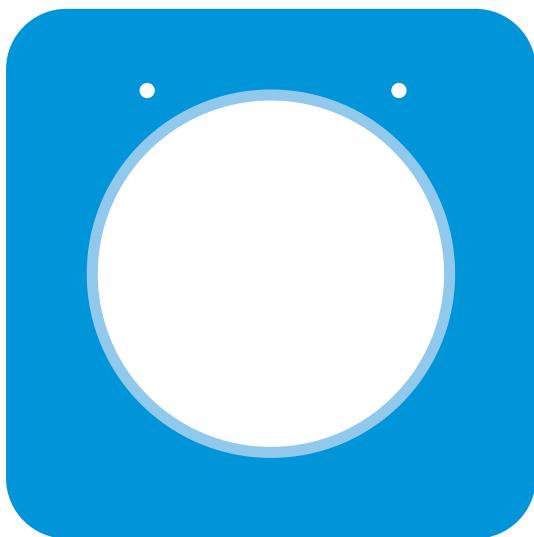


しょうがっこう  
**小学校プログラミング**  
**シンプルセット**



そう ごう  
**総合**

1. プログラミングってなんだろう？
2. 明かりをつけよう
3. イルミネーションライトをつくろう
4. 信号機をつくろう

ねん  
年

くみ  
組

ばん  
番

なまえ  
名前

## もくじ 目次

# 1. プログラムってなんだろう?

そういいじゅぎょうじかん  
想定授業時間 1コマ(45分)

## 学習内容

- 身近なものの中でコンピューターが使われているものを探す。
- 身近な機械にどんなプログラムが組み込まれているか考える。
- プログラム・プログラミングとは何かを理解する。

## 達成目標

## 活動内容

### 導入

活動(かつどう)	活動内容(かつどうないよう)	ページ
生活の中にあるコンピューター	● コンピューターで動く身のまわりのものを挙げ、どれほど多くのものがコンピューターで動いているのかに気付かせる。	P.6

### 基本課題

活動(かつどう)	活動内容(かつどうないよう)	ページ
コンピューターの動き	● 導入で上がった事例から一つを例にとって、それがどのような動きをしているのか具体例を挙げる。	P.7～8

### まとめ

活動(かつどう)	活動内容(かつどうないよう)	ページ
プログラム・プログラミングとは	● どのような仕組みでコンピューターが動いているのか理解する。 ● プログラム・プログラミングとは何か理解する。	P.9

## 2. 明かりをつけよう

想定授業時間 1コマ(135分)

この章ではプログラミングの基礎となる3つの要素である順次処理、繰り返し、条件分岐を学ぶことができます。

### 学習内容

- LEDの点灯、消灯 (順序処理)・LEDの点滅 (繰り返し)・ボタン制御 (条件分岐)。

### 達成目標

- プログラムの3つの基本要素 (順次処理、繰り返し、条件分岐) を理解する。
- 意図したプログラムになるよう試行錯誤することで、プログラミング的思考を育む。

### 活動内容

## 導入

活動(かつどう)	活動内容(かつどうないよう)	ページ
LEDを光らせてみよう	<ul style="list-style-type: none"> <li>● ブロックをクリックして LED の点灯、消灯の制御ができることを確認する。</li> <li>● LED の色を自由に変える。</li> </ul>	P.10 ～17

## 基本課題

活動(かつどう)	活動内容(かつどうないよう)	ページ
LEDを点滅させよう	<ul style="list-style-type: none"> <li>● LED を 1 秒間点灯させるプログラムを作成し、順次処理について説明する。</li> </ul>	P.18 ～24
LEDをくり返し点滅させよう	<ul style="list-style-type: none"> <li>● 上記のプログラムをもとに、LEDを点滅させるプログラムを作成するにはどうすればよいかグループで話し合いながらプログラミングに取り組ませる。</li> <li>● よくある間違い例を挙げて解説しながら繰り返し処理について説明する。</li> </ul>	

## 基本課題②

活動(かつどう)	活動内容(かつどうないよう)	ページ
ボタンを押してLEDを光らせよう	<ul style="list-style-type: none"> <li>● ボタンのON/OFFによってLEDを制御するプログラムを作成する。</li> <li>● フローチャートを用いて条件分岐について説明する。</li> <li>● よくある間違い例を挙げて解説する。</li> </ul>	P.25 ～28

## まとめ

活動(かつどう)	活動内容(かつどうないよう)	ページ
ボタンを使って明るさを調整しよう	<ul style="list-style-type: none"> <li>● 変数について学ぶ。</li> <li>● ボタンを押した回数を変数を使って数える仕組みを理解する。</li> <li>● ボタンを押すと明るさが変化するプログラムを作成する。</li> </ul>	P.29 ～33

## もくじ 目次

### 3. イルミネーションライトをつくろう

想定授業時間 2コマ(90分)

この章ではLEDの色、秒数、順番を自由に変えることで2章で学んだLEDの制御の復習・定着度確認をすることができます。

#### 学習内容

- LEDの光らせ方を自由に考え、考えた通りにプログラミングする。

#### 達成目標

- 意図したプログラムになるよう試行錯誤することで、プログラミング的思考を育む。
- グループで協力して一つの作品を生み出す。

#### 活動内容

### 基本課題

活動(かつどう)	活動内容(かつどうないよう)	ページ
いろ 色や点灯時間を変え てみよう	<ul style="list-style-type: none"><li>● 1台のアーテックリンクスを使って、どのように LED を点灯させたいのか計画を立てる。</li><li>● 立てた計画通りに点灯させるプログラムを作成する。</li></ul>	P.34 ～36

### まとめ

活動(かつどう)	活動内容(かつどうないよう)	ページ
たくさんのメインユニットを光らせよう	<ul style="list-style-type: none"><li>● 複数台のアーテックリンクスを使いグループで協力して 10 秒間のイルミネーションライトのプログラムを作成する。</li></ul>	P.37 ～39

## もくじ 目次

# 4. 信号機をつくろう

想定授業時間 2コマ(90分)

この章では、信号機の動きをプログラミングすることで、信号機の仕組みを学ぶことができます。

### 学習内容

- 自動車用信号機、歩行者用信号機、押しボタン式信号機のプログラムを作成して、信号機の仕組みを学ぶ。

### 達成目標

- 意図したプログラムになるよう試行錯誤することで、プログラミング的思考を育む
- 信号機のような身近な機械がプログラムによって制御されていることをプログラミングを通して理解する。

### 活動内容

## 導入

活動(かつどう)	活動内容(かつどうないよう)	ページ
信号機の動きを整理しよう	●自動車用、歩行者用信号機の光る順番を整理する。	P.40 ～41

## 基本課題

活動(かつどう)	活動内容(かつどうないよう)	ページ
信号機をつくろう	●自動車用と歩行者用信号機の違いを理解しながら、プログラムを作成する。 ●よくある間違い例を挙げて解説する。	P.42 ～44

## まとめ

活動(かつどう)	活動内容(かつどうないよう)	ページ
押しボタン式信号機をつくろう	●ボタンを押すことで信号が青に変わる押しボタン式信号機のプログラムを作成する。 ●双方向通信の機能を用いて、自動車用信号機と歩行者用信号機が連動する仕組みを再現する。	P.45 ～50

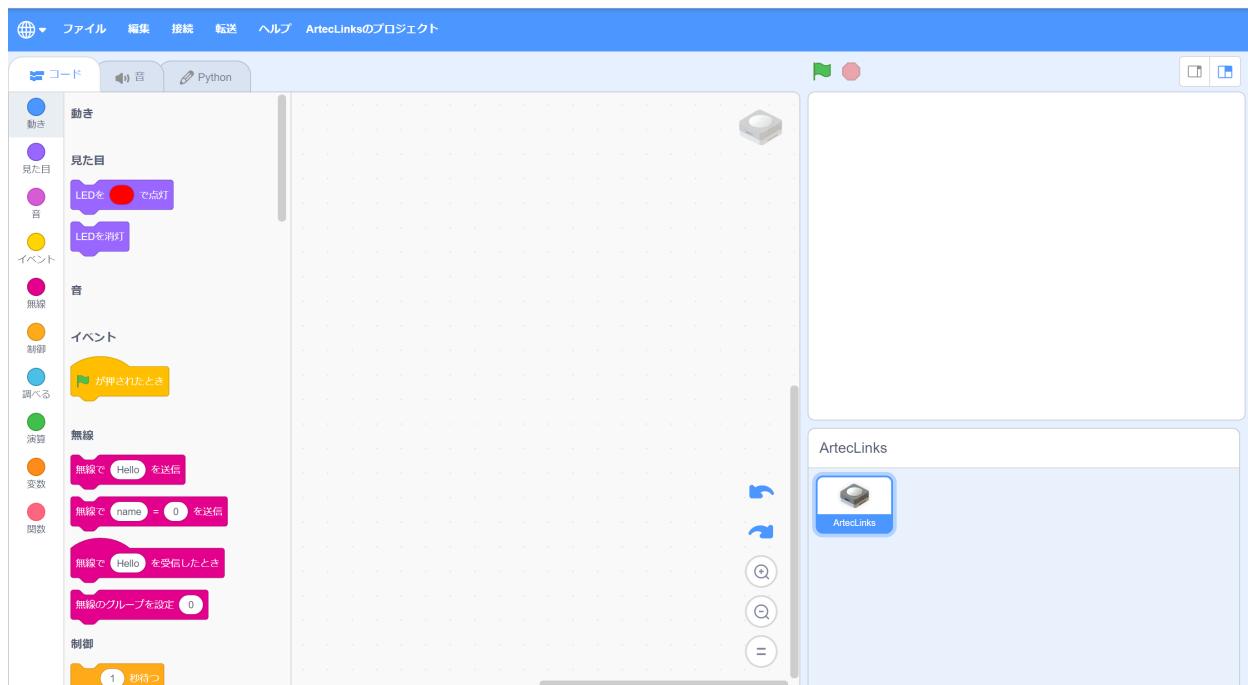
## ソフトウェアについて

このテキストでは、専用のソフトウェアを使用します。下記サイトにアクセスして、インストール版のソフトウェアをダウンロードする、または、オンライン版のWEBアプリを開いて使用してください。

<https://www.artec-kk.co.jp/arteclinks/software/>



下記の画面になればソフトウェアの起動は完了です。



# 1. プログラムってなんだろう?

## 生活の中にあるコンピューター

身の回りにはコンピューターで動いているものがたくさんあります。

パソコンやタブレットだけでなく、駅の改札や洗濯機など「わたしたちの生活を便利にするもの」のなかにはコンピューターが使われています。

### 考えてみよう

下の2つの絵の中からコンピューターが使われていると思うものを選んで書いてみましょう。

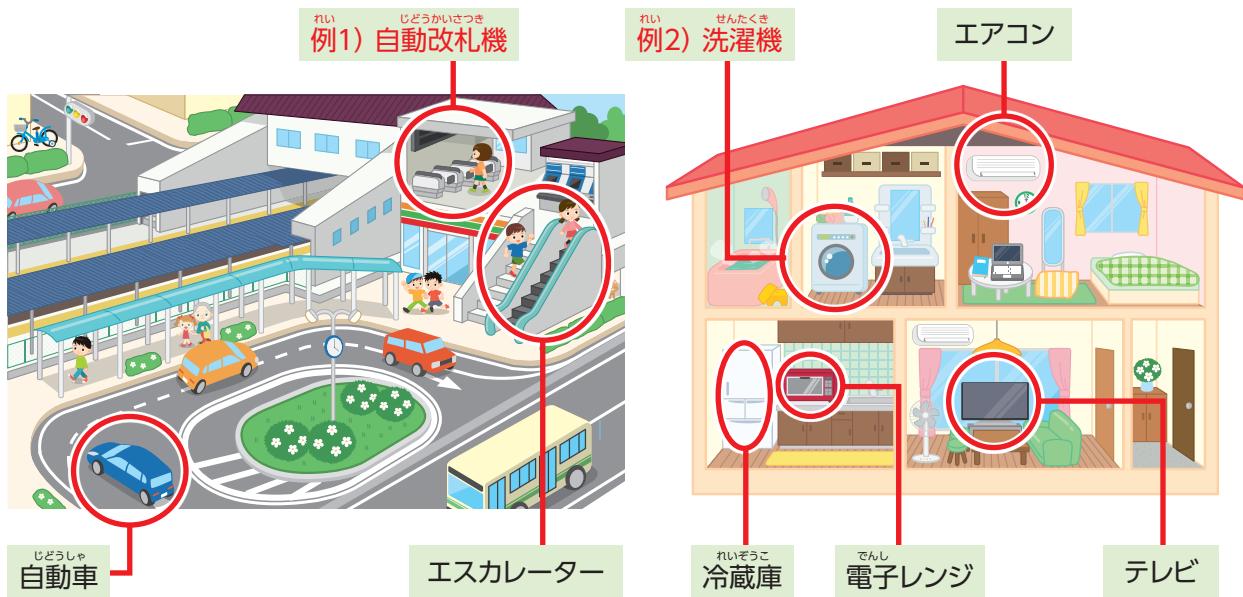
#### 例1) 自動改札機



#### 例2) 洗濯機



# コンピューターの動き



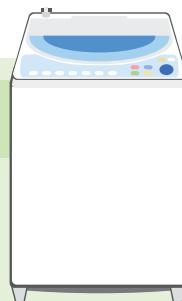
じつ  
なか  
はい  
実はすべてのものの中にコンピューターが入っています。

