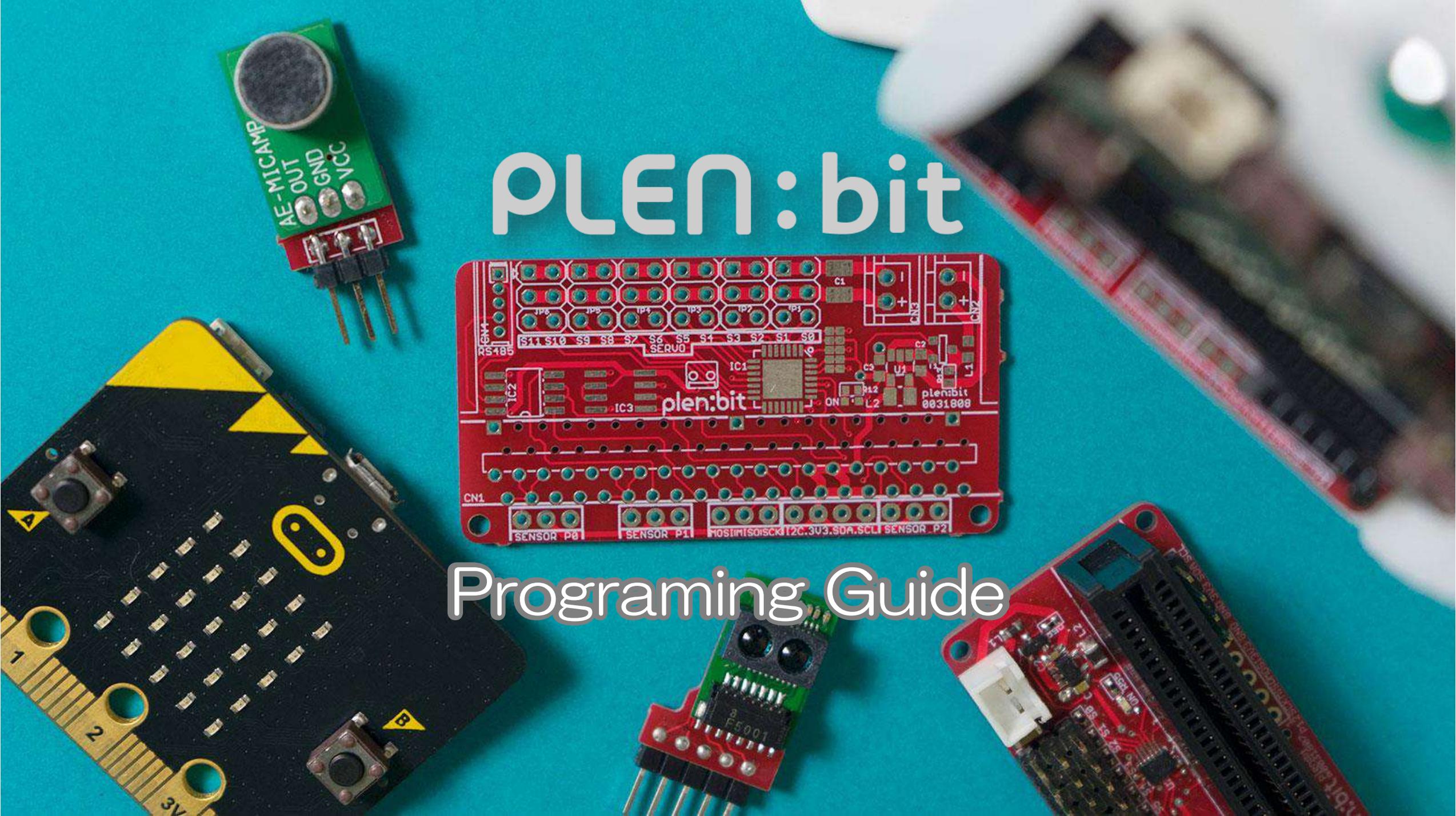


PLEN:bit

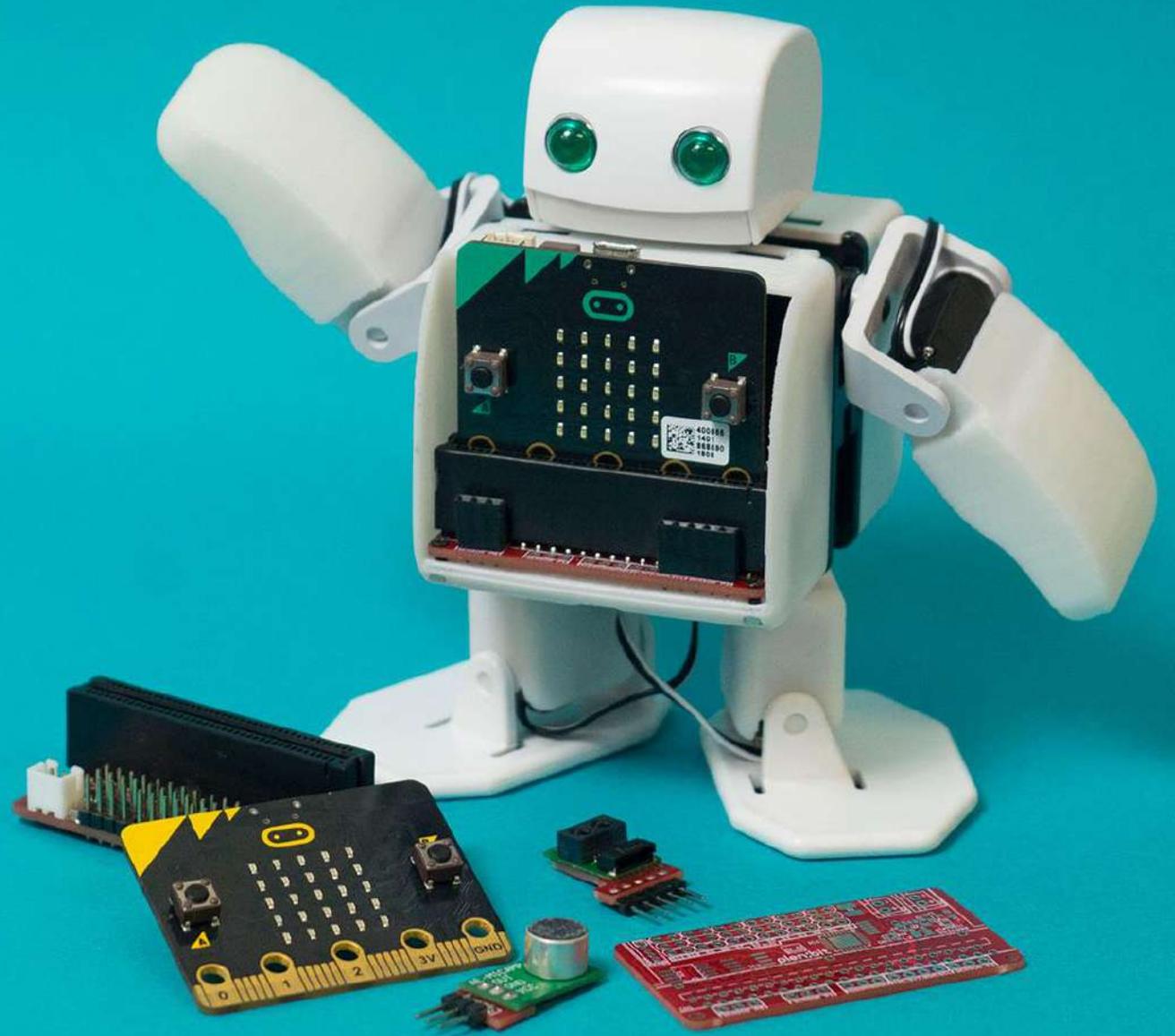
Programming Guide



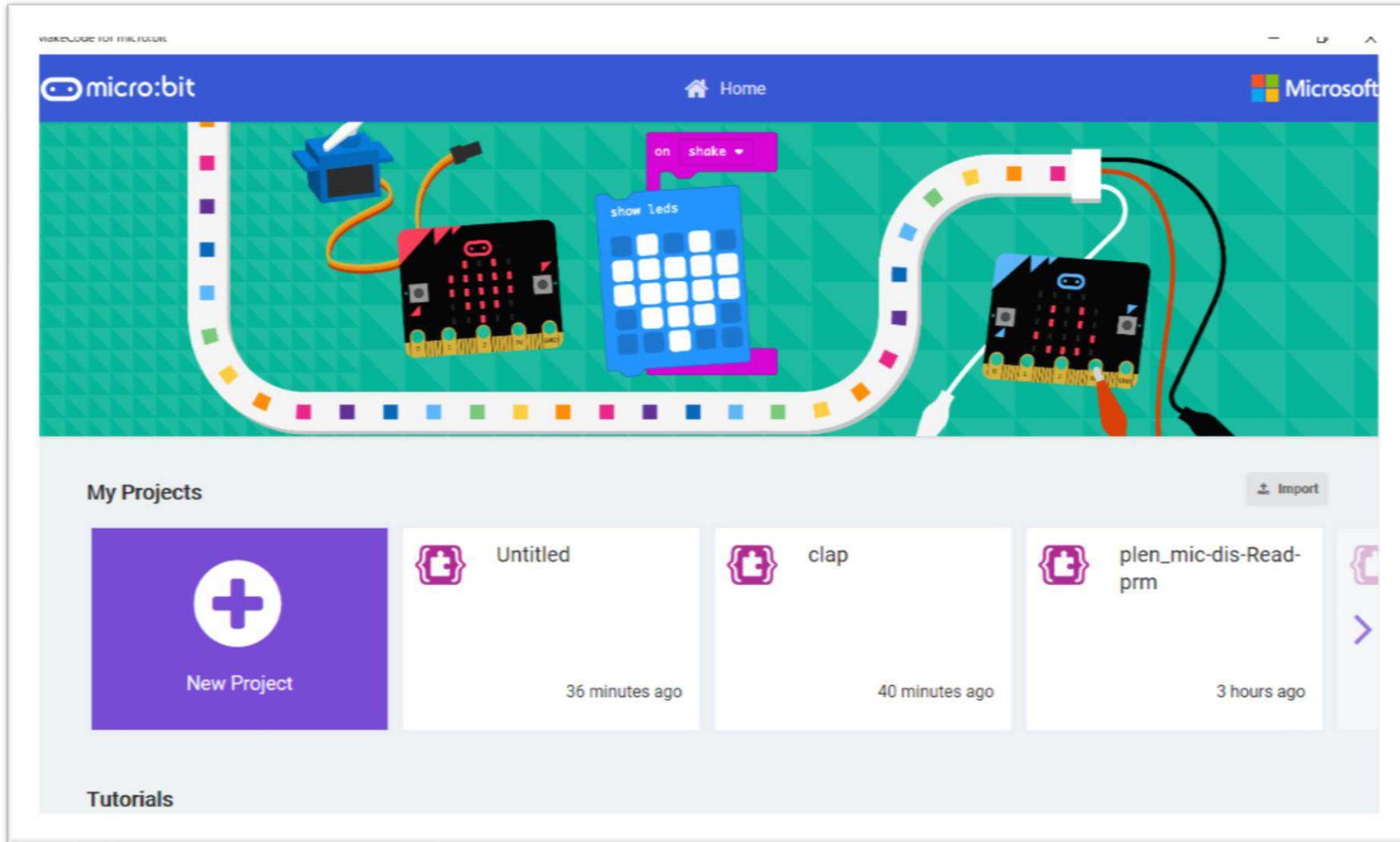
About PLEN:bit

PLEN:bit is a small humanoid robot with movable arms and legs.

