

Scratchベースで動かそう! スタディーノで Studuino 多電子工作

第8回 お絵描きロボカー

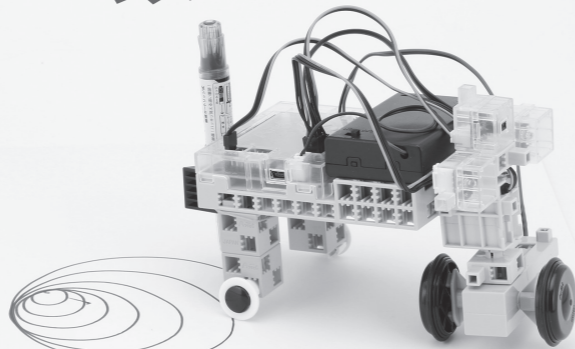
タートルグラフィックスって知ってるかな? 紙の上を歩くカメ(タートル)にペンをつけると、歩いた跡が線で描かれるよね。こんなふうに、ペンを備えたカメ型のロボットをプログラムで動かして、絵を描くことをタートルグラフィックスっていうんだ。今回は、前回つくったロボットを改造して、ペンで絵が描けるようにしてみよう。プログラムにしたがって、正方形や星形、渦巻きなど、いろいろな絵が描けるよ。

監修・原案/青山学院大学客員教授 阿部和広
協力/NPO法人 CANVAS 文/塩野祐樹 工作/石井モルナ

「コカねっ!」のスペシャルページで復習しよう

www.kodomonokagaku.com/magazine/studuino/

ワークショップ
開催決定!
(詳細はP71下)



用意するもの

前回つくった「障害物よけロボカー」に加えて、サインペンと大きな紙(画用紙や模造紙、カレンダーの裏など)、基本三角ブロック2個を追加で用意しよう。あと、定規と分度器も使うよ。ロボカーをテーブルや床で動かすときは、汚さないように新聞紙を敷くといいかもしれないね。油性のマジックは消しにくいから気をつけよう。ホワイトボードマーカーと紙のホワイトボード(ビニールコートされた厚紙)を使うと、何回も描けて便利だよ。

- 前回の障害物よけロボカー部品
スタディーノ ロボティスト用カバー台座付き…1個
LED…2個(色は自由)
赤外線フォトリフレクタ…1個
センサー接続ケーブル…3本
アテックブロック 四角…4個、目…2個
サーボモーター…1個
DCモーター…1個
電池ボックス…1個

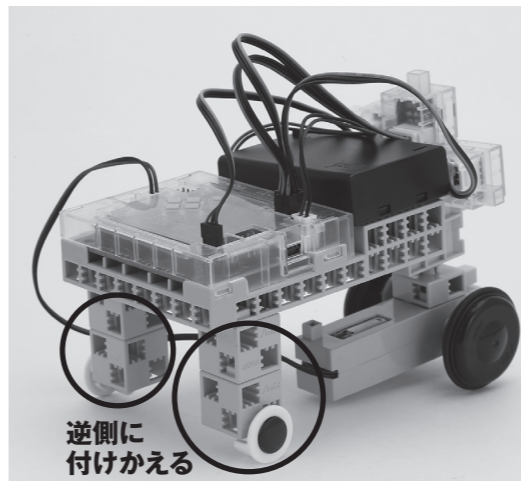
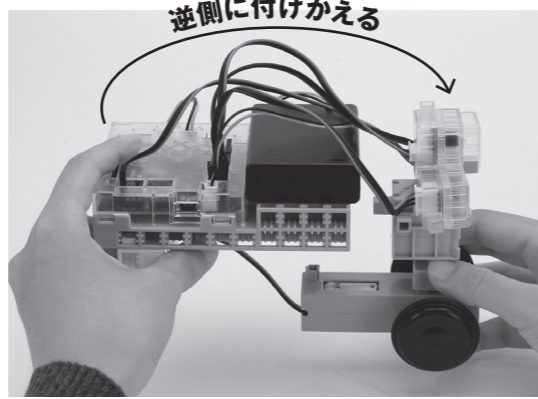
- 今回追加されたもの
アテックブロック 三角…2個
サインペン…1本
大きな紙(画用紙や模造紙、カレンダーの裏など)
定規
分度器