## 考えてみよう

うご  
なか  
ひと  
えら  
うご  
かんが  
コンピューターで動いているものの中から一つを選んで、どんな動きをするものなのかを考えてみましょう。

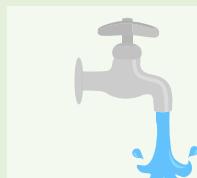
### もののなまえ 例) 洗濯機

せんたくもの せんざい い あと お  
洗濯物と洗剤を入れた後、スタートボタンが押されると…



せんたくもの りょう おう  
洗濯物の量に応じて

みず  
水をそそぐ



あら  
洗う



あら  
すすぎ洗い



みず  
だっさい  
脱水



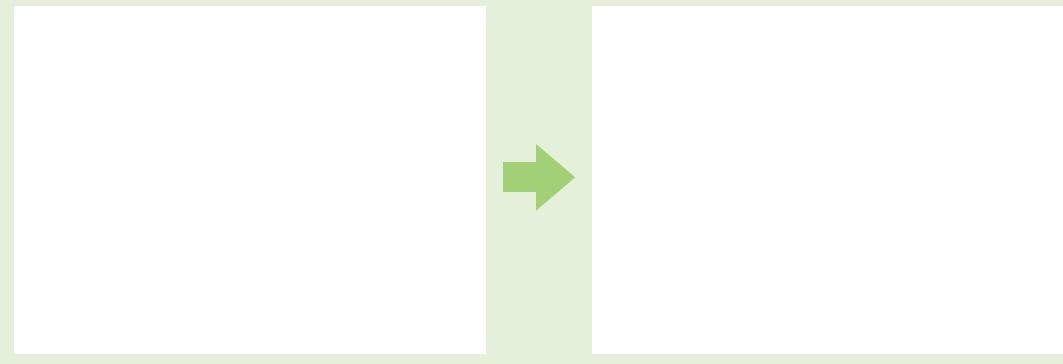
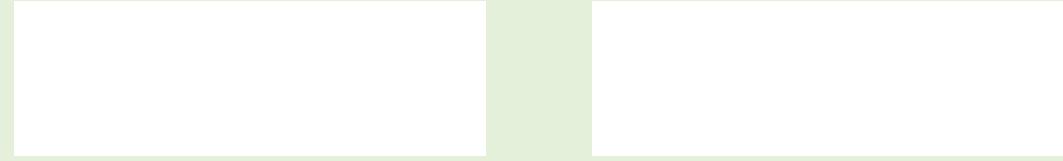
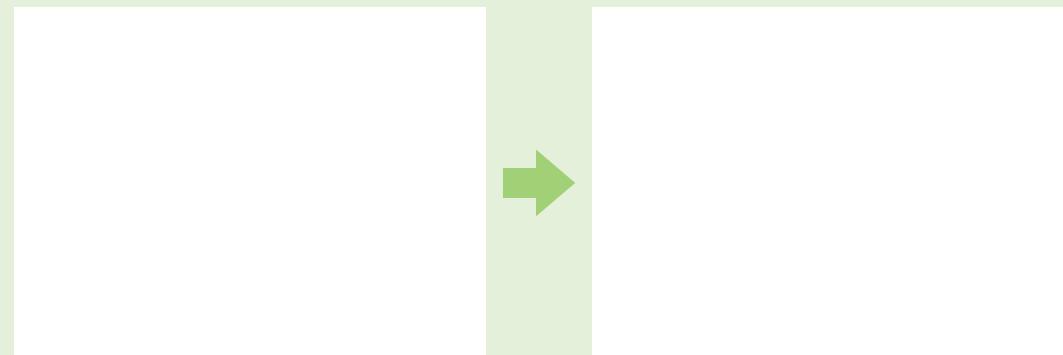
せんたくかんりょう おと し  
洗濯完了を音でお知らせ



もののかたえ



うごきのじゅんばん

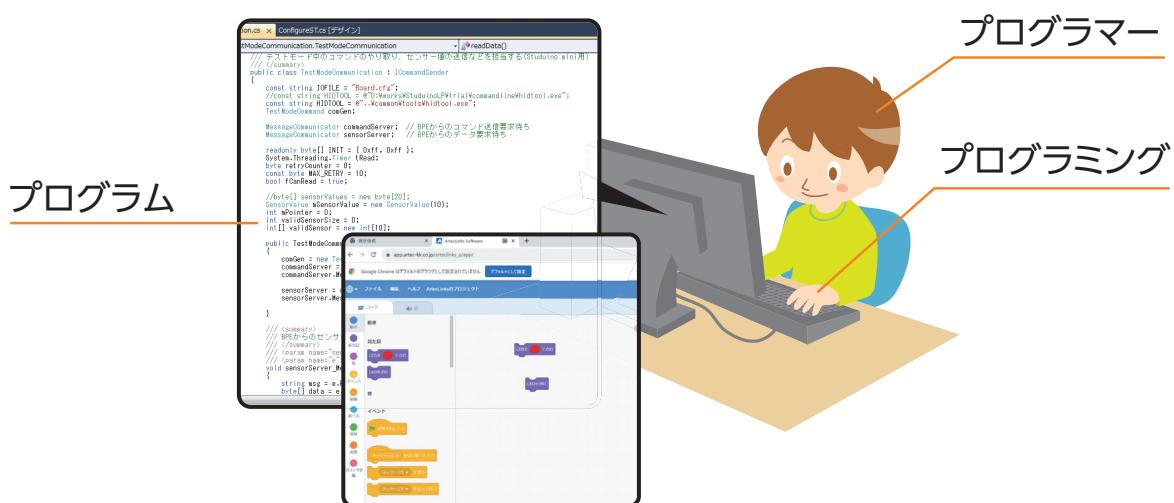


# プログラム・プログラミングとは

前のページで考えた「どんなとき、どんなうごきをするか」はあらかじめ人間がコンピューターに「お願い」をして、覚えさせる必要があります。



プログラム ..... コンピューターに覚えさせる「お願い」  
プログラミング ..... お願いをつくること  
プログラマー ..... お願いをつくる人



## 2. 明かりをつけよう

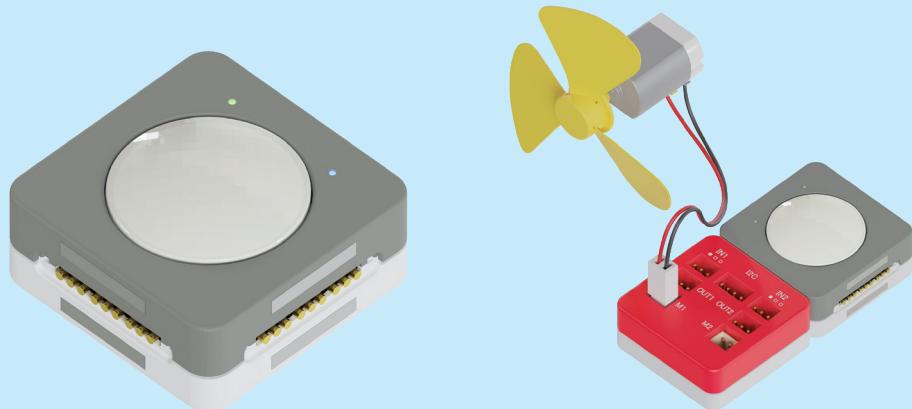
アーテックリンクス のLEDをプログラミングで動かしてみましょう。LEDを自動で点灯・消灯させたり、自由に色を変えたりしてみましょう。



# アーテックリンクスについて

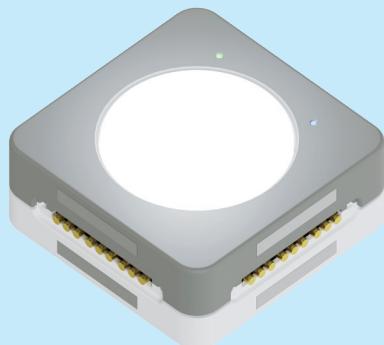
## メインユニットってなに？

プログラムで動く、小さなコンピューターです。メインユニットを使うと、さまざまなセンサー やメインユニットに搭載されたLED、モーターと組み合わせて機械やロボットなどをつくる ことができます。また、メインユニットの白い丸の部分はLEDになっており、押しボタンとし ても使えるようになっています。



## LED ってなに？

豆電球と同じように電気を流すと光る電気部品です。発光ダイオードともいいます。LEDには さまざまな種類があり、赤・緑・青・白などいろいろな色の光を出すことができます。  
たとえば、信号機や自動車のヘッドライト、液晶テレビのバックライトなどにはたくさん LED が使われています。



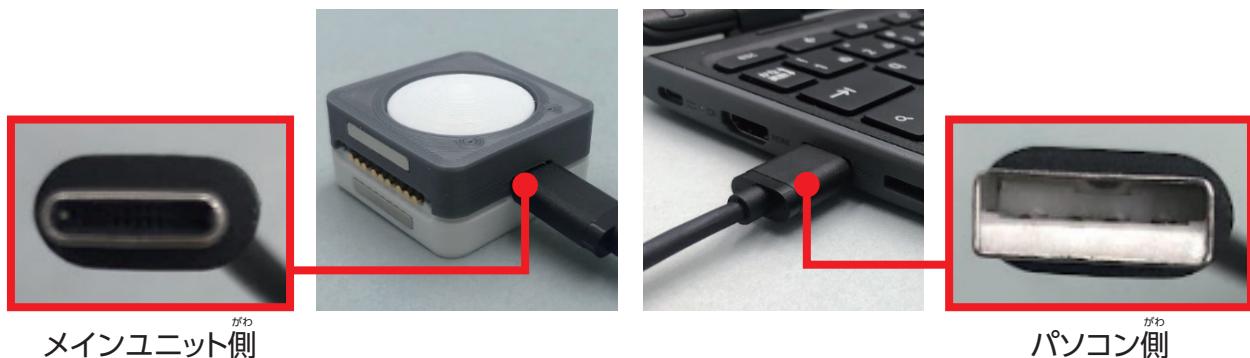
# LEDを光らせてみよう

アーテックリンクスとパソコンを接続してLEDを光らせてみましょう。



## 接続方法

- ① ソフトウェアを起動させます。
- ② メインユニットとパソコンをUSBケーブルで接続します。



メインユニット側

パソコン側

- ③ 画面左上のメニューから「接続」をクリックします。



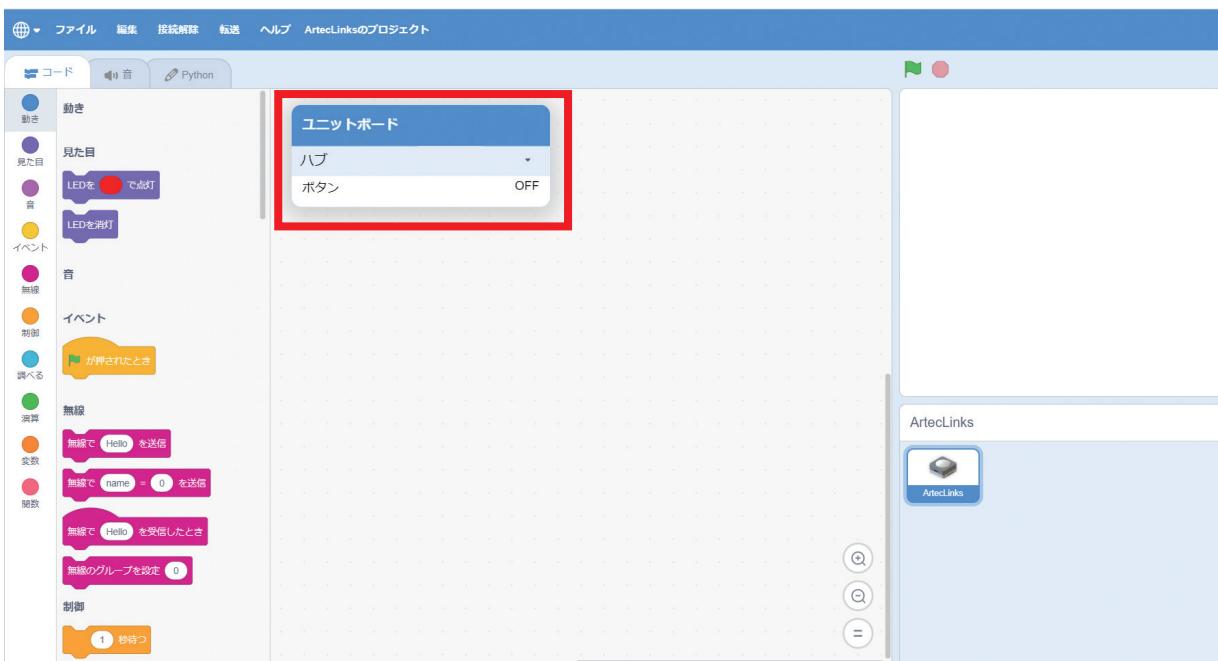
④「USB JTAG…」を選び、「接続」をクリックします。  
画面上に「ユニットボード」が表示されたら接続完了です。

app.artec-kk.co.jp ガシリアルポートへの接続を要求しています

USB JTAG/serial debug unit (COM4)

接続

キャンセル



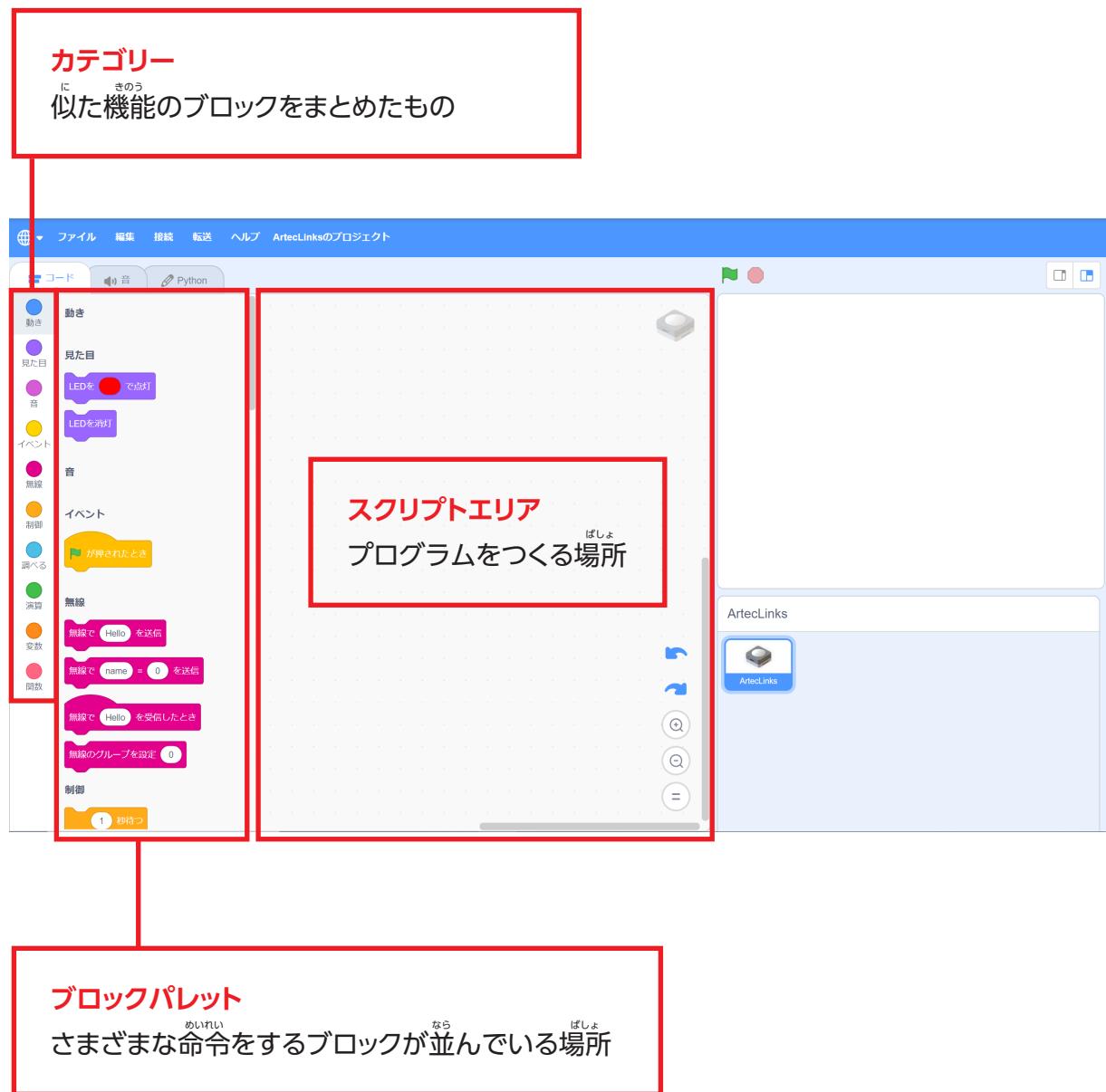
接続に失敗する場合は、USBケーブルを抜きさししてもう一度②から実行してみましょう。