Anyone can easily use PLEN:bit through programming and onboard sensors.

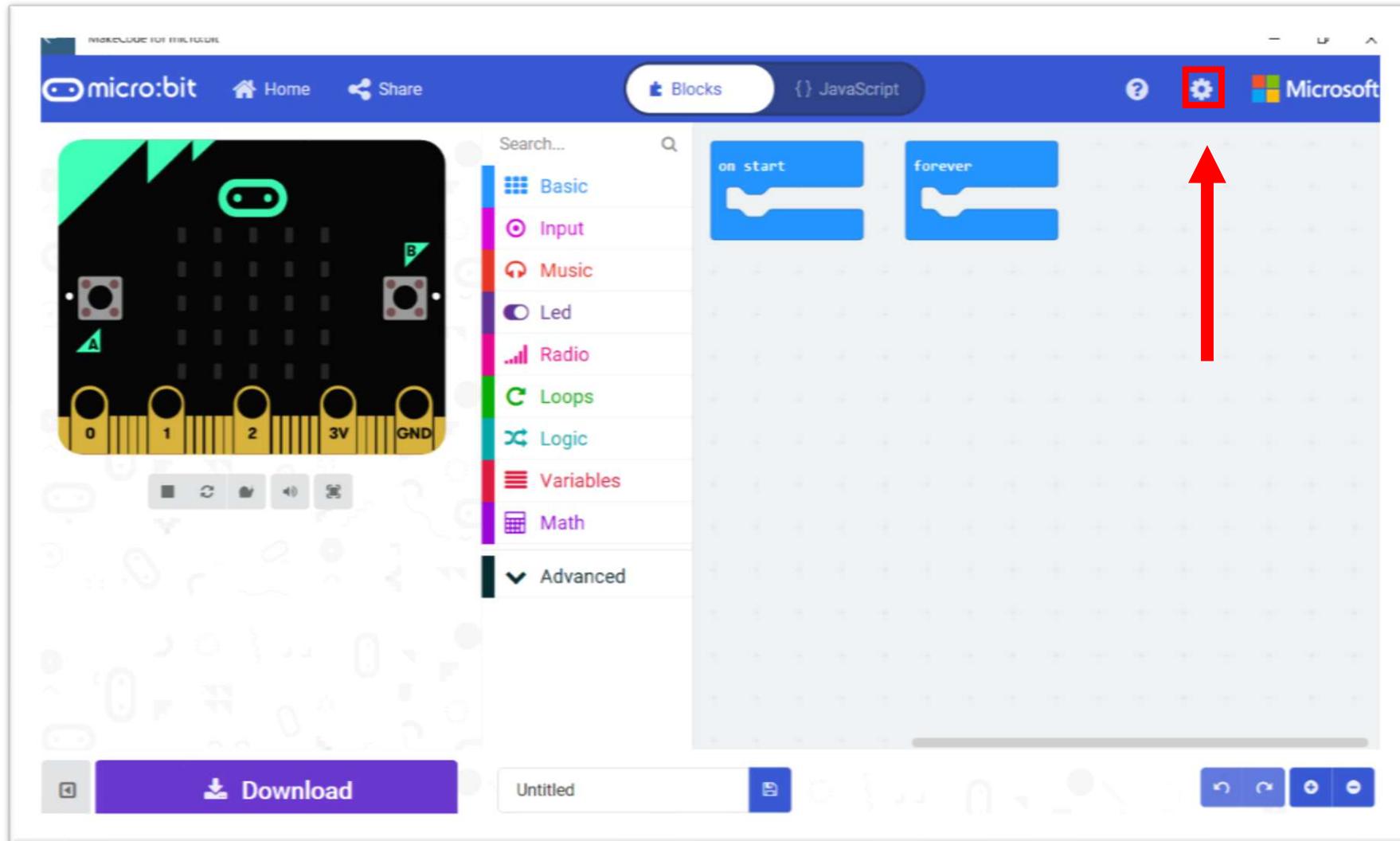


How to **PLEN:bit** – Connect to micro:bit –



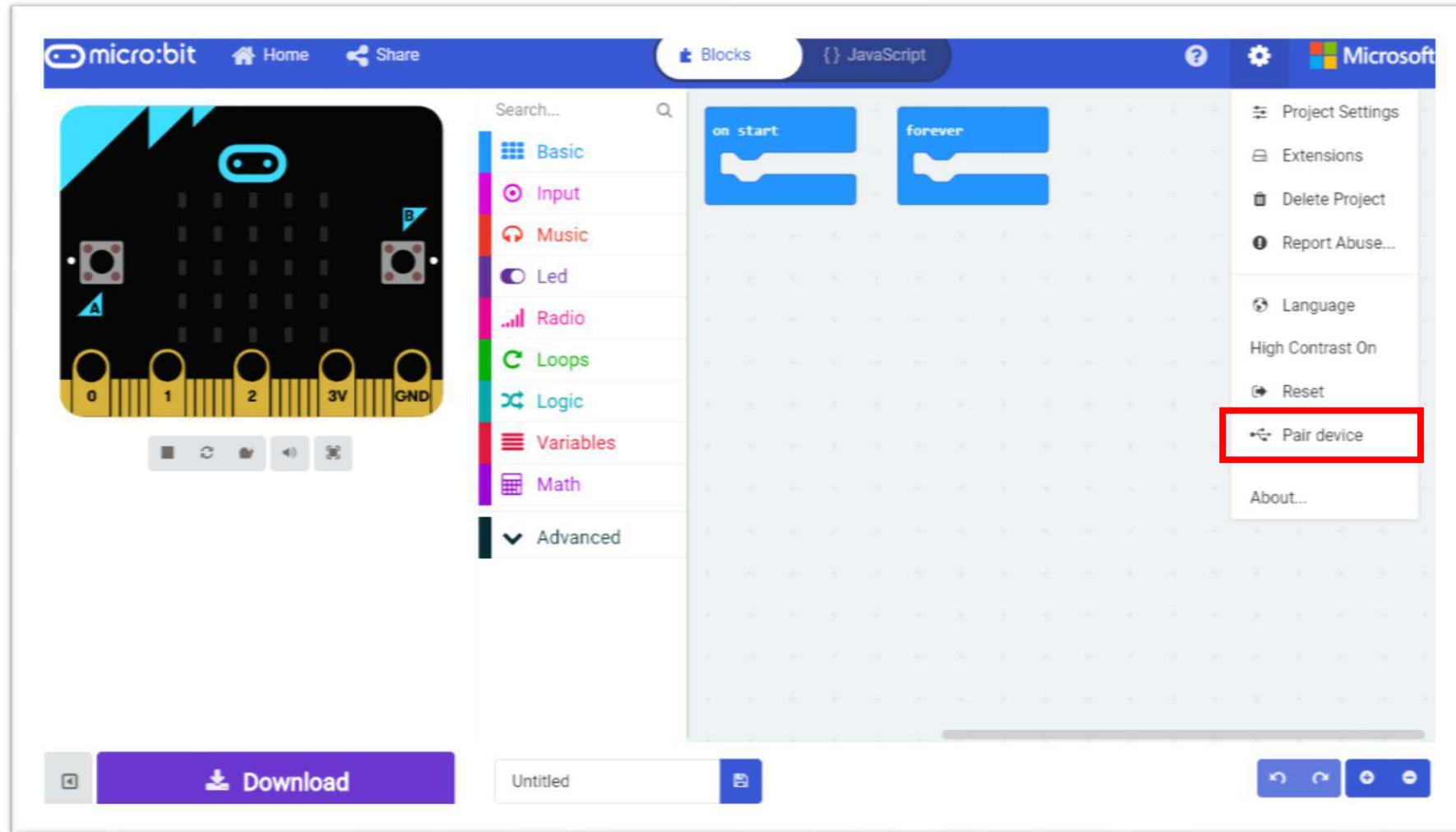
Go to Access <https://makecode.microbit.org/> & Click the 「+」 button

