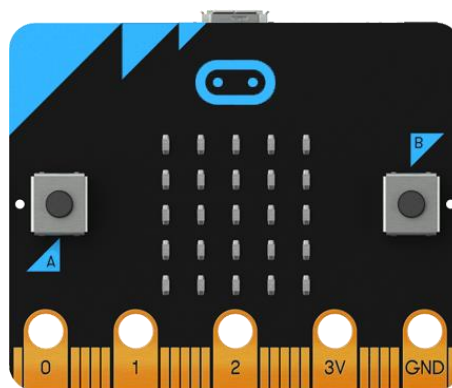


ゆめほたる環境科学技術塾

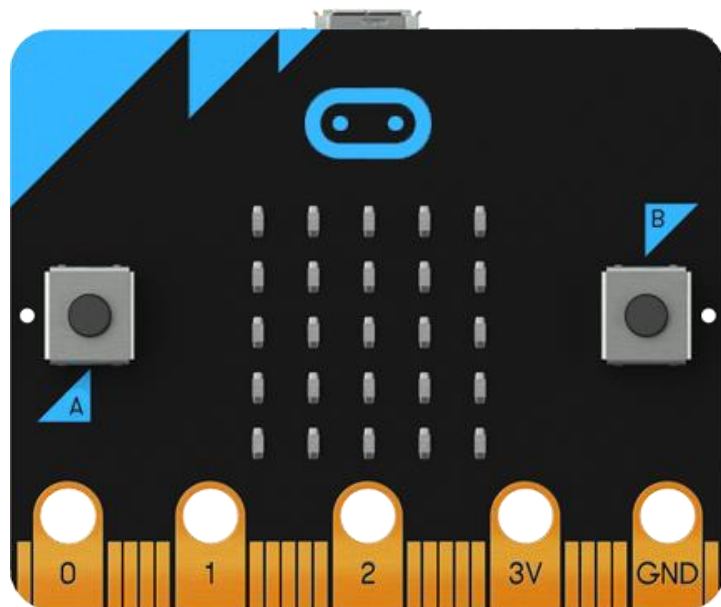
micro:bitプログラミング ～基礎～



ゆめほたる環境科学技術クラブ

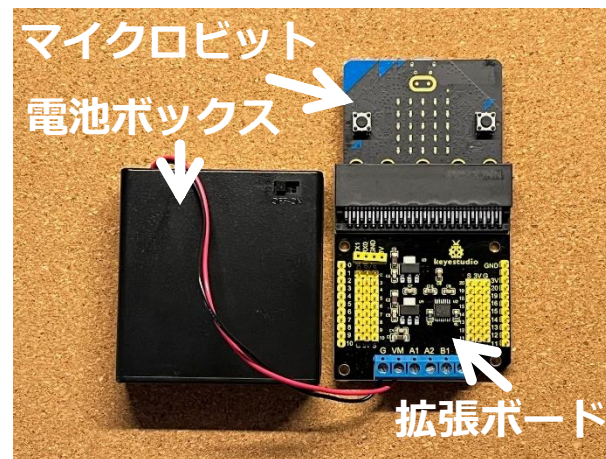
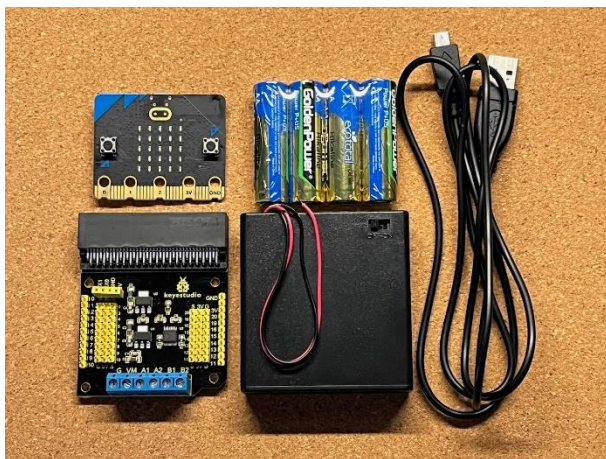
マイクロビットについて

- マイクロビットは、ちいさなコンピュータです。
- パソコンでつくったプログラムを、マイクロビットに入れてうごかします。
- LEDやセンサなどの部品がついていて、いろいろなプログラムをうごかすことができます。