ロボカーの改造

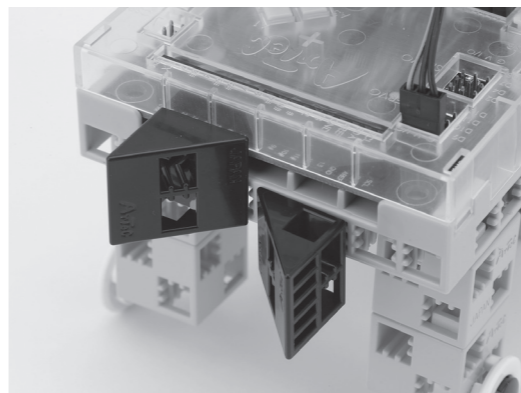
では、ロボカーを改造しよう。障害物よけロボカーでは、向きを素早く変えられるように(回頭性をよくするために)、後輪側に電池ボックスを置いていたけど、今回は、前輪(DCモーターのついた駆動輪)に荷重をかけて、正確に動かせるようにするよ。

まず、電池ボックスを止めていた輪ゴムを外そう。次に、前輪のサーボモーターと後輪のブロックの固まり(左右とも)を台座から外す。そうしたら、台座の前後を逆にして、新しい前側に前輪のサーボモーター、新しい後ろ側の左右に後輪のブロックの固まりを差し込もう。

電池ボックスは、前と同じ場所に置くよ。前輪のセンサーケーブルが電池ボックスに当たるので、輪ゴムがなくても安定するはずだ。



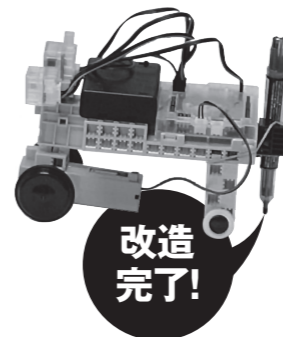
次に、基本三角ブロック2個を、台座の後ろ、下の写真の場所に差し込もう。



最後に、サインペンを三角ブロックの間に置いて、輪ゴムを後輪に通して止めよう。このとき、しっかりと固定するのではなく、手で押すと動くくらいのゆるさにするのがポイントだ。サインペンのキャップは、まだつけたままでいいよ。



配線は、DCモーターの白いコネクターをM1からM2に変えるだけで、他は変更なしだ。M2にするのはケーブルの長さに余裕がないからだ。電池ボックスのスイッチはOFFにすること。これで、ロボカーの改造は完了だ!



ブロックプログラミング 環境の準備と設定

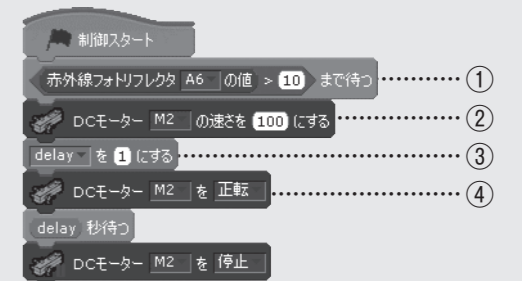
スタディーノ基板とパソコンをUSBケーブルでつなぎ、パソコンでブロックプログラミング環境を起動しよう。起動したら、[編集]メニューの「入出力設定…」を、配線した通りに設定しよう。今回変わったのはDCモーターをM2にしたことだけだけど、念のために全部の配線を表に載せておこう。

続いて、電池ボックスのスイッチをONにして、[編集]メニューの[モーター校正…]で、前輪がまっすぐになるようにD9の角度を調整しよう。終わったら、電池ボックスのスイッチをOFFにしよう。

部品	端子
DCモーター	M2 (今回変わったのはここだけ)
サーボモーター	D9
LED(赤)	A4
LED(緑)	A5
赤外線フォトリフレクタ	A6
電池ボックス	POWER

直線を描く

ロボカーのテストもかねて、最初に直線を描いてみよう。プログラムはこんな感じだ。



[プログラム解説]

- ①いきなり動き出さないように、赤外線フォトリフレクタに手をかざすまで待つ。
- ②DCモーターの速さを100にする。
- ③DCモーターを動かす秒数を変数delayにセット。
- ④DCモーターをdelay秒動かして止める。

プログラムを入力したら、[実行]メニューの[プログラム作成・転送]でスタディーノに転送しよう。

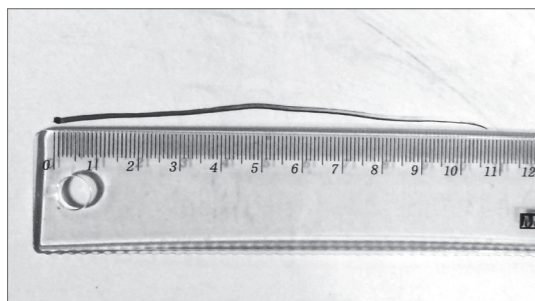
転送が終わったら、USBケーブルを抜き、サインペンのキャップを外す。このとき、ペンの先が2つの後輪を結んだ線よりも少し下に飛び出すようにペンの位置を調整するのがコツだ。

