

小中学生のための micro:bit を利用した

プログラミング教室（第 1 回）資料

氏 名 _____

主催 NPO 法人 学習開発研究所

後援 京田辺市教育委員会

講習(2)：無線通信で信号機を制御してみよう（テーマ 8）

日 時：2025 年 7 月 27 日(日) 13:30～16:00

場 所：京田辺市中央公民館 2F 第 3・4 研修室

問い合わせ先：ild-kemsyu@u-manabi.org

担当者：NPO 法人 学習開発研究所 三輪、高橋

テーマ 8 無線通信で信号機を制御してみよう

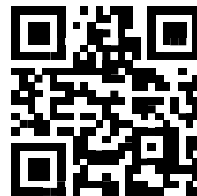
目 次

- | | |
|------------------------|--------------------------------|
| 1. micro:bit のプログラム | プログラム prei1-1,prei1-2 |
| 2. プログラムの基本(順次構造、反復構造) | プログラム prei2-1,prei2-2,prei2-3 |
| 3. プログラムの基本(分岐構造) | プログラム prei2-41,prei2-4,prei2-5 |
| 4. 信号機の点灯と制御 | プログラム prei8-1,prei8-2,prei8-3 |
| 5. 無線通信による信号機の制御 | プログラム prei8-5,prei8-6 |

注) 例題番号は、「小中学生のためのプログラミング～学習ガイド」の例題番号に対応しています。
本文中の例題の右端の、「☆」は応用、「★」は発展、「無印」で基礎の問題を示しています。

<小中学生のためのプログラミング教室>

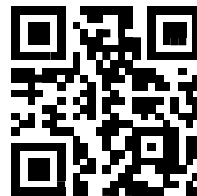
<https://u-manabi.net/ild-pkouza/>



<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)

<https://u-manabi.net/microbit/>



1. micro:bit のプログラム

【例題 1-1】 ハートの表示 (プログラム prei1-1)

「最初だけ」ブロックを利用して、ハートを表示してみよう。



「最初だけ」ブロック

*「基本」-「LED 画面に表示」LED をタップして、ハート形に見えるように LED を ON にする。

【例題 1-2】 ハートの点滅 (プログラム prei1-2)

「ずっと」ブロックに変更して、ハートを点滅してみよう。



「ずっと」ブロック

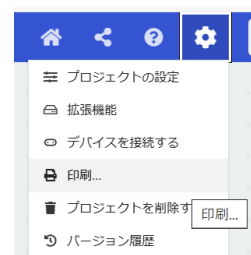
*「基本」-「一時停止」時間を 500 (ミリ秒) にする。

*「表示を消す」

*「基本」-「一時停止」時間を 500 (ミリ秒) にする。

<参考> プログラムの印刷

エディタの画面 (右上端) にある歯車 (⚙️) から「印刷」を選択する。少し経過して「プログラムを印刷する」の画面が表示され、ブロックのプログラムを印刷できる。



2. プログラムの基本(順次構造、反復構造)

【例題 2-1】 アイコンの表示 (プログラム prei2-1)