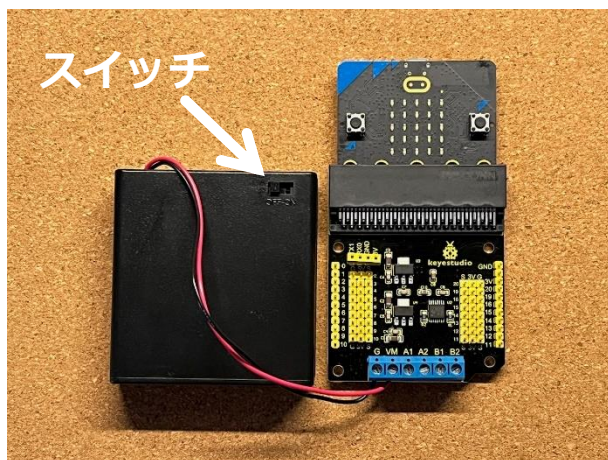
マイクロビットについて

- 今回のセットにはマイクロビット本体のほかに、拡張ボードや電池ボックスなどがはいつています。
- 拡張ボードと電池ボックスはマイクロビットに取り付けてあります。
- 電池ボックスのフタをあけて電池をいれてください。プラスとマイナスをまちがえないようにしてください。



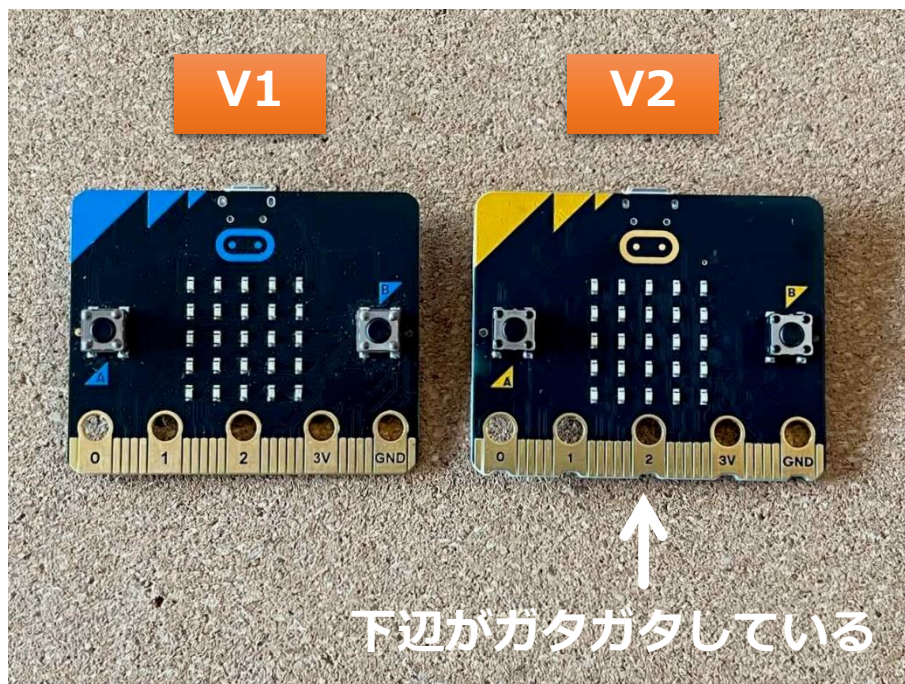
マイクロビットについて

- つかわないときは電池ボックスのスイッチをオフにしてください。
- パソコンからマイクロビットにプログラムをかきこむときには、パソコンとマイクロビットをUSBケーブルでつなぎます。



マイクロビットのバージョン

- 今回はマイクロビット V2 (v2.2) を使います。
- ゆめほたるの貸し出し用マイクロビット (体験会でも使用) は V1 (v1.5) です。
- V1 と V2 では、できることに少しちがいがあります。

