購入するには)

スイッチサイエンス <https://www.switch-science.com/>
Amazonやモノタロウ などでも部品が手に入る

(いろんな質問、疑問があつたら)

<https://adox.jp/kids-programming/>
の「プログラミング教室についてのお問い合わせから」メールで聞いてください。

もっと複雑なことができるプログラミングについて 勉強してみよう

○Scratch : スクラッチ

子ども向けのビジュアルプログラミング言語【Scratch(スクラッチ)】が作れる。

小さい子供でもPCだけあれば視覚的にプログラミングを学習可能。

○Python : パイソン→Micro Python

Python の利用場面は多岐に渡っていて、Web アプリケーションだけでなく、中でも機械学習やディープラーニング用のライブラリが Python には充実している事から、最近ではAIやデータ解析処理の場面において利用され注目度の高い言語です。

○JAVA : ジャバ

JavaScript の主な用途は HTML と合わせて使用して動きのある WEB ページを開発するために使用します。

・HTML:Hyper Text Markup Language(ハイパーテキストマークアップランゲージ)の頭文字をとったもので、「Webページを作成するときに使われるコンピューター言語」です。

○Ruby : ルビー

Ruby とは、日本人により開発されたプログラミング言語であり可読性を重視した構文になっていることが特徴です。

○C : シー

C はシステムの共通言語として様々なプロットフォームで使用されています。

C で作成したプログラムはどのプログラム言語よりも高速に動作させることができます。

いろいろなプログラミング言語があるから、何か一つ勉強してみよう



Category すべて開く

新商品 (116)

- ④ [スイッチサイエンス製品\(288\)](#)
- ④ [スイッチエデュケーション製品\(46\)](#)
- ④ [スイッチサイエンスマーケットプレイス \(委託商品\) \(564\)](#)
- ④ [Rapiro\(26\)](#)
- ④ [MakerBot\(65\)](#)
- ④ [Arduino\(286\)](#)
- ④ [SparkFun\(479\)](#)
- ④ [Seed\(207\)](#)
- ④ [Adafruit\(268\)](#)
- ④ [Pololu\(157\)](#)
- ④ [Pimoroni\(41\)](#)
- ④ [Kitronik\(21\)](#)
 - > micro:bit関連製品(21)
- ④ [Digi International\(21\)](#)
- ④ [Raspberry Pi\(260\)](#)
- ④ [Mbed\(75\)](#)
- ④ [Intel\(39\)](#)
 - > SPRESENSE(10)
- ④ [micro:bit\(97\)](#)
 - ④ [純正品・互換機\(21\)](#)
 - ④ [拡張基板\(47\)](#)
 - ④ [その他\(49\)](#)
- ④ [DFRobot\(1050\)](#)
- ④ [STMicroelectronics\(29\)](#)
- ④ [M5Stack\(55\)](#)
- ④ [KORG\(15\)](#)
- ④ [littleBits\(38\)](#)
- ④ [MESH\(25\)](#)
- ④ [FeliCa・NFC\(16\)](#)
- ④ [XBee\(49\)](#)
- ④ [Farnell\(10\)](#)

BBC micro:bit用プロトタイピングセット

仕入先変更により、本製品の型番を変更しました。

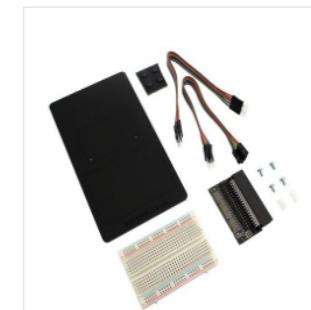
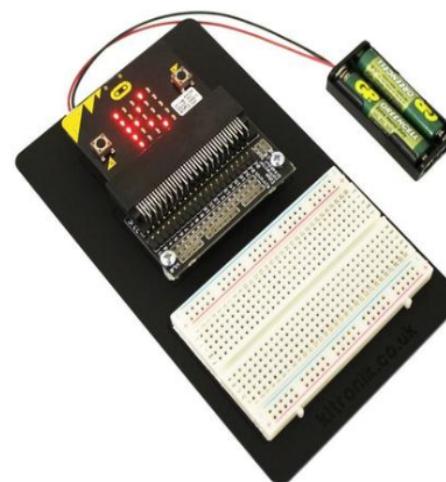
旧型番 : PIMORONI-KIT5609

