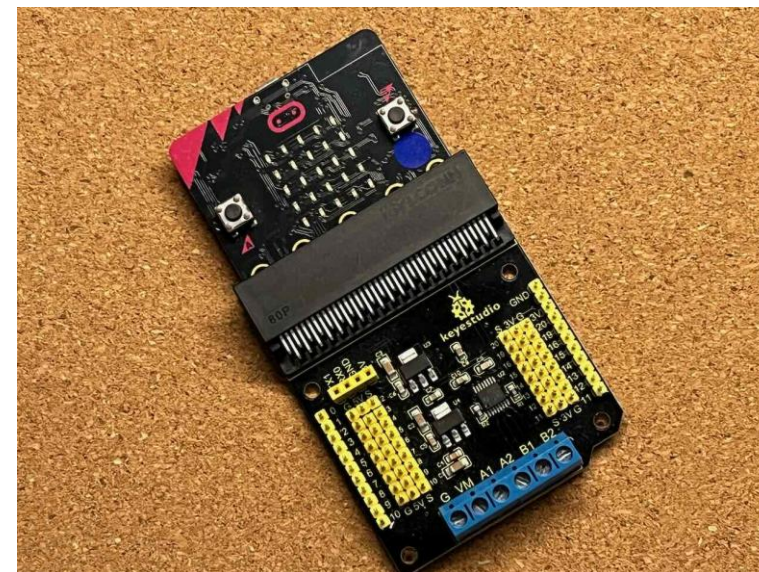
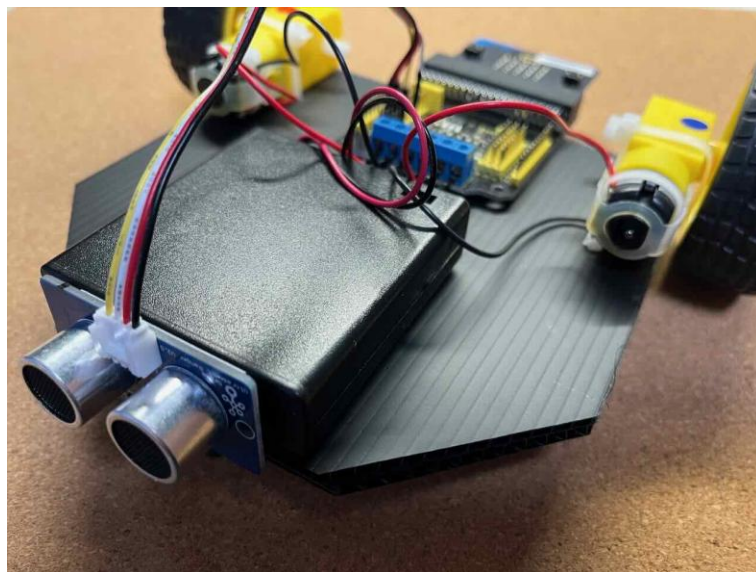