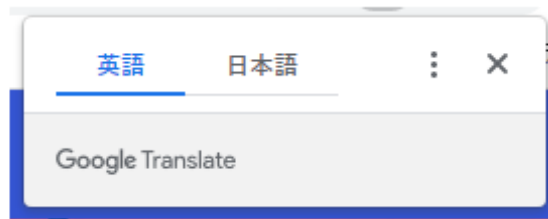


MakeCodeエディタの立ち上げ方

- プログラムは「MakeCodeエディタ」で作ります。
- Webブラウザ（Edge、Chrome、Safariなど）で、以下のページをひらきます。

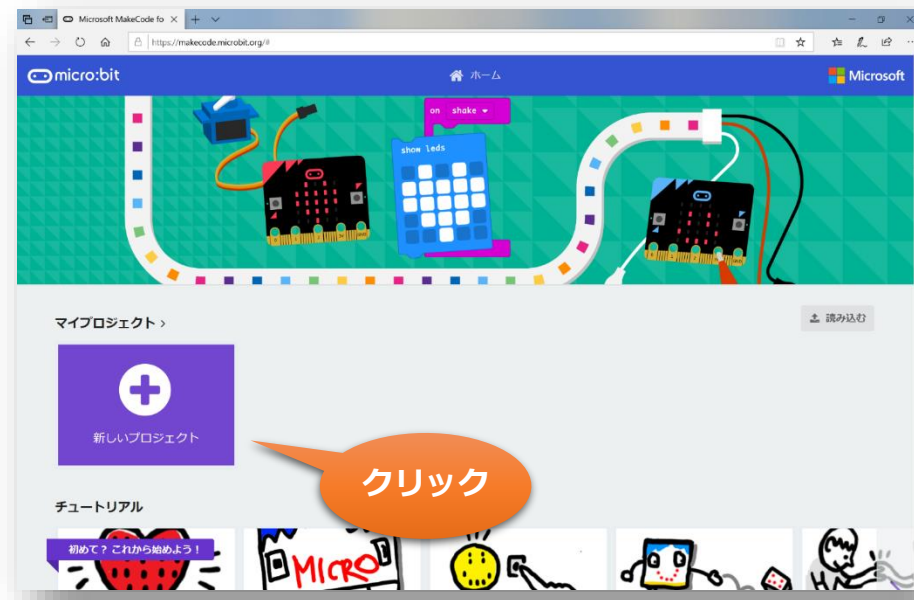
<https://makecode.microbit.org/>

- もしもページが英語で表示されたら、ページいちばん下の「Language」で日本語をえらびます。
- もしもウィンドウの右上にこのようなフォームが表示されたら、「×」をクリックしてとじるか、「⋮」をクリックして「このサイトは翻訳しない」をえらびます。