電池ボックスのスイッチをONにして、紙の上にロボカーを置き、赤外線フォトリフレクタに手をかざすと、ロボカーが1秒動いてから止まるはずだ。ちゃんと線は引けたかな? もし引けないようなら、ペンの位置を調整して、もう一度試してみよう。再スタートは、リセットボタンを押して、赤外線フォトリフレクタに手をかざそう。

ちゃんと線が引けたら、電池ボックスのスイッチをOFFにして、サインペンにキャップをつけておこう。

動かす秒数(delay)の調整

今は適当な長さの線を描いたけど、長さを決めて線が描けるように、動かす秒数(delay)を調整するよ。紙に描かれた線の長さを定規で測り、その数値を記録しよう。次の写真は実際に試したものだ。約10.7cmだね。



1秒間で10.7cm動いたということは、1cm動くのにかかった秒数は、1(秒)を10.7(cm)で割った数になる。ここで、動かしたい距離(線の長さ)をlengthとすると、その数にlengthをかければdelayが求まるね。

$$\text{delay} = 1 \div 10.7 \times \text{length}$$

これをプログラムに当てはめ、同時に、再利用しやすいようにforward関数として定義しよう。

```

制御スタート
赤外線フォトリフレクタ A6 の値 > 10 まで待つ
DCモーター M2 の速さを 100 にする
length を 10 にする
forward 関数をコールする
サーボモーター D9 を 90 度にする
0.5 秒待つ
delay を 1 / 10.7 * length にする
DCモーター M2 を 正転
delay 秒待つ
DCモーター M2 を 停止
    
```

【プログラム解説】

- ① 動かす距離(線の長さ)をlengthにセットしてforward関数を呼び出す。ここでは10cm。
- ② 確実に直進するようにハンドルをまっすぐ(90°)に向ける。
- ③ $1 \div 10.7 \times \text{length}$ の答えをdelayにセットする。
- ④ DCモーターをdelay秒動かして止める。

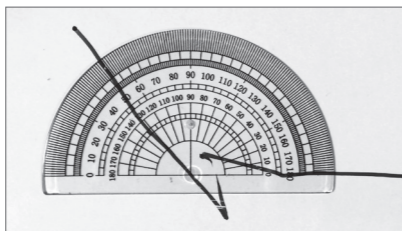
プログラムを入力したらスタディーノに転送して、さっきと同じように紙の上でロボカーを動かしてみよう。線の長さは大体10cmになったかな? 今回のしくみでは、長さの厳密な制御はできない。また、電池の消耗具合や床との摩擦などでも結果は変わるので、数cmの誤差は出ると思う。

確認できたら、電池ボックスのスイッチをOFFにして、サインペンにキャップをつけておこう。

右回りする

次に、ロボカーが決められた角度で右回りするにはどうすればよいかを考えてみよう。まず、ハンドルを右に切って(サーボモーターの角度を180°にして)、あとは直線のときと同じように、1秒間DCモーターを回転させたときの角度を測り、1°あたりの秒数を計算してdelayの値を求めればいいね。

これを実際に試したのが次の写真だ。曲がるときにちょっとペンが動いているけど、約50°だね。



式はこのようになる。角度の変数はangleにしたよ。

$$\text{delay} = 1 \div 50 \times \text{angle}$$

プログラムはこうだね。右回りの関数はturnRightとした。

```

制御スタート
赤外線フォトリフレクタ A6 の値 > 10 まで待つ
DCモーター M2 の速さを 100 にする
length を 10 にする
forward 関数をコールする
angle を 90 にする
turnRight 関数をコールする
length を 10 にする
forward 関数をコールする
turnRight 関数
サーボモーター D9 を 180 度にする
0.5 秒待つ
delay を 1 / 50 * angle にする
DCモーター M2 を 正転
delay 秒待つ
DCモーター M2 を 停止
    
```

【プログラム解説】

- ① forward関数を呼び出して、10cmの線を描く。
- ② turnRight関数を呼び出して、90°右回りする。
- ③ forward関数を呼び出して、10cmの線を描く。
- ④ ハンドルを右に切る(サーボモーターを180°にする)。
- ⑤ $1 \div 50 \times \text{angle}$ の答えをdelayにセットして、その秒数DCモーターを動かして止める。