さとやまプログラミングクラブ2025

マイクロビットで「ロボカー」をつくろう！

第1回

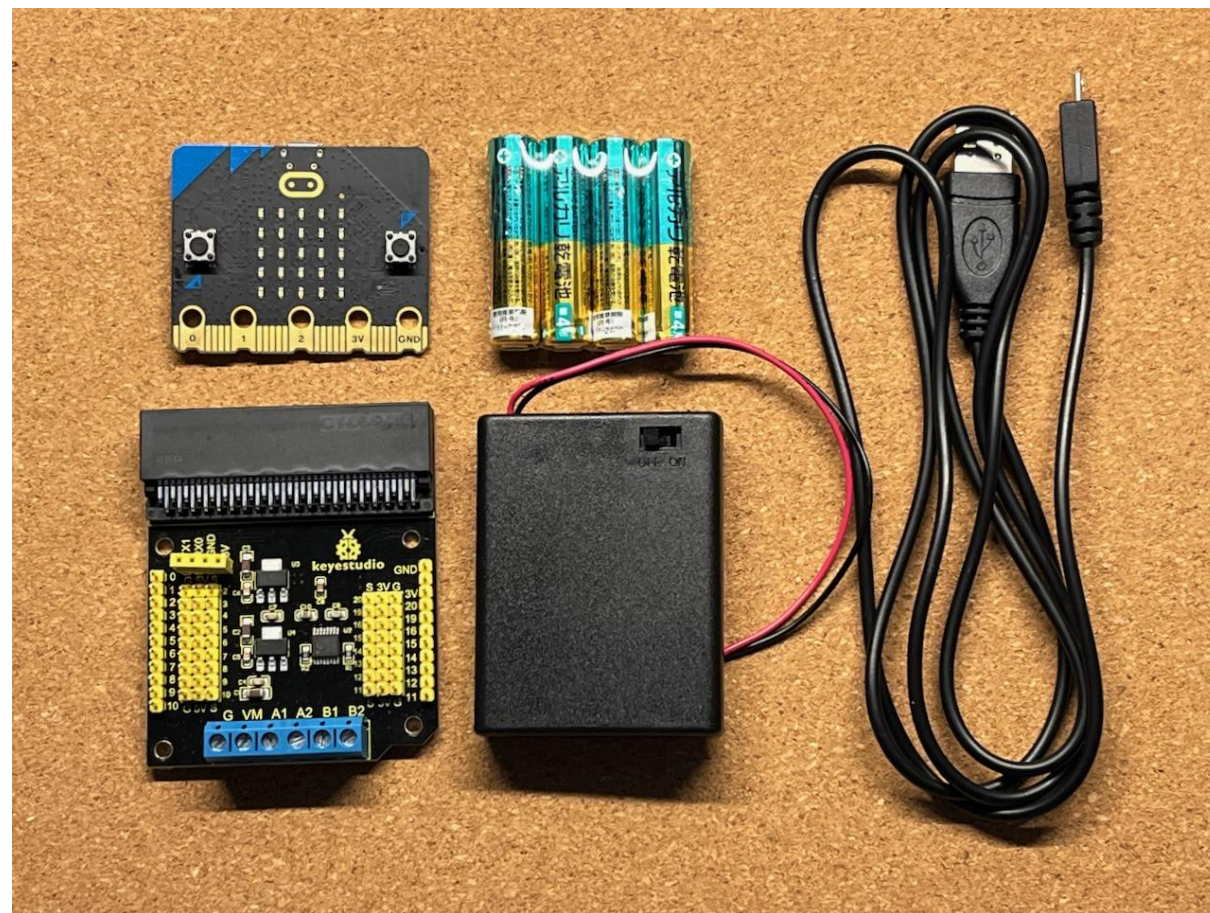


今日やること

- 「マイクロビット」についてまなびます。
- パソコンでマイクロビットのプログラムをつくってみます。
- マイクロビットにプログラムをかきこんでみます。
- 「超音波センサ」ではかった距離を、マイクロビットで確認してみます。
- マイクロビットで「モーター」をうごかしてみます。

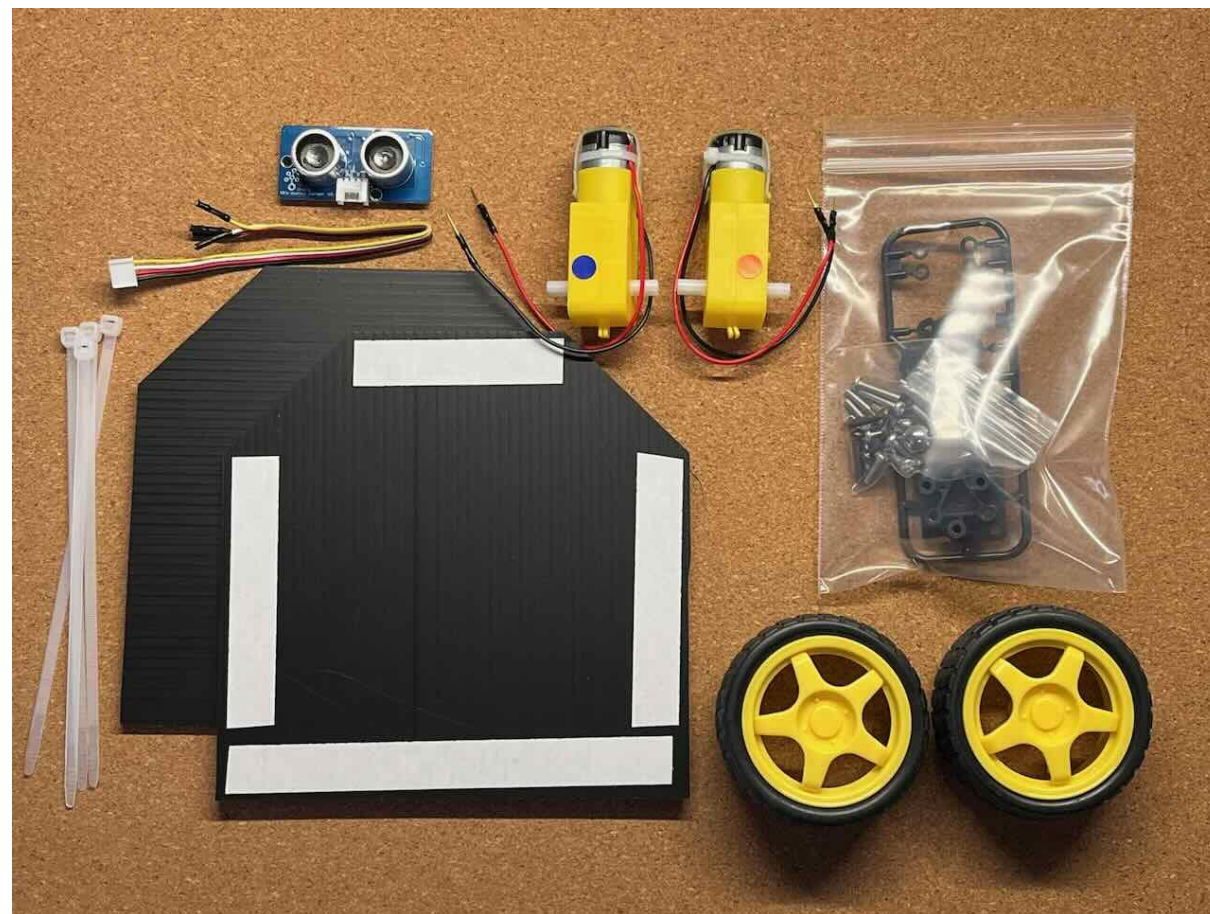
部品の確認 (マイクロビットセット)

