

# micro:bitによるプログラミング (3)

担当：稲川孝司（帝塚山学院大学）

## 実習内容

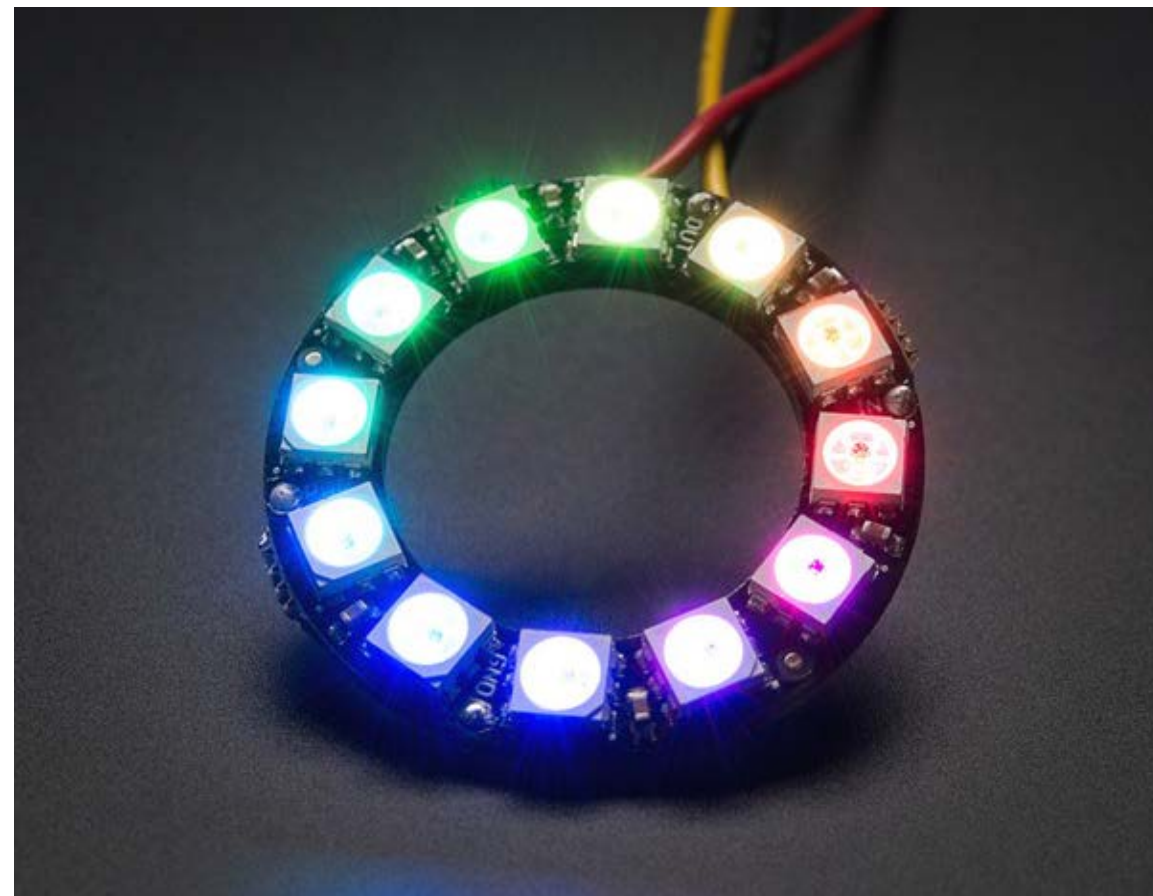
- NeoPixelによるフルカラーLED制御

## 引用・参考文献

高橋参吉、喜家村奨、稲川孝司：micro:bitで学ぶプログラミング  
ブロック型からJavaScriptそしてPythonへ、コロナ社（2019.9）。

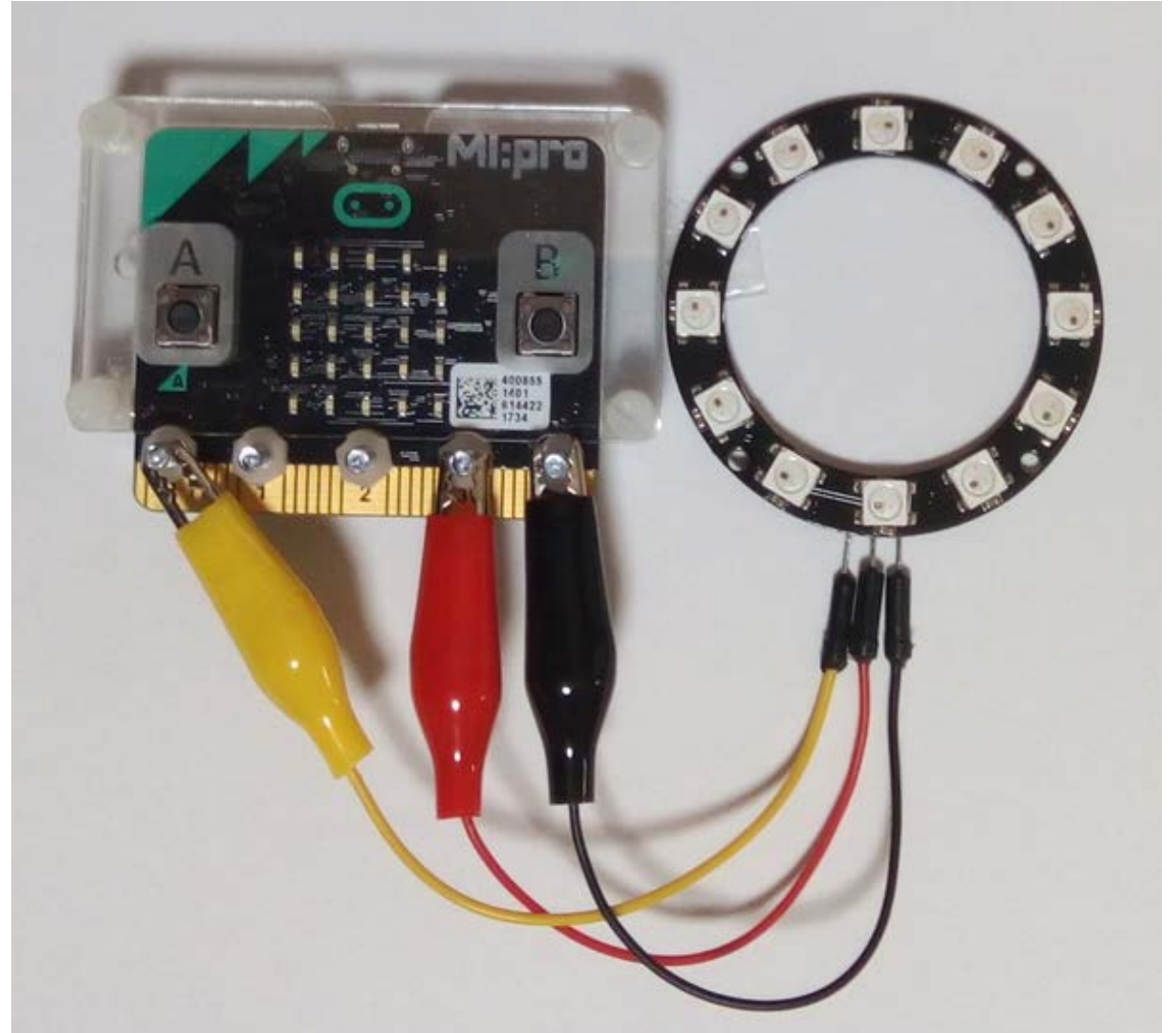