How to **PLEN:bit** – Connect to micro:bit –



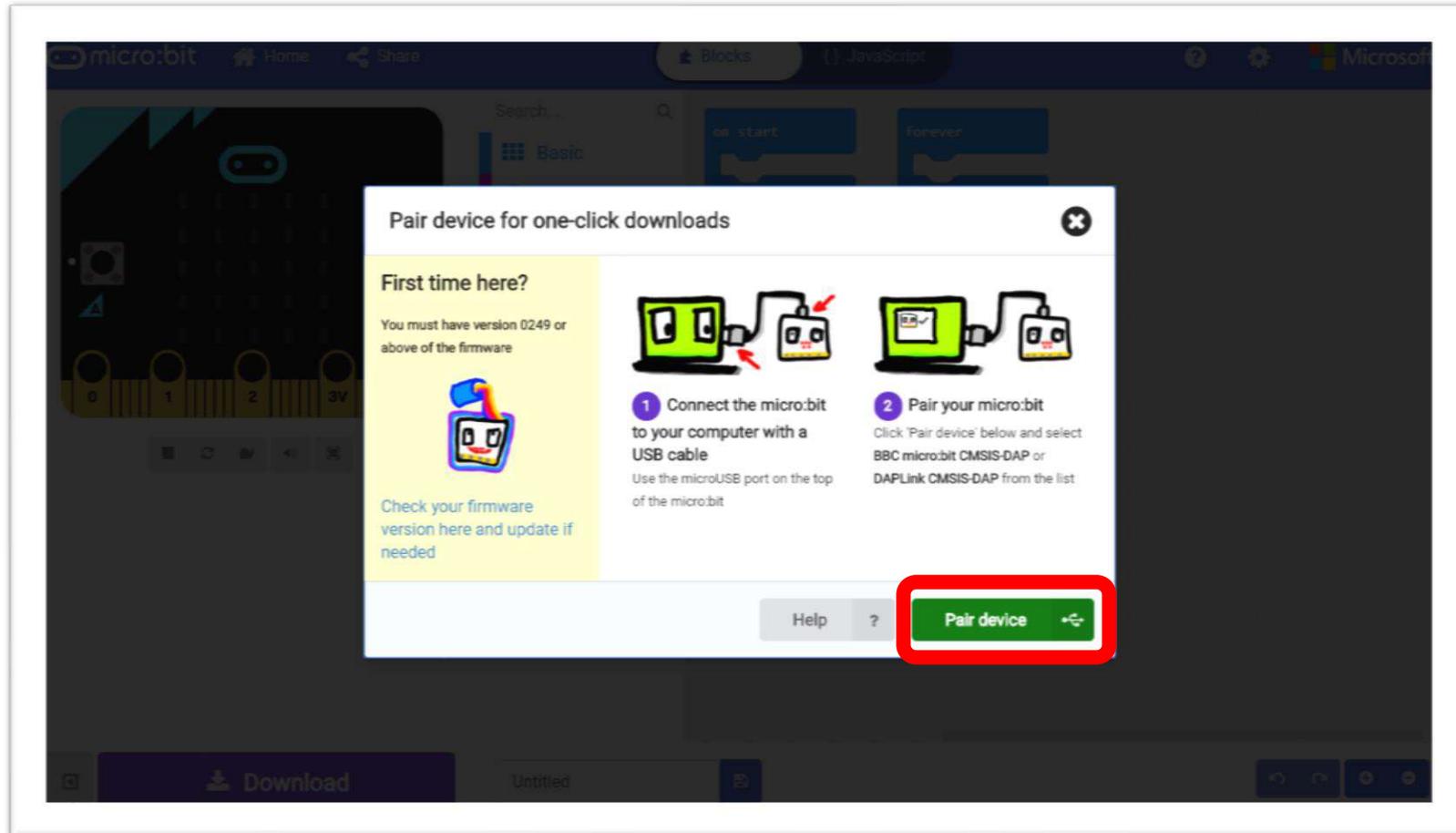
Click the "Gear Mark" at the top right inside the screen

