

# ワークショップ

## micro:bit でペンライトを光らせよう

### 大阪公立大学 稲川孝司

## 1. はじめに

最近、コンサートで観客が曲に合わせてペンライトを振ることが多くなっている。当初は、観客が1色で光るものを複数個持参し、曲に合わせて各自が選択した色で振っていた。その後、1本で複数の色で光るペンライトができ、曲に合わせて手元で色を選択して、ペンライトを振っている。しかし、初心者にとっては、どの場面でどの色のペンライトを選び、それを使ってどのように振舞えばよいかよくわからない。

そこで、micro:bit と NeoPixel を使って、様々な色で光るペンライトプログラムを工夫して作成し、その後、管理者が全体を制御して、統一感のあるライティングができるペンライト制御を行う無線での送・受信プログラムを作成し、実際に演奏に参加する。

**なお、シミュレータが良くできており、本物がなくても同等に動作する。**

## 2. micro:bit で NeoPixel を動かす

### 2.1 micro:bit と NeoPixel の接続

NeoPixel の DIN 端子に micro:bit の P0 信号線、5 VDC 端子に micro:bit の 3 V、GND 端子に micro:bit の GND を接続します (図 1)。

なお、micro:bit の基板上の 3V 端子から流せる電流は  $V1 < 90\text{mA}$   $V2 < 200\text{mA}$  なので、直接駆動する場合は、NeoPixel の数の上限に注意すること。

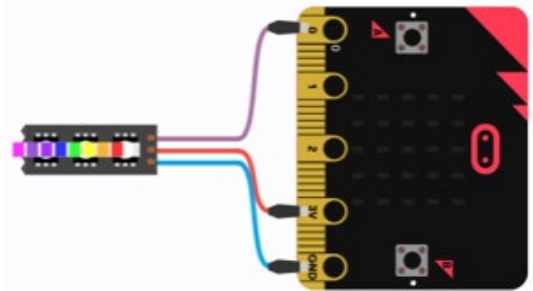


図 1 NeoPixel と micro:bit の接続

### 2.2 NeoPixel ライブラリの追加

micro:bit で NeoPixel を利用するにはライブラリの追加が必要で、まず ([MakeCode for micro:bit](#)) に行き、ツールボックスの下にある「拡張機能」(図 2)をクリックして、表示されたネット上の拡張機能から「NeoPixel」(図 3)を探して、クリックしてライブラリを追加する。ライブラリが追加されると、図 4 に示す NeoPixel の命令が使えるようになる。

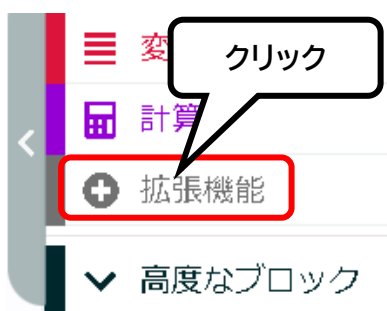


図 2 拡張機能



図 3 NeoPixel ライブラリ



図 4 追加された命令群

### 3 基本的な NeoPixel プログラムの作成

#### 【例 1】 NeoPixel 全体を赤色で光らせてみよう。(rei-1)

- 1) 新しいプロジェクト⇒名前を「rei1」と入力し、作成ボタンをクリックする。
- 2) 「NeoPixel」ブロックから「変数を NeoPixel モードにする」を「最初だけ」の中に追加する。
- 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 8 個なので、ブロック内の 24 を 8 に変更する。
- 5) 「NeoPixel」ブロックから「strip を赤色に点灯する」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。



#### 【例 2】 NeoPixel 全体を好きな色で 500 ミリ秒ごとに点滅させてみよう。(rei-2)

- 1) ファイル名を「rei-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 8 個なので、ブロック内の 24 を 8 に変更する。
- 5) 「ずっと」ブロックの中に「strip を赤色に点灯する」ブロックを追加する。
- 6) 基本から「一時停止 (ミリ秒) 500」ブロックを追加する
- 7) 「strip を赤色に点灯する」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) 500」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、正しく点滅することを確認する。