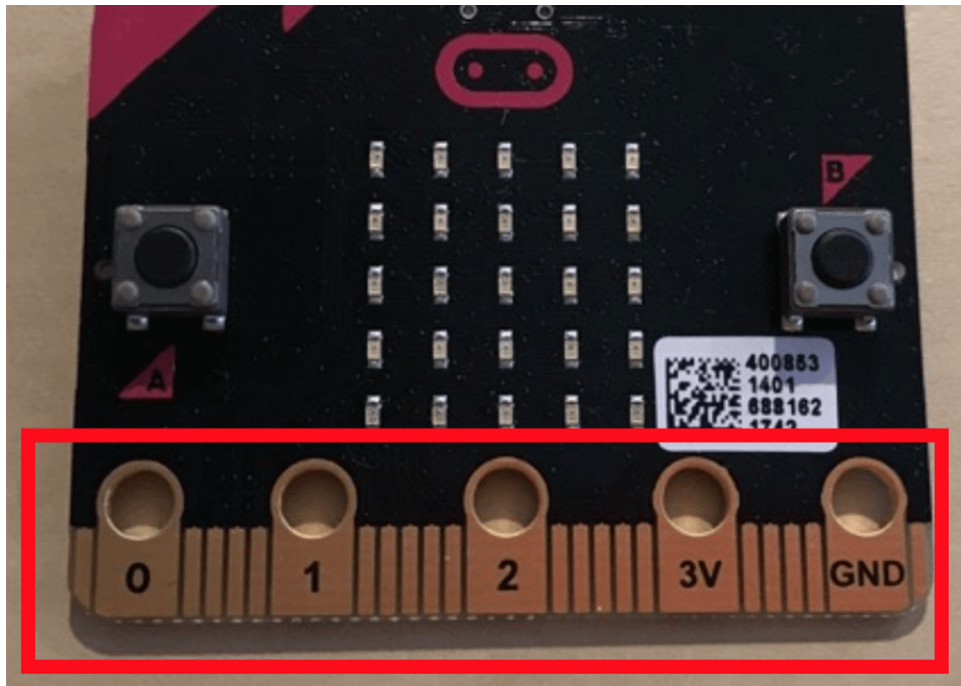
## NeoPixelとは

- NeoPixelは、1つのセルにRGBの3つのLEDとその制御回路が入っており、シリアル通信でRGBの値を送ることで、簡単に多くのLEDをフルカラーで光らせることができるLEDの集合体です。RGBそれぞれを256段階に調整でき、 $256*256*256=16777216$ 色で光ります。
- 今回は、12個のLEDがRING型に接続された製品を使って、様々な色で光らせる実習を行います。