How to PLEN:bit – Connect to micro:bit –



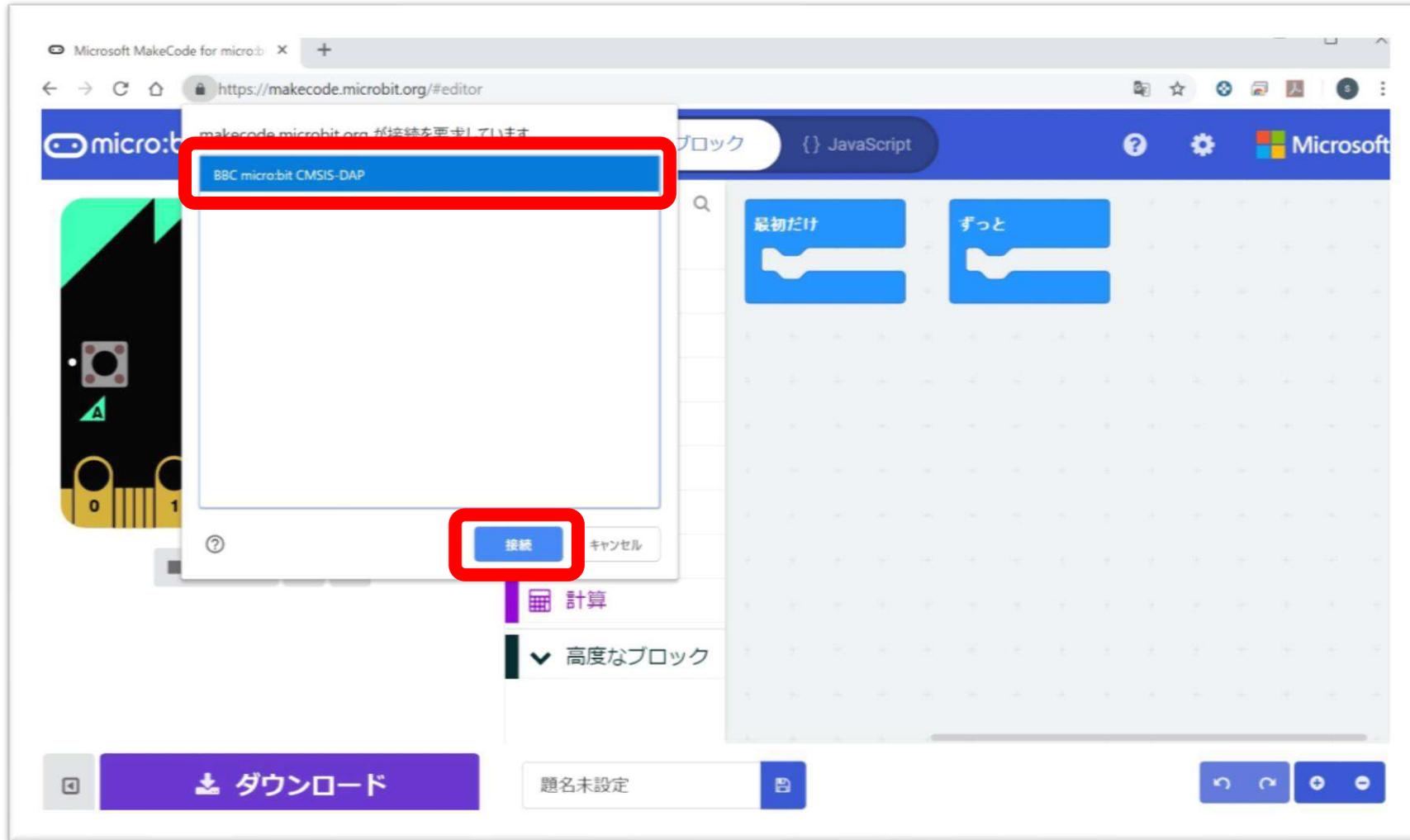
Click “Pair device” from the menu

How to **PLEN:bit** – Connect to micro:bit –



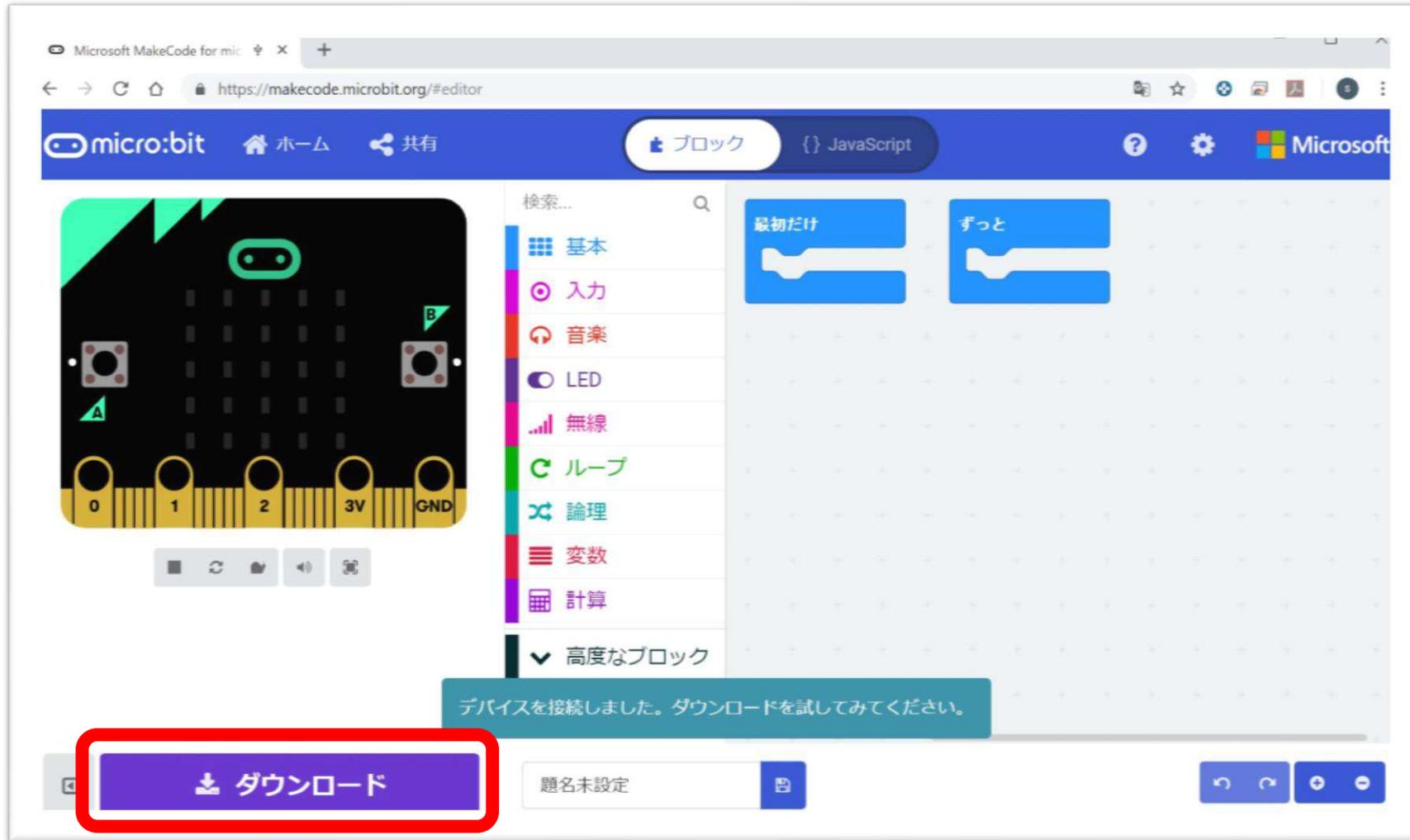
Click “Pair device” Button

How to PLEN:bit – Connect to micro:bit –



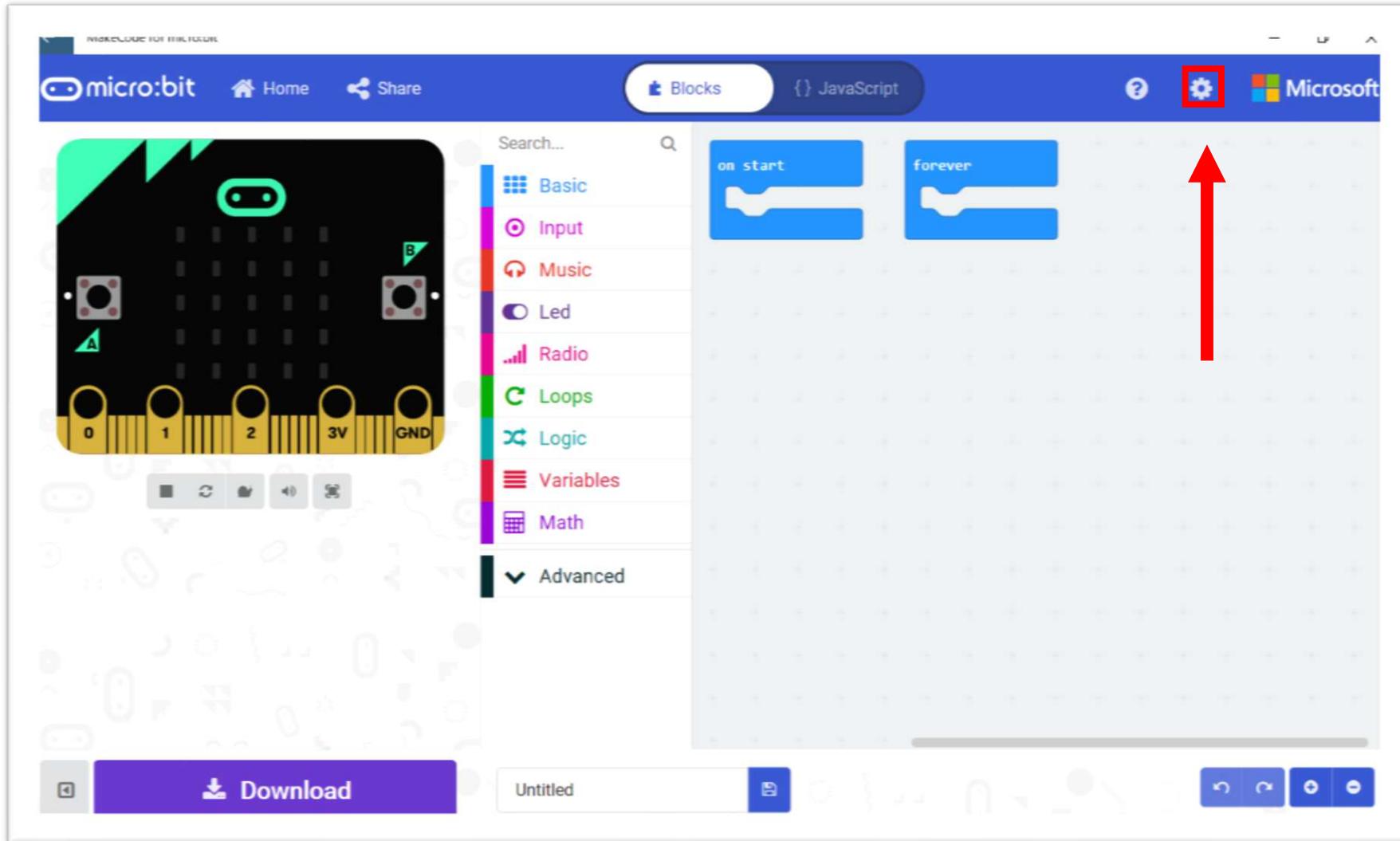
Select to micro:bit & Click “Pair Connect” Button

