

Scratchベースで動かそう! Studuino^{スタディーノ}で電子工作

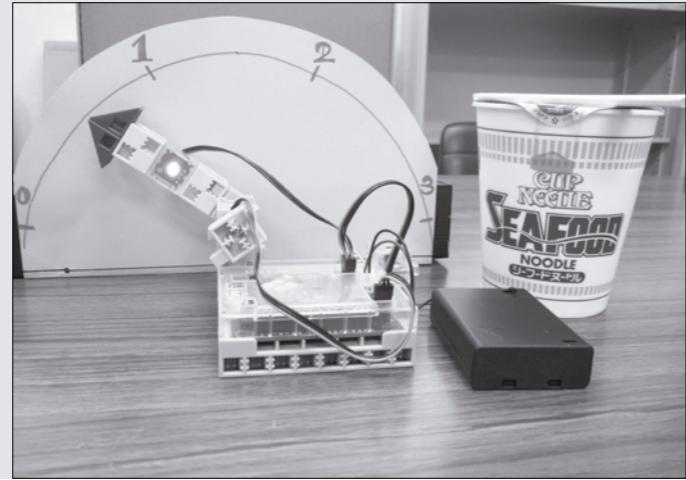
第5回 ラーメンタイマー

この連載が始まって早くも5回目!これまでの4回では、「あっち向いてホイ」で人間と対決できるロボットをつくったね。でも、スタディーノの可能性はそれだけじゃない。スタディーノは、ブロックを組み替えたりプログラムを書き換えたりすることで、ゲームだけでなく、みんなの生活に役立つ道具をつくることもできるんだ。今回はそのような道具のひとつ、時間を計ることのできるラーメンタイマーをつくってみよう。

「コカねっと!」のスペシャルページで復習しよう

www.kodomonokagaku.com/magazine/studuino/

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

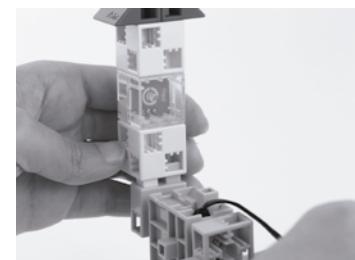


ラーメンタイマーのしくみ

力アップめんをつくるとき、種類によっていろいろだけど、お湯を入れてから3分間とか5分間とか待つよね。今回は、この時間をカウントダウンで知らせるタイマーをつくるよ。

カウントダウン中は、矢印形に組んだブロックをサーボモーターで時計の針のように動かして、時間の経過がわかるようにしてみよう。カウントする時間の長さは、スタディーノのボタンでセットできるようにするぞ。セットした時間が来たら、LEDの光とサーボモーターの動きで知らせるようにしよう。タイマーのスタートとストップを、赤外線フォトリフレクタに手をかざすことでできるようすれば、ボタンを触らないでいいから便利かも!