教員の方へ

※このページ以降は、基本的には「ユニットボード」が表示されていることを前提に解説されております。

# 画面の見方

ソフトウェアの画面はいくつかのエリアに分かれています。



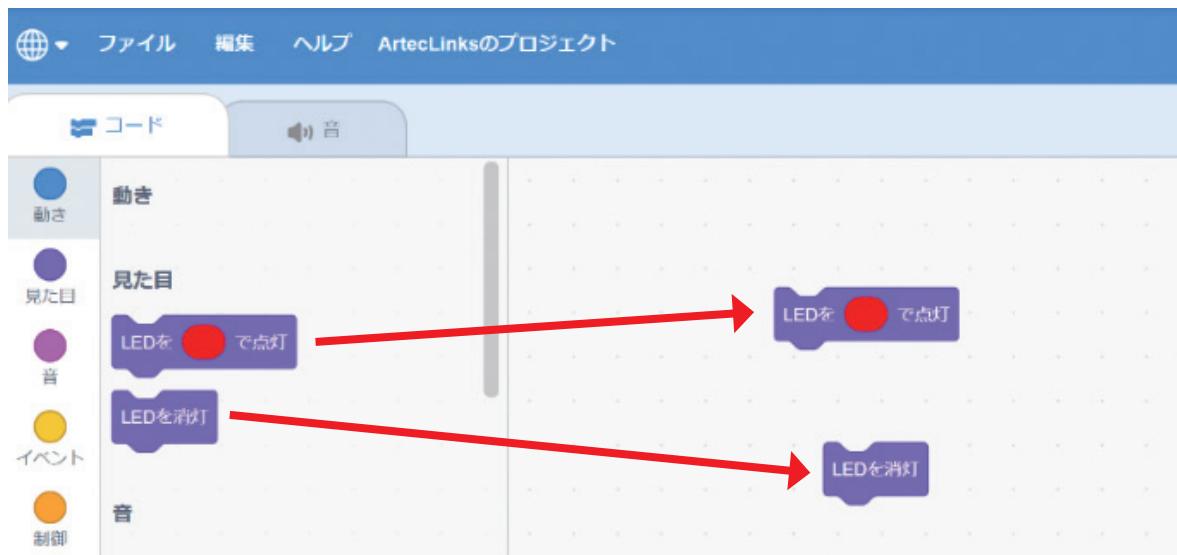
がめん

みかた

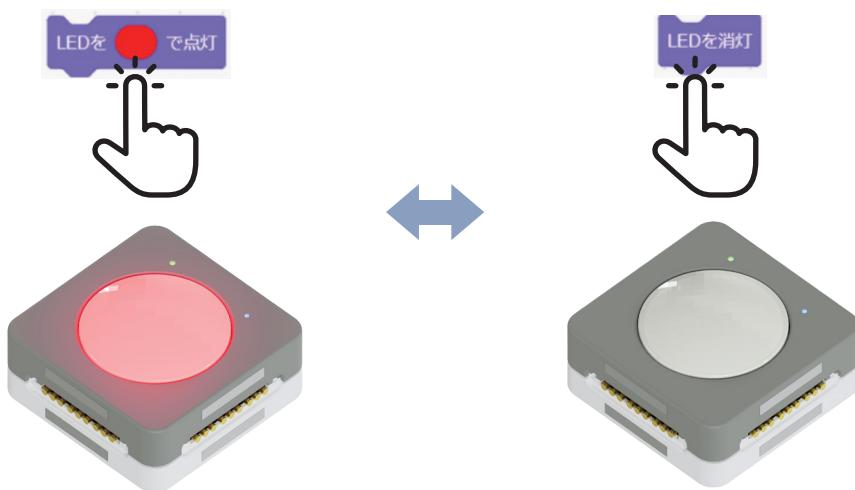
# 点灯・消灯させるLEDのプログラムをつくろう

## かんが 考えてみよう

- ①  カテゴリーから  と  をドラッグ&ドロップでスクリプトエリアに並べましょう。



- ②   をクリックするとアーテックリンクスのLEDが点灯・消灯することを確認しましょう。



## 教員の方へ

※アーテックリンクスとパソコンなどの制御端末が正しく接続されていないとプログラムは実行されません。ユニットボードが表示されているか確認してください。

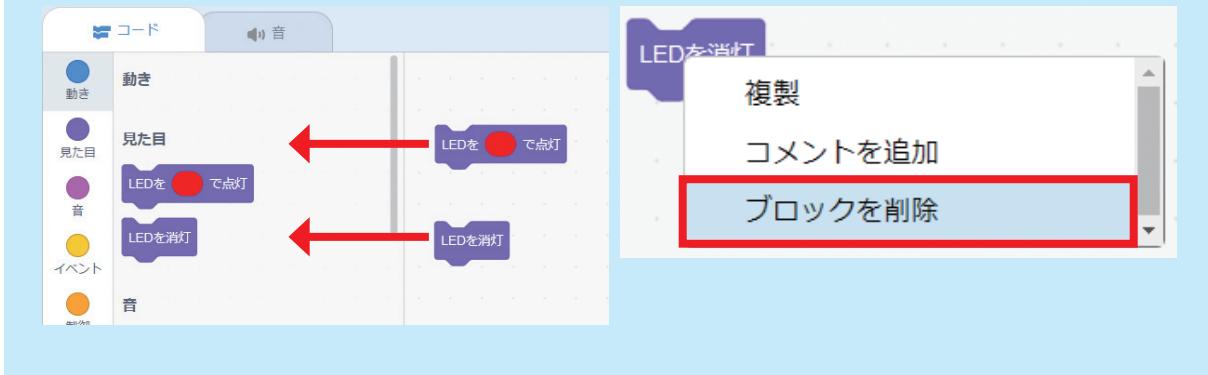
## ユニットボード

メインユニット	▼
ボタン	OFF

## ブロックを消す方法

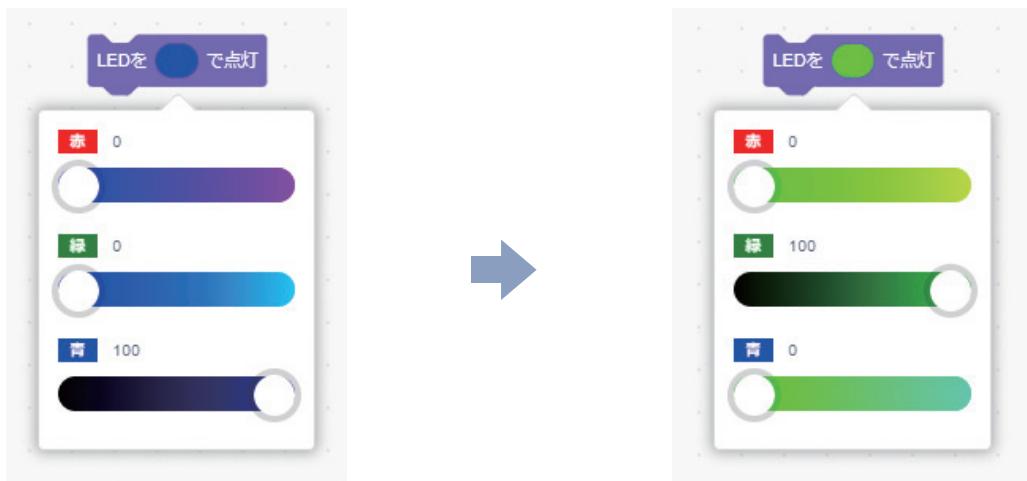
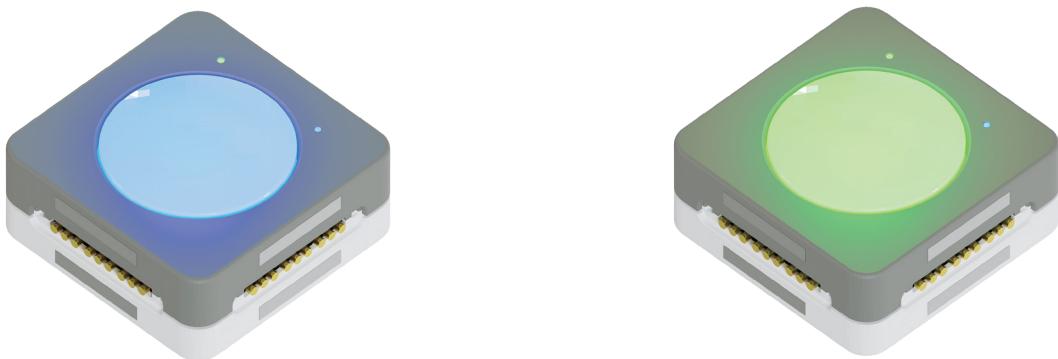
スクリプトエリアに置いたブロックは、ブロックパレットにドラッグ&ドロップすると消すことができます。

ブロックの上で右クリックをして、表示された画面から「ブロックを削除」をクリックして消すこともできます。



③ をクリックしましょう。

○を動かすとLEDの色が変わることを確認しましょう。



## 発展 LEDの色を数値で指定する

数値を使って細かく色の指定をすることもできます。

- ① 画面左上の「編集」をクリックします。
- ② 「ブロックの表示・非表示」をクリックします。
- ③ 「LED RGB数値指定」をクリックするとブロックが追加されます。



追加されるブロック



・光の強さを0~100で指定することができます。



# LEDを点滅させよう

## やってみよう！

LEDを1秒間点灯させた後、消灯させるプログラムをつくりましょう。

### プログラムの動き



### 使うブロック



イベント



見た目



制御



### プログラム例



## かくにんしよう

が押されたとき かステージエリアにある  
をクリックするとプログラムが実行されます。  
LEDが1秒間光ることを確認しましょう。



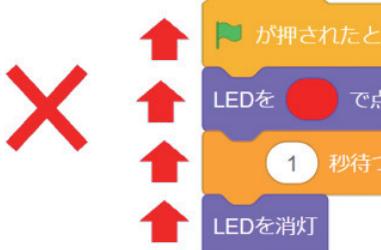
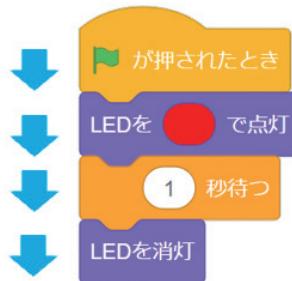
### ブロックの見た目