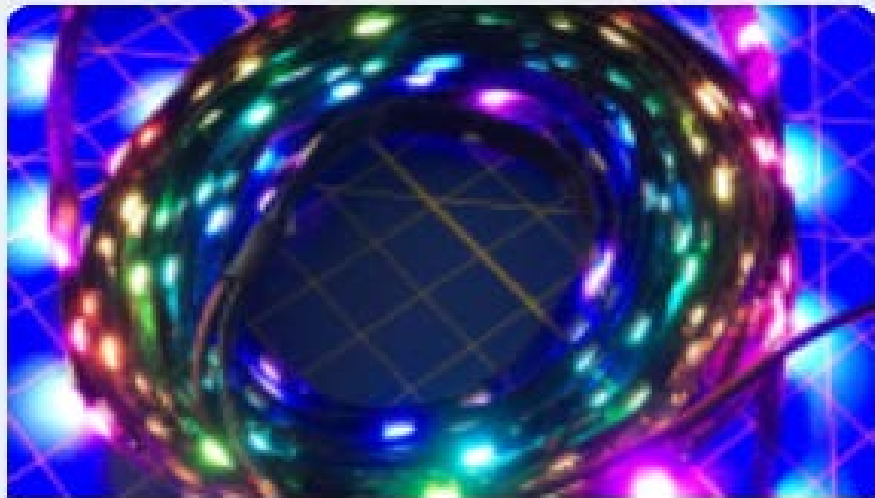
## 【準備1】 Micro:bitとNeoPixelの接続

Micro:bitとリング型NeoPixelを  
右図のように接続する。  
(黒色はGND、赤色は3V、黄色はP0)



## 【準備2】 NeoPixelライブラリーの入手

- 1) ブラウザで <https://makecode.microbit.org> サイトに行く。
- 2) 新しいプロジェクト⇒名前を「rei3-1」と入力し、作成ボタンをクリックする。
- 3) 一番下の「高度なブロック」⇒「+拡張機能」⇒「neopixel」の**写真をクリック**する。
- 4) プログラム作成画面上に、水色のNeoPixelのブロック群が追加される。



neopixel



## 【例題3-1】 NeoPixelを光らせてみよう。(ファイル名:rei3-1)

### <手順>

- 1) 新しいプロジェクト⇒名前を「rei3-1」と入力し、作成ボタンをクリックする。
- 2) 「最初だけ」ブロックの中に「変数をNeoPixelモードにする」を追加する。
- 3) NeoPixelを接続している端子はP0なので、P0であることを**確認**する。
- 4) NeoPixelの数は12個なので、ブロック内の24を12に**変更**する。
- 5) 「Neopixel」ブロックから「赤色に点灯する」ブロックを「最初だけ」に追加する
- 6) 作成したプログラムをmicro:bitに書き込み、赤色で点灯することを確認する。

最初だけ

変数 strip を 端子 P0 に接続しているLED 12 個のNeoPixel (モード RGB (GRB順) ) にする

strip を 赤 色に点灯する

24を12に変更

ここをクリックすると、好きな色に変えることができる

## 【例題3-2】 LEDを好きな色で100ミリ秒ごとに点滅させてみよう。 (ファイル名:rei3-2)

### <手順>

- 1) ファイル名を「rei3-2」と入力する。
- 2) 「最初だけ」ブロックの中に「変数をNeoPixelモードにする」ブロックをいれる
- 3) NeoPixelを接続している端子はP0なので、P0であることを確認する。
- 4) NeoPixelの数は12個なので、ブロック内の24を12に**変更**する。  
-----
- 5) 「ずっと」ブロックの中に「赤色に点灯する」ブロックを追加する。
- 6) 基本から「一時停止（ミリ秒）100」ブロックを追加する
- 7) 「赤色に点灯する」ブロックを追加し、消灯するために**色をblackに変更**する。
- 8) 再度、基本から「一時停止（ミリ秒）100」ブロックを追加する
- 9) 作成したプログラムをmicro:bitに書き込みます。