組み立て

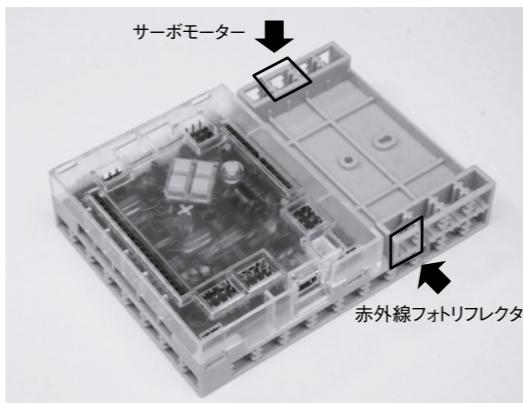


針をブロックで組み立てよう!
針をブロックで組み立て(つくり方は右カミ参照)、サーボモーターの回転軸側の穴に差し込む。

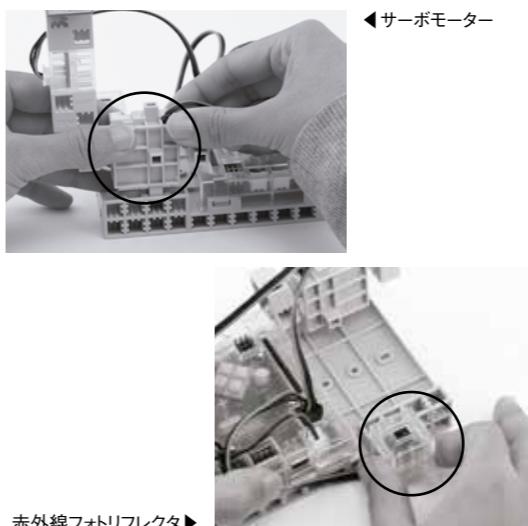


ブロックの組み方は自由だけど、いちばん下のボッチが写真と同じ位置にくるようにしよう。

サーボモーターと赤外線フォトリフレクタを、下の写真で印をつけたスタディーノ基板の穴に差し込む。サーボモーターの自由に回転する軸は、どこにも差し込まないでOKだ。



印をつけた穴に、サーボモーターと赤外線フォトリフレクタを取りつけよう。



赤外線フォトリフレクタ▶

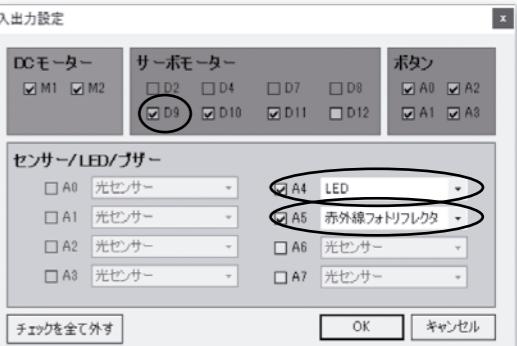
各ブロックをスタディーノ基板に取りつけたら、センサーケーブルをつなごう。LEDのセンサーケーブルはスタディーノ基板のA4に、赤外線フォトリフレクタのケーブルはA5に、サーボモーターのケーブルはD9に差そう。いつものように灰色の線が内側だ。

電池ボックスのスイッチをOFFにしてからPOWER端子に差し込めば、組み立ては完了だ!



プログラミング環境の設定

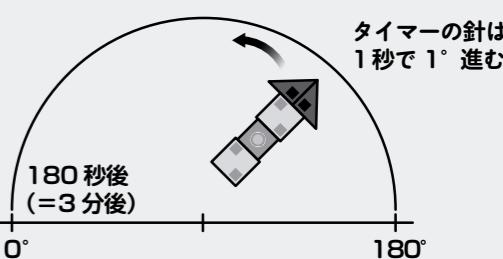
USBケーブルで、スタディーノ基板とパソコンをつなごう。パソコンでロックプログラミング環境を起動したら、[編集]メニューの[入出力設定…]を開き、配線通りに設定しよう。



それが終わったら、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、ブロックでつくった針が垂直になるように[D9]の角度を調整しよう。



いちばん簡単な3分タイマー



まずは、3分間を計るだけのタイマーのプログラムをつくってみよう。

サーボモーターは、0°から180°まで動かすことができる。時間の経過がわかるように、最初は針の位置を180°にセットして、3分たったら0°になるように動かせばよいだろう。

1分は60秒だから、3分は60秒×3分=180秒になる。ということは、1秒に動かす角度は $180^\circ \div 180\text{秒} = 1^\circ$ だ。この角度を、1秒経過するごとに最初の180°から引いて、179°、178°、177°……と針の角度を設定していくばいいんだね。

サ 一ポモーターの角度は変数「angle」で表すようにしよう。

[変数] カテゴリーの「新しい変数を作る」で、「angle」を作つておこう。

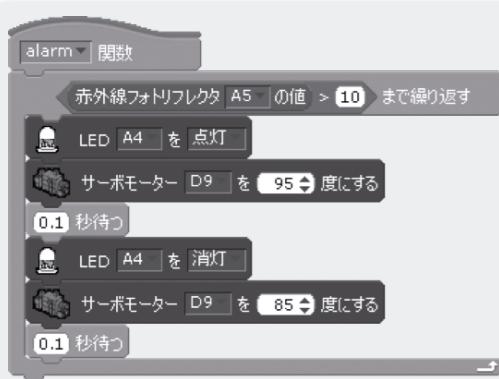
これをプログラムで表すとこんな感じになる。



[実行] メニューの「[テストモード開始]」でテストモードにしてから、緑の旗をクリックして試してみよう。1秒ごとに針が少し動いて、3分(180秒)たつたら0°で止まつたかな?

アラームを光と動きで知らせよう

これだけだと、時間が来たことがわかりにくいから、光と動きでアラームを表そう。そのための関数をつくつて、プログラムの一番最後に呼び出す(コールする)ようにすればいいね。このとき、赤外線フォトリフレクタに手をかざすとアラームが止まるようにしよう。関数の名前は「alarm」にしたよ。



組み立てたら、[alarm関数をコールする]ブロックをクリックして試してみよう。0.1秒おきに、LEDを点滅させながら、サーボモーターで針を小刻みに動かしているのがわかるかな。赤外線フォトリフレクタに手を近づけて、センサーの値が10より大きくなったらアラームは止まる(繰り返さなくなる)。

確認できたら、[alarm関数をコールする]ブロックを、さつきのタイマーのプログラムの最後につけて実験してみよう。実験するたびに3分待つのは大変だから、とりあえず180をもっと小さな値に変えて試してもいいよ。

時間のセット

3分のような決められた時間だけでなく、スタディーノ本体のボタンを使って、自由に時間がセットできるようにしてみよう。セットする時間は1分単位で、最大5分。A2ボタンを押すと増えて、A1ボタンで減るようにするのがいいかな。

セットした時間は針の位置でわかるようにしよう。時間のセットが終わつたら、赤外線フォトリフレクタに手を近づけてタイマーをスタートだ。

これらをまとめた関数「set」をつくつていこう。この関数はちょっと長くて複雑だから、じっくり説明するよ。

追加の変数は、セットする時間(分)を表す「minutes」、それを秒に変換した「seconds」、セット可能な最大の時間(分)「maxMinutes」だ。それぞれ、[変数] カテゴリーの「新しい変数を作る」でつくつておこう。