プログラム実行中の場合、ブロックが黄色枠で囲まれます。



### 順次処理

プログラムは必ず上から順番に実行されます。



上から順番にプログラムが実行されることを順次処理と言います。

## アレンジしよう

1 秒待つ の値は自由に変えることができます。  
値を変えて、LEDが光る時間を自由に変えてみましょう。

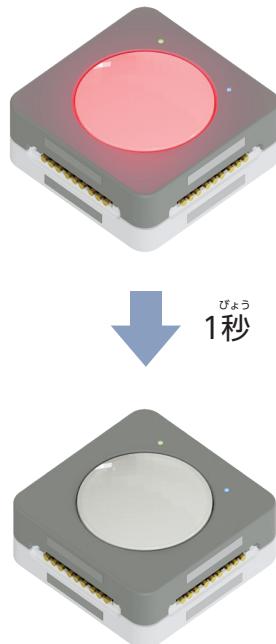
クリックして値を入力



# LEDをくり返し点滅させよう

# かんが 考 えて み よ う

LEDをくり返し点滅させるにはどうしたらよいか考えてみましょう。



かんが  
考えたアイデアを下の欄に書いてみましょう。

## プログラムのアイデア

教員の方へ

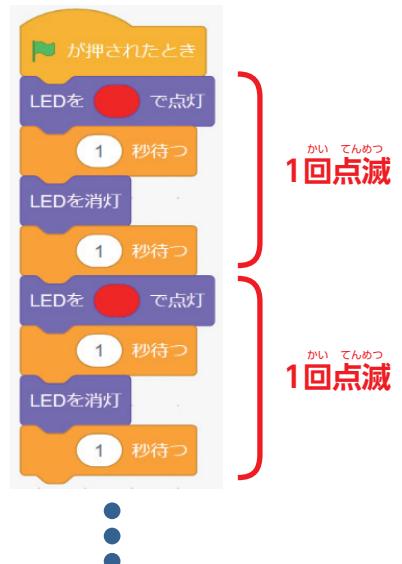
※自由にプログラムを試しながらグループで考えさせましょう。

\*18ページでつくったプログラムを何度も実行すればくり返し点滅させられることに気づかせましょう。

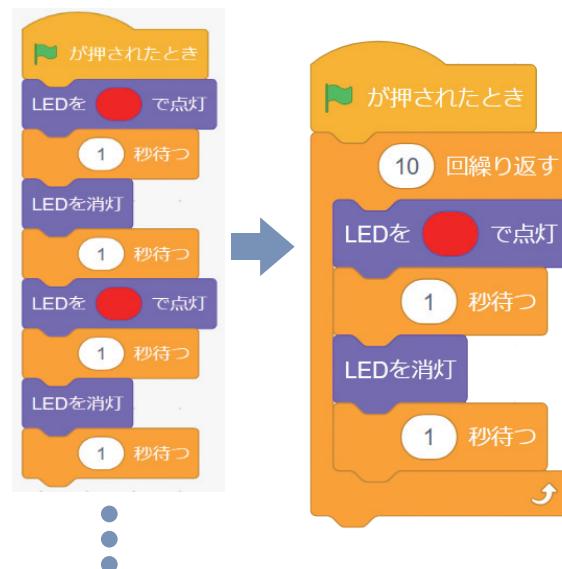
ブロックを繰り返したい回数つなげれば、くり返し点滅させることができます。

しかし、たくさんつなげると、プログラムが長くなつて見づらくなつてしまします。

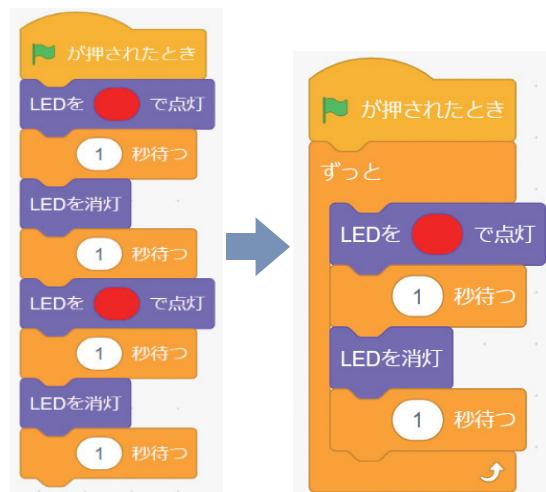
制御 の  
みじか  
しのプログラムを短くできます。



### 例) 10回くり返し点滅させる プログラム



### 例) ずっとくり返し点滅させる プログラム



の数値を変えることで、くり返しの回数を変えることができます。



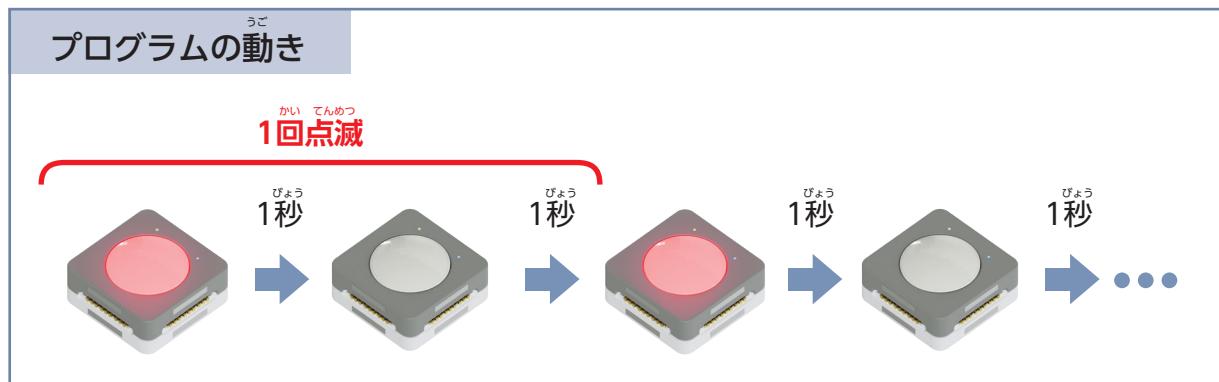
のなか  
の中を  
10回くり返す



のなか  
の中を  
5回くり返す

## やってみよう！

LEDを4回点滅させるプログラムをつくってみましょう。



つか  
使うブロック

制御

1秒待つ

10回繰り返す

LEDを ( ) で点灯

LEDを消灯

が押されたとき

が押されたとき

かくにん

れい  
プログラム例

## かくにんしよう

かくにん  
じっこう  
をクリックするとプログラムが実行されます。

かい てんめつ  
LEDが4回点滅することを確認しましょう。

かえ  
くり返し

なんど おな てじゅん かえ とくい  
プログラムは何度も同じ手順をくり返すことが得意です。

みぎ みがん かい かい  
右のプログラムは **□** 部分を10回くり返すプログラムです。

かえ  
くり返しのプログラムをつくることができます。

なんど おな じっこう かえ い  
何度も同じプログラムが実行されることを**くり返し**と言います。



## ポイント

くり返しのプログラムを作る場合、図のような間違いをする場合があります。



よくある間違い



正しいプログラム



なぜ の後ろに がないプログラムではうまく点滅しないのでしょうか？



部分が高速で実行されてしまい、消灯した後すぐに点灯してしまうからです。



を使わずに  
プログラムをつくると…



消灯した後にすぐ  
点灯してしまう

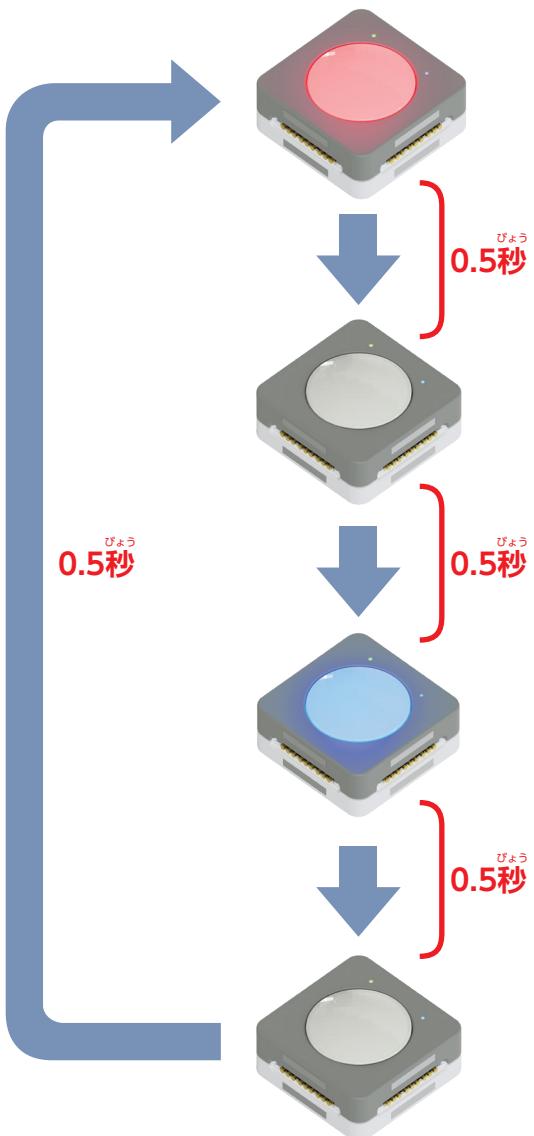
## アレンジしよう

点滅のプログラムができたら、点滅の秒数を変えたり点滅する色を増やしたりしてアレンジしてみましょう。

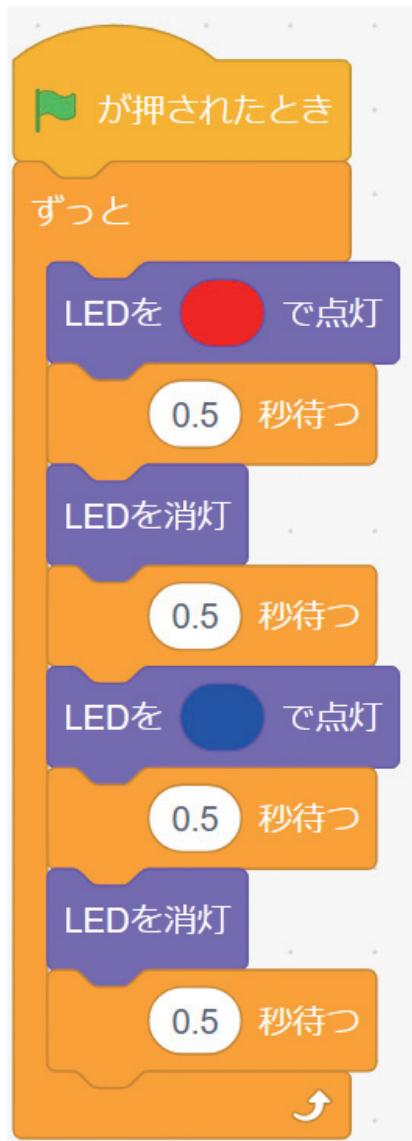
アレンジ  
の例

赤と青のLEDライトが短い間隔で点滅するプログラム

### プログラムの動き



### プログラム例

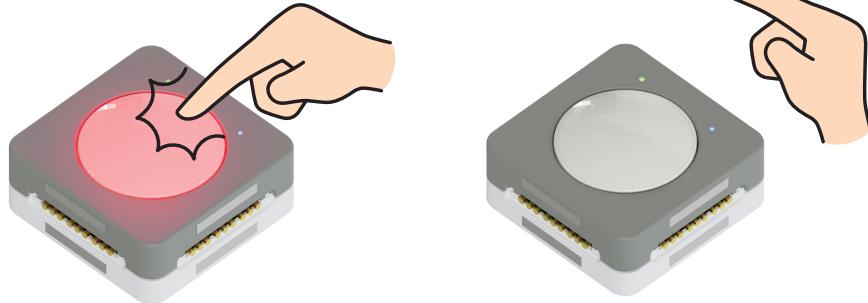


# かえ てんめつ LEDをくり返し点滅させよう

## やってみよう！

お あいだ てんとう  
ボタンを押している間だけLEDが点灯するプログラムをつくってみましょう。

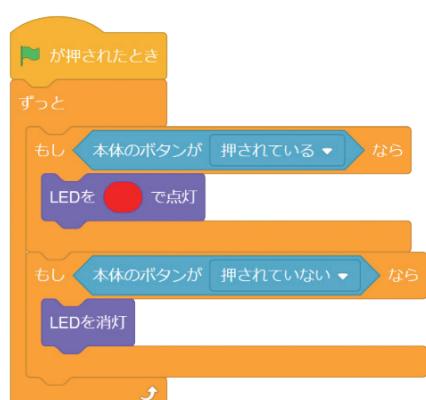
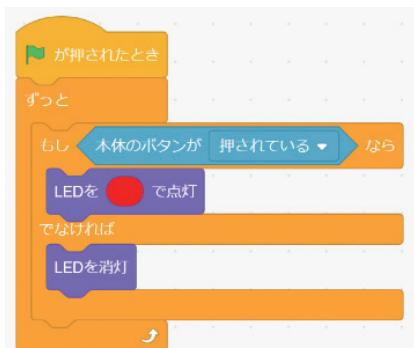
### プログラムの動き



### 使うブロック

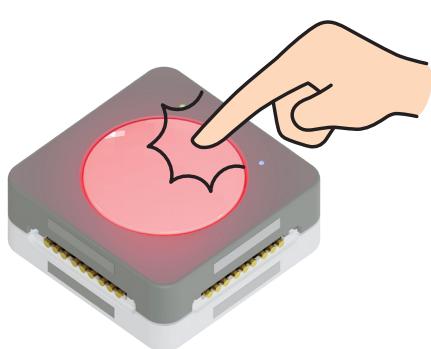


### プログラム例



## かくにんしよう

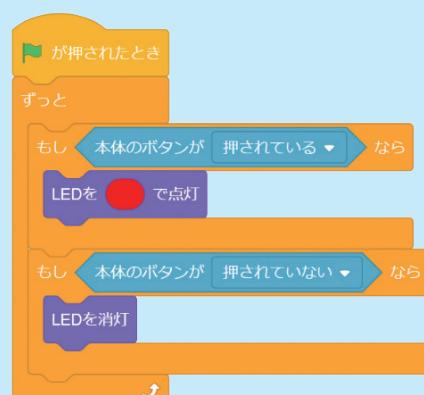
ユニットボードでボタンの値を確認してみましょう。



ボタンを押すとLEDがONになります。指を離すとLEDがOFFになります。

### 条件分岐

「もし～だったら」と条件を設定して、その条件に当てはまる場合と当てはまらない場合で実行するプログラムを変えることを**条件分岐**と言います。



ボタンが押されている

本体のボタンが 押されている ▶

Yes

No

点灯

もし 本体のボタンが 押されている ▶ なら  
LEDを [red v] で点灯

消灯

でなければ  
LEDを消灯



## ポイント

くり返しのプログラムをつくる場合、図のような間違いをする場合があります。



よくある間違い



正しいプログラム

```

when green flag clicked
repeat until [button A pressed? v]
    if [button B pressed? v] then
        led on
    else
        led off
end
end

```

```

when green flag clicked
repeat (5)
    if [button B pressed? v] then
        led on
    else
        led off
end
end

```



が抜けているプログラムではなぜうまく動作しないのでしょうか？



A コンピューターが高速でプログラムを実行することによって、ボタンが押された瞬間に本体のボタンが押されているかどうかを確認するプログラムが終了してしまうからです。

スタート



すぐに  
終了

```

when green flag clicked
repeat (5)
    if [button B pressed? v] then
        led on
    else
        led off
end
end