How to PLEN:bit – Connect to micro:bit –



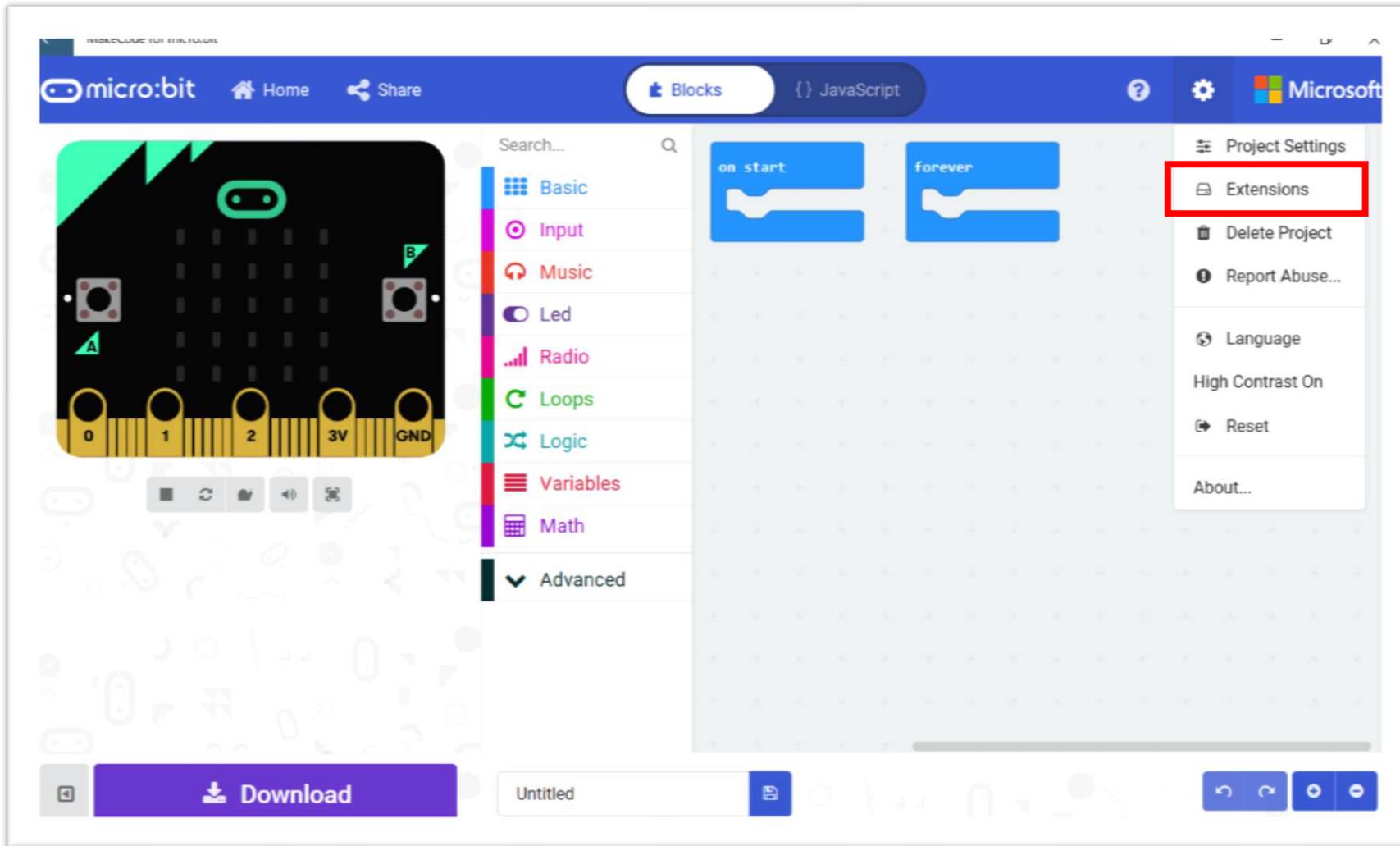
Click "Download" to confirm that the program is transferred

How to PLEN:bit – Add PLEN:bit Block –



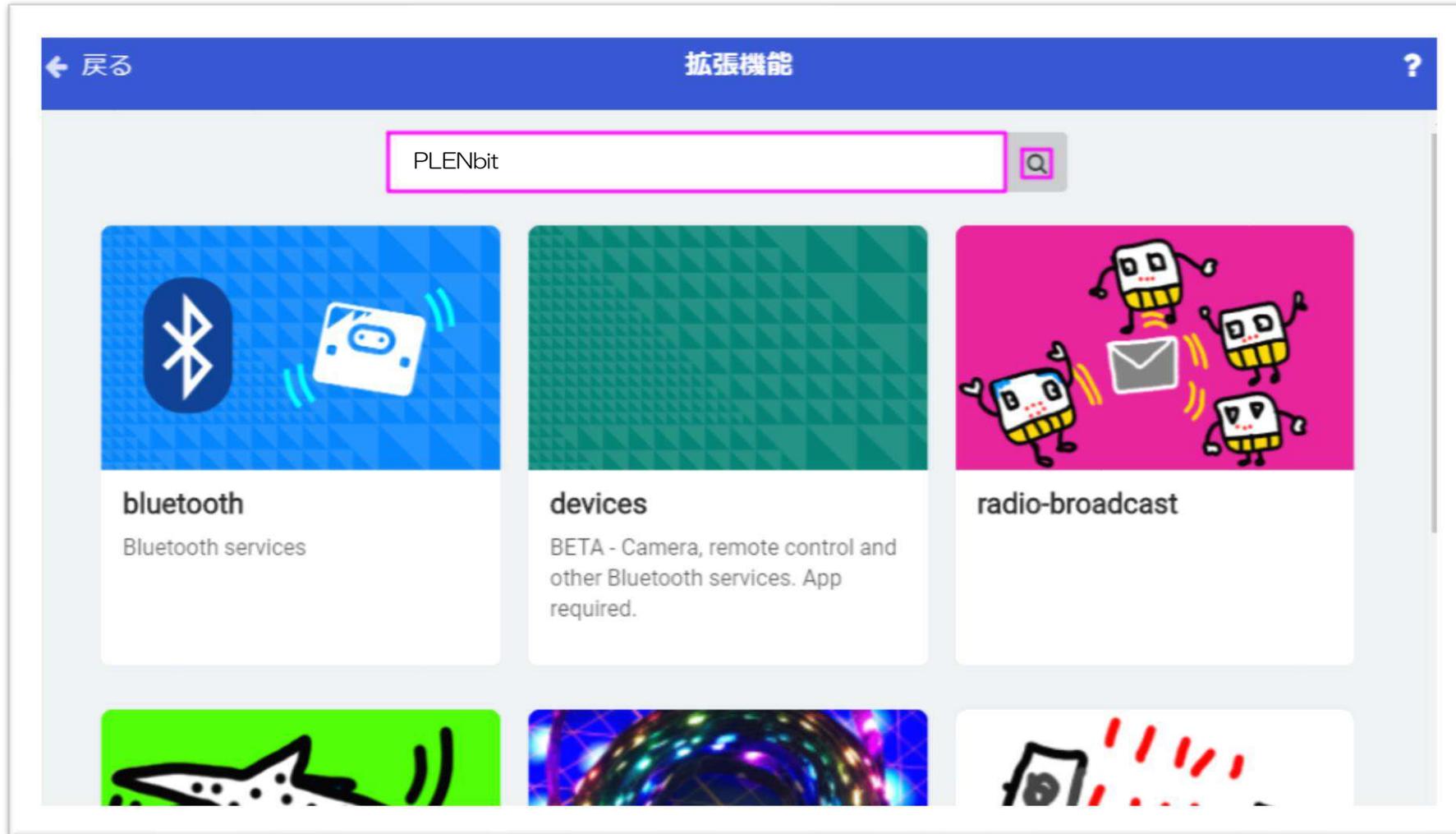
Click the "Gear Mark" at the top right inside the screen