【例3】 NeoPixel の 8 個の LED を、個別に様々な色で点灯させてみよう。(rei-3)



図7 例3のプログラム

※「strip の”0”番目の LED を”赤色”に設定する」ブロックでは、何番目にある LED を何色に点灯させるかを設定する。この時、「●番目」の値は、1 ではなく **0 から始まる** ことに注意！ micro:bit と接続してある側が「0 番目」です。

【例4】 NeoPixel で光らせる色を自分で作ってみよう。(rei-4)

RGB の値はボタンを押して作ります。

A ボタンで R(赤)、B ボタンで G(緑)、A+B ボタンで B(緑)の値を変えて、好きな色で光らせてみます。

動作は、各ボタンを押すと、RGB の数値が 10 ずつ増えて、micro:bit の画面にグラフ表示され、その RGB の値で、NeoPixel が光ります。

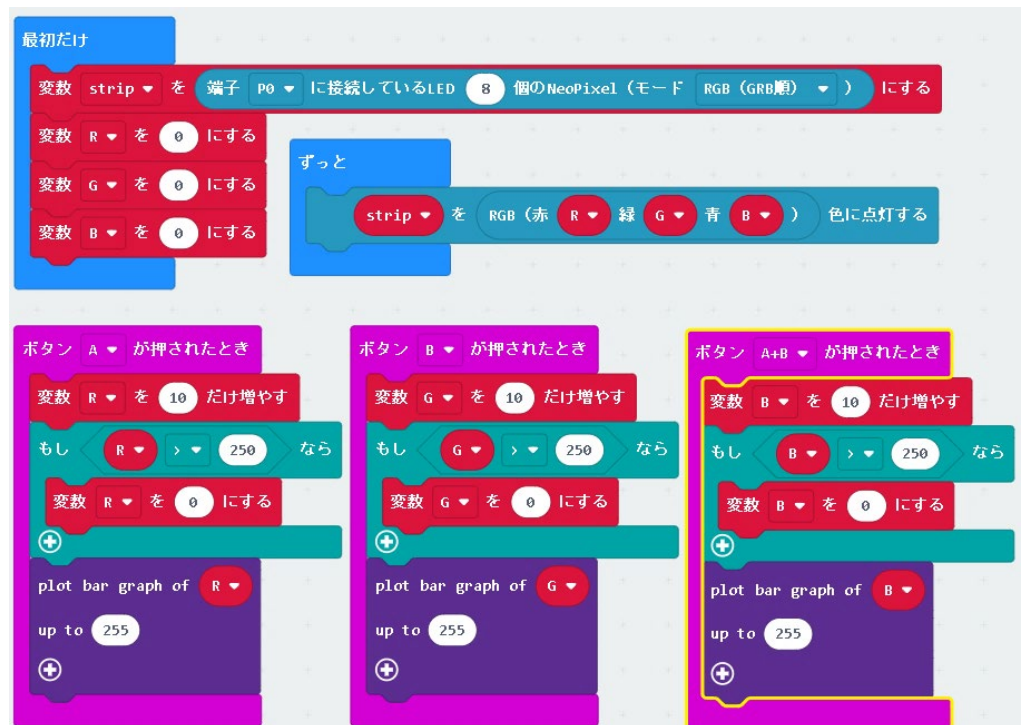


図8 例4のプログラム

**【例 5】 配列を利用して NeoPixel の 8 個をすべて同じ色で光らせる。**

最初は赤色で、A ボタンを押すごとに、赤、だいたい、黄、緑、青、あい、紫、白、黒、緑、の順に点灯させ、また、B ボタンを押すと逆順で色を光らせることもできるように設定する。プログラムは、最初だけのブロック内で色のデータを配列に記録させておき、変数 S を使って A ボタンを押すと値が 1 増えて次の色に、B ボタンを押すと値が 1 減って前の色に戻るように、配列の場所を指定して色のデータを取り出し、その色で NeoPixel を発光させている。(rei-5)