部品	個数
マイクロビット	1
拡張ボード (KEYE STUDIO KS4033)	1
USBケーブル	1
単三x4電池ボックス	1
単三乾電池	4



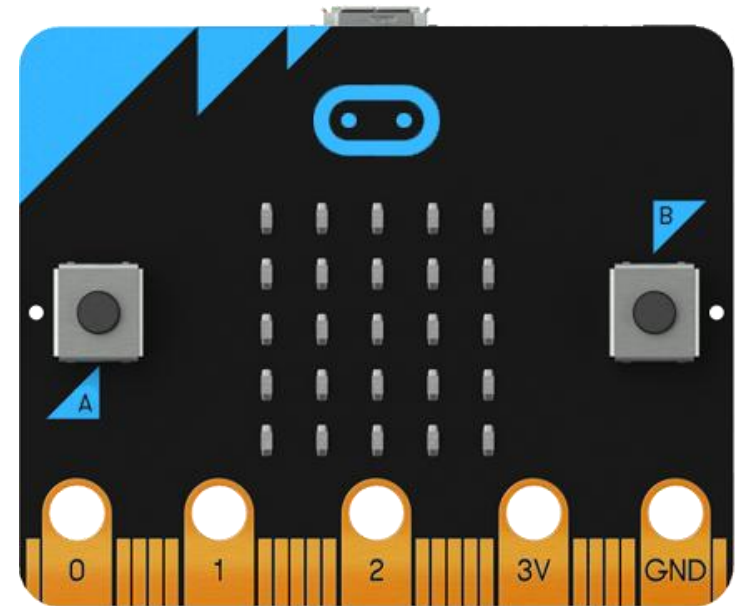
部品の確認（ロボカー）

部品	個数
モーター	2
タイヤ	2
超音波センサ	1
超音波センサ用ケーブル	1
プラダン	2
ボールキャスター	1
結束バンド	4



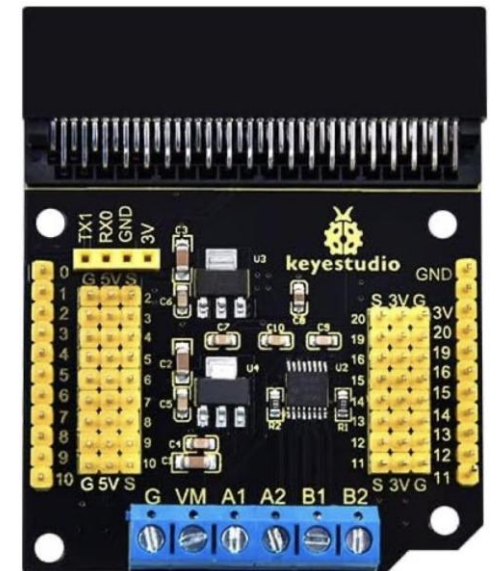
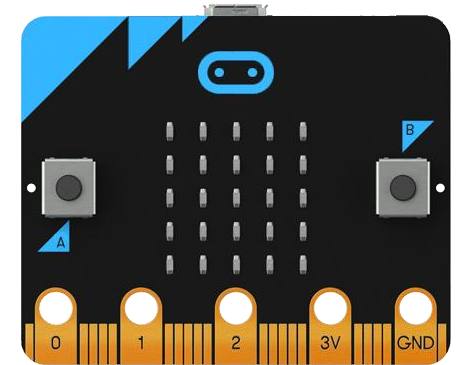
マイクロビットについて

- マイクロビットは、ちいさなコンピュータです。
- プログラムはパソコンでつくり、それをマイクロビットにかきこんでうごかします。
- マイクロビットにプログラムをかきこんだあとは、パソコンからとりはずしても、マイクロビットだけでうごきます。
- 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つくってうごかすことができます。