How to PLEN:bit – Add PLEN:bit Block –



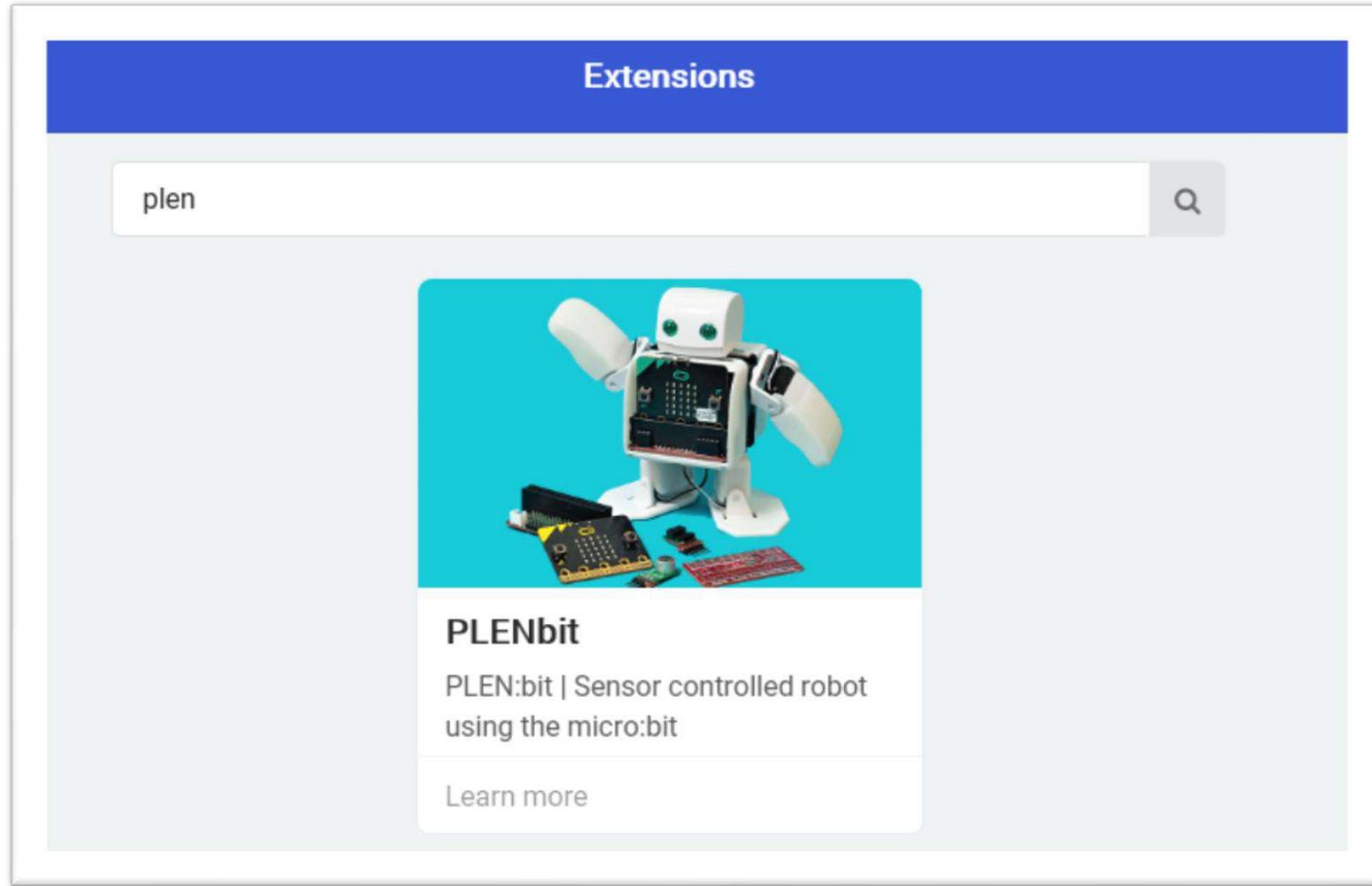
Click "Extensions" from the menu

How to **PLEN:bit** – Add PLEN:bit Block –



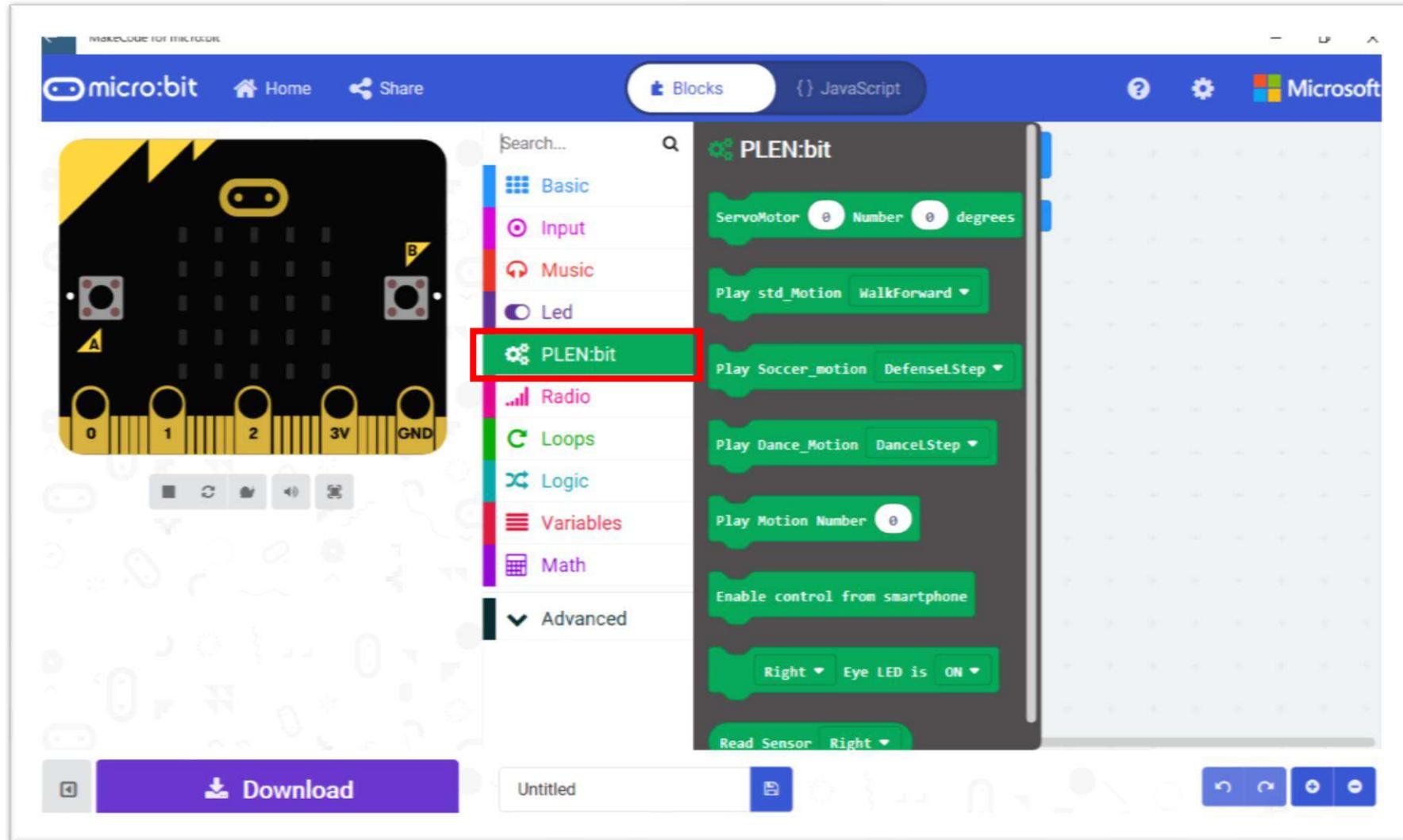
Input to “PLENbit” & Search

How to **PLEN:bit** – Add PLEN:bit Block –



Select “PLENbit” from the search results

How to **PLEN:bit** – Add PLEN:bit Block –



Block list "PLEN: bit" will be added

PLEN:bit Block List

ServoMotor Number degrees

Play std_Motion

Play Soccer_motion

Play Dance_Motion

Play Motion Number

Enable control from smartphone

eye led is

read sensor button side

- Select motor & Change degree
- Make the robot move “Standard motion”
- Make the robot move “Soccer motion”
- Make the robot move “Dance motion”
- Make the robot move the selected number
- Enable control from PLEN Connect(APP)
- PLEN:bit eye LED on / off control
- Read Sensor (A side or B side)

PLEN:bit Programming | PLEN:bit Basic

```
on start
  ServoMotor_initial
  show icon [grid icon]
```

```
on button A pressed
  Play std_Motion Walk Forward
```

```
on button B pressed
  Play std_Motion Arm PataPata
```

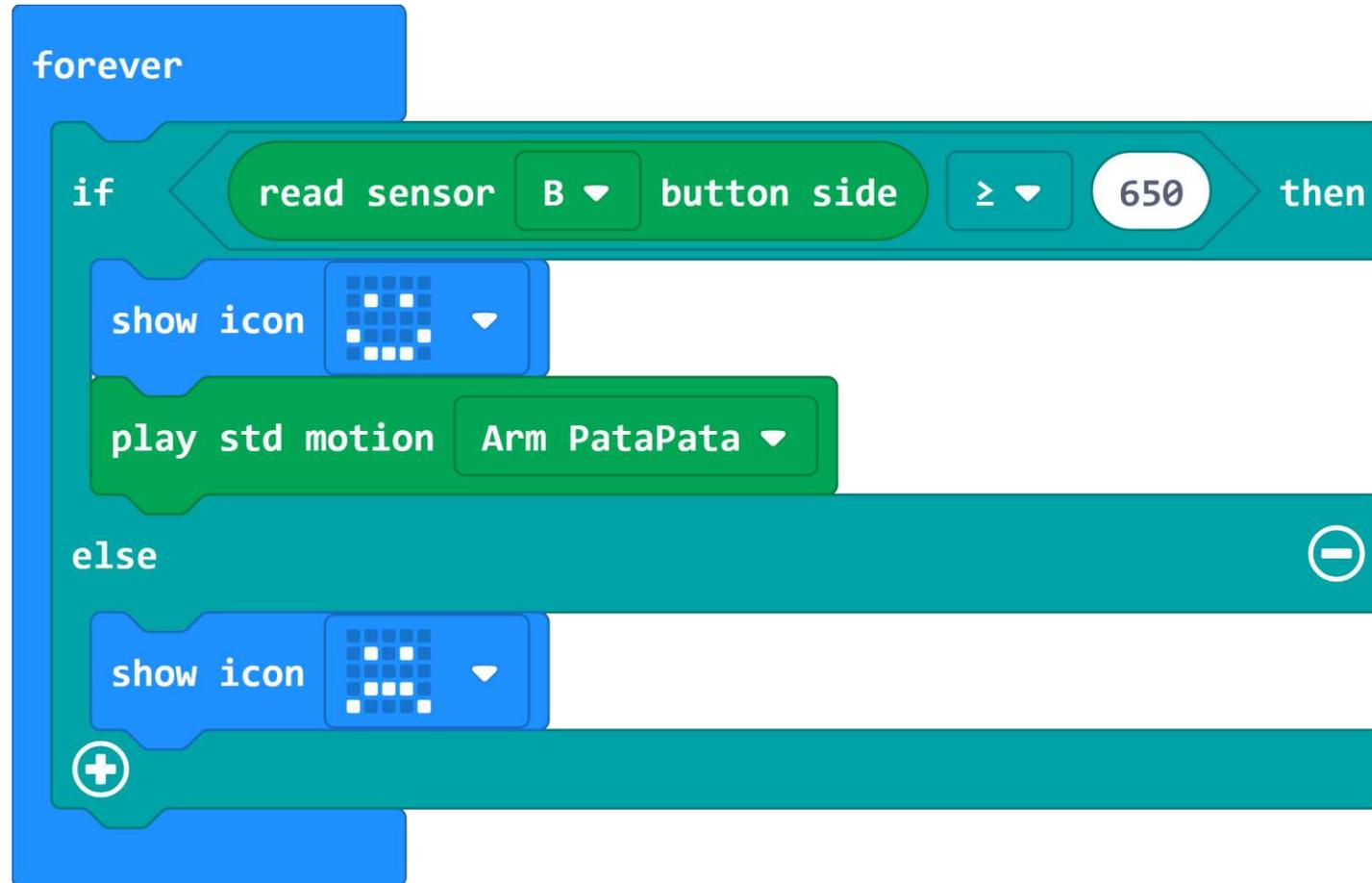
```
forever
  Eye LED is ON
  pause (ms) 100
  Eye LED is OFF
  pause (ms) 100
```

```
on button A+B pressed
  Play Soccer_motion Right Kick
```