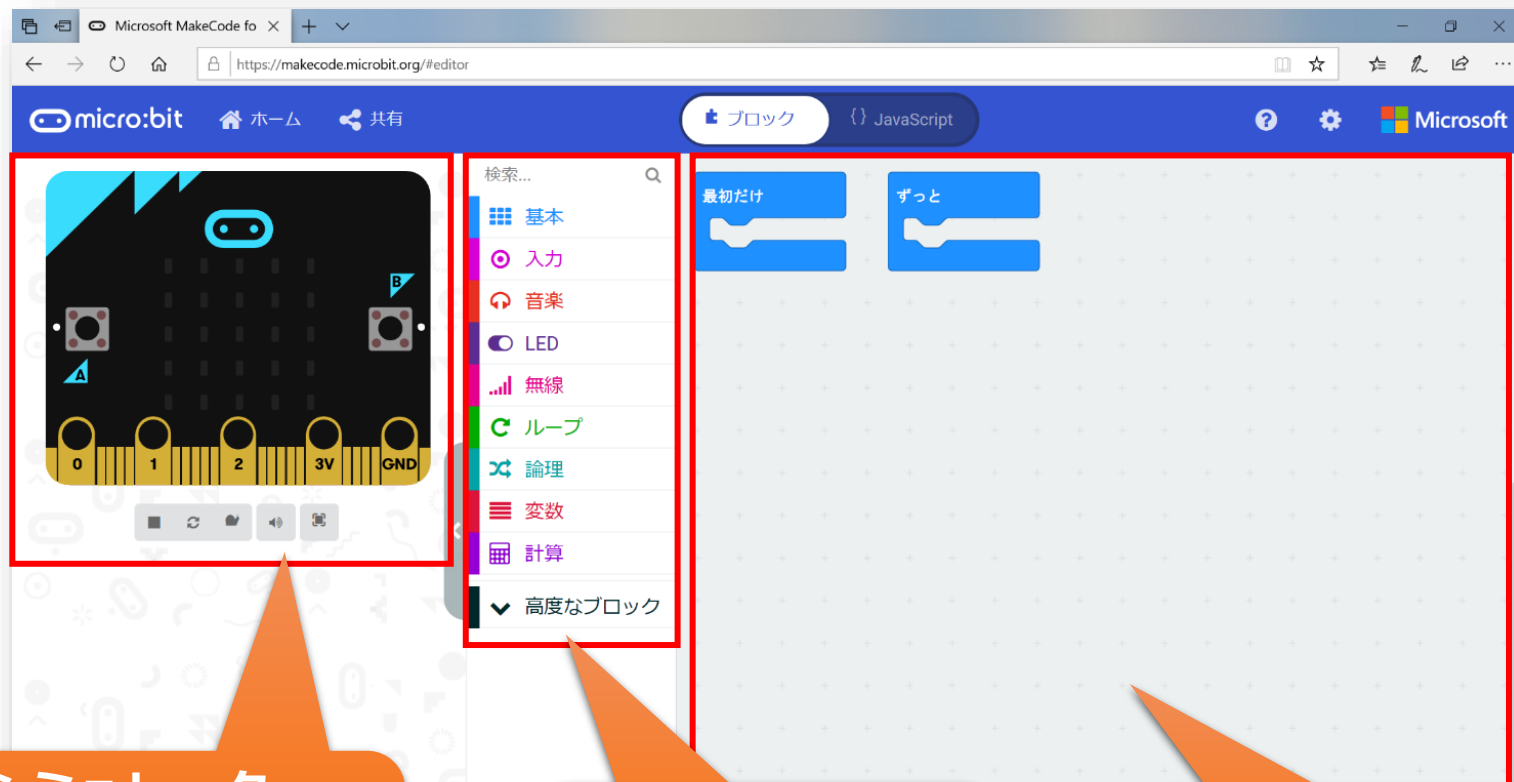
MakeCodeエディタの立ち上げ方

- これが「MakeCodeエディタ」のホーム画面です。
- 「新しいプロジェクト」をクリックすると、プロジェクトに名前をつけるウィンドウが表示されます。
- すきな名前を入力して「作成」をクリックすると、あたらしいプログラムをつくる画面がひらきます。



プログラムの作り方

- これがプログラムをつくる画面です。



シミュレータ

マイクロビットに書きこむ前にプログラムの動作を確認できます。

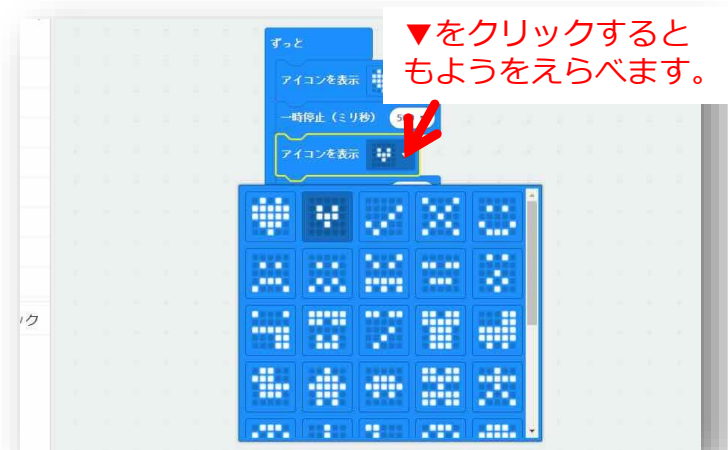
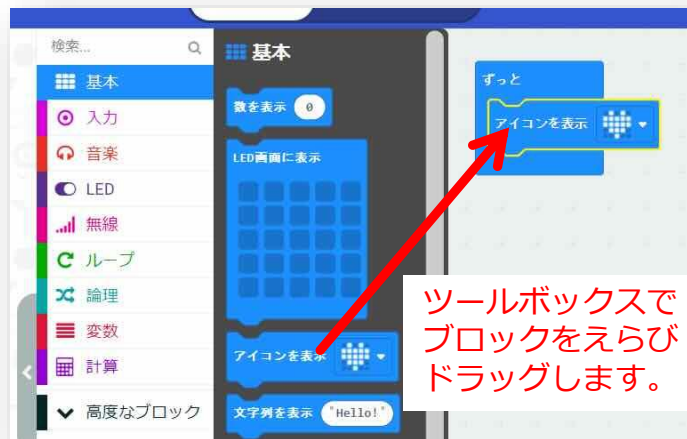
ツールボックス

いろいろなブロックが
おいてあります。

ワークスペース

ここでプログラムをつ
くれます。

プログラムのつくり方



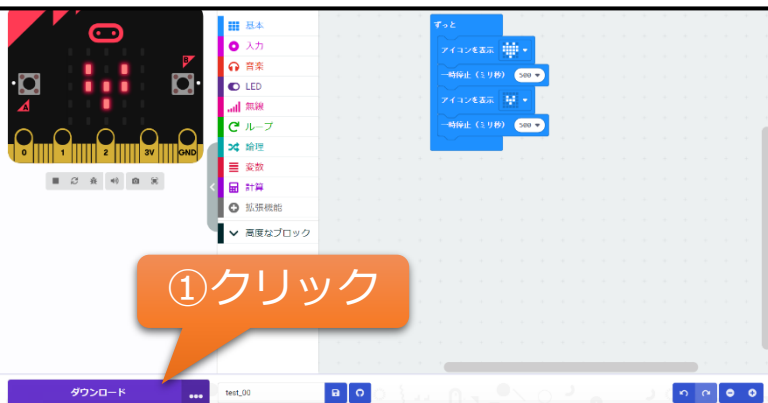
プログラムのダウンロード方法

- つくったプログラムをマイクロビットにかきこむ方法はふたつあります。
 - 1) プログラム作成画面から、そのままマイクロビットにかきこむ方法。
 - 2) プログラムをいちどパソコンにダウンロードしてから、マイクロビットにかきこむ方法。
- かきこめる方法は、つかっているパソコンやWebブラウザによってちがいます。
- もしも①でかきこめるなら①の方法をつかってください。①がダメなら②の方法をつかってください。