## 【例題3-2】 (ファイル名:rei3-2)

最初だけ

変数 **strip** を 端子 **P0** に接続しているLED **12** 個のNeoPixel (モード **RGB (GRB順)**) にする

ずっと

**strip** を **赤** 色に点灯する

一時停止 (ミリ秒) **100**

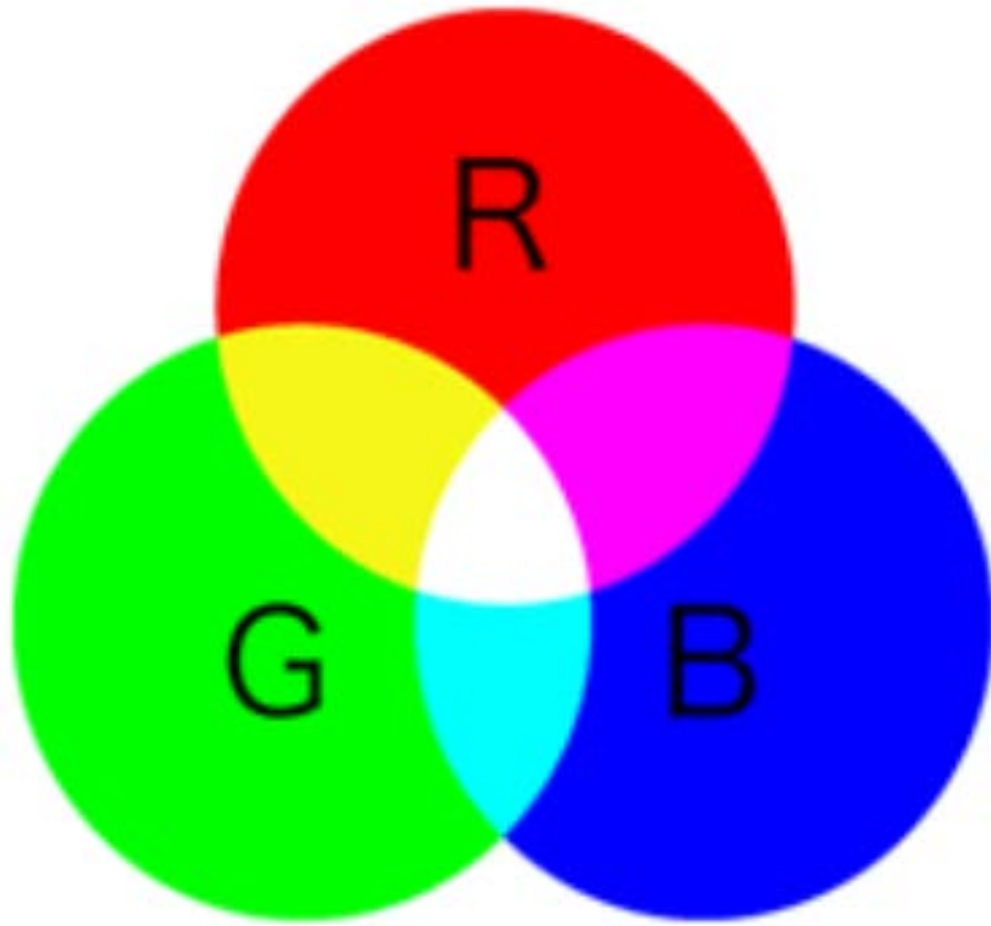
**strip** を **black** 色に点灯する

一時停止 (ミリ秒) **100**

Neopixelのその他ブロックからRGB(赤255、緑255、青255)を選ぶと、RGBそれぞれを256段階に変えて表示させることができる。

RGB (赤 **255** 緑 **255** 青 **255**)

光の3原色 R(赤) G(緑) B(青)



赤 緑 青  
256色 × 256色 × 256色  
= 16777216色  
≒ 1677万色

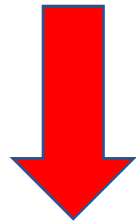
10進数 ⇔ 16進数  
0 ~ 255 ⇔ 00 ~ FF



# 原色大辞典 <https://www.colordic.org/>

floralwhite #fffaf0	blue #0000ff	palegreen #98fb98	darkorange #ff8c00	thistle #d8bfd8
linen #faf0e6	dodgerblue #1e90ff	lightgreen #90ee90	goldenrod #daa520	magenta #ff00ff
antiquewhite #faebd7	cornflowerblue #6495ed	springgreen #00ff7f	peru #cd853f	fuchsia #ff00ff
papayawhip #ffefd5	deepskyblue #00bfff	mediumspringgreen #00fa9a	darkgoldenrod #b8860b	violet #ee82ee
blanchedalmond #ffe4cd	lightskyblue #87cefa	lawngreen #7cfc00	chocolate #d2691e	plum #dda0dd
bisque #ffe4c4	skyblue #87ceeb	chartreuse #7fff00	sienna #a0522d	orchid #da70d6
moccasin #ffe4b5	lightblue #add8e6	greenyellow #adff2f	saddlebrown #8b4513	mediumorchid #ba55d3
navajowhite #ffdead	powderblue #b0e0e6	lime #00ff00	maroon #800000	darkorchid #9932cc
peachpuff #ffdab9	paleturquoise #afeeee	limegreen #32cd32	darkred #8b0000	darkviolet #9400d3
mistyrose #ffe4e1	lightcyan #e0ffff	yellowgreen #9acd32	brown #a52a2a	darkmagenta #8b008b
lavenderblush #fff0f5	cyan #00ffff	darkolivegreen #556b2f	firebrick #b22222	purple #800080

skyblue  
#87ceeb



16進数であることを示す0xを先頭につける



自動的に10進数になる

## 【コラム：LEDの番号の割り当てについて】

NeoPixelは、順番にLEDを点灯させるためにそれぞれに番号が割り当てられている。

12個のリング型の場合は、右図のように信号線に近いほうから時計回りに順に0から11が割り当てられている。



個別に色を設定して点灯する方法は、



で番号を指定し、n番目のLEDを好きな色で点灯させることができる。また、



で、点灯しているLEDを次にずらすことができる。

【例題3-3】 12個のLEDを個別に好きな色に指定して、点滅させてみよう。  
(ファイル名:rei3-3)

<手順>

- 1) ファイル名を「rei3-3」と入力する。
- 2) 「最初だけ」ブロックの中に「変数をNeoPixelモードにする」ブロックをいれる。
- 3) 今回のNeoPixelの数は12個なので、ブロック内の24を12に変更する。  
-----
- 4) 「ずっと」ブロックの中に、NeoPixelの「その他」ブロックから「stripの0番目のLEDを赤色に設定する」ブロックをいれる。
- 5) 「設定」ブロックを右クリックして複製し、計12個の「設定」ブロックを作成。
- 6) 設定する番号を0から11の順に変更し、色は好きな色に変更する。
- 7) 「stripを設定した色で点灯する」ブロックと「一時停止(ミリ秒)200」ブロックをいれる。
- 8) 「stripをblack色で点灯する」ブロックと「一時停止(ミリ秒)200」ブロックをいれる。

