

地域ICTクラブ

山中児童センター 10月25日（木）

(株)インテトラス 中裕也、四津谷瞬

本日の流れ



micro:bitで動くバギー(車)を動かそう！

micro:bitを使った自分の作品を考えよう！

授業のルール

参加してくれたみんなへ

- 途中で休憩時間をとります。
疲れた子はいつでも休憩をしてもかまいません。
- 飲み物もOKです。机にこぼさないように注意してください。
- わからないことがあったら近くの大人に質問してください。
- まわりの子が困っていたら手伝ってあげてください。
みんなでおもしろいものを作りましょう。



<大人の方々へ>

- 子どもと一緒に考えて、アドバイスしてあげてください。
- わからないことがあれば中や四津谷に聞いてください。

機材について

- パソコンやマイクロビットはみんなで使います。
やさしく使ってください。
- もし「動きが変だ」「壊れたかもしれない」と思ったらすぐに
言ってください。



前回の振り返り

前回やったことを思い出して、今回の授業に取り組もう！

サーボモーターってなに？

- 回転などがコントロールできるモーター
- 制御信号で位置や方向、角度を変えることができる
- バギーキットの「FS90R」というモータは360度回転し回る方向とスピードを変えることができる。バギーの左右の車輪にそれぞれついている。



オレンジ

制御信号

赤

プラス側

茶色

GND(マイナス側)

バギーを動かす（前進、後進、停止）



■「サーボ出力する 端子P〇 角度〇」はサーボモーターを入力した角度によって動かす。

■360度回転サーボモーターのときは回る方向とスピードが変わる。

👉左のプログラムの動き（①から⑥を繰り返す）

①前進する



②1000ミリ秒待つ(1秒前進する)



③後進する



④1000ミリ秒待つ(1秒後進する)



⑤停止する



⑥1000ミリ秒待つ(1秒停止する)

※ 前進と後進が逆の場合は モーターとmicro:bitの接続コードがクロスしているので、コードを接続しなおすと直る。

※「一時停止(ミリ秒)」

バギーを動かす (左回転、右回転)



■「サーボ出力する 端子P○ 角度○」はサーボモーターを入力した角度によって動かす。

■360度回転サーボモーターのときは回る方向とスピードが変わる。

☞左のプログラムの動き (①から④を繰り返す)

①左回転する



②1000ミリ秒待つ(1秒左回転する)



③右回転する



④1000ミリ秒待つ(1秒右回転する)

【参考】バギーを動かす(パッケージ追加)

- 「高度なブロック」⇒「パッケージを追加する」をクリックする
- 「kitronik」と入力する

パッケージを追加する... ?

kitronik

「kitronik-servo-lite」をクリックする
↓
「Servo:Lite」パッケージが追加される

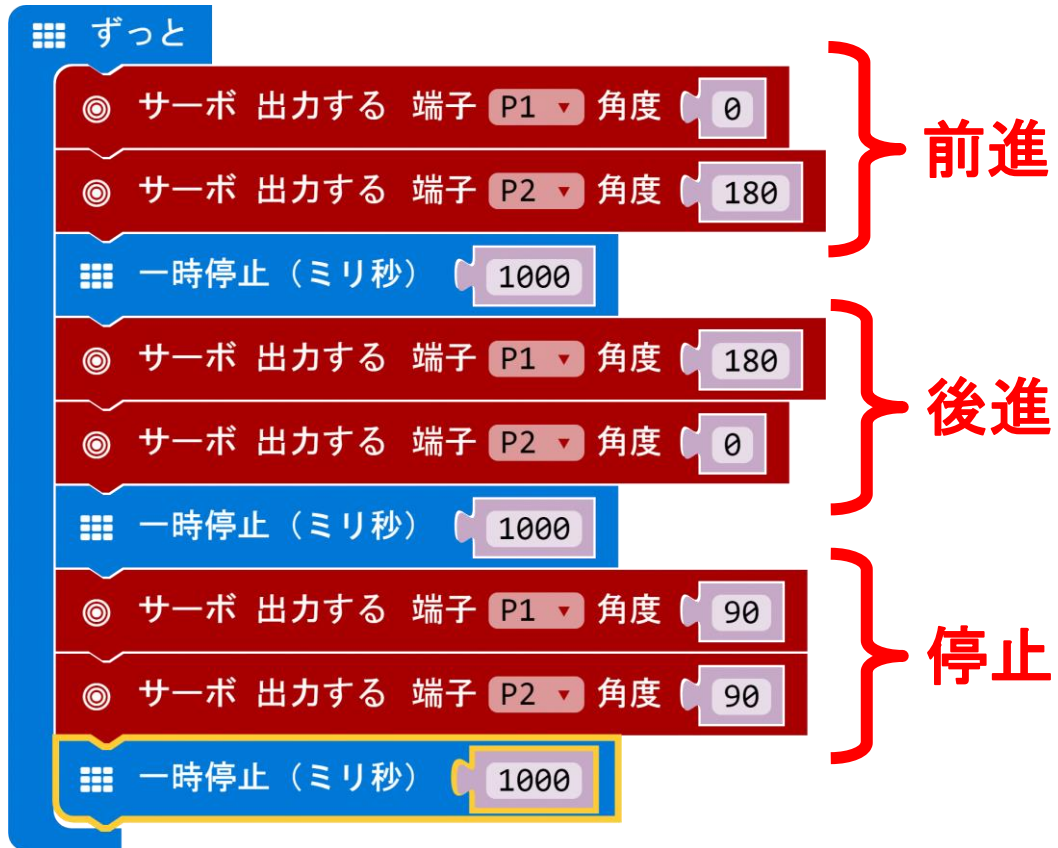
kitronik-servo-lite
Blocks for driving Kitronik micro:bit boards