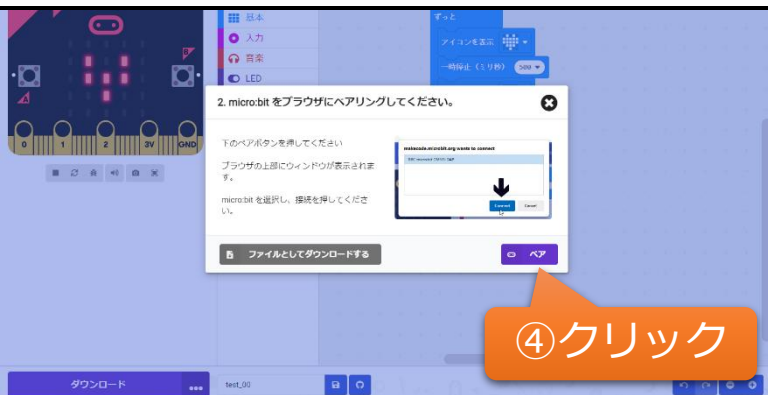
ダウンロード方法①

- パソコンやWebブラウザによって方法がちがいます。

① 「ダウンロード」 をクリックします



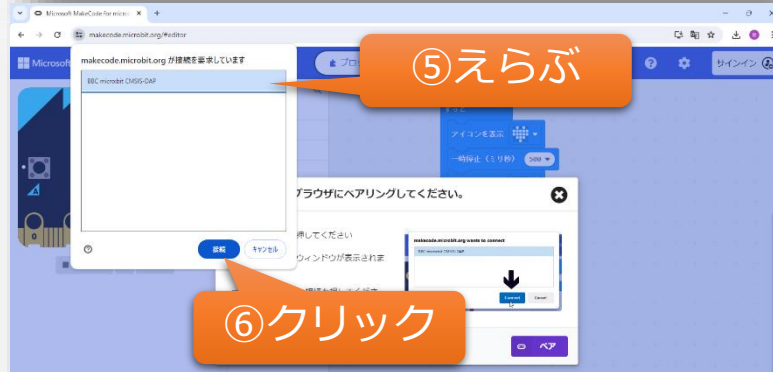
④ 「ペア」 をクリックします



② PCとmicro:bitをUSBケーブルでつなぎます
③ 「次へ」 をクリックします



⑤ えらぶ



⑤ 「BBC micro:bit」 をえらびます
⑥ 「接続」 をクリックします 11

ダウンロード方法① ~つづき

- パソコンやWebブラウザによって方法がちがいます。

⑦ 「ダウンロード」をクリックします



⑧ 2回めからは画面左下の「ダウンロード」のクリックだけでダウンロードできます



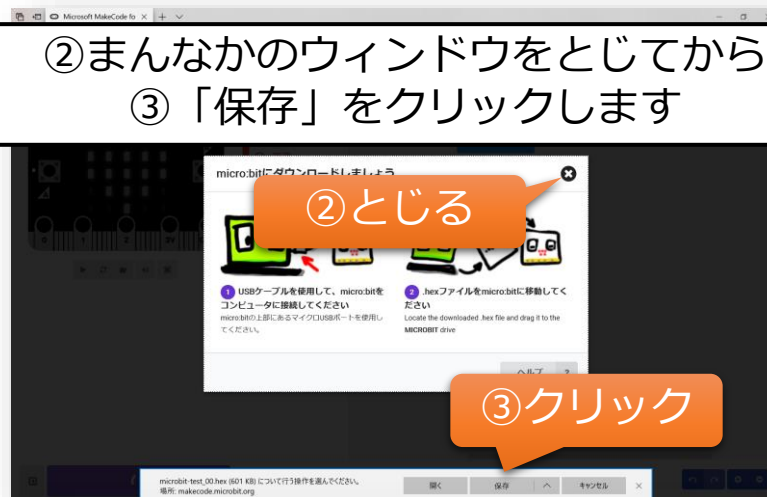
ダウンロード方法②

- パソコンやWebブラウザによって方法がちがいます。

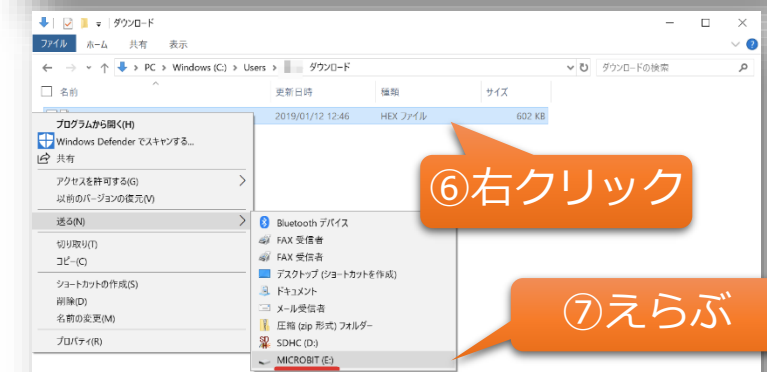
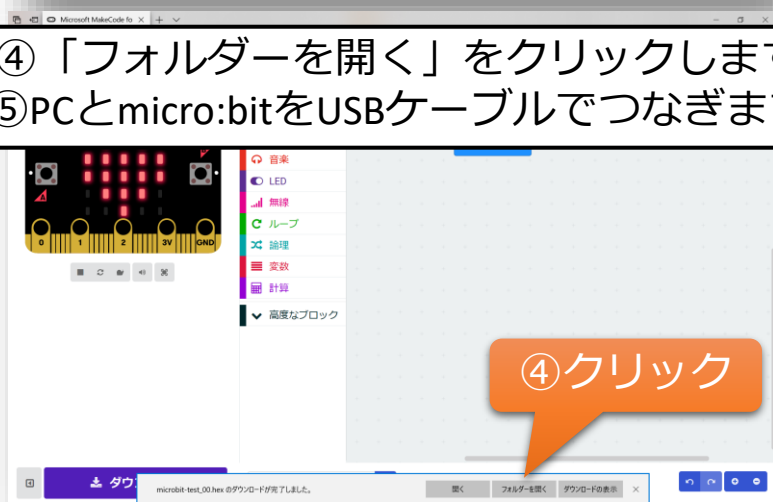
①「ダウンロード」をクリックします



②まんなかのウィンドウをとじてから
③「保存」をクリックします



④「フォルダーを開く」をクリックします
⑤PCとmicro:bitをUSBケーブルでつなぎます



⑥ファイルを右クリックし
⑦「送る」>「MICROBIT」をえらびます

キッチンタイマーをつくろう