最初だけ

# 【例題3-3】 (ファイル名:rei3-3)

変数 strip を 端子 P0 に接続しているLED 12 個のNeoPixel (モード RGB (GRB順) ) にする

ずっと

```
strip の 0 番目のLEDを 赤 色に設定する
strip の 1 番目のLEDを だいたい 色に設定する
strip の 2 番目のLEDを 黄 色に設定する
strip の 3 番目のLEDを 緑 色に設定する
strip の 4 番目のLEDを 青 色に設定する
strip の 9 番目のLEDを あい 色に設定する
strip の 10 番目のLEDを すみれ 色に設定する
strip の 11 番目のLEDを 黄 色に設定する
```

```
strip を設定した色で点灯する
```

```
一時停止 (ミリ秒) 200
```

```
strip を black 色に点灯する
```

```
一時停止 (ミリ秒) 200
```

複製する

【例題3-4】 「1個分ずらす」ブロックを使って、点灯した緑のLEDを右回転させてみよう。  
(ファイル名:rei3-4)

<手順>

- 1) ファイル名を「rei3-4」と入力する。
- 2) 「最初だけ」ブロックの中に「変数をNeoPixelモードにする」ブロックをいれる  
-----
- 3) 「ずっと」ブロックの中に、NeoPixelの「その他」ブロックから「0番目のLEDの色を設定する」ブロックをいれる。
- 4) ループから「くりかえし」ブロックを入れ、回数を12回に設定する。
- 5) 「くりかえし」ブロックの中に、NeoPixelから「指定した色で点灯する」ブロックをいれる。
- 6) さらに、基本から「一時停止(ミリ秒)50」ブロックをいれる。
- 7) そのブロックの下に、NeoPixelから「色をLED1個分ずらす」ブロックをいれる。