```

ボタンが押された瞬間には本体のボタンが押されていないので、LEDを消灯が実行されてプログラムが終了してしまう。

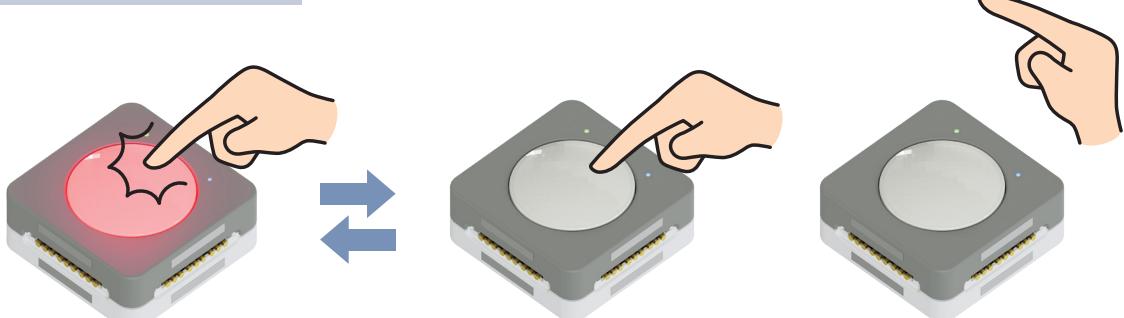


でプログラムをくり返し実行することによって、ボタンが押されているかどうかを何回も確認し続けることができます。

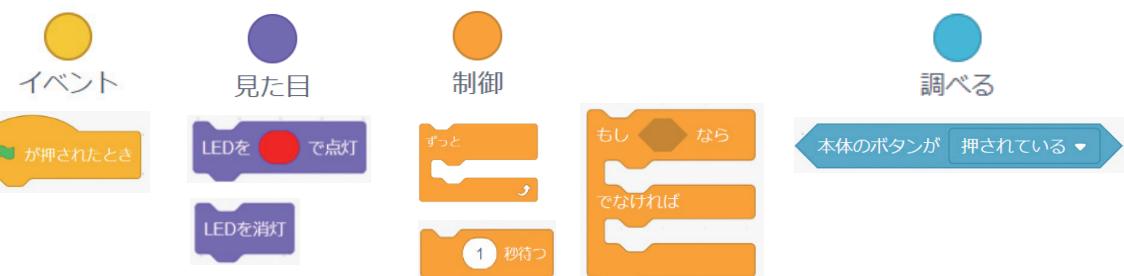
## アレンジしよう

お あいだ てんめつ  
ボタンを押している間、LEDが点滅するプログラムをつくってみましょう。

### うご プログラムの動き



### つか 使うブロック



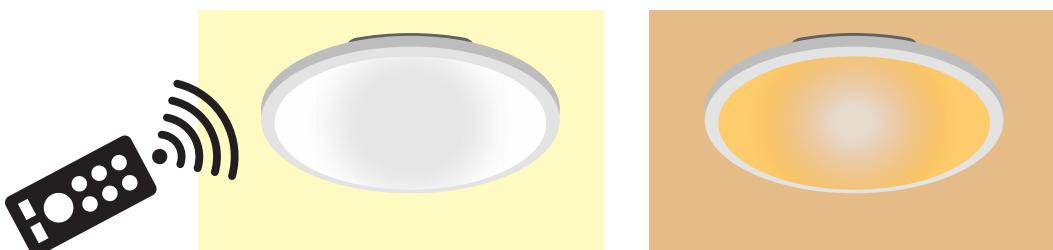
### れい プログラム例



# 生活の中にあるコンピューター

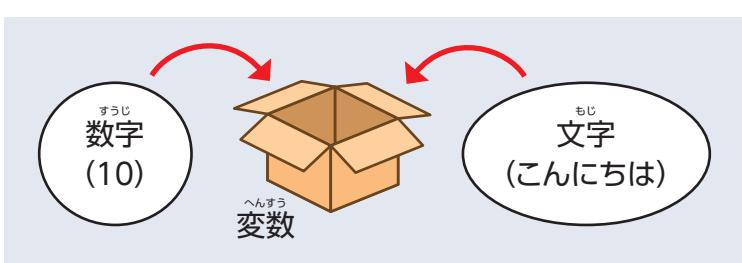
照明にはリモコンを使って光の明るさを調整できるものがあります。

ボタンを押す回数によってLEDの明るさを変えるプログラムをつくってみましょう。

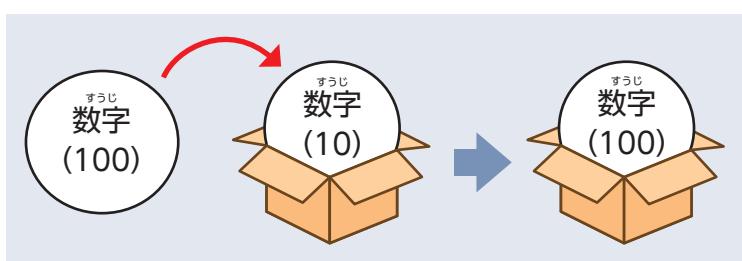


ボタンを押した回数を調べるために変数が必要です。

## 画面の見方



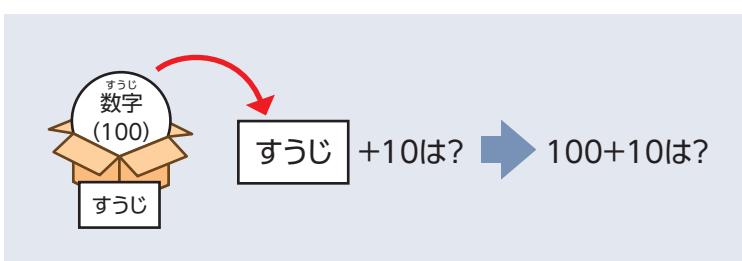
変数は数字や文字を「1つだけ」入れ  
ることができる箱です。



箱の中に既に数字や文字が入って  
いるとき、新しい数字や文字を入れ  
ると上書きされます。



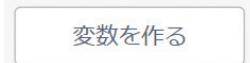
何が入っているか分かるように、箱  
には名前を付けることができます。

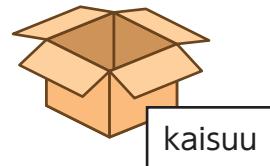


付けた名前で呼ぶことで、中身の数  
字や文字を使うことができます。

変数のしくみを使って、ボタンを押す回数によって明るさが変わるライトをつくりましょう。

## かくにんしよう

- ①  の  **変数を作る** を押してkaisuu(回数)という変数をつくりましょう。



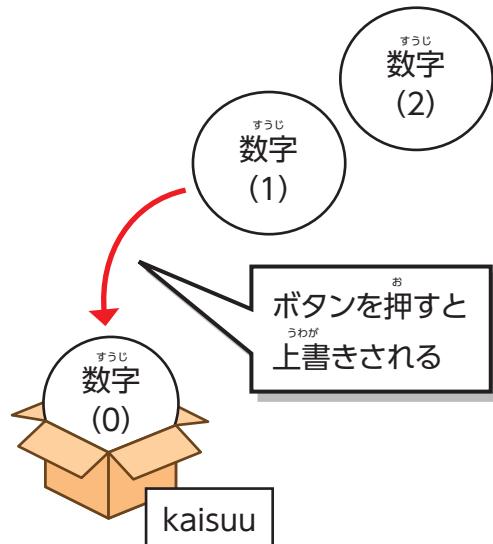
- ②  が押されたとき、「kaisuu」を0にしましょう。



- ③ ボタンを押すたびに「kaisuu」が1ずつ大きくなるようにしましょう。



- ④  の下に表示された「kaisuu」の数字がボタンを押すたびに増えしていくことを確認しましょう。

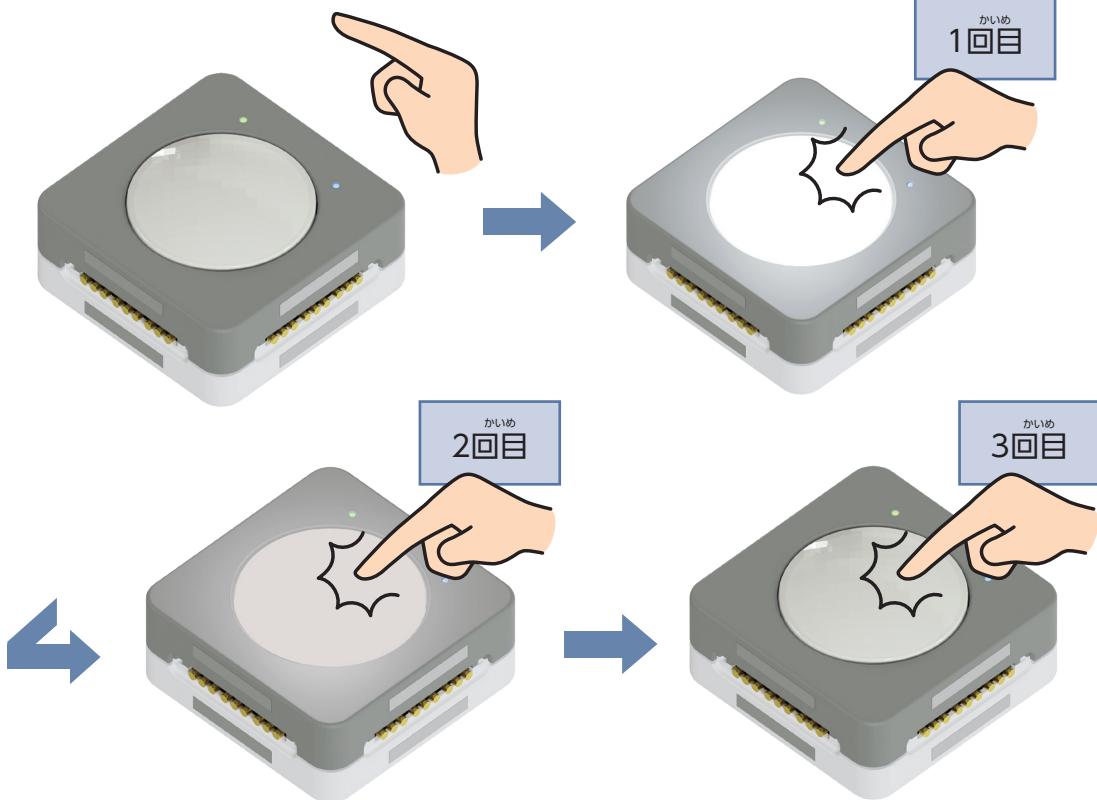


このように変数のしくみを使ってボタンを押した回数を数えることができます。

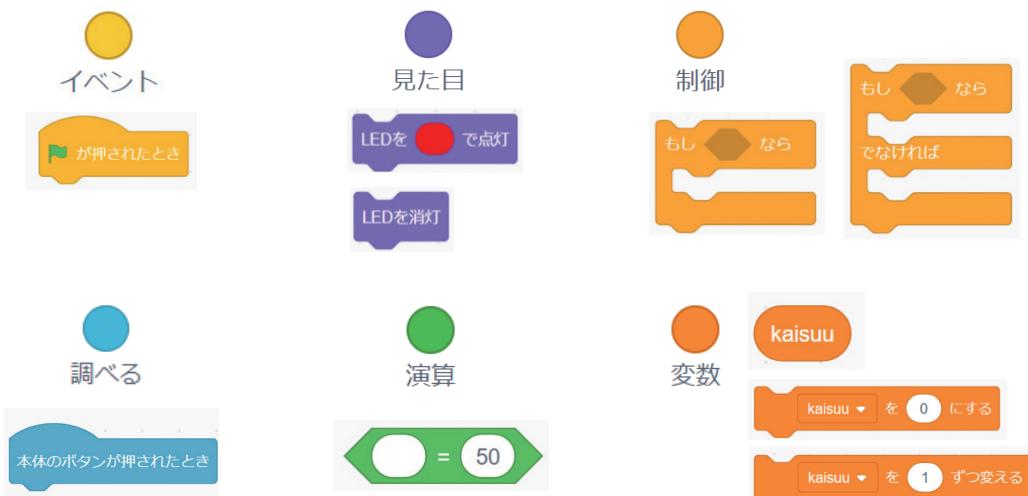
## かくにんしよう

かい お しろいろ てんとう かい お あか よわ かい  
1回ボタンを押すとLEDが白色で点灯、2回ボタンを押すとLEDの明るさが弱まり、3回ボタンを  
お お しおとう お お お お お お  
押すと消灯するというしくみのライトをつくってみましょう。

### プログラムの動き



### 使うブロック

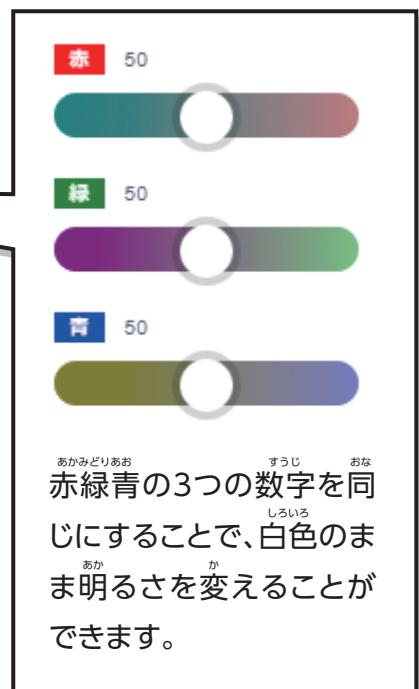
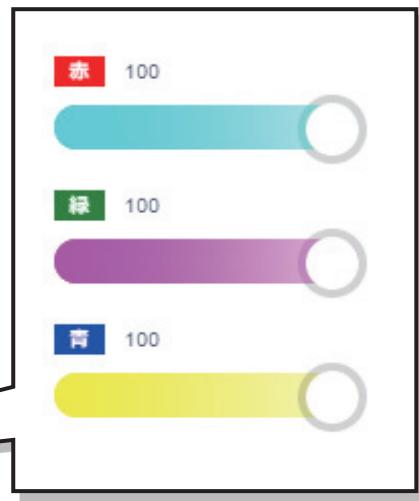