3つのアイコン(「ハート」「小さいダイヤモンド」「しかく」)を表示してみよう。

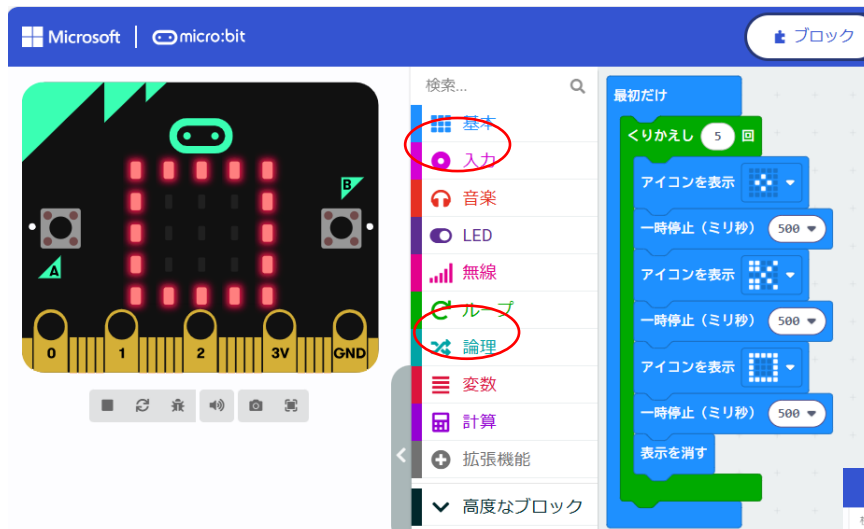


「最初だけ」ブロック

- *「基本」-「アイコン表示」(ハート)、「基本」-「一時停止」
- *「基本」-「アイコン表示」(小さいダイヤモンド)、「基本」-「一時停止」
- *「基本」-「アイコン表示」(しかく)、「基本」-「一時停止」
- *「基本」-「表示を消す」

【例題 2-2】 アイコンの繰り返し表示 (プログラム prei2-2)

アイコンを 3 つ(「小さいダイヤモンド」「はさみ」「しかく」)を繰り返し 5 回表示してみよう。



「最初だけ」ブロック

- *「基本」-「アイコン表示」の「はさみ」を選択する。
 - *「ループ」-「くりかえし」数値 0 を 5 にする。
- じゃんけんの「グー」「チョキ」「パー」をこの 3 つのアイコンで表示する。
- 反復構造は、繰り返し構造ともいう。



【例題 2-3】 カウンターの利用 (プログラム prei2-3)

「変数 カウンター」を利用して、表示してみよう。



「最初だけ」ブロック

*「ループ」の「変数 カウンター ~ くりかえす」数値 0 を 4 にする。

変数「カウンター」は、「0 から始まり、「0, 1, 2, 3, 4」の 5 回繰り返す。

注) 変数は数値や文字を入れるための領域

3. プログラムの基本 (分岐構造)

【例題 3-1】 「グー」「パー」の表示 (プログラム prei2-41)

変数 $c=0$ のときは、「グー」を表示し、 $c=1$ ときは、「パー」を表示してみよう。



「最初だけ」ブロック

*「変数」-「変数を追加する」変数 c にする。

*「変数」-「変数 c を 0 にする」

*「論理」-「条件判断」(もし なら ~ でなければ ~)

*「論理」-「くらべる」「 $0 = \nabla 0$ 」を「 $c = \nabla 0$ 」にする。

分岐構造は、選択構造ともいう。

順次構造、反復構造(繰り返し構造)、分岐構造(選択構造)をプログラムの基本構造という。

【例題 3-2】乱数の利用 (プログラム prei2-4)

乱数を利用して、変数 c に 0 から 1 の数値を入れるように変更してみよう。

「ずっと」ブロック

- *「変数」-「変数を追加する」
- *変数の名前を c にする。
- *「計算」-「0~10 までの乱数」数値の 10 を 1 にする。
(乱数は 0、1 のいずれか)

【演習】「グー」「チョキ」「パー」の表示 (プログラム prei2-5)

例題 3-2 のプログラムを「グー」「チョキ」「パー」を表示するように変更してみよう。

1 を追加する

チョキを追加する

ブロックを追加する

<参考> micro:bit の Python への自動変換プログラム

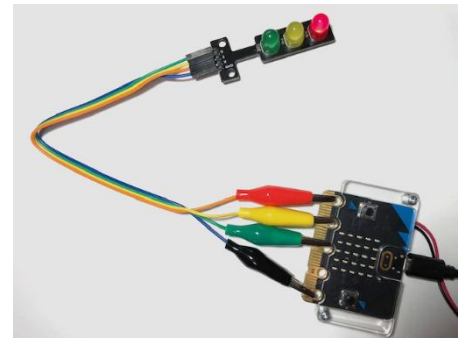
(1) 反復構造 (prei2-3) <pre>for カウンター in range(5): basic.show_icon(IconNames.SMALL_DIAMOND) basic.pause(500) basic.show_icon(IconNames.SCISSORS) basic.pause(500) basic.show_icon(IconNames.SQUARE) basic.clear_screen()</pre>	(2) 分岐構造 (prei3-1) <pre>c = 0 if c == 0: basic.show_icon(IconNames.SMALL_DIAMOND) else: basic.show_icon(IconNames.SQUARE)</pre>
---	--