kitronik-motor-driver
Blocks for driving the Kitronik micro:bit motor driver board

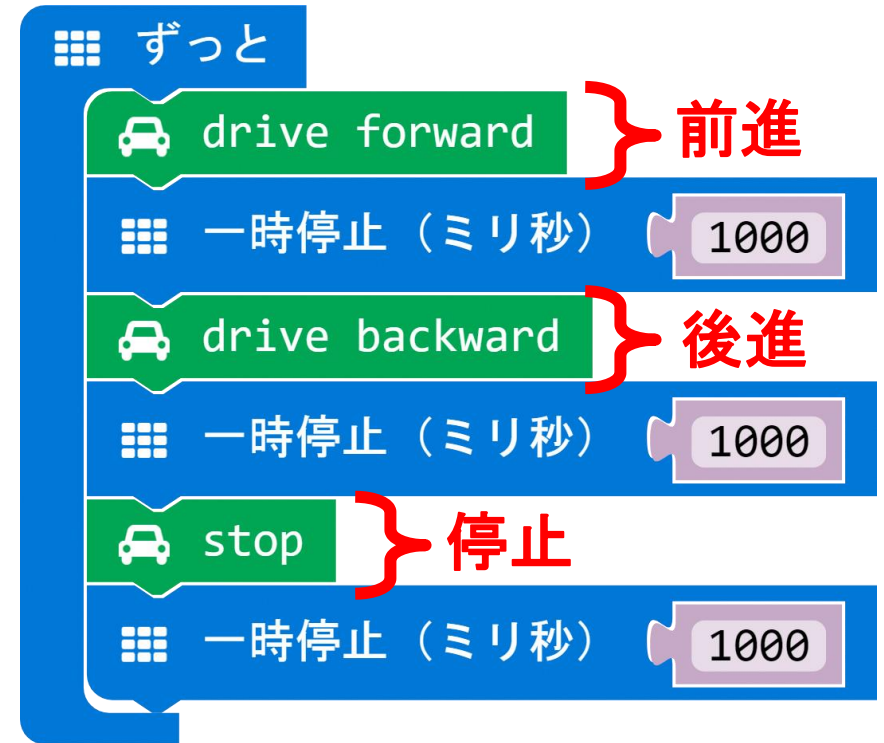
kitronik-I2C-16-servo
Blocks for driving the Kitronik I2C 16 servo expansion board
(www.kitronik.co.uk/5612)