## プログラム例

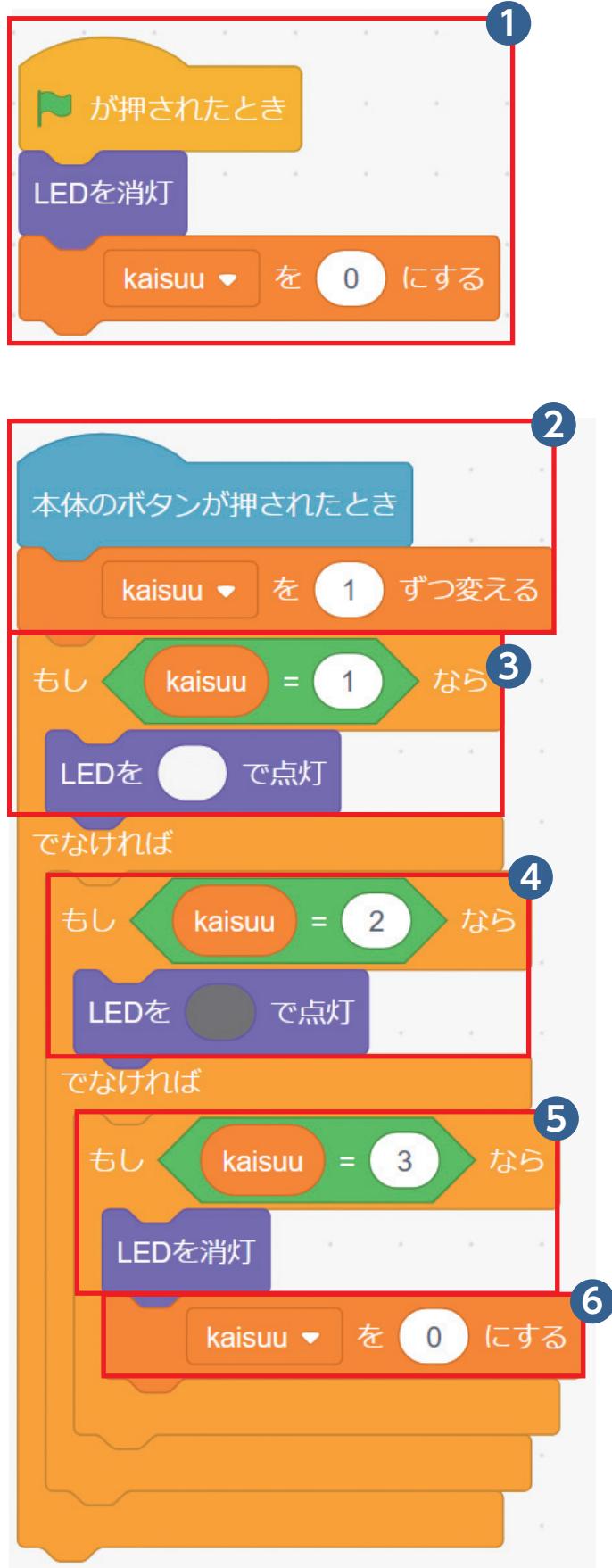
```

[Flag pressed] when [LED turn off]
  [kaisuu ▾] set [0] to [kaisuu]
[Body button pressed] when [kaisuu] change by [1]
  if [kaisuu = 1] then
    [LED turn on]
  else if [kaisuu = 2] then
    [LED turn on]
  else if [kaisuu = 3] then
    [LED turn off]
  [kaisuu ▾] set [0] to [kaisuu]

```



# プログラムの説明



- 1 が押されたときLEDを消灯して変数を0にします。  
ボタンを押した回数を0にするリセットボタンの役割をします。
- 2 ボタンを押すと変数が1ずつ大きくなります。
- 3 1回ボタンを押すとLEDが点灯します。
- 4 2回ボタンを押すとLEDの明るさが暗くなります。
- 5 3回ボタンを押すとLEDが消灯します。
- 6 変数を0にすることで③～⑤のプログラムをくり返し実行できるようにしています。

### 3. イルミネーションライトをつくろう

LEDが決まった色と順番で点灯と消灯を繰り返すイルミネーションライトをつくりましょう。



# かくにんしよう

## かくにんしよう

LEDをどのように点灯させたいか考えてみましょう。



LED の色	例) 赤	例) 青	例) 緑	例) 黄
点灯時間	例) 1 秒	例) 2 秒	例) 3 秒	例) 4 秒

考えた色と時間を下の表に書きましょう。

LED の色						
点灯時間	秒	秒	秒	秒	秒	秒

## 補足 プログラムをコピーする方法

複製を使うと同じプログラム(ブロック)を簡単につくることができます。



① **LEDを [red] で点灯**  
を右クリック

② **複製**をクリック

③ **クリックしたブロック**  
につながっているブロック  
すべてが全てコピーされる

## やってみよう！

考えた順番で点灯するプログラムを考えましょう。

プログラム例



LED の色	例) 赤	例) 青	例) 緑	例) 黄
点灯時間	例) 1 秒	例) 2 秒	例) 3 秒	例) 4 秒



教員の方へ

※生徒が考えた順番通りにプログラムが作れているか確認してください。  
早くできた場合は、色や時間を自由に変更させてください。

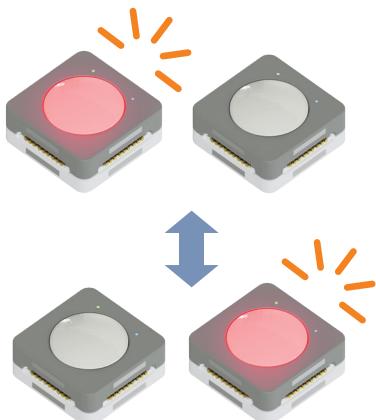
# たくさんの中継ユニットを光らせよう

たくさんの中継ユニットを光らせて、イルミネーションライトをつくりましょう。  
中継ユニットのLEDを10秒間自由に光らせるプログラムをつくりましょう。

## 考えてみよう

LEDをどのように点灯させたいか考えてみましょう。

交互に光らせるには  
どうしたらよいだろう?



左から順番に光らせるには  
どうすればよいだろう?



## 教員の方へ

※クラスの人数、所持台数によって使用台数は調整してください。

## やってみよう！

表を使うと、たくさんのLEDの光らせ方を整理しやすくなります。

グループで話し合いながら、LEDの光らせ方を表にまとめてみましょう。

点灯時間	1台目のLEDの色	2台目のLEDの色	3台目のLEDの色	4台目のLEDの色
0秒 1秒				
2秒				
3秒				
4秒				
5秒				
6秒				
7秒				
8秒				
9秒				
10秒				

### 補足 秒数の数値

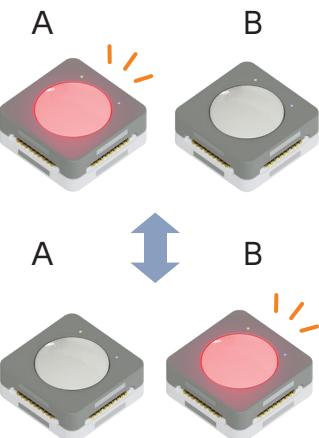
秒数は小数でも入力できます。



かんせい ボタンを一斉に押して光らせてみましょう。

## プログラム例

### ①交互に点滅するプログラム



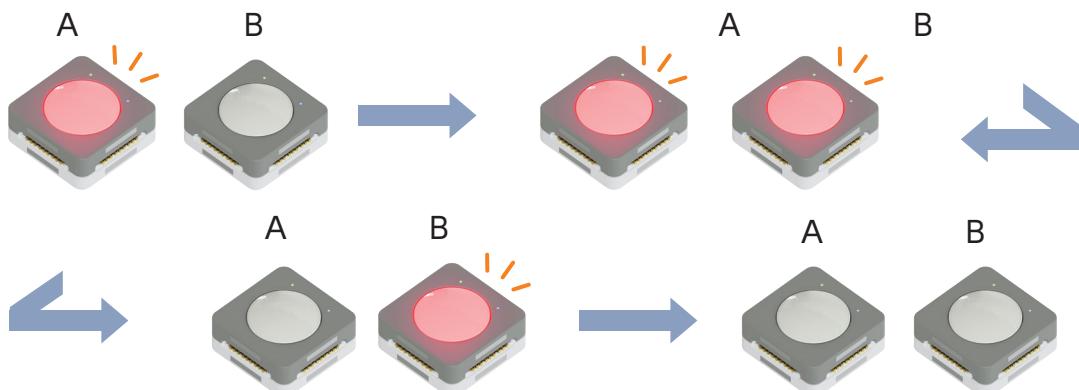
Aのプログラム

```
が押されたとき
ずっと
LEDを [赤] で点灯
[0.5] 秒待つ
LEDを消灯
[0.5] 秒待つ
```

Bのプログラム

```
が押されたとき
ずっと
LEDを消灯
[0.5] 秒待つ
LEDを [赤] で点灯
[0.5] 秒待つ
```

### ②順番に点灯して消灯するプログラム



Aのプログラム

```
が押されたとき
LEDを [赤] で点灯
[1] 秒待つ
LEDを消灯
```

Bのプログラム

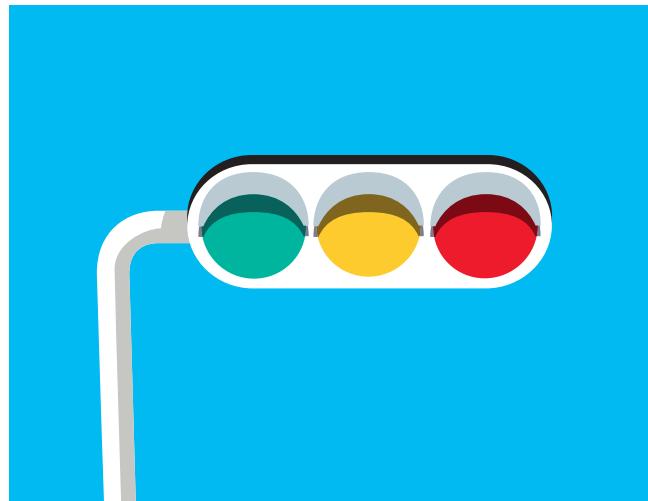
```
が押されたとき
LEDを消灯
[0.5] 秒待つ
LEDを [赤] で点灯
[1] 秒待つ
LEDを消灯
```