4. 信号機の点灯と制御

micro:bit 用の信号機は、図のような、赤、黄、緑の LED が 3 つ付いたものである。図の信号機は、自作のものであるが、市販品もある。

micro:bit との接続は、P0 端子に赤の LED、P1 端子に黄の LED、P2 端子に緑の LED を接続する。接続線の色や「みのむしクリップ」も、図のように、同じ色にしておくと、わかりやすい。

P0:赤 P1:黄 P2:緑 GND:黒



後半は、無線通信の機能を使って、自動で信号機の制御を行う。

【例題 4-1】信号の点滅(プログラム prei8-1)

信号の赤、黄、緑の LED を 1 秒ごとに、点灯、消灯を繰り返すプログラムを作成してみよう。



「ずっと」ブロック

「高度なブロック」の「入出力端子」の「デジタルで出力する 端子」を使って制御する。

- *デジタルで出力する端子 P0 値 1 → 赤点灯(1 秒間)
- *デジタルで出力する端子 P0 値 0 → 赤消灯
- *デジタルで出力する端子 P1 値 1 → 黄点灯(1 秒間)
- *デジタルで出力する端子 P1 値 0 → 黄消灯
- *デジタルで出力する端子 P2 値 1 → 緑点灯(1 秒間)
- *デジタルで出力する端子 P2 値 0 → 緑消灯

【例題 4-2】スイッチボタンの利用 (プログラム prei8-2)

A ボタンを押したとき、信号の赤を 1 秒間点灯し消灯、それを 3 回繰り返す (点滅)、その後、緑を点灯させるプログラムを作成してみよう。なお、赤の点滅のときには、アイコンの「バツ」、緑の点灯のときには、アイコンの「ハート」を表示する。



「最初だけ」ブロック

*デジタルで出力する端子 P0 値 1 → 赤点灯

「ボタン A」

*アイコン「バツ」を表示

*デジタルで出力する端子 P0 値 0 → 赤消灯 (1 秒間)

デジタルで出力する端子 P0 値 1 → 赤点灯 (1 秒間)

を 3 回繰り返す。

*アイコン「ハート」を表示

*デジタルで出力する端子 P2 値 1 → 緑点灯 (6 秒間)

*デジタルで出力する端子 P2 値 0 → 緑消灯

*表示を消す。

【例題 4-3】2 台の信号機の点滅 (プログラム prei8-3)

つぎのような信号機の点滅のプログラムを作成してみよう。

A ボタンを押したときは、緑を 6 秒間点灯し、黄を 1 秒間点灯し消灯、それを 3 回繰り返す (点滅)。その後、赤を 4 秒間点灯させる。B ボタンを押したときは、赤を 6 秒間点灯し、黄を 1 秒間点灯し消灯、それを 3 回繰り返す (点滅)。その後、緑を 4 秒間点灯させる。

この例の信号機は、直交する交差点で衝突しない様にするためのプログラムの例である。2 台の micro:bit にダウンロードして、A さんは A ボタンを押して、B さんは B ボタンを押して、2 人で点滅の状態を確かめてみるとよい。