It's a program using for assembly!

Let's program in blocks as above

PLEN:bit Programming | Distance sensor Basic



Let's program in blocks as above

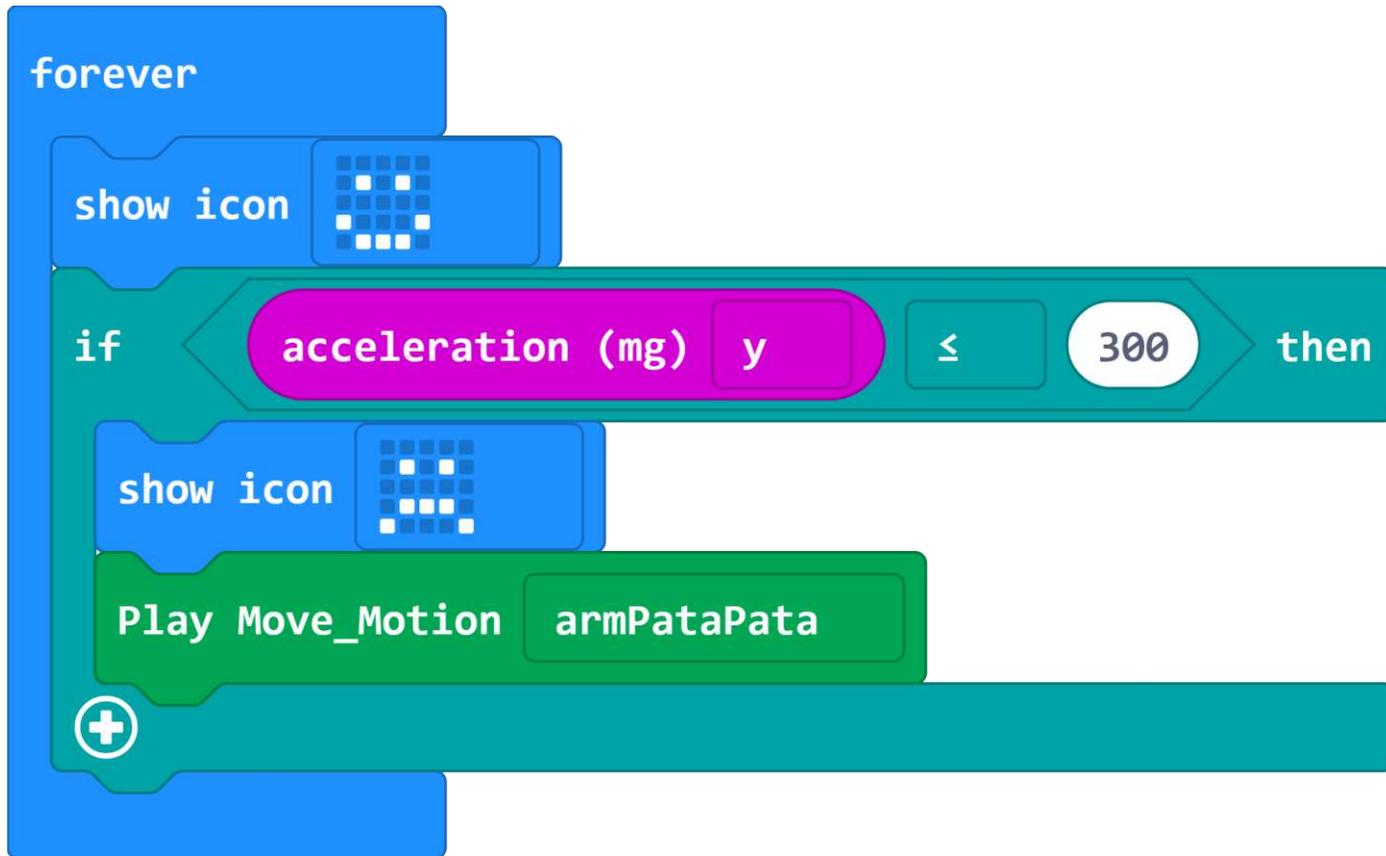
PLEN:bit Programming | Sound sensor Basic

```
on start
  show icon [grid icon]
```

```
forever
  if [read sensor A button side] >= [900] then
    show icon [grid icon]
    play std motion [Arm PataPata]
    show icon [grid icon]
```