最初に「angle」を0にして、針を0度の位置に動かし、「minutes」も0分にしている。そして、赤外線フォトリフレクターに手をかざすまで(センサーの値が10より大きくなるまで)、ボタンの入力処理を繰り返しているよ。

この繰り返しの中で、A2ボタンが押されたら(A2ボタンの値が0になつたら)「minutes」の値を1増やし、A1ボタンが押されたら1減らしているのがわかるかな。それぞれのボタンが離される(値が1になる)まで待っているのは、ボタンを押している間、値が増え続けるのを防止するためだ。待つことによって、ボタンを1回押すごとに、確実に「minutes」の値が1ずつ増えたり減つたりするようにしている。

ボタンによる設定のあと、「minutes」を「maxMinutes」と比較しているのは、設定した値が最大値を超えないようにするためだ。もし、ボタンを押すすぎて最大値を超えた場合は、最大値に戻している。同じように、「minutes」を「0」と比較しているのは、最小値より小さくならないようにするためだ。つまり、「maxMinutes」が3のとき、「minutes」は絶対に0以上、3以下になる。

ここまでできたら、 $180 \div \text{maxMinutes}$ で1分あたりの角度を求め、それにボタンで設定された「minutes」を掛けて、針が指す角度「angle」をセットして、サーボモーターを動かしている。例えば、「minutes」が2で、「maxMinutes」が3だとすると、 $2 \times (180 \div 3) = 120^\circ$ にサーボモーター(針)が動くことで、いま設定している時間がわかるようにしているんだ。

赤外線フォトリフレクタに手がかざされて繰り返しを抜けたら、「minutes」に60を掛けて、秒数に変換した値を「seconds」にセットしている。

メインプログラムの変更

「set」関数と「alarm」関数を呼び出すようにメインプログラムを変更しよう。合わせて、タイマーが動いていることがわかりやすいように、1秒おき(点灯0.5秒、消灯0.5秒)にLEDを点滅するように変えてみたよ。



最初に「maxMinutes」を3にしているけど、例えば、これを5に変えると最大で5分まで計れるようになるよ。

次に、「set」関数を呼び出して、計る時間を「seconds」にセットだね。

そして、「seconds回繰り返す」の中で、針を動かしている。1秒ごとの角度「angle」は、 $(180 \div (60 \times \text{maxMinutes})) \times (-1)$ で求めているのがわかるかな。「maxMinutes」が3のときは-1、5のときは-0.6になるね。

繰り返しが終わると、「alarm」関数で時間が来たことを知らせて終了だ。

難しい部分を関数にすることで、メインプログラム自体は簡単になっているね。

ラーメンタイマー完成!

[実行] メニューから「[プログラム作成・転送]」を選んで、パソコンから切り離してもタイマーが使えるようにしよう。転送が終わつたら、USBケーブルを抜いて、リセットボタンを押そう。A2ボタンとA1ボタンで時間をセッティングして、赤外線フォトリフレクタに手をかざすとタイマーがスタートするぞ。時間が来てアラームが作動したら、赤外線フォトリフレクタに手をかざして止めよう。もう一度計りたいときは、リセットボタンだ。早速、カップめんをつくって食べてみてね! 使い終わつたら、電池ボックスのスイッチをOFFにするのを忘れずに。

今回つくったサーボモーターと針を使って、時間以外にも、いろいろな情報を示すことができる。例えば、扇形に数字を書いた紙を壁に貼り、毎日に



本連載の内容ができるキットがKoKa Shopで販売中!!

本連載2016年1月号~8月号で紹介する連載内容が遊べるキットが子供の科学のオンラインショップ「KoKaShop」で好評販売中!

KoKaスタディーノプログラミングセット 1万800円

KoKaスタディーノ追加セット 3880円



このキット1つで、2016年1月号~8月号で紹介する連載内容が遊べるよ。



「KoKaスタディーノ基本セット」を購入した人向けの追加キット。追加購入することで、さらに8月号の連載内容まで遊べるように。

子供の科学の通販サイト「KoKa Shop」 shop.kodomonokagaku.com

スタディーノを置いて乱数で針を動かせばルーレットになるし、「勉強時間」や「お手伝い」、「ゲームの時間」などと書いて、それを針で示すなど、いろいろ応用してみよう。



今回つくった3分タイマーの裏側に数字を書いた画用紙を立て、残り何分かがわかるようにしたよ。

応用編 普通の電子部品でつくろう

連載第2回の応用編で紹介したLED、3回で紹介したマイクロサーボ、4回で紹介した赤外線フォトリフレクタの配線をすべて使って、このタイマーを作つてみよう!

用意するもの

- スタディーノ基板
- mini USBケーブル
- フレッドボード
- 電池ボックス&単三形アルカリ乾電池
- マイクロサーボSG-90
- LED 赤
- 赤外線フォトリフレクタ [RPR-220]
- 抵抗100Ω, 220Ω, 15kΩ
- ジャンパー線 オス-オス 2本 オス-メス 5本