なお、配列の要素を RGB (赤、青、緑) にして、フルカラーから選ぶこともできる。

The image shows a Scratch-like block-based programming environment. The code is as follows:

- 最初だけ (Initially):**
  - 変数 strip を 端子 P0 に接続している LED 8 個の NeoPixel (モード RGB (GRB順)) にする
  - 変数 s を 0 にする
  - 変数 max を 9 にする
  - 変数 配列 を [赤, だいたい, 黄, 緑, 青, あい, すみれ, 紫, 白, black] にする
- ずっと (Forever):**
  - 数を表示 s
  - strip を 配列 の s 番目の値 色に点灯する
- ボタン A が押されたとき (When button A is pressed):**
  - 変数 s を 1 だけ増やす
  - もし s > max なら
    - 変数 s を 0 にする
- ボタン B が押されたとき (When button B is pressed):**
  - 変数 s を -1 だけ増やす
  - もし s < 0 なら
    - 変数 s を max にする

Two callout boxes are present:

- Microbit の配列の添字は0から始まる (Microbit array indices start from 0)
- 配列の要素は LED の色 (Array elements are LED colors)

図9 例5のプログラム

【例6】NeoPixelのライブラリに「レインボーパターンに点灯する」という命令がある。その命令を使って、ボタンAを押したら下向きに、ボタンBを押したら上向きに、レインボーパターンで移動させながら光らせてみよう。(rei-6)

```

最初だけ
  変数 strip を 端子 P0 に接続しているLED 8 個のNeoPixel (モード RGB (GRB順) ) にする

ボタン A が押されたとき
  変数 dir を 1 にする

ボタン B が押されたとき
  変数 dir を -1 にする

ずっと
  strip をレインボーパターン (色相 1 から 360 ) に点灯する
  くりかえし 8 回
    strip に設定されている色をLED dir 個分ずらす (ひとまわり)
    strip を設定した色で点灯する
    一時停止 (ミリ秒) 200
  
```

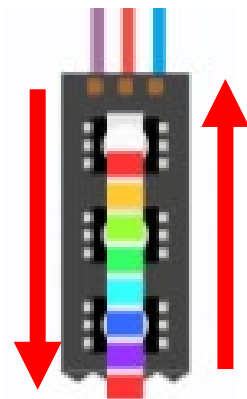


図10 例6のプログラム

【例7】例6で、8個のLEDをそのままレインボーパターンで点灯させると色相を単調なので、レインボーパターンの色相をnからn+180に細かく分割して光らせ、滑らかなレインボーカラーで点灯させてみよう。(rei-7)

```

最初だけ
  変数 strip を 端子 P0 に接続しているLED 8 個のNeoPixel (モード RGB (GRB順) ) にする

ずっと
  変数 n を 0 ~ 360 に変えてくりかえす
    strip をレインボーパターン (色相 n から n + 180 ) に点灯する
    一時停止 (ミリ秒) 30
    変数 n を 10 だけ増やす
  
```



図11 例7のプログラム

## 4. ペンライト受信機のプログラム(receive)

様々な色で点灯可能なペンライトを、コンサート会場で観客が好き勝手に使用されると会場の雰囲気も白けるため、最近では管理者が無線でデータを送り、統一したライティングをするようになってきた。ここでは、無線を使ったペンライト受信機を作成する。  
事前準備：無線のグループと NeoPixel の設定をする。