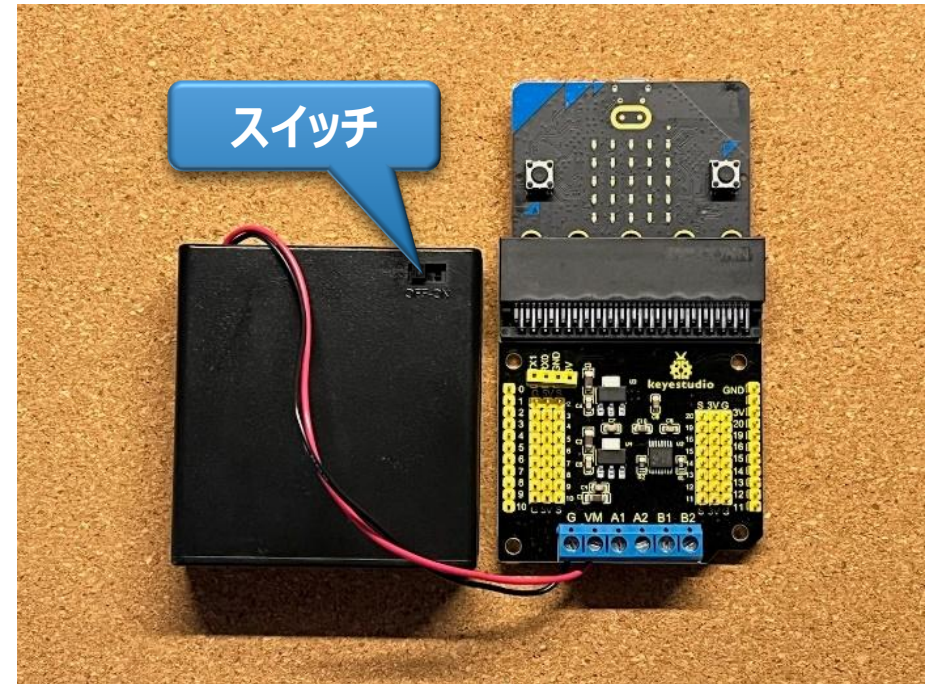
拡張ボードについて

- 今回は、マイクロビットを「拡張ボード」にとりつけてつかいます。
- 拡張ボードに電池ボックスをとりつけ、電池でマイクロビットをうごかします。
- 拡張ボードには端子がたくさんついており、それをつかってマイクロビットとほかの部品をつなぐことができます。
- 拡張ボードにモーターをとりつけると、マイクロビットからの命令でモーターをうごかすことができます。



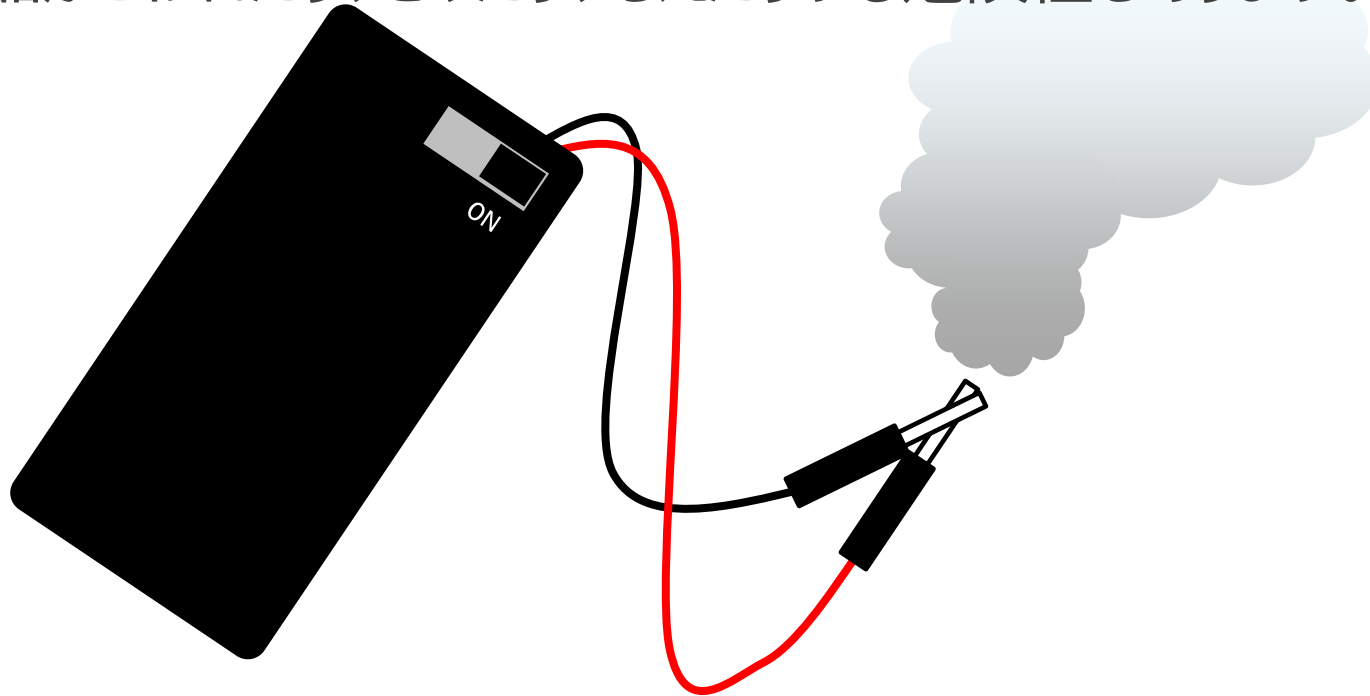
準備作業

- マイクロビットを拡張ボードにさしこんでください。
- 電池ボックスの黒いコードを、拡張ボードの青い端子のうち「G」とかいてあるところにとりつけてください。
- 電池ボックスの赤いコードを、拡張ボードの青い端子のうち「VM」とかいてあるところにとりつけてください。
- 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。



注意事項

- 使わないときは、電池ボックスのスイッチをオフにしてください。
- オンのままにしていると、電池がすぐになくなるだけでなく、もしもプラスとマイナスがショートすると、部品がこわれたり、とけたり、もえたりする危険性もあります。