## 教員の方へ

※グループで考えたLEDの光らせ方通りに正しくプログラミングが動いていることを確認してください。

## 4. 信号機をつくろう

プログラムでLEDの光らせ方を制御して、信号機をつくりましょう。

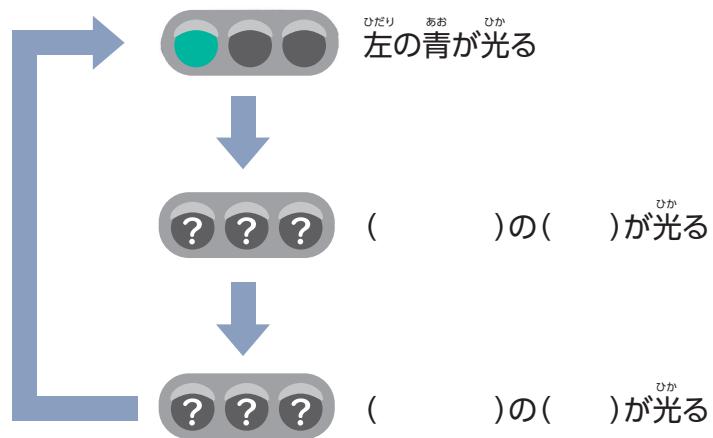


# 信号機の動きを整理しよう

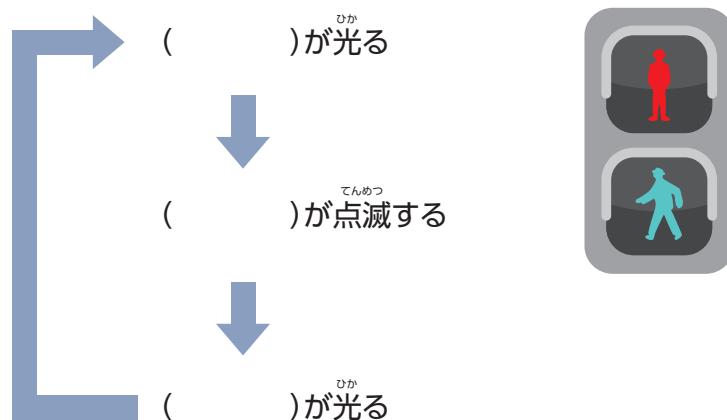
## やってみよう！

信号機の光る順番を整理してみましょう。

### 自動車用信号機



### 歩行者用信号機



# 信号機をつくろう

## やってみよう！

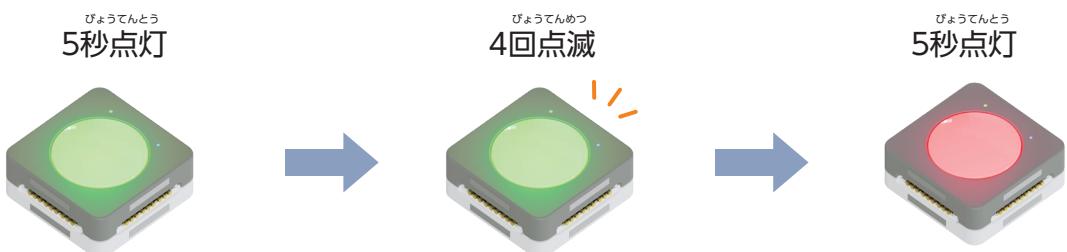
自動車用と歩行者用の信号機をつくりましょう。

### プログラムの動き

#### 自動車用信号機



#### 歩行者用信号機



### 使うブロック



イベント  
が押されたとき



見た目

LEDを で点灯

LEDを消灯



制御  
10 回繰り返す

## プログラム例

自動車用信号機  
じどうしゃようしんごうき



歩行者用信号機  
ほこうしゃようしんごうき



### 教員の方へ

※まずは簡単な自動車用信号機から作成させてください。  
作成できた生徒から、歩行者用信号機に挑戦させてください。



## ポイント

繰り返しのプログラムをつくる場合、図のような間違いをする場合があります。

**X よくある間違い**



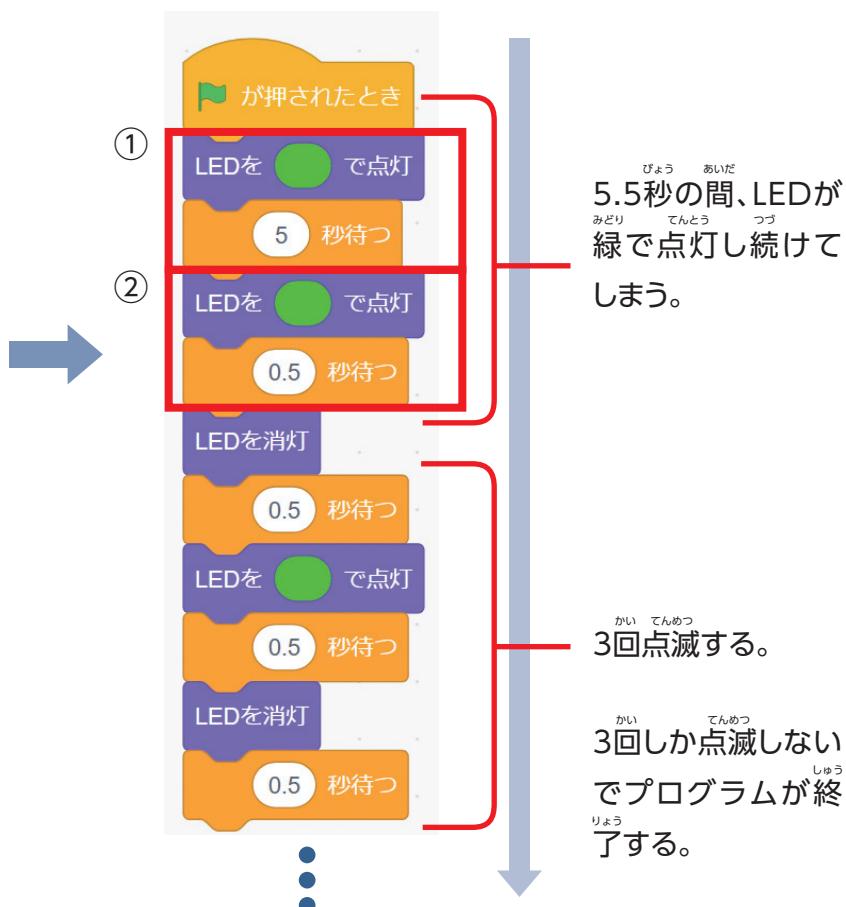
**O 正しいプログラム**



なぜ「4回繰り返す」ブロックを使っていても3回しか点滅しないのでしょうか。



を使わずにプログラムをつくると、①緑で5秒点灯と②点滅1回目の0.5秒点灯が続いて5.5秒点灯になるからです。



### 押しボタン式信号機をつくろう

## やってみよう！

ほこうしゃ お しんごう あお か お しき しんごうき  
歩行者がボタンを押すと信号が青に変わる押しボタン式信号機をつくり  
ましょう。



## プログラムの動き

## ①本体の ほんたい

ボタンを  
押す

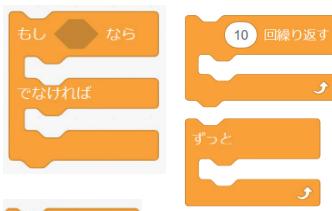
あか てんとう

### ③緑で5秒点灯後

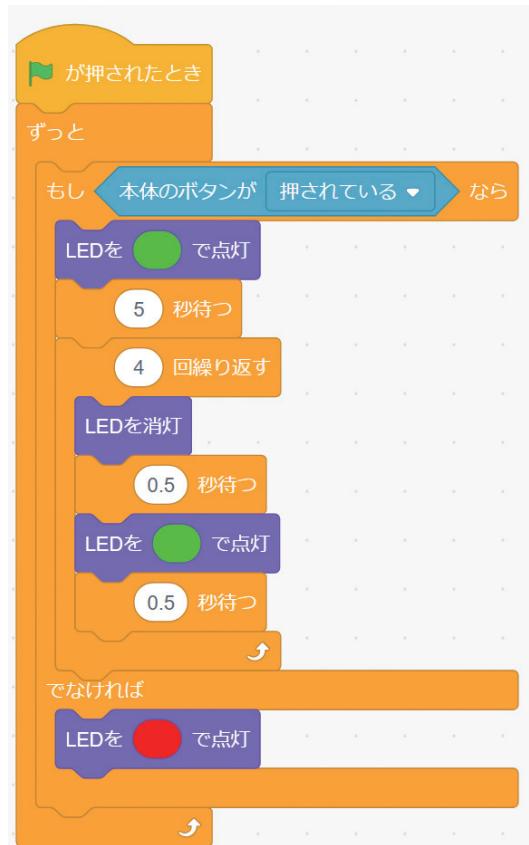
かい てんめつ

④赤で点灯

## つか 使うブロック



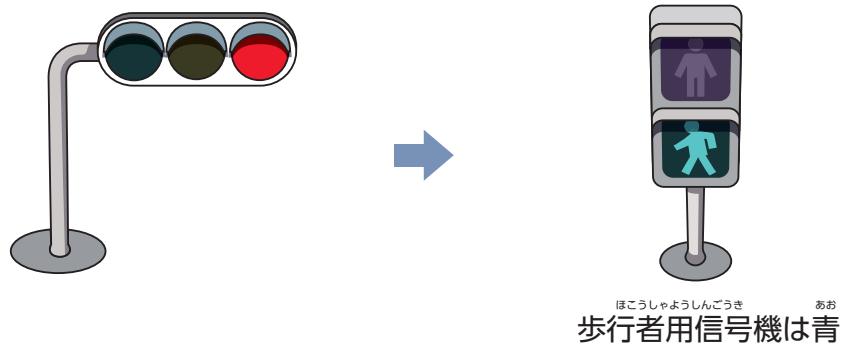
れい



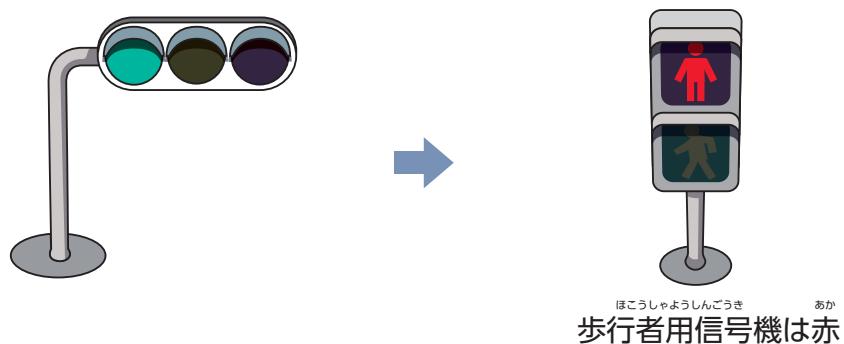
## やってみよう！

これまでに作成した自動車用信号機と歩行者用信号機を連動させてみましょう。  
本物の信号機も、自動車用信号機のライトの色によって、歩行者用信号機のライトの色が変わります。

### 自動車用信号機が赤のとき

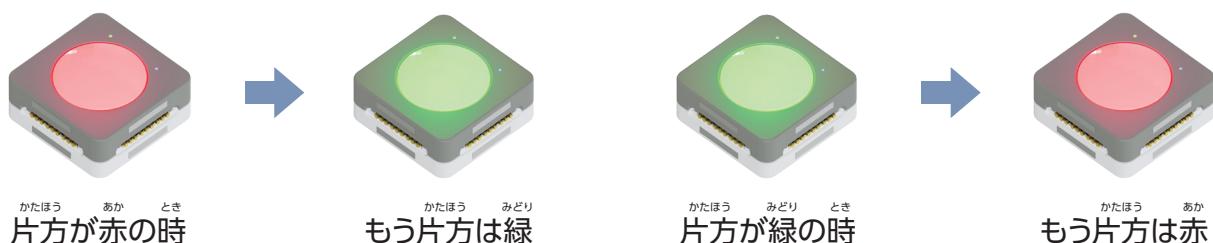


### 自動車用信号機が青のとき



## 考えてみよう

片方のメインユニットのLEDの色によって、もう片方のメインユニットのLEDの色を変えるにはどうすればよいでしょうか。



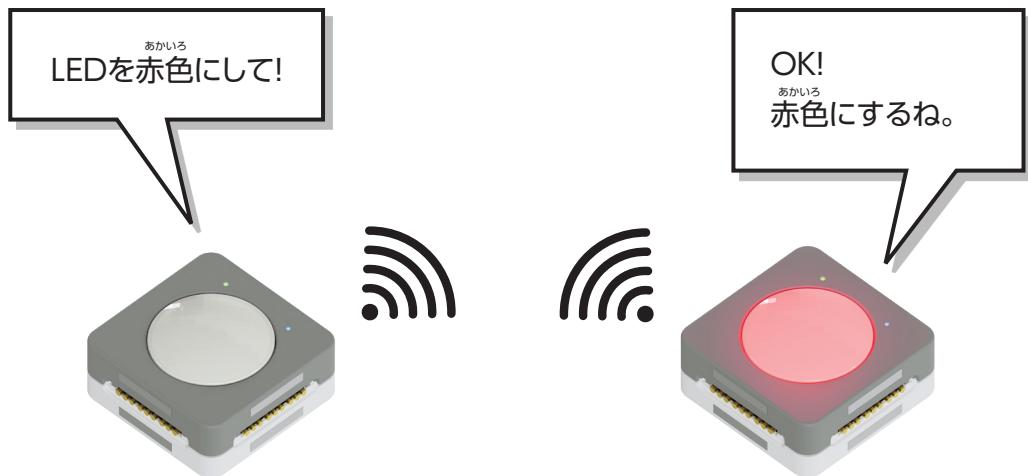
片方が赤の時

もう片方は緑

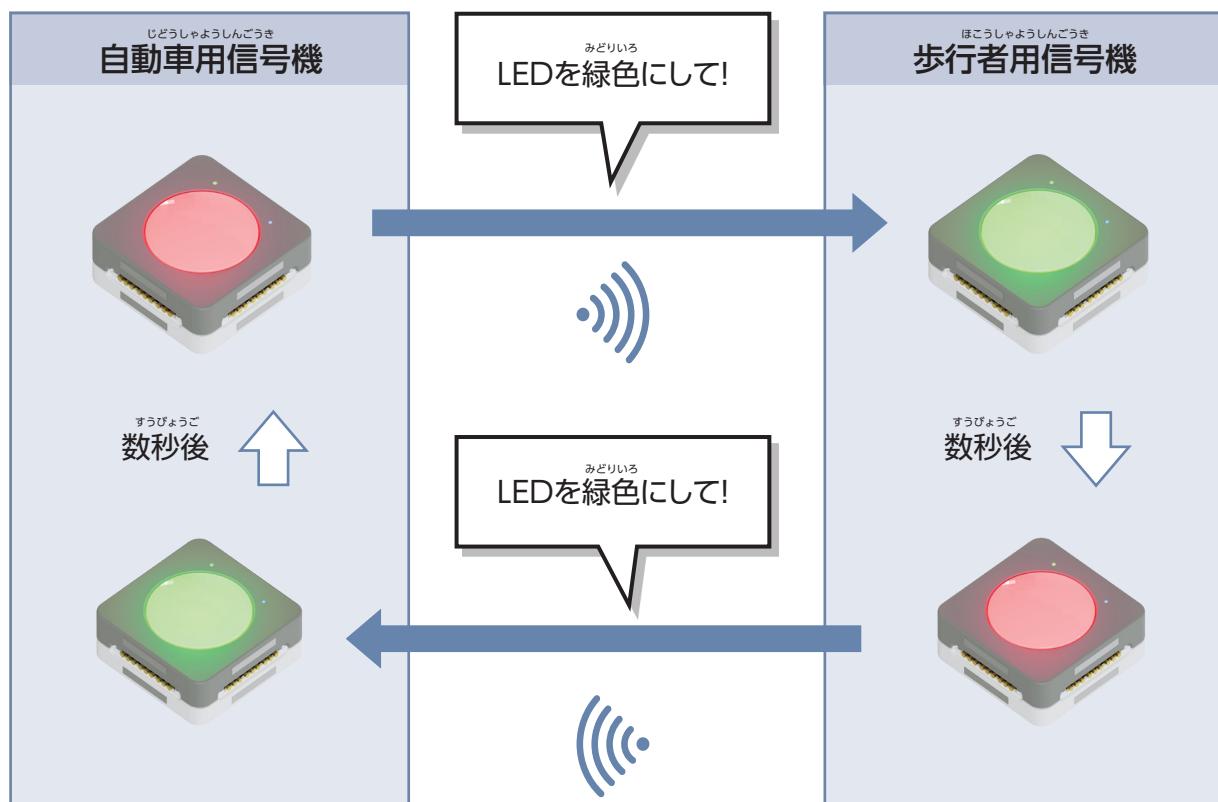
片方が緑の時

もう片方は赤

メインユニットどうしは、無線で通信することができます。



この機能を使って、自動車用信号機と歩行者用信号機を連動させてみましょう。



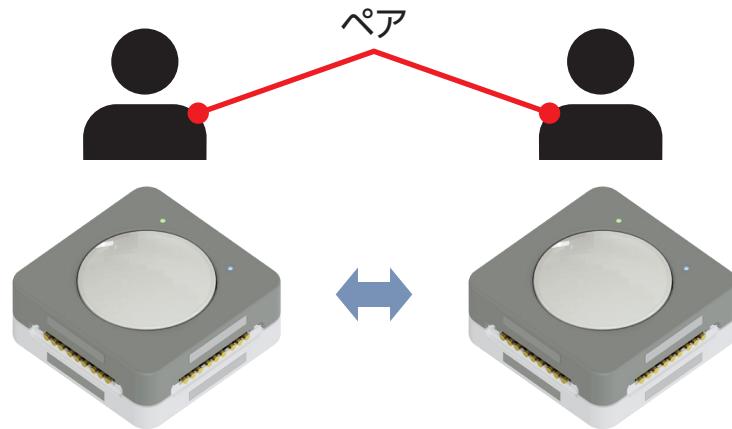
### 補足 有線と無線の通信と双方向通信

- 有線通信** : パソコンとメインユニットをUSBケーブルでつないでいるように、「線でつながって」通信すること。
- 無線通信** : メインユニットどうしの通信のように「線でつながずに」通信すること。
- 双方向通信** : 信号機どうしの通信のように「お互いに」信号を送りあう通信のこと。

# 無線の接続方法

2台のメインユニットを無線で接続してみましょう。

(1) 通信するメインユニットを決めて、ペアをつくりましょう。



(2) 0~255の中から「ペアで同じ数字」を決めて 無線の  
無線のグループを設定 0 に数字を入力しましょう。

(3) が押されたとき と 無線のグループを設定 0 をつなげましょう。



教員の方へ

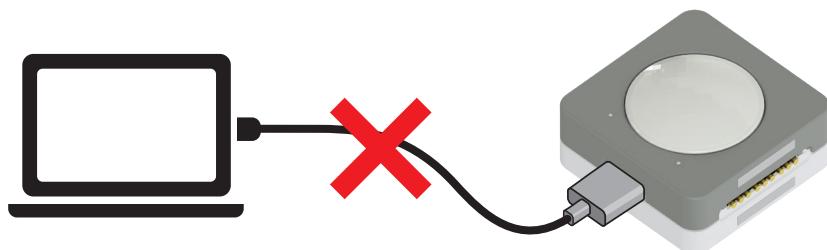
※ 通信させるメインユニットどうしは、必ず同じネットワークグループの数字を設定してください。