「最初だけ」ブロック

*すべてのデジタルで出力する端子 (P0、P1、P2) の値は、「0」(赤、緑、黄:消灯)にする。

「ボタン A」 *デジタルで出力する端子 P2 値 1 → 緑点灯(6 秒間) デジタルで出力する端子 P2 値 0 → 緑消灯 *デジタルで出力する端子 P1 値 1 → 黄点灯(1 秒間) デジタルで出力する端子 P1 値 0 → 黄消灯(1 秒間) を 3 回繰り返す。 *デジタルで出力する端子 P0 値 1 → 赤点灯(4 秒間) *デジタルで出力する端子 P0 値 0 → 赤消灯	「ボタン B」 *デジタルで出力する端子 P0 値 1 → 赤点灯(6 秒間) デジタルで出力する端子 P0 値 0 → 赤消灯 *デジタルで出力する端子 P1 値 1 → 黄点灯(1 秒間) デジタルで出力する端子 P1 値 0 → 黄消灯(1 秒間) を 3 回繰り返す。 *デジタルで出力する端子 P2 値 1 → 緑点灯(4 秒間) *デジタルで出力する端子 P2 値 0 → 緑消灯
--	--

<参考文献>

高橋、喜家村、稲川: micro:bit で学ぶプログラミング、pp.45-52、コロナ社(2019)

5. 無線通信による信号機の制御

【例題 5-1】信号機の点灯(プログラム prei8-5) ☆

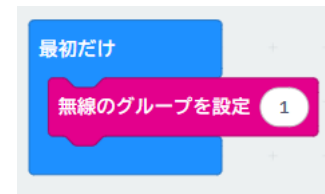
例題 4-1 の考え方に、無線通信の機能を追加することによって、自分の状態を相手に送信し、相手が受信することによって、同じ色(赤、黄、緑)の LED を点灯・消灯するプログラムを作成してみよう。

注) micro:bit ヘダウンロードする場合は、例題 5-2 のプログラムで行う。

「最初だけ」ブロック

*「無線」-「無線グループの設定」を選択し、初期設定の 1 にしておく。

注) いくつかのグループで行うときは、無線のグループ番号を変え、2 台以外は同じ番号にしないこと。



「ずっと」ブロック

4-1 のプログラムにおいて、赤、黄、緑の LED の点灯の順序と時間を変更している。また、LED の点灯後は、無線で、緑は「2」、黄は「1」、赤は「0」の数値を送る。