このプログラムをスタディーノに転送して、ロボカーで線を描いて確認してみよう。回る角度が90°にならないときは、「 $(1/50) \times \text{angle}$ 」の50を変えよう。90°以上回るときはこの数を大きく、90°以下のときはこの数を小さくして、うまくいくまでプログラムの転送と角度の測定を繰り返そう。これも長さと同じように、多少の誤差は仕方ないね。

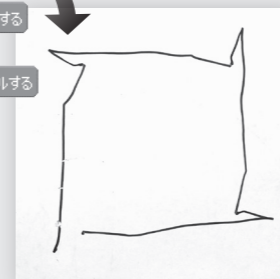
多角形を描く

好きな長さの線を、好きな角度で描けるようになったので、いろいろな多角形を描いてみよう。

正方形は、4つの辺が同じ長さで、4つの頂点の角度が直角(90°)の四角形だね。1辺の長さが10cmなら、10cm進んで、右に90°曲がる…を4回繰り返せばいいかな。正五角形の場合は、それぞれの頂点で72°回る。星形(五芒星)だと、頂点で曲がる角度は2倍の144°になる。正多角形の頂点で曲がる角度と、頂点の数には関係があるので、その法則を調べてみるとおもしろいよ。

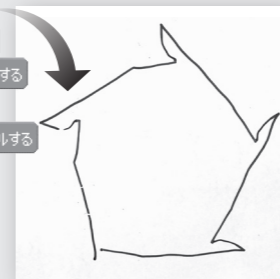
```

制御スタート
赤外線フォトリフレクタ A6 の値 > 10 まで待つ
DCモーター M2 の速さを 100 にする
4 回繰り返す
length を 10 にする
forward 関数をコールする
angle を 90 にする
turnRight 関数をコールする
    
```



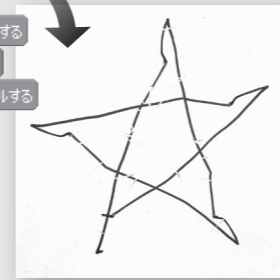
```

制御スタート
赤外線フォトリフレクタ A6 の値 > 10 まで待つ
DCモーター M2 の速さを 100 にする
5 回繰り返す
length を 10 にする
forward 関数をコールする
angle を 72 にする
turnRight 関数をコールする
    
```



```

制御スタート
赤外線フォトリフレクタ A6 の値 > 10 まで待つ
DCモーター M2 の速さを 100 にする
5 回繰り返す
length を 10 にする
forward 関数をコールする
angle を 144 にする
turnRight 関数をコールする
    
```

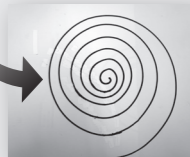


最後に、渦巻き模様を描くプログラムを紹介しよう。解説はしないので、なぜこうなるのかは自分で考えてみてね。

```

制御スタート
赤外線フォトリフレクタ A6 の値 > 10 まで待つ
angle を 180 にする
サーボモーター D9 を angle 度にする
0.5 秒待つ
DCモーター M2 の速さを 100 にする
DCモーター M2 を 正転
400 回繰り返す
サーボモーター D9 を angle 度にする
angle を -0.05 ずつ変える
0.1 秒待つ
DCモーター M2 を 停止
    
```

コカねっと! で動くようすをチェック!



応用編 普通の電子部品でつくろう



前回の「障害物よけロボカー」でつくったロボットをベースにカメラの甲羅をつけて、お絵描きカメラロボットに変身させよう!

詳しくはコカねっと! のスタディーノページへGO

ワークショップ開催決定!

「KoKa スタディーノプログラミングキット」でロボットづくりに挑戦!

本連載の著者・阿部和広先生と一緒に「KoKa スタディーノプログラミングキット」を使ったロボットをつくってみよう!

日時: 2016年7月25日(月)
(1) 10:30-12:30, (2) 14:00-16:30

会場: 日本科学未来館
参加費*: (1) 1万円, (2) 1万5000円

定員: 各回30名
対象年齢: 小3以上

内容: (1) スタートアップコース: スタディーノの基本説明&自由制作 (2) マスターコース: 新しい部品を使用して、オリジナルロボットを作成

詳細・応募方法: http://www.canvas.ws/kenkyujo/laboratory/scrobo_2016.html から応募

※参加には「KoKa スタディーノプログラミングキット」が必要です。すでにお持ちの方はご持参ください。お持ちでない方はKoKa Shopでお求めいただくか、当日、会場でも販売しています。