回路を組んだり、実験することが容易になる、BBC micro:bit用のプロトタイピングセットです。

エッジコネクタピッチ変換基板には、3 Vおよび0 V (GND) のラインと、どこにも接続されていない3箇所のプロトタイピングエリアを用意しています。これらにより、スイッチやセンサ、ブルアップ抵抗、ブルダウン抵抗などが簡単に着脱可能です。エッジコネクタピッチ変換基板の詳細は[こちらを参照ください](#)。

同梱のブレッドボードと併せて使用することで、付属品のジャンパワイヤを使って追加部品の実装が簡単に行えます（はんだ付けは不要です）。

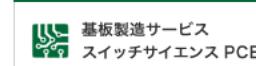
利用例



名前	BBC micro:bit用プロトタイピングセット
コード番号	KITRONIK-5609
SKU#	3179
送料区分	<u>500</u>
税込単価	2,084 円
数量	<input type="text" value="1"/> カートに追加
在庫	多数 (即日出荷可能)
次回入荷	未定
短縮URL	ssci.to/3179
公開日	2017年3月29日

[ツイート](#)

[いいね！0](#)

**Category** すべて開く**新商品** (120)

- [スイッチサイエンス製品](#)(288)
- [スイッチエデュケーション製品](#)(47)
- [スイッチサイエンスマーケット](#)ブレイス (委託商品) (564)
- [Rapiro\(26\)](#)
- [MakerBot\(65\)](#)
- [Arduino\(287\)](#)
- [SparkFun\(480\)](#)
- [Seeed\(207\)](#)
- [Adafruit\(272\)](#)
- [Pololu\(157\)](#)
- [Pimoroni\(41\)](#)
- [Kitronik\(21\)](#)
 - > [micro:bit関連製品](#)(21)
- [Digi International\(21\)](#)
- [Raspberry Pi\(260\)](#)
- [Mbed\(75\)](#)
- [Intel\(39\)](#)
 - > [SPRESENSE](#)(10)
- [micro:bit](#)
 - [純正品・互換機](#)(21)
 - > [拡張基板](#)(47)
 - > [その他](#)(50)
- [DFRobot\(1050\)](#)
- [STMicroelectronics\(29\)](#)
- [M5Stack\(55\)](#)
- [KORG\(15\)](#)
- [littleBits\(38\)](#)
- [MESH\(25\)](#)
- [FeliCa・NFC\(16\)](#)
- [XBee\(49\)](#)
- [Ennacraft\(50\)](#)

micro:bit用エッジコネクタピッチ変換基板

仕入先変更により、本製品の型番を変更しました。

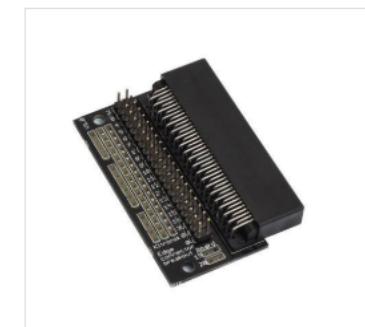
旧型番 : PIMORONI-KIT5601B

micro:bit下部のエッジコネクタ主要端子全てにアクセスすることを可能にする、ピッチ変換基板です。ジャンパワイヤを介して、回路やハードウェアの実装などが簡単にできます。スイッチサイエンス製chibi:bitでも利用可能です。

追加のI/Oライン、A、Bボタンへのアクセス、LEDマトリックス、I²Cバスなど、micro:bitの各端子に関する詳細は、[データシート](#)をご参照ください。SCLとSDA端子は、見分けやすくするため、ボードの端に引き出しています。

当製品には、3 Vおよび0 V (GND) のラインと、どこにも接続されていない3箇所のプロトタイピングエリアを用意しています。これらにより、スイッチやセンサ、プルアップ抵抗、プルダウン抵抗などが簡単に着脱可能です。

本基板をご使用になる際、下記のようにBBC micro:bitをしっかりとコネクタに挿入してください（少し力を入れて押し込む必要があります）。



名前	micro:bit用エッジコネクタピッ... チ変換基板
コード番号	KITRONIK-5601B
SKU#	3181
送料区分	150
税込単価	831 円
数量	<input type="text" value="1"/> カートに追加
在庫	多数 (即日出荷可能)
次回入荷	未定
短縮URL	ssci.to/3181
公開日	2017年3月29日

[ツイート](#)[いいね！ 0](#)