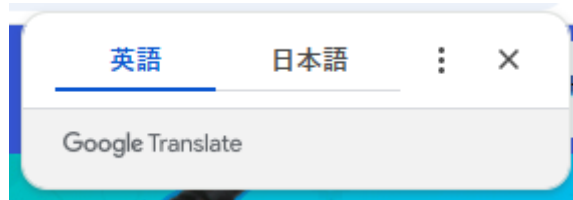


MakeCodeエディタのたちあげかた

- マイクロビットのプログラムは「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エディタのたちあげかた

- 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。
 - ❖ 「プロジェクト」というのがプログラムのことです。
- 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。

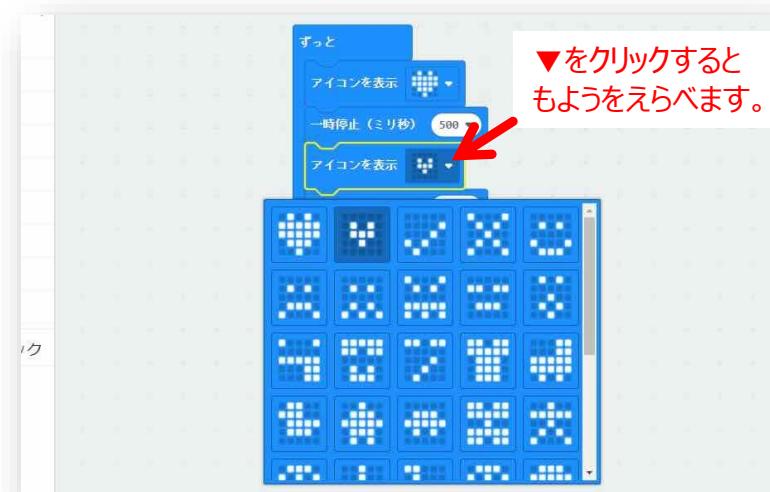
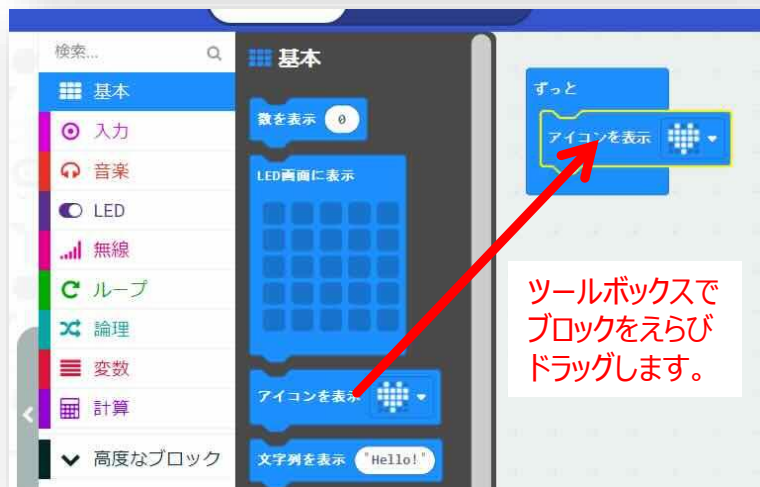


プログラム作成画面

The screenshot shows the Microsoft MakeCode editor for Micro:bit. The browser address bar displays <https://makecode.microbit.org/#editor>. The interface features a blue header with the 'micro:bit' logo, navigation links for 'ホーム' (Home) and '共有' (Share), and tabs for 'ブロック' (Blocks) and 'JavaScript'. A search bar is located in the top center. On the left, a simulator shows a Micro:bit board with pins labeled 0, 1, 2, 3V, and GND. Below the simulator are icons for running, refreshing, and undo. The center panel contains a '検索...' (Search) bar and a list of block categories: 基本 (Basic), 入力 (Input), 音楽 (Music), LED, 無線 (Wireless), ループ (Loops), 論理 (Logic), 変数 (Variables), 計算 (Math), and 高度なブロック (Advanced Blocks). The right panel shows a workspace with a grid background and two blue buttons labeled '最初だけ' (Only at the start) and 'ずっと' (Forever). Three blue callout boxes are overlaid on the interface:

- 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かた

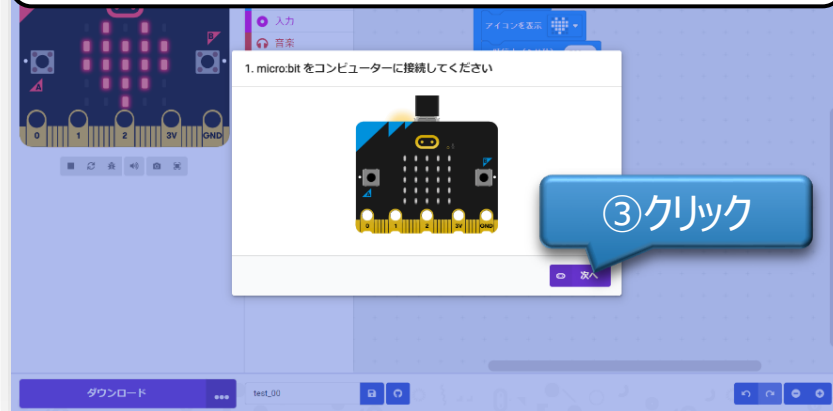


マイクロビットへのプログラムのかきこみかた

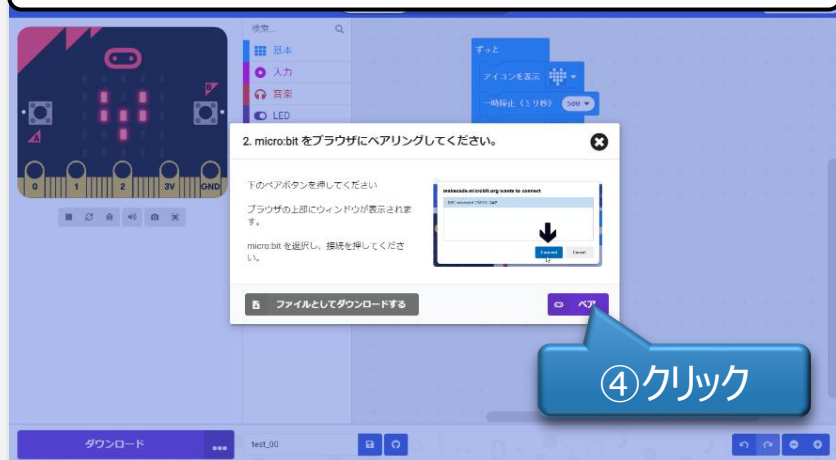
①「ダウンロード」をクリックします



② PCとマイクロビットをUSBケーブルでつなぎます
③「次へ」をクリックします



④「ペア」をクリックします



⑤「BBC micro:bit」をえらびます
⑥「接続」をクリックします



マイクロビットへのプログラムの書きこみかた

⑦「ダウンロード」をクリックします



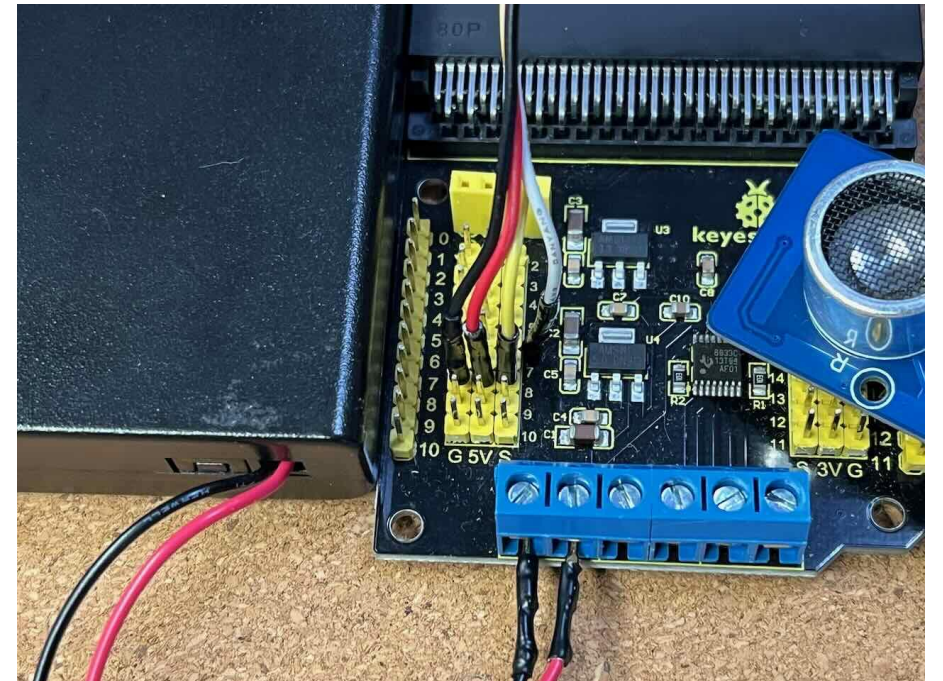
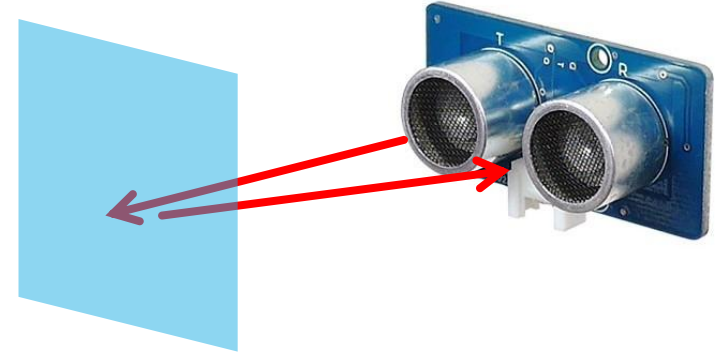
⑧ 2回目からは画面左下の「ダウンロード」のクリックだけでダウンロードできます