*デジタルで出力する端子 P2 値 1 → 緑点灯 (6 秒間)、無線で数値 2 を送る。

*デジタルで出力する端子 P2 値 0 → 緑消灯

*デジタルで出力する端子 P1 値 1 → 黄点灯 (2 秒間)、無線で数値 1 を送る。

*デジタルで出力する端子 P1 値 0 → 黄消灯

*デジタルで出力する端子 P0 値 1 → 赤点灯 (4 秒間)、無線で数値 0 を送る。

*デジタルで出力する端子 P0 値 0 → 赤消灯

「無線で受信したとき」

*「無線」-「Receive」の「無線で受信したとき receivedNumber」を選択する。

*この変数「receivedNumber」は、変数名がある箇所を論理ブロックの箇所にドラッグ&ドロップで入力する

*デジタルで出力する端子 すべて (P0、P1、P2) を「0」にする。

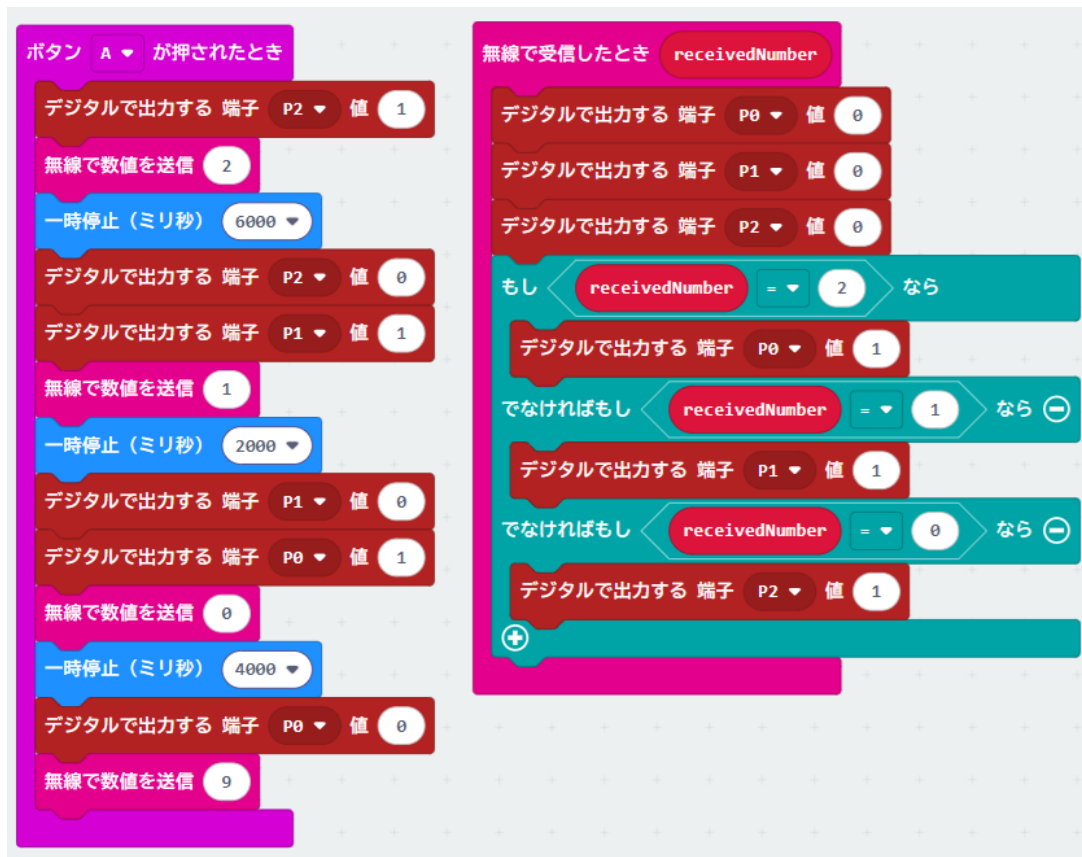
*receivedNumber が 2 のとき、*デジタルで出力する端子 P2 値 1 → 緑点灯

*receivedNumber が 1 のとき、*デジタルで出力する端子 P1 値 1 → 黄点灯

*receivedNumber が 0 のとき、*デジタルで出力する端子 P0 値 1 → 赤点灯

【例題 5-2】信号機の制御(プログラム preii8-6)★

例題 5-1 のプログラムに、A ボタンを押して、自分の状態を相手に送信したとき、相手は、交差点の信号機を想定し、異なる色の LED (赤のときは緑、黄のときは同じく黄、黄のときは赤) を点灯・消灯するプログラムにしてみよう。



「最初だけ」ブロック

- *「無線」-「無線グループの設定」を 1 にしておく。
- *デジタルで出力する端子をすべて(P0、P1、P2)を「0」にする。
- *「無線で数値を送信 9」(9 は、0、1、2 以外の数値であればよい)を送ることにより、相手側のデジタルで出力する端子もすべて(P0、P1、P2)を「0」(初期設定)にする。

「ボタン A」

最後の無線で数値を送信 9」を追加している。

- *無線で数値 9 を送り、相手側のデジタルで出力する端子を初期設定する。

「無線で受信したとき」

- *「無線」-「Receive」の「無線で受信したとき receivedNumber」を選択する。
- *デジタルで出力する端子 すべて(P0、P1、P2)を「0」にする。
- *receivedNumber が 2 のとき、*デジタルで出力する端子 P0 値 1 → 赤点灯
- *receivedNumber が 1 のとき、*デジタルで出力する端子 P1 値 1 → 黄点灯
- *receivedNumber が 0 のとき、*デジタルで出力する端子 P2 値 1 → 緑点灯