# 【例題3-4】 (ファイル名:rei3-4)

最初だけ

変数 strip を 端子 P0 に接続しているLED 12 個のNeoPixel (モード RGB (GRB順) ) にする

ずっと

strip の 0 番目のLEDを 緑 色に設定する

くりかえし 12 回

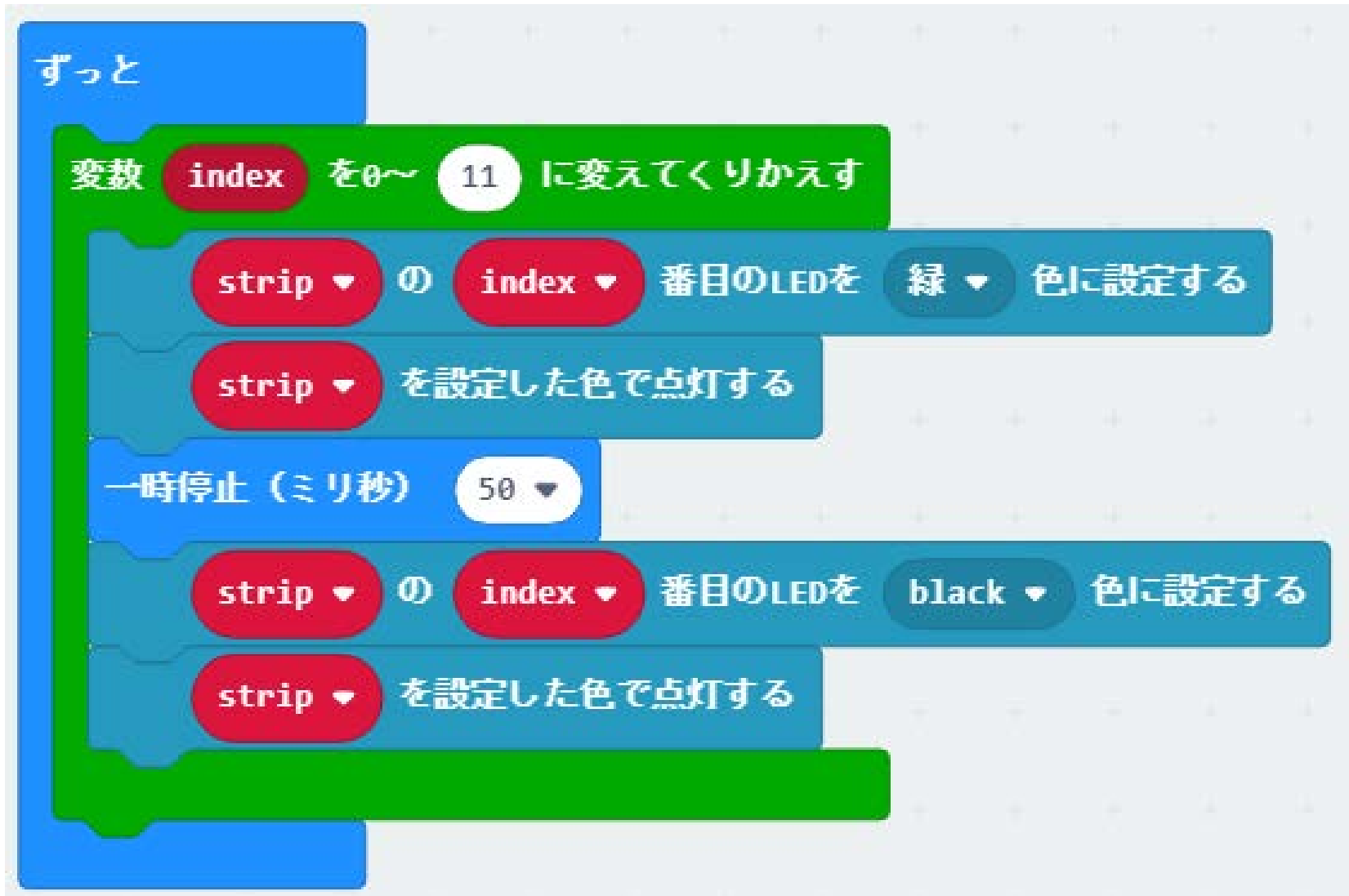
strip を設定した色で点灯する

一時停止 (ミリ秒) 50

strip に設定されている色をLED 1 個分ずらす



【例題3-4別解】（演習課題1） 「ループ」からの繰り返しのブロックと変数を使って、例題3-4と同じ動きをするプログラムを作成してみよう。  
（ファイル名:rei3-4b）※最初だけブロックは省略している



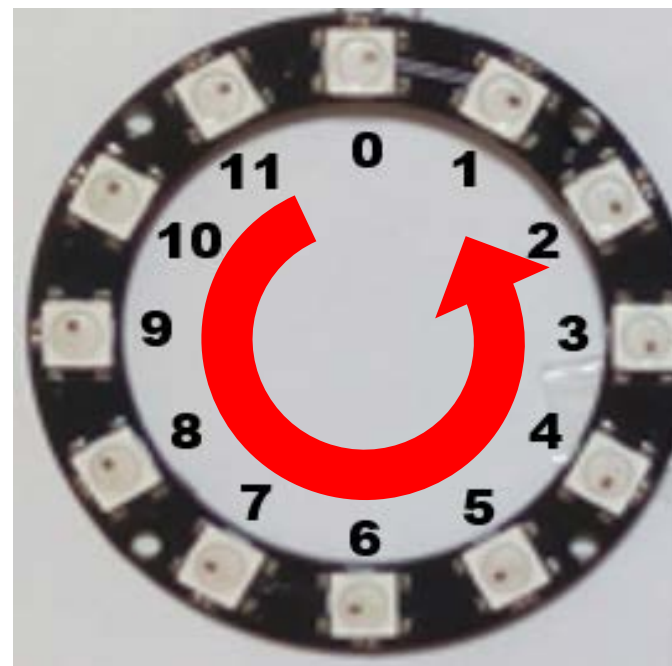


【例題3-5】 1つだけLEDを点灯させて、左回転させてみよう。

(ファイル名:rei3-5)

(※例題3-4において、11番目から0番目が点灯するように、次の2か所を修正すればよい。) ※最初だけブロックは省略している

The image shows a Scratch script for controlling an LED strip. The script is enclosed in a blue 'ずっと' (Forever) loop block. The first block is a 'strip' block with '11' circled in red, indicating the LED index to be set to '緑' (Green). This is followed by a green 'くりかえし' (Repeat) block set to '12' times. Inside the repeat loop, there are three blocks: 'strip' block with 'を設定した色で点灯する' (Turn on with the set color), a blue '一時停止 (ミリ秒)' (Pause) block set to '50', and another 'strip' block with '-1' circled in red, indicating the index to be shifted by -1. A blue callout box labeled '変更箇所' (Change location) has two red arrows pointing to the circled '11' and '-1'.



【例題3-5別解】（演習課題2）例題3-4別解を利用して、左回転させるために、0⇒11ではなく、11⇒0の順に点灯させるように、繰り返しブロックの変数を工夫してみよう。

（ファイル名:rei3-5b） ※最初だけブロックは省略している



The image shows a Scratch script for controlling an LED strip. It starts with a 'ずっと' (Forever) loop. Inside the loop, there is a '変数 index を0～11 に変えてくりかえす' (Repeat with variable index from 0 to 11) block. The script then performs two actions: setting the LED at index 11 to green and then setting the LED at index 11 to black. A red arrow points from a blue callout box labeled '変更箇所' (Change location) to the '11' value in both the 'strip' blocks, indicating the change from 0 to 11.