# プログラムを転送する方法

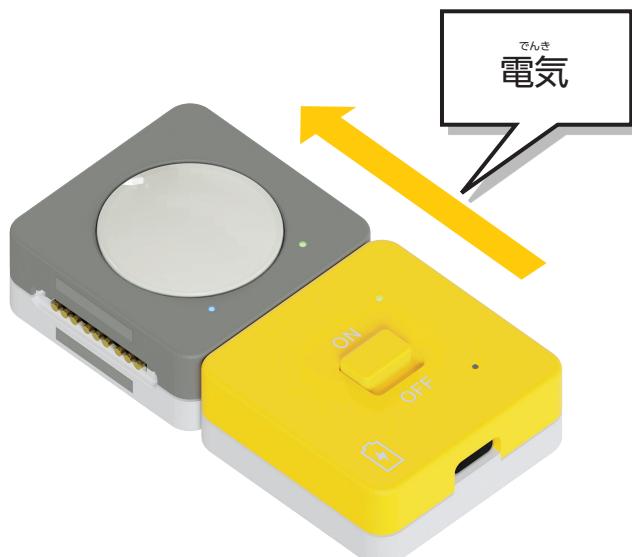
「転送」とは、デバイスでつくったプログラムをメインユニットに保存することです。

無線のブロックを使うためには、**プログラムを転送する**必要があります。

つくったプログラムを転送してメインユニットに保存することで、メインユニットとデバイスをUSBケーブルでつながなくとも、プログラムが動くようになります。



ただし、ケーブルでつながない代わりにバッテリーを接続して電気を送る必要があります。  
(バッテリーがない場合は、USBケーブルを使ってデバイスやACアダプタ(5V,1~1.5A)から電気を送ってください。)



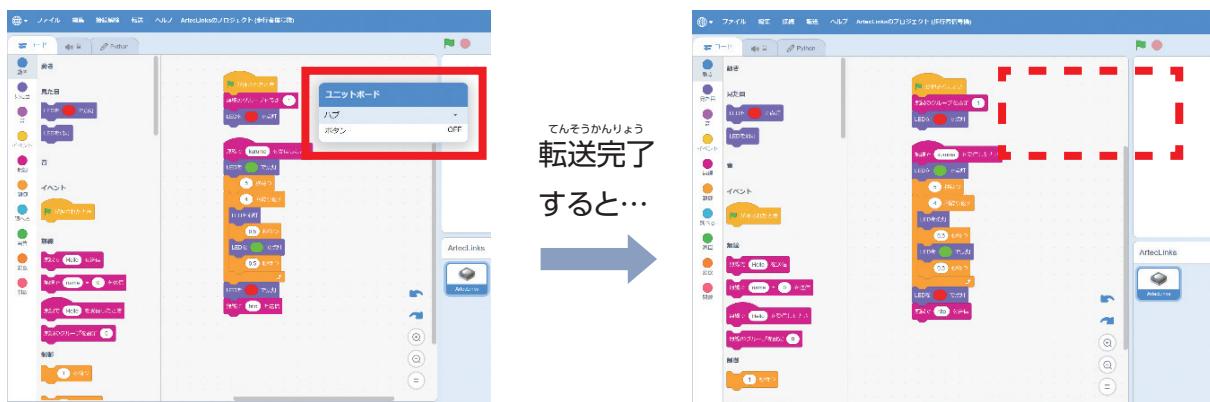
プログラムを変更する場合は、もう一度メインユニットとデバイスを接続して変更したプログラムを転送する必要があります。

## やってみよう！

①作成したプログラムを表示したまま「転送」をクリックして、メインユニットにプログラムを送ります。



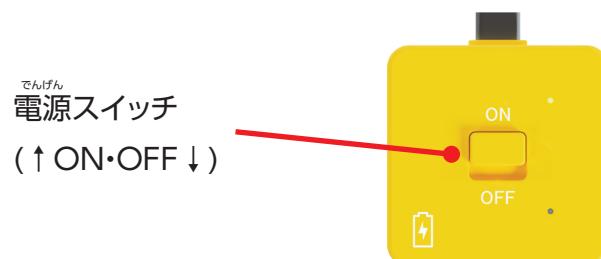
メインユニットにプログラムが転送されると、自動でデバイスとの接続が解除されます。接続が解除されるとユニットボードが画面から消えます。



②メインユニットからUSBケーブルを取り外し、バッテリーをつなぎます。

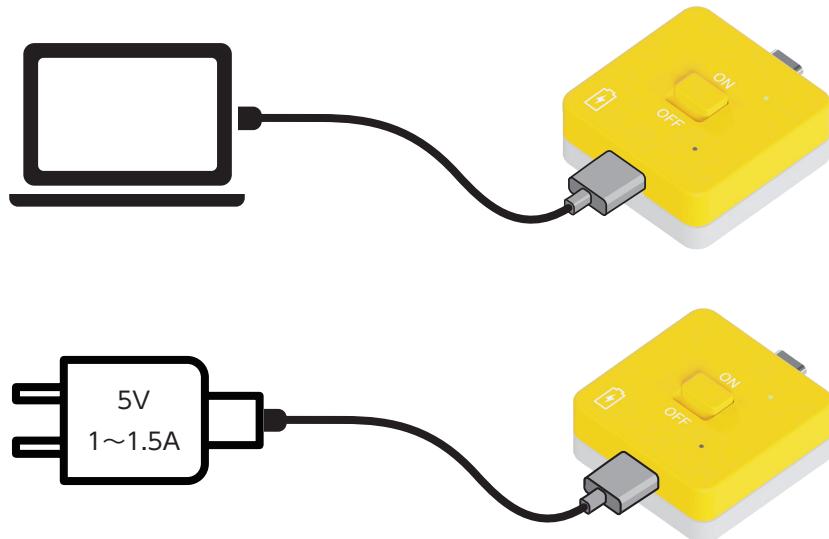


電源スイッチをONにするとバッテリーからメインユニットに電気が送られて、プログラムが動くようになります。

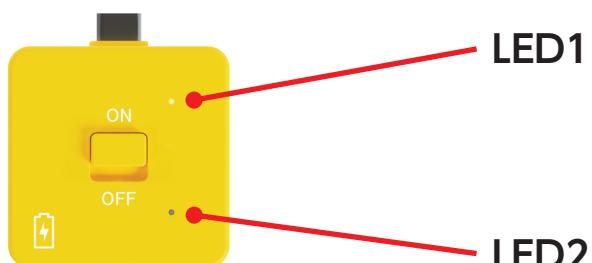


# バッテリーの使い方

バッテリーを使用するときは、あらかじめ充電しておく必要があります。バッテリーとデバイスをUSBケーブルで接続すると充電できます。

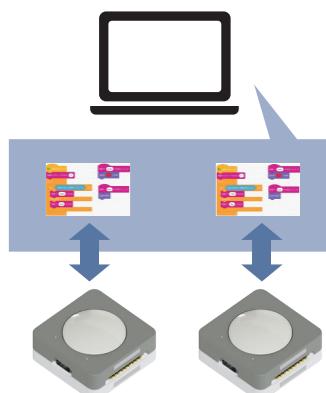


バッテリーの電源をONにするとLED1が緑色に点灯し、充電が少なくなると赤色に点灯します。充電中はLED2が赤色に点灯し、電気が満タンになると青色に点灯します。



## 補足 1台のパソコンで双方向通信をする方法

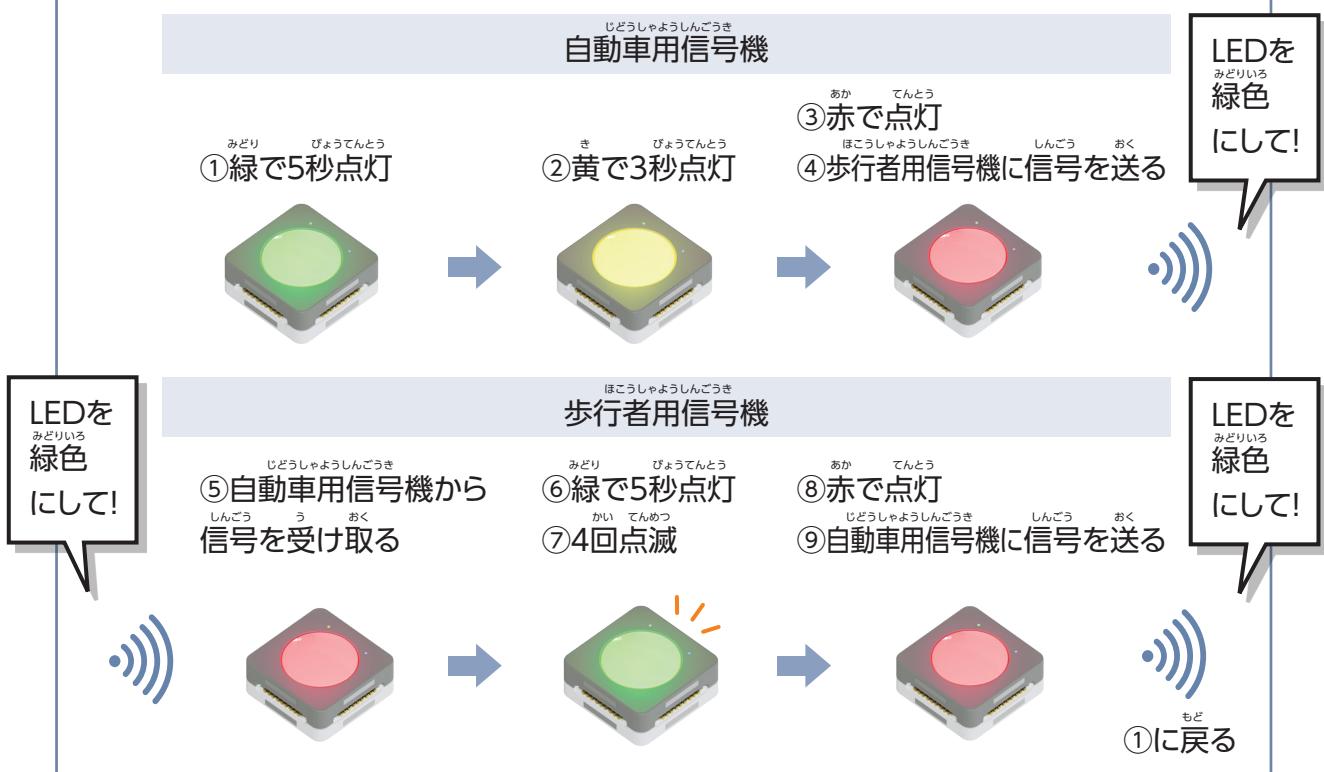
1台のパソコンでも、ソフトウェアを2つ起動して2台のメインユニットを接続することで、同じように双方双方向通信を行うことができます。



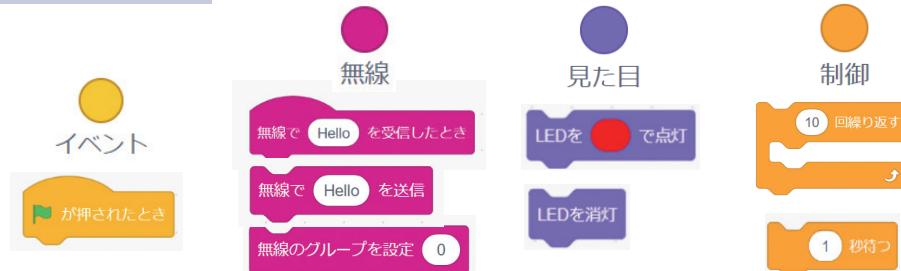
## やってみよう！

双方通信を使って2つの信号機のLEDを制御するプログラムをつくりましょう。

### プログラムの動き



### 使うブロック



### 無線カテゴリーのブロック

無線で Hello を受信したとき  
同じグループの から信号(Hello)を受信したとき、下に続くプログラムを実行する

無線で Hello を送信  
同じグループに信号(Hello)を送信する

## プログラム例

### 自動車用信号機



### 歩行者用信号機



### 教員の方へ

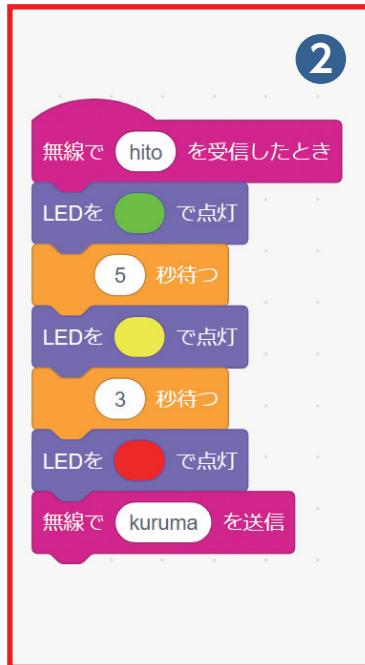
※ ブロックで送受信する信号の名前(英数字)にするように気をつけてください。



の「Hello」の部分)は必ず半角

# プログラムの説明

## 自動車用信号機



① バッテリーの電源スイッチをONになると  
つながるブロックのプログラムが実行されます。  
LEDが緑→黄→赤と点灯し歩行者用信号機に信号を送ります。

## 歩行者用信号機



① 歩行者用信号機は赤に点灯します。  
② 同じグループの信号機から信号 (kuruma)を受け取ったときLEDが緑に点灯します。自動車用信号機はkuruma、歩行者用信号機はhitoという信号を送ります。

**ArTeC Links**  
アーテックリンクス

## 小学校プログラミング シンプルセット 総合

テキストに関するお問い合わせ

株式会社 **アーテック** お客様相談窓口  
 ◀Webからのお問い合わせはこちら  
<https://www.artec-kk.co.jp/contact/>  
 お電話でのお問い合わせはこちら  
TEL 072-990-5656

B049851 K0324B