超音波センサをつかう

- 「超音波センサ」は、前にあるモノとの距離をはかることができるセンサです。
- 「右目」から超音波をだし、モノにあたってはねかえってきたのを「左目」でうけます。超音波をだしてからうけるまでの時間をはかることで、モノまでの距離がわかります。
- 超音波センサと拡張ボードを以下のようにつなぎます。

超音波センサ	拡張ボード
黒のコード	G
赤のコード	5V
白のコード	どこにもつなぎません
黄のコード	8



超音波センサをつかう

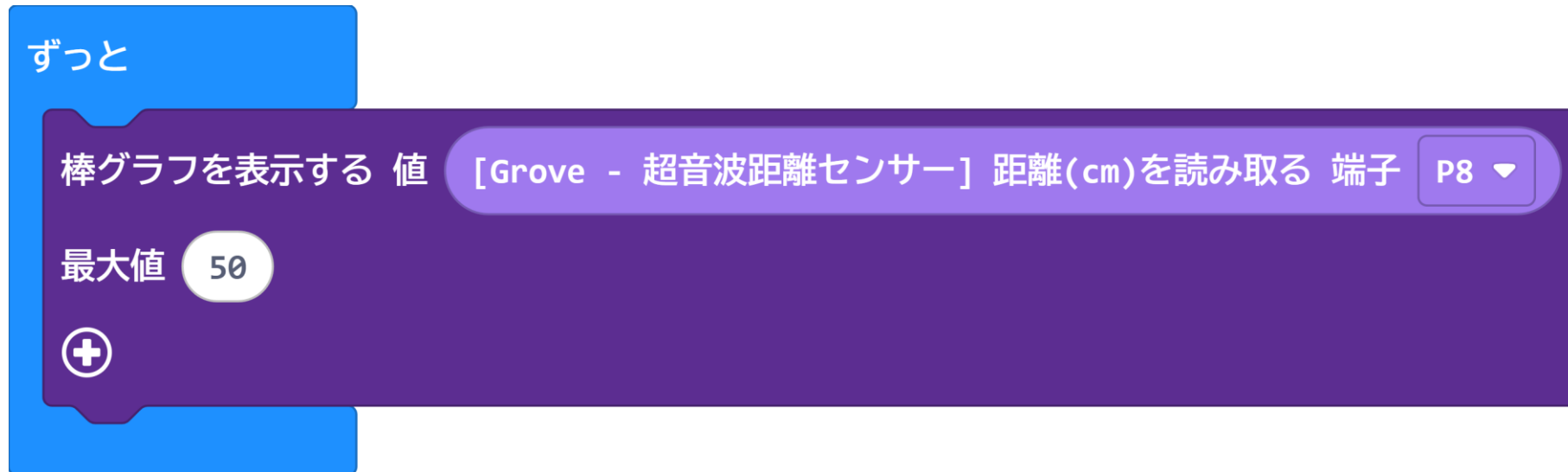
- ツールボックスで「拡張機能」をクリックします。
- 検索窓に「grove」と入力します。
- 表示された中から「grove」をクリックします。
- ツールボックスに「Grove」があらわれます。
- 「Grove」の中にある「[Grove-超音波距離センサー] 距離(cm)を読み取る」ブロックをつかって距離をはかります。



The screenshot shows the Scratch IDE interface. On the left, the 'Grove' category is selected in the toolboxes. The main workspace displays a script for a '4-Digit' display. The script includes several 'strip' blocks for controlling the display. Below these, under the 'Ultrasonic' category, there are three blocks for reading distance from an ultrasonic sensor. A red arrow points to the block '[Grove - 超音波距離センサー] 距離(cm)を読み取る 端子 P0', which is the block mentioned in the instructions.

超音波センサをつかう

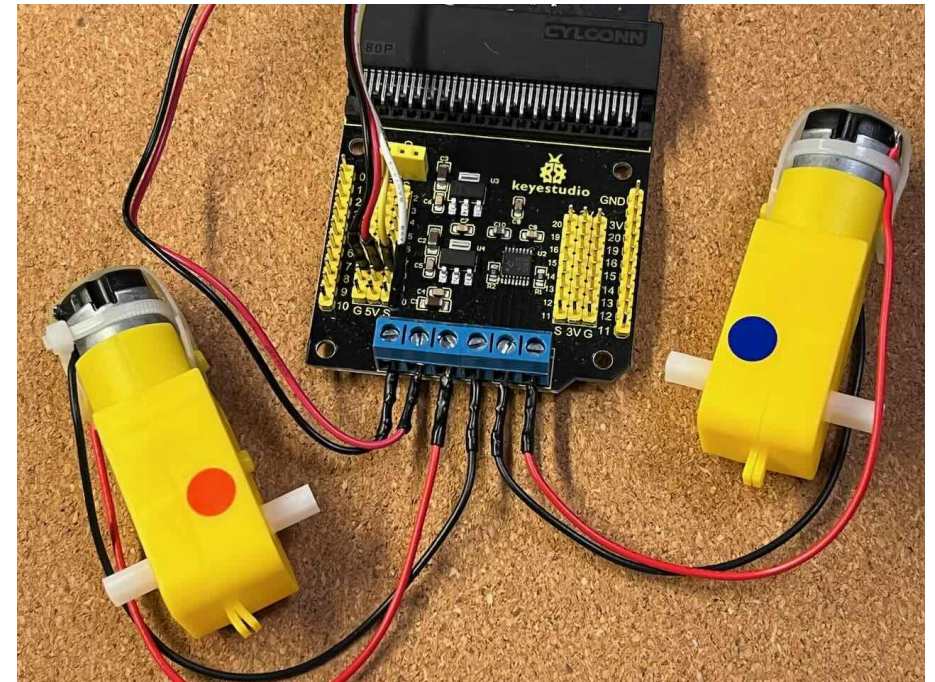
- MakeCodeエディタで以下のプログラムをつくり、マイクロビットにかきこんでください。
- 超音波センサのまえに手をかざすと、距離によってLEDのひかる数が変わります。