- 今回は、みなさん自身に考えてもらって、プログラムをつくってもらいます。
- つくるのは「**キッチンタイマー**」です。
 - AボタンとBボタンをいっしょにおすと、タイマーがうごきはじめます。
 - 時間がくると音がなり、アイコンがひかります。
 - AボタンとBボタンで、はかる時間をかえられるようにします。
 - タイマーがうごいている間はアイコンがひかるようにします。
- パスタのゆで時間やカップラーメンをつくる時間をはかるのに役立ちます。

さいしょのプログラム

- AボタンとBボタンをいっしょにおしたら、「えがお」アイコンがひかり、「ピコーン！」と音になるプログラムをつくってください。
 - 「入力」に「ボタン A が押されたとき」ブロックがあります。「A」のところをクリックすると「A」「B」「A+B」をえらぶことができます。
 - 「基本」に「アイコンを表示」ブロックがあります。アイコンをクリックすると、好きなアイコンをえらぶことができます。
 - 「音楽」に「鳴らす メロディダダダム バックグラウンドで一度だけ」ブロックがあります。「ダダダム」のところをクリックすると、好きなメロディをえらぶことができます。

プログラムができたら、ただしくごくか
シミュレータでたしかめてください

さいしよのプログラム～作成例



改良プログラム 1

- AボタンとBボタンをいっしょにおしてから、アイコンがひかるまでに、10秒まつように改良してください。
 - 「基本」に「一時停止（ミリ秒）」ブロックがあります。
 - 「ミリ秒」とは、 $1/1000$ 秒のことです。なので、1秒まちたいときには「1000」と入力します。