【参考】バギーを動かす（「Servo:Lite」での前進、後進、停止）

入出力端子



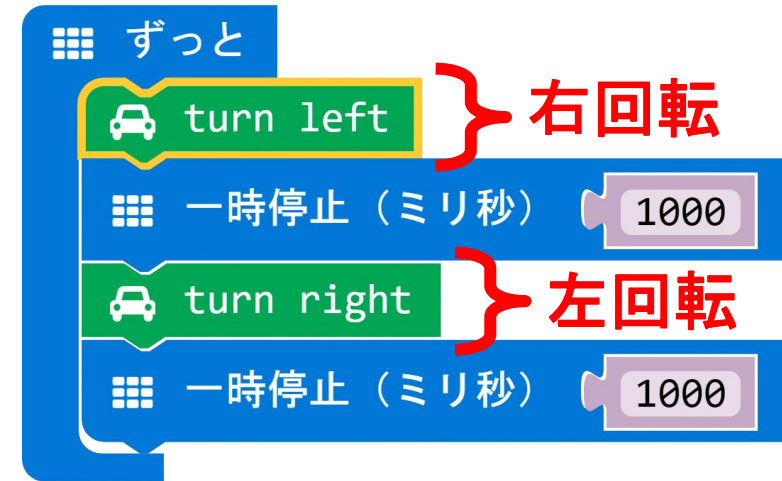
Servo:Lite



【参考】バギーを動かす（「Servo:Lite」での前進、後進、停止）

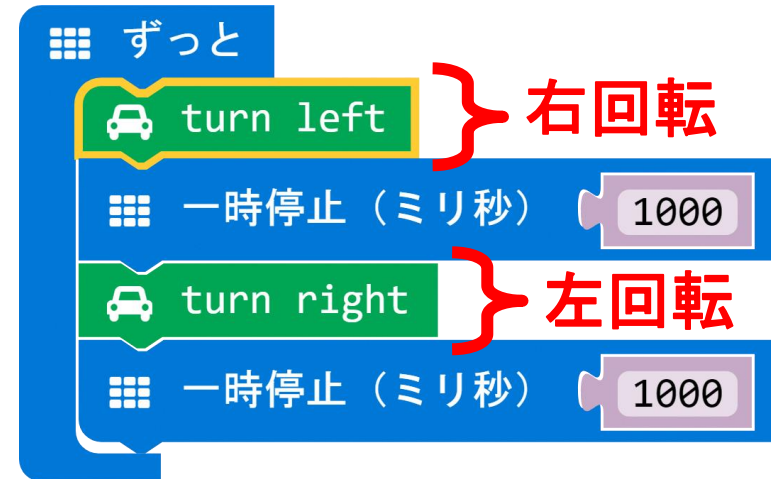
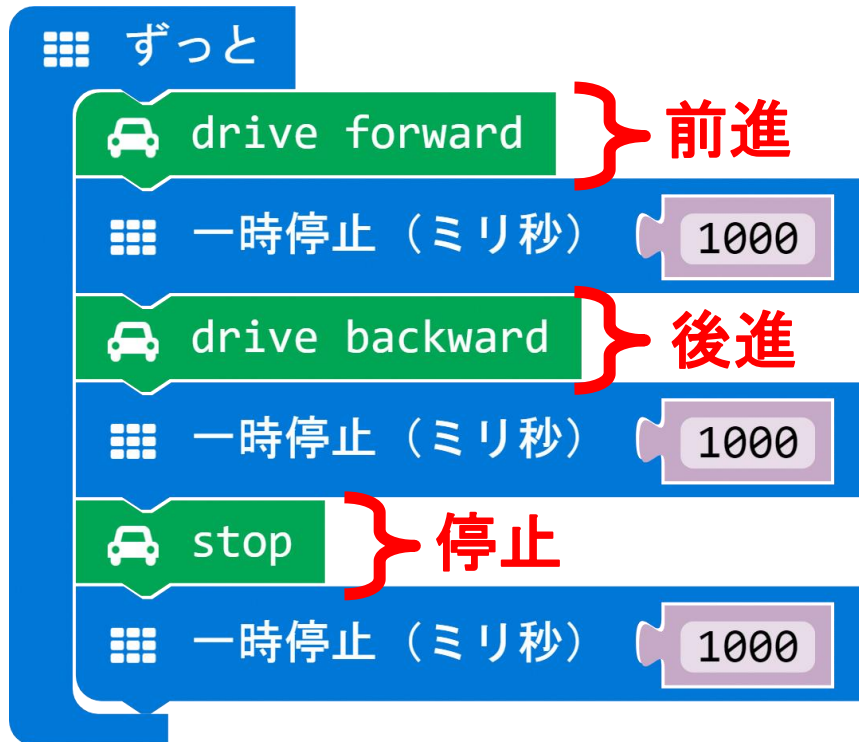
入出力端子

Servo:Lite



【参考】バギーを動かす（「Servo:Lite」を使おう）

Servo:Liteのブロックを使ってバギーを動かそう



バギーのLEDを点灯させよう！

バギーのLEDを点灯する(パッケージ追加)

- 「高度なブロック」⇒「パッケージを追加する」をクリックする

パッケージを追加する... ?

検索または、プロジェクトのURLを入力...

devices ベータ
BETA - Camera, remote control and other Bluetooth services. App required.

bluetooth
Bluetooth services

neopixel
AdaFruit NeoPixel driver

「neopixel」をクリックする
↓
「Neopixel」パッケージが追加される

バギーのLEDを点灯する(1色で点灯) 1/2

最初だけ

変数 pixel を 端子 P0 に接続しているLED 5 個のNeoPixel (モード RGB (GRB順)) にする

ずっと

pixel を 赤 色に点灯する

同じ名前の変数にする

LEDの数を入力

色を選ぶ

1. 「変数」-「変数を追加する」で好きな名前(例は、「pixel」)を入力してOKをクリック
2. 「変数」-「変数 変数▼を0にする」を「最初だけ」の中に入れる
3. 「変数▼」の▼をクリックして作った変数の名前を選ぶ