動作1：管理者から無線で送られてきた数値（0～9）を画面に表示する。

動作2：送られてきた数値ごとの処理（機能）プログラムを作成する。

動作3：ユーザに拡張機能（A ボタン、B ボタン）が使えるようにする。

（例：変数:time を 900 に設定し、A ボタンが押されている間は time を 300 にする）

（例：変数:dir を 1 に設定し、B ボタンが押されている間は dir を -1 にする）

最初だけ

無線のグループを設定 1

変数 strip を 端子 P0 に接続しているLED 8 個のNeoPixel (モード RGB (GRB順)) にする

無線で受信したとき receivedNumber

数を表示 receivedNumber

変数 s を receivedNumber にする

ずっと

変数 time を 900 にする

もし ボタン A が押されている なら

変数 time を 300 にする

+

変数 dir を 1 にする

もし ボタン B が押されている なら

変数 dir を -1 にする

+

ずっと

もし s = 0 なら

strip をレインボーパターン (色相 1 から 360) に点灯する

でなければもし s = 1 なら

でなければもし s = 2 なら

でなければもし s = 3 なら

でなければもし s = 4 なら

でなければもし s = 5 なら

でなければもし s = 6 なら

でなければもし s = 7 なら

でなければ

ここに、各自が作成したプログラムを書く

拡張機能例

グループ毎に異なる番号を割り振る

動作チェック用

図 1 2 受信プログラム (microbit-penlight-recvie)

【s=0の例】NeoPixel全体をレインボーパターンで点灯する。(動作チェック用)



図13 S=0のプログラム例

【s=1の例】NeoPixel全体を赤、だいたい、黄、緑、青の順に、色を指定してtimeで定義された時間点灯し、色を1つずつ変更する。

拡張機能で変数timeを定義しているので、一時停止の時間がそのままでは900ms、Aボタンを押すと300msと短くなり、Aボタンを押している間は色の変化が早くなる

【s=2の例】レインボーパターンで点灯し、色を1つずつずらす。

8個のLEDをレインボーパターン命令そのもので点灯すると色が単調になってしまうので、レインボーパターンの色相をnからn+180で定義して光らせ、360まで繰り返して光らせることで、滑らかなレインボーカラーで順に点灯させている。

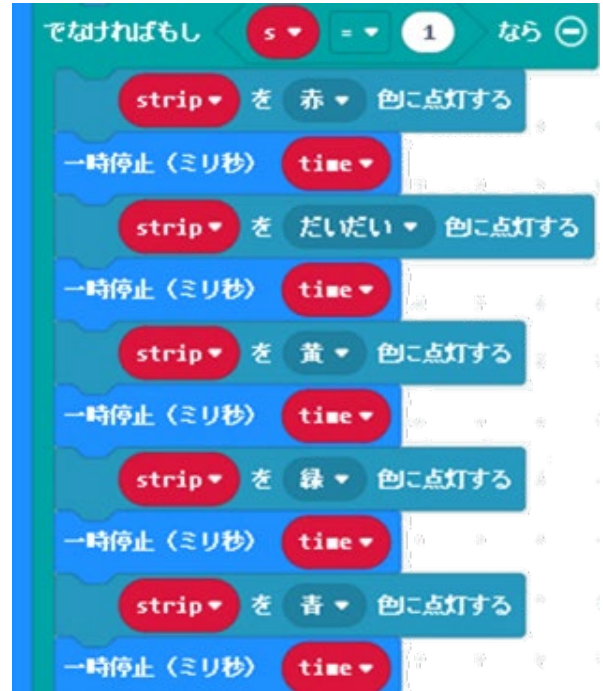


図14 S=1のプログラム例



図15 S=2のプログラム例

※グループ内で相談して、S=3からS=9までのプログラムを分担して作成する。

## 5. ペンライト送信機のプログラム(send)

設定：無線のグループを他のグループとは異なる数値（0～255）に設定する。

動作：機能の数値（0～9）を画面に表示し、A ボタンで手前に、B ボタンで次に移動。A+B ボタン（両方同時押し）で決定して、機能の数値を無線でグループ内の相手に送信する。micro:bit の画面表示は、2文字以上になると文字列をスクロールして見にくくなるので、機能の数値は1桁のみに制限している。