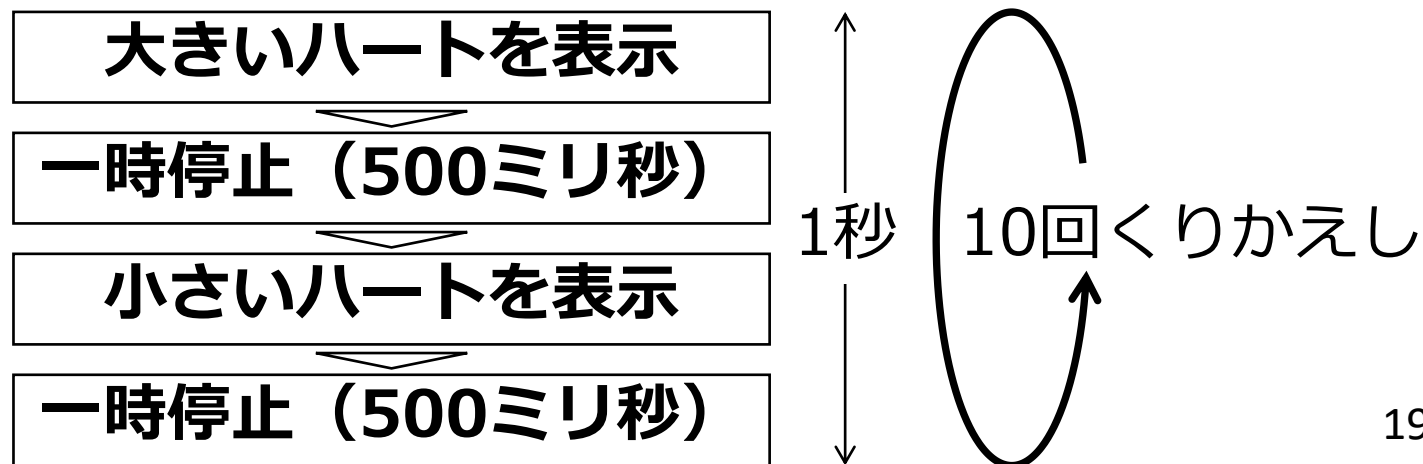
プログラムができたら、ただしくごくか
シミュレータでたしかめてください

改良プログラム 1 ～作成例

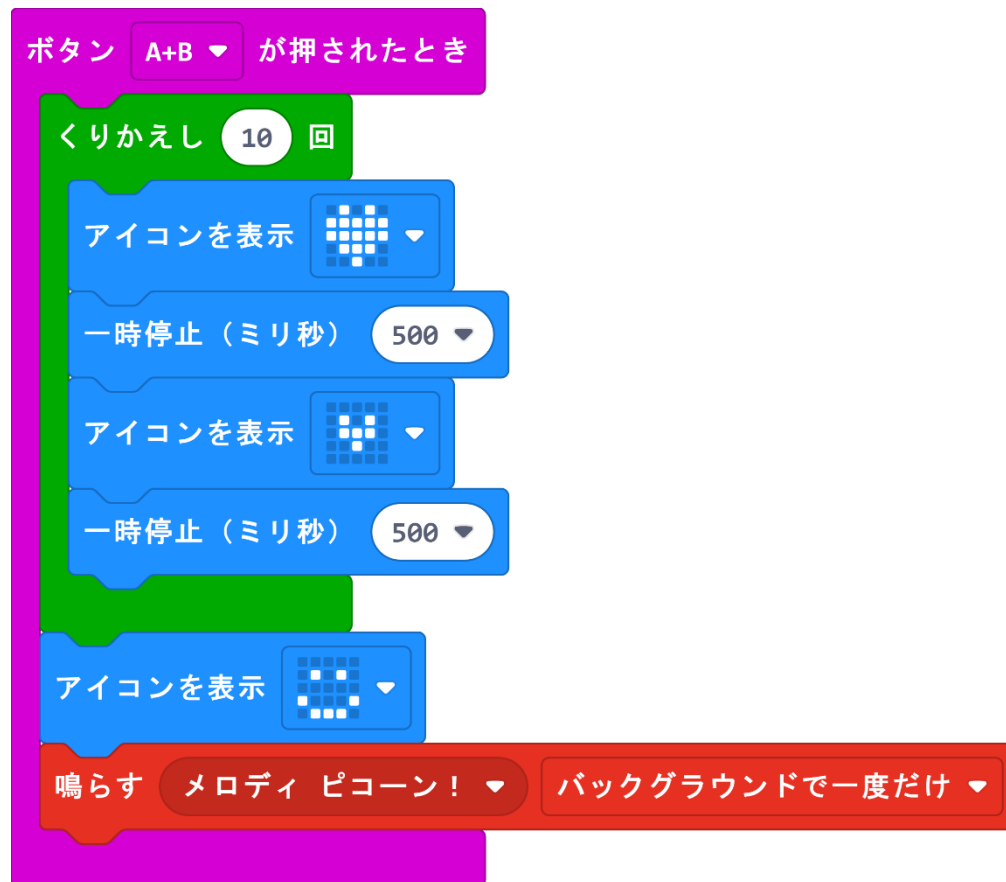


改良プログラム2

- 10秒まっているあいだ、なにもおこらないのはつまらないので、そのあいだ、0.5秒おきに「ハート」アイコンが大きくなったり小さくなったりするように改良してください。
 - 「ループ」に「くりかえし 4 回」ブロックがあるので、これをつかってください。
 - 以下のうごきを、「くりかえし」ブロックで10回くりかえします。