モーターをうごかす

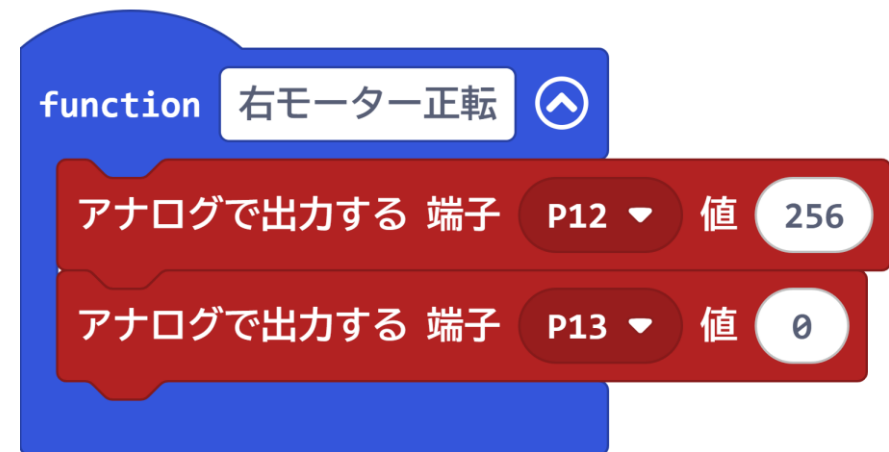
- 今回つかっている拡張ボードには、モーターを制御する機能があります。
- マイクロビットからの命令で、ふたつのモーターをまわしたり、とめたり、まわる向きをかえたりできます。
- ふたつのモーターと拡張ボードの青い端子を以下のようにつなぎます。

モーター		拡張ボード
右モーター (赤いシールがはってある)	赤のコード	A1
	黒のコード	A2
左モーター (青いシールがはってある)	黒のコード	B1
	赤のコード	B2



モーターをうごかす

- ひとつのモーターは、ふたつの出力信号をつかって制御します。
 - 右モーター：出力端子「P12」と「P13」
 - 左モーター：出力端子「P15」と「P16」
- あとからつかいやすいように、「関数」というものをつかってモーターをうごかす命令をつくっておきます。
- ツールボックスで「高度なブロック」>「関数」をクリックします。
- 「関数を作成する」をクリックします。
- 「doSomething」とかいてあるところに「右モーター正転」とかいて「完了」をクリックします。
- 「高度なブロック」>「入出力端子」>「アナログで出力する」をつかって、右のような関数をつくります。



モーターをうごかす

- おなじように、以下のような関数をつくります。

The image displays six code blocks arranged in a 2x3 grid, each representing a function for controlling a motor. Each block is a blue rounded rectangle with a white header containing the function name and a blue arrow icon. Below the header are two red rounded rectangles representing analog output blocks, each with a white header 'アナログで出力する 端子' (Analog output terminal) and a white value field.

- 右モーター正転 (Right Motor Forward):** Terminal P12 value 256, Terminal P13 value 0.
- 右モーター逆転 (Right Motor Reverse):** Terminal P12 value 0, Terminal P13 value 256.
- 右モーター停止 (Right Motor Stop):** Terminal P12 value 0, Terminal P13 value 0.
- 左モーター正転 (Left Motor Forward):** Terminal P15 value 256, Terminal P16 value 0.
- 左モーター逆転 (Left Motor Reverse):** Terminal P15 value 0, Terminal P16 value 256.
- 左モーター停止 (Left Motor Stop):** Terminal P15 value 0, Terminal P16 value 0.

モーターをうごかす

- つくった関数をつかって、右のプログラムをつくります。できたらマイクロビットにかきこんでください。
- マイクロビットのボタンをおすとモーターがまわります。
 - A+B：ふたつのモーターが正転
 - A：右モーターが逆転
 - B：左モーターが逆転
- ボタンをはなすとモーターがとまります。
- マイクロビットでモーターの制御ができました！

```
ずっと
もし ボタン A+B が押されている なら
  呼び出し 右モーター正転
  呼び出し 左モーター正転
でなければもし ボタン A が押されている なら
  呼び出し 右モーター逆転
  呼び出し 左モーター停止
でなければもし ボタン B が押されている なら
  呼び出し 右モーター停止
  呼び出し 左モーター逆転
でなければ
  呼び出し 右モーター停止
  呼び出し 左モーター停止
+
```