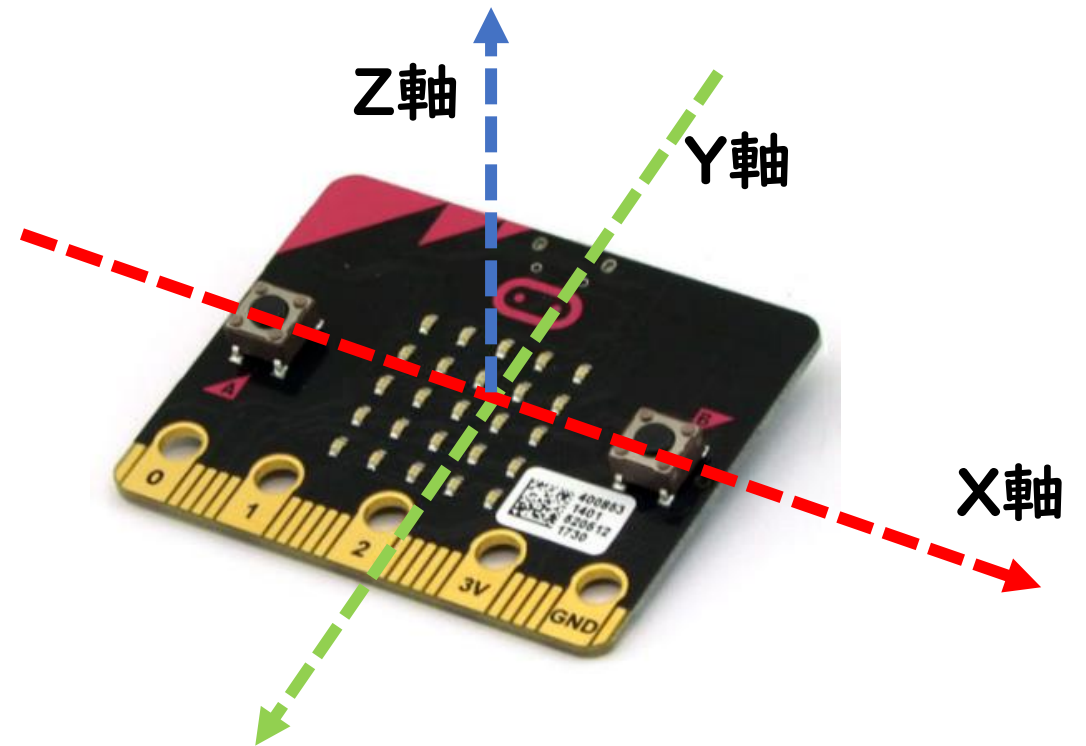
```
ずっと  
変数 index を0～11 に変えてくりかえす  
strip の 11 番目のLEDを 緑 色に設定する  
strip を設定した色で点灯する  
一時停止 (ミリ秒) 50  
strip の 11 番目のLEDを black 色に設定する  
strip を設定した色で点灯する
```

## 【コラム：加速度センサについて】

micro:bitには、加速度センサが搭載されており、図のようにx,y,zの3方向の加速度を計測することができます。

地上では下向きの重力加速度があるので、それを計測してmicro:bitを3次元的に動かすだけで値を変化させることができます。

また、動きや振動、衝撃なども検知することができます。



**【例題3-6】** micro:bitを傾けて、搭載されている加速度センサの3つの値を読み込んでRGBの3つの色に対応したLEDを点灯させ、micro:bitを動かして色を変えてみよう。  
(ファイル名:rei3-6)

**<手順>**

- 1) 「最初だけ」ブロックの中に「変数をNeoPixelモードにする」ブロックをいれる。  
-----
- 2) ずっとブロックの中に、NeoPixelのその他から「stripの0番目を赤色にする」ブロックを入れます。
- 3) 赤の部分を、NeoPixelのその他の「RGB(赤255、緑255、青255)」に置き換える
- 4) 数値の255の部分を加速度X、加速度Y,加速度Zに置き換える。
- 5) Neopixelから「Stripを指定した色で点灯する」ブロックをいれる。
- 6) 基本から一時停止(ミリ秒) 100をいれる
- 7) NeoPixelから「stripに設定されている色を1個分ずらす」ブロックを入れる。

## 【例題3-6】 (ファイル名:rei3-6)

最初だけ

変数 `strip` を 端子 `P0` に接続しているLED 12 個のNeoPixel (モード `RGB (GRB順)`) にする

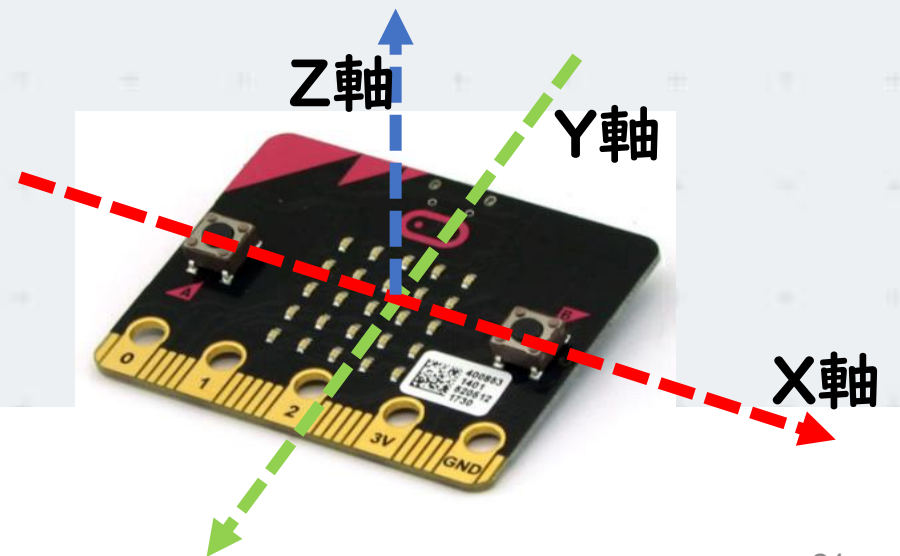
ずっと

`strip` の 0 番目のLEDを RGB (赤 `加速度 X` 緑 `加速度 Y` 青 `加速度 Z`) 色に設定する

`strip` を設定した色で点灯する

一時停止 (ミリ秒) 100

`strip` に設定されている色をLED 1 個分ずらす



【例題3-7】 レインボーパターンで光らせてみよう。  
NeoPixelのライブラリの中に「レインボーパターンで点灯する」という命令があり、それを使うと、レインボーパターンで光らせることができます。  
(ファイル名:rei3-7)