改良プログラム2～作成例



ここまでの確認

- AボタンとBボタンをいっしょにおしたら「ハート」アイコンが10秒間ピカピカし、そのあとで「えがお」アイコンがひかって「ピコーン！」と音になるプログラムができました。
- これで「**キッチンタイマー**」のできあがりです！
- ただし、このキッチンタイマーではかれる時間は「10秒」だけです。
- はかる時間をかえられるようにするために「**変数**」をつかいます。「変数」というのは、プログラムの中でかえることができる値のことです。

改良プログラム3

- さっきのプログラムで、くりかえしの回数で「10」と書いていたところを、変数をつかうように改良してください。
 - 「変数」の「変数を追加する」をクリックします。「作成する変数の名前」に「はかる時間」と入力して「OK」をクリックしてください。これで「はかる時間」という名前の変数ができました。
 - 「基本」に「最初だけ」ブロックがあります。また「変数」に「変数 はかる時間 を 0 にする」ブロックがあります。これらを使って、最初に変数「はかる時間」の値を「10」にします。
 - くりかえしの回数は、変数「はかる時間」で指定します。

プログラムができたら、ただしくごくか
シミュレータでたしかめてください

改良プログラム3～作成例

The image shows two parts of a Scratch script. On the left, a blue '最初だけ' (Only at the beginning) block contains a red '変数 はかる時間 を 10 にする' (Set variable 'はかる時間' to 10) block. A red arrow points to the '10' value. On the right, a purple 'ボタン A+B が押されたとき' (When button 'A+B' is pressed) block contains a green 'くりかえし はかる時間 回' (Repeat 'はかる時間' times) block, which is highlighted with a red arrow. Below the repeat block are two blue 'アイコンを表示' (Show icon) blocks, each followed by a blue '一時停止 (ミリ秒) 500' (Pause 500 milliseconds) block. At the bottom of the script is a red '鳴らす メロディ ピコーン! バックグラウンドで一度だけ' (Play melody 'ピコーン!' once in the background) block.

この改良では、プログラムのかきかたがかわるだけで、
プログラムのうごきはなにもかわりません

改良プログラム 4

- Bボタンをおすたびに、変数「はかる時間」が10ずつふえるように、またAボタンをおすと、変数「はかる時間」が10にもどるように改良してください。
- 変数の値をかえたときには、その値が表示されるようにしてください。
 - 「入力」に「ボタン A が押されたとき」ブロックがあります。「A」のところをクリックすると「A」「B」「A+B」をえらぶことができます。
 - 「変数」に「変数 はかる時間 を 1 だけ増やす」「変数 はかる時間 を 0 にする」ブロックがあります。
 - 「基本」に「数を表示」ブロックがあります。

プログラムができたら、
マイクロビットにかきこんでください

改良プログラム 4 ～作成例

The image displays a Scratch script with the following blocks:

- 最初だけ** (Only at the beginning):
 - 変数 **はかる時間** を **10** にする (Set variable **はかる時間** to **10**)
- ボタン **B** が押されたとき** (When button **B** is clicked):
 - 変数 **はかる時間** を **10** だけ増やす (Increase variable **はかる時間** by **10**)
 - 数を表示 **はかる時間** (Show number **はかる時間**)
- ボタン **A** が押されたとき** (When button **A** is clicked):
 - 変数 **はかる時間** を **10** にする (Set variable **はかる時間** to **10**)
 - 数を表示 **はかる時間** (Show number **はかる時間**)
- ボタン **A+B** が押されたとき** (When button **A+B** is clicked):
 - くりかえし **はかる時間** 回 (Repeat **はかる時間** times)
 - アイコンを表示 (Show icon)
 - 一時停止 (ミリ秒) **500** (Pause for **500** milliseconds)
 - アイコンを表示 (Show icon)
 - 一時停止 (ミリ秒) **500** (Pause for **500** milliseconds)
 - アイコンを表示 (Show icon)
 - 鳴らす **メロディ ピコーン!** **バックグラウンドで一度だけ** (Play melody **メロディ ピコーン!** **バックグラウンドで一度だけ**)

まとめ

- マイクロビットのつかいかたを確認しました。
- マイクロビットで「キッチンタイマー」をつくりました。
- 基本プログラムについて学びました。
 - **順次**^{じゅんじ}：かいた順番に作業すること。
 - **分岐**^{ぶんき}：条件にあてはまるかどうかで作業の内容をかえること（もし～なら）。
 - **反復**^{はんぷく}：ある作業をなんどもくりかえすこと。
 - **アルゴリズム**：「順次」「分岐」「反復」をくみあわせてつくった作業の手順。
 - **変数**^{へんすう}：数や文字などを入れておく「ハコ」のようなもの。

ゆめほたる環境科学技術塾

micro:bitプログラミング ～基礎～

おわり

ゆめほたる環境科学技術クラブ