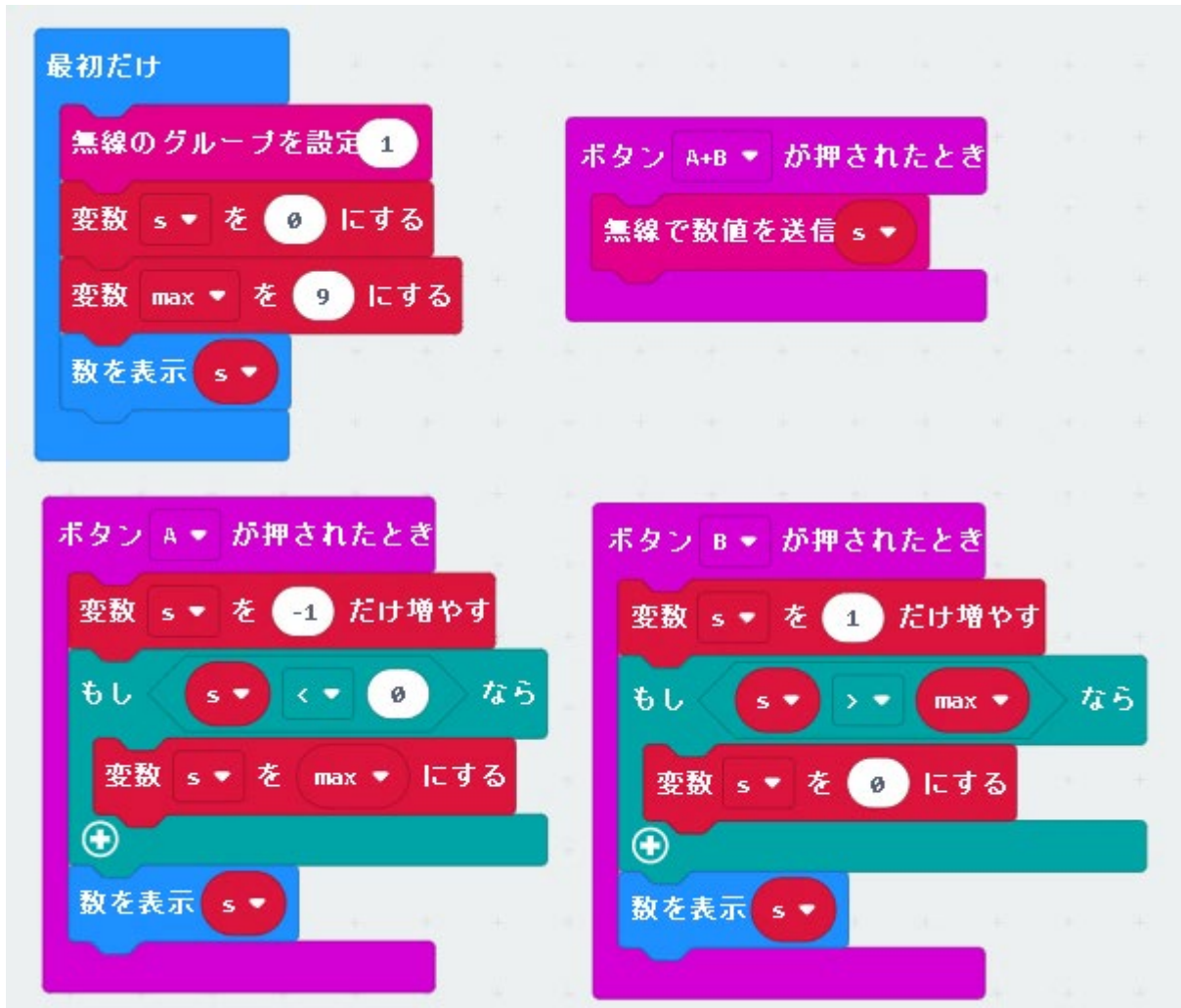


図 1 6 送信プログラム 1 (microbit-penlight-send)

## 6. ペンライト演奏参加

1. みんなで、曲を決める
2. どのタイミングで、どのような動作のライティングをするかを皆で相談して決める。
3. ペンライトソフトを動かしてリハーサルを行う。
4. 実際に演奏に参加する（1名は操作）