<手順>

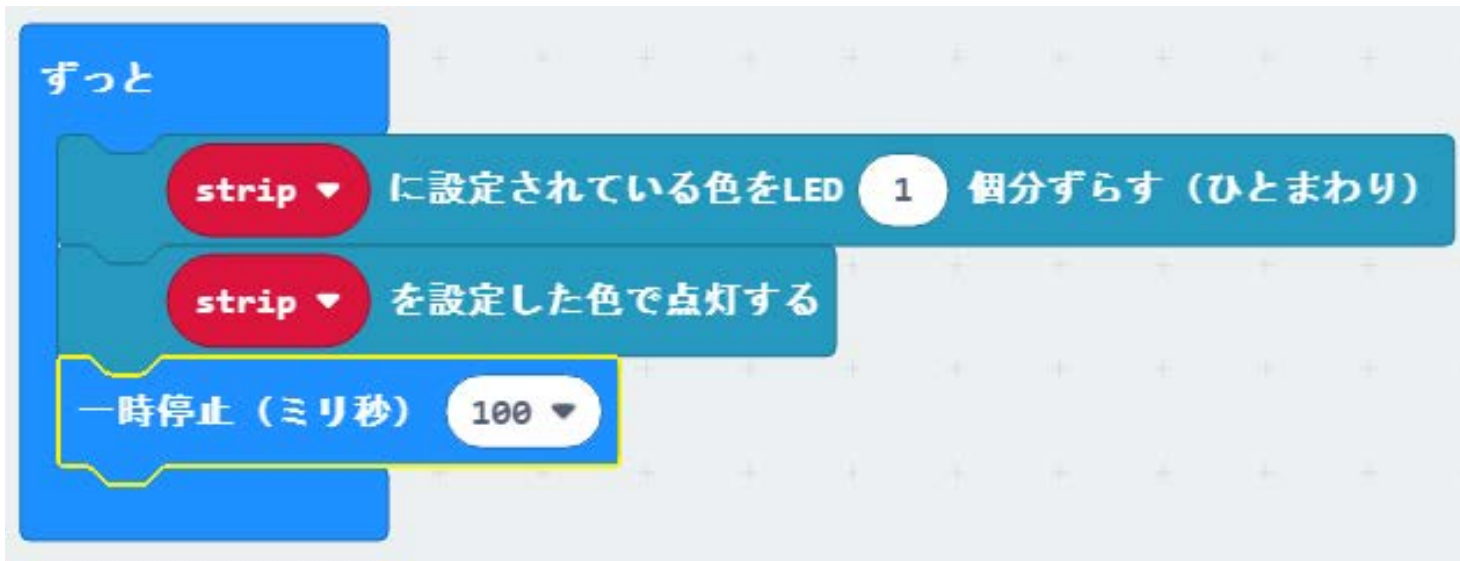
- 1) 「最初だけ」ブロックの中に、「変数をNeoPixelモードにする」ブロックをいれる。
- 2) 今回のNeoPixelの数は12個なので、ブロック内の24を12に変更する。
- 3) 次に「Stripをレインボーパターンに点灯する」ブロックをいれる。



【例題3-8】 レインボーパターンで右回転させよう。  
例題3-7ではレインボーパターンの色が固定されていたので、「ずっと」ブロックの中に「1個分ずらす」ブロックを追加して、色が順番に右回転するようにしてみよう。 (ファイル名:rei3-8)

<手順>

- 1) 「最初だけ」のブロックは、例題3-7をそのまま使います。
- 2) 「ずっと」ブロックの中に、Neopixelから「LED1個分ずらす (ひとまわり)」ブロックと「設定した色で点灯する」ブロックをいれます。
- 3) 基本から「一時停止 (ミリ秒) 100」ブロックをいれます。



【例題3-9】 レインボーパターンで左回転させよう。  
例題3-8ではレインボーパターンで右回転したので、ここでは左回転する  
ようにしてみよう。 (ファイル名:rei3-9)

<手順>

- 1) ブロックは、例題3-8を**基本的にそのまま使います**。
- 2) ずっと」ブロックの中の「LED 1個分ずらす (ひとまわり)」を「**-1**個分ずらす」に**変更**します。

The image shows a Scratch script for controlling an LED strip. It is divided into two sections: '最初だけ' (Initially) and 'ずっと' (Forever).

**最初だけ (Initially):**

- 変数 strip を 端子 P0 に接続しているLED 12 個のNeoPixel (モード RGB (GRB順)) にする
- strip をレインボーパターン (色相 1 から 360) に点灯する

**ずっと (Forever):**

- strip に設定されている色をLED **-1** 個分ずらす (ひとまわり)
- strip を設定した色で点灯する

A red circle highlights the '-1' value in the 'ずっと' block, with a red arrow pointing to it from a blue callout box labeled '変更箇所' (Change location).



## プログラムのダウンロードについて

### <プログラムに対する注意事項>

本テキストで利用している例題プログラムなどは、NPO法人学習開発研究所の下記のWebサイトからダウンロードしてください。

本書の中で記載しているJavaScriptのファイル名、例えば、rei〇〇は、保存ファイル名では、microbit-rei〇〇.hex、 になっています。

<http://www.u-manabi.org/microbit/>