(次のページにつづく)

バギーのLEDを点灯する(1色で点灯) 2/2

最初だけ

変数 pixel を 端子 P0 に接続しているLED 5 個のNeoPixel (モード RGB (GRB順)) にする

ずっと

pixel を 赤 色に点灯する

色を選ぶ

4. 「Neopixel」—「端子 P0▼に接続しているLED 24 個のNeopixel (モード RGB (GRB順))」を「変数 変数▼を 0 にする」の「0」に入れる
5. 「端子 P0▼に接続しているLED 24 個のNeopixel (モード RGB (GRB順))」の **24** を **5** に変える
6. 「Neopixel」—「変数▼を 赤▼ 色に点灯する」を「ずっと」の中に入れる

バギーのLEDを点灯する(LED別に色を変える)

The image shows a Scratch script with two main sections: "最初だけ" (Only at the start) and "ずっと" (Forever loop).
1. "最初だけ" section:
- Block 1: "変数 pixel を 端子 P0 に接続しているLED 5 個のNeoPixel (モード RGB (GRB順)) にする" (Set the variable pixel to the 5 NeoPixels connected to terminal P0, mode RGB (GRB order)).
- Block 2: "pixel の 0 番目のLEDを 赤 色に設定する" (Set the 0th LED of pixel to red).
2. "ずっと" section:
- Block 3: "pixel を設定した色で点灯する" (Light up pixel with the set color).
A red callout box points to the "0" in the second block, stating: "バギーのLED(5個)は左から0、1、2、3、4番目となる" (The baggie's LEDs (5 total) are numbered 0, 1, 2, 3, 4 from left to right).

- 「Neopixel」—「変数▼の 0番目のLEDを 赤▼ 色に設定する」を「最初だけ」の「変数 pixel を～にする」の下に入れる ※このブロックは 設定する なので 点灯しない ことに注意する
- 「変数▼の 0番目のLEDを 赤▼ 色に設定する」の 0 を変えると光らせるLEDを選べる(0だと一番左のLED)

この作業を色を分けたいLEDの数だけ行う

- 「Neopixel」—「変数▼ を設定した色で点灯する」を「ずっと」の中に入れる(変数の名前は同じにする)

マイクロビットどうして通信して
バギーを動かそう！

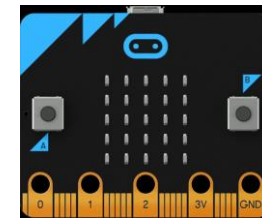
マイクロビットで通信

- マイクロビット同士は無線（むせん）で通信することができる

例えば、マイクロビットのAボタンを押すと、他のマイクロビットのLEDに「A」と表示するなどができる

- マイクロビットでマイクロビット（バギー）を動かそう

- ・ Aボタンを押すと、バギーが左に曲がる
- ・ Bボタンを押すと、バギーが右に曲がる
- ・ A+Bボタンを押すと、バギーが前に進む



まずは、『【参考】 バギーを動かす（パッケージ追加）』のページを見て「Servo:Lite」パッケージを追加しよう！

マイクロビットで通信(データを送る)

最初だけ

無線のグループを設定 1

・「無線」-「無線のグループを設定 1」
グループの数値はデータを受け取るマイクロビットと
合わせないと通信できない

◎ ボタン A が押されたとき

無線で文字列を送信 “ hidari ”

・「無線」-「無線で文字列を送信 “ ”」

◎ ボタン B が押されたとき

無線で文字列を送信 “ migi ”

◎ ボタン A+B が押されたとき

無線で文字列を送信 “ mae ”

マイクロビットで通信(データを受け取る)

最初だけ

無線のグループを設定 1

- ・「無線」-「無線のグループを設定 1」
グループの数値はデータを送るマイクロビットで設定した数値と合わせないと通信できない

無線で受信したとき receivedString

- ・「無線」-「無線で受信したとき receivedString」
receivedStringは、受け取った文字列

もし (receivedString = "hidari")

なら turn left

もし (receivedString = "migi")

なら turn right

もし (receivedString = "mae")

なら drive forward

- ・「論理」-「もし~なら」
- ・「変数」-「receivedString」
- ・「文字列」-「" "」

受け取った文字列から判断(はんだん)して
右や左、前に動かす

LEDとモーターでバギーを好きに
プログラミングしよう

マイクロビットで作る作品の アイデアを考えよう

スケジュール

8月23日（木）

マイクロビットをつかってみよう

9月27日（木）

マイクロビットで動く車を作ろう

10月25日（木）

マイクロビットで動く車で遊ぼう、作品のアイデアを考えよう

11月22日（木） ←次回、12月13日（木）

作品のアイデアをまとめてくる

作品を制作する！

12月15日（土）、16日（日）

NT加賀2018に出展する！