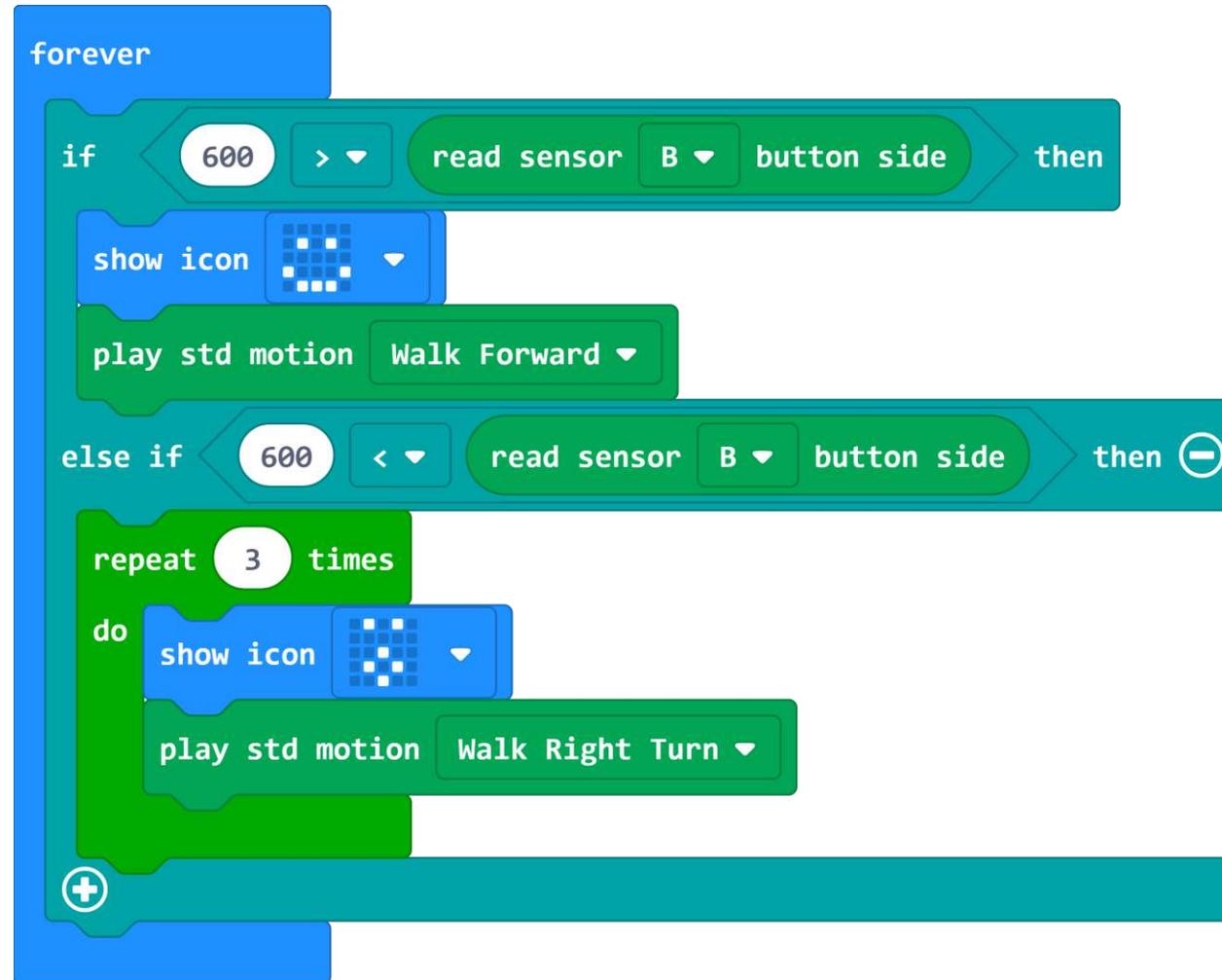
Let's program in blocks as above

PLEN:bit Programming | call help when fall down



Let's program in blocks as above

PLEN:bit Programming | Dodge the wall



Let's program in blocks as above

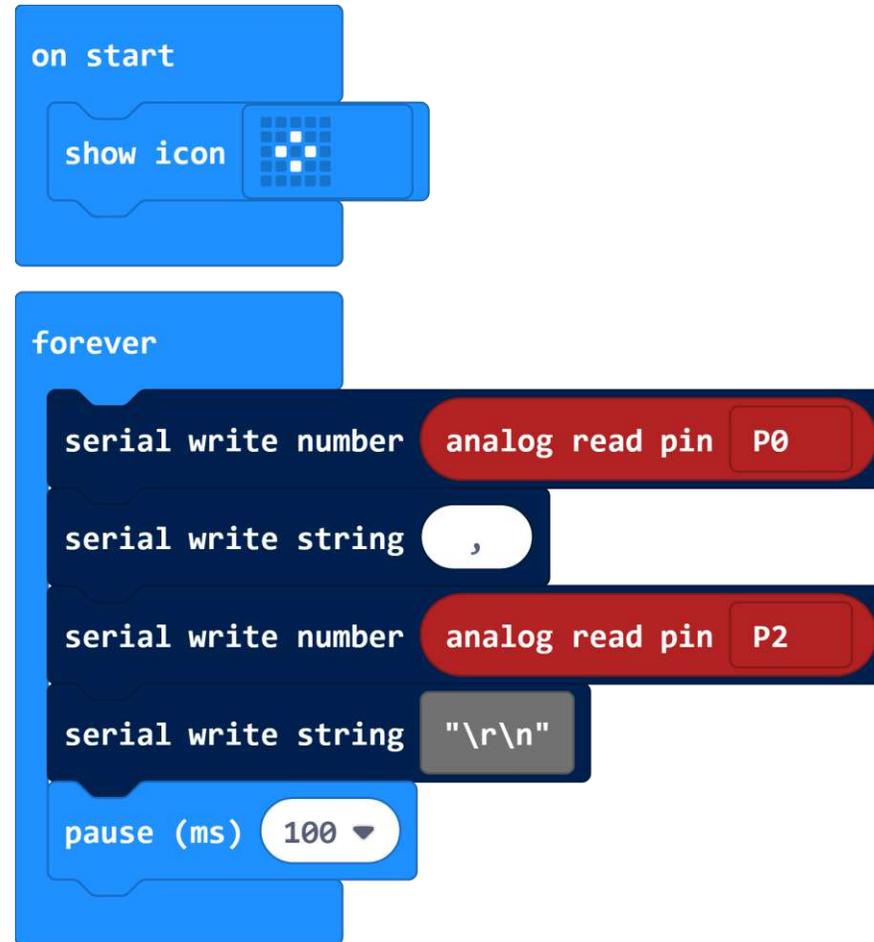
PLEN:bit Programming | Walk towards the north

```
forever
  set direction to direction
  if direction <= 20 or direction >= 340 then
    show arrow North
    Play std_Motion Walk Forward
  else if direction > 20 and direction <= 180 then
    show arrow East
    Play std_Motion Walk Left Turn
  else if direction > 180 and direction < 340 then
    show arrow West
    Play std_Motion Walk Right Turn
```

The image shows a Scratch-style block programming environment. A blue 'forever' loop block contains several conditional blocks. The first is an 'if' block with two conditions: 'direction <= 20' or 'direction >= 340'. Inside this 'if' block are a 'show arrow North' block and a 'Play std_Motion Walk Forward' block. The second is an 'else if' block with conditions 'direction > 20' and 'direction <= 180'. Inside are 'show arrow East' and 'Play std_Motion Walk Left Turn' blocks. The third is another 'else if' block with conditions 'direction > 180' and 'direction < 340'. Inside are 'show arrow West' and 'Play std_Motion Walk Right Turn' blocks. A plus sign icon is visible at the bottom left of the loop block, indicating it can be expanded.

Let's program in blocks as above

PLEN:bit Programming | Let's know value of sensor



Let's program in blocks as above

PLEN:bit Programming | Adjust ServoMotor position

```
最初だけ
サーボモーター初期値設定
アイコンを表示

ずっと
もし ボタン A が押されている なら
  関数を呼び出す servoAdjust
でなければもし ボタン B が押されている なら
  基本モーション 前に進む の再生
```

```
関数 servoAdjust
変数 adjNum を 0 にする
変数 servoNum を 0 にする
数を表示 servoNum
変数 loop を 真 にする
もし loop ならくりかえし
  もし ボタン A+B が押されている なら
    save positon servoNum adjNum
    変数 servoNum を 1 だけ増やす
    変数 adjNum を 0 にする
    数を表示 servoNum
    でなければもし ボタン A が押されている なら
      変数 adjNum を 1 だけ増やす
      変数 adjNum を servo adjust servoNum adjNum にする
    でなければもし ボタン B が押されている なら
      変数 adjNum を -1 だけ増やす
      変数 adjNum を servo adjust servoNum adjNum にする
    でなければもし servoNum > 7 なら
      アイコンを表示
      一時停止 (ミリ秒) 2000
      変数 loop を 偽 にする
```



https://makecode.microbit.org/_6WgCH61mChM7

- How to use
1. Push A to start correction
 2. Push A or B to move each servo
 3. Push A+B to switch to next servo
 4. Loop
 5. Ends when smile is displayed
 6. Reset, then Push B to walk
- If PLEN does not fall over, setting is complete

Download and use the program from the URL

PLEN:bit Programming | Receive IR



```
forever
  if read sensor A button side < 512 then
    show icon [grid icon]
    play std motion Arm PataPata
    show icon [grid icon]
  +
```

If it is IR remote control, anything can be used

Infrared detection program using an IR sensor. When you press the button on the infrared remote control, your arm will move.

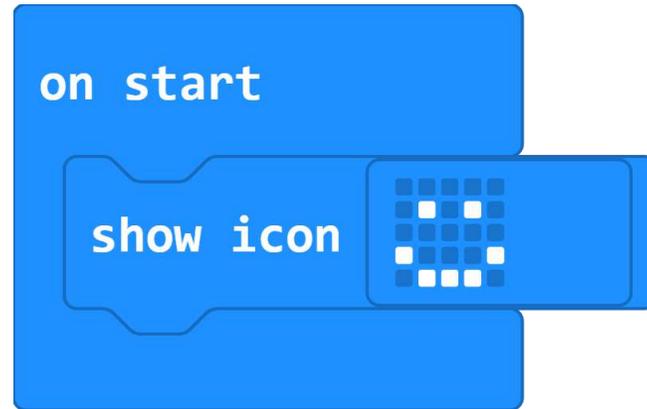
PLEN:bit Programming | Welcome!



```
forever
  if 512 < read sensor B button side then
    show icon [grid icon]
    play std motion Arm PataPata
    show icon [grid icon]
    pause (ms) 2000
  +
```

It is a human detection program using a PIR sensor. When a person moves in front of a PLEN:bit, the arm is move.

PLEN:bit How to BLE ver. head board



Download the app and use it

PLEN Connect

iPhone

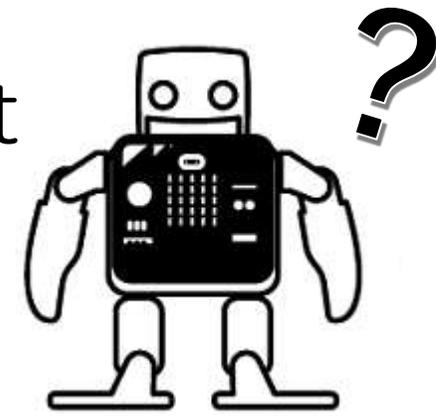
<https://itunes.apple.com/jp/app/plen-connect/id990980940?mt=8>

Android

<https://play.google.com/store/apps/details?id=jp.plen.plenconnect2>

You can control from PLEN Connect(APP)

PLEN:bit HELP! | FAQ & Support



1. PLEN:bit block is different ? ?

PLEN:bit's extensions is updated from time to time.

If something is wrong, try adding it again.

2. If you have any questions, please contact one of the following.

- PLEN:bit Slack community u0u0.net/YJzp
- PLEN Support <https://plen.jp/wp/contact/>