

スモウルビープログラム集



テルミン

はじめに

テルミンは、1919年にロシアの発明家レフ・セルゲーエヴィチ・テルミンが発明した世界初の電子楽器です。 スモウルビーとセンサーを組み合わせて、テルミン嵐の電子楽器を作ってみましょう。

センサーを試してみる

まずはスモウルビーとセンサーを組み合わせたプログラムを作って、どのように動くのか見てみよう。



ここまでできたら **『**実行 してみよう♪ センサーの情報に合わせて**歯**面の中の猫が左右に動くはずです。

もしかすると「ハードウェアの準備に失敗しました」と表示されるかもしれません。そのときは、センサーが正しくつながっていないかもしれませんので、プログラムを終っていないかもしれませんので、プログラムを終っていないからセンサーをつなぎなおしてください。

めいれい **命令ブロック**

センサー A0

<調べる>

「センサー [Ã0] 」ブロック

これはセンサーの情報を表現するブロックです。センサーをつないだところに合わせて着 $0 \sim 7.5$ を選びます。センサーの情報は $0 \sim 1.0.2.4$ です。センサーによっては、一番がさい情報が0よりも大きかったり、一番大きい情報が1.0.2.4 よりも小さかったりします。センサーの特徴に合わせてプログラムを作ります。



プログラム(1)

それではテルミン嵐の電子楽器を作ります。 一度「リセット」してプログラムを消してから、好きなキャラクターを選び、プログラムを作ります。

プログラム①が終わったら、そのまま続けてプログラム②を作ってください。間違ってプログラム①を消さないでくださいね♪

ハードウェアを準備する

```
ま行ポタンがクリックされたとき
ずっと
もし センサー AO ≦ 200 ならば
((ド) か と言う
(ピアノのドマ の音を鳴らす
を繰り返す
```

命令ブロック

センサー A0マ ≤ (200)

く 条 件 >

「「値 1] ≦ 「値 2] 」ブロック

これは2つの。値を比べることを表現するブロックです。「もし[条準]ならば」ブロックと組み合わせて、「値 1]が「値 2] 以下のときにプログラムを実行する、といったことを装現します。 どの組み合わせでは「センサー Λ 0 の情報が200 以上だったとき」を装現しています。

^{ホな} 同じようなブロックには、

- 「 $\begin{bmatrix} \overset{\delta t \cdot \cdot}{\mathbf{d}} & \mathbf{1} \end{bmatrix}$ < $\begin{bmatrix} \overset{\delta t \cdot \cdot}{\mathbf{d}} & \mathbf{2} \end{bmatrix}$ 」**ブロック** = $\begin{bmatrix} \overset{\delta t \cdot \cdot}{\mathbf{d}} & 1 \end{bmatrix}$ より $\begin{bmatrix} \overset{\delta t \cdot \cdot}{\mathbf{d}} & 2 \end{bmatrix}$ が小さい
- 「「**値 1**] =「**値 2**] | **ブロック**=「値 1] より「値 2] と同じ
- 「[**値1**]>[**値2**]」**ブロック**=[値1]より[値2]が大きい
- \bullet 「「 $\overset{bc}{\mathbf{6}}$ 1] \geq 「 $\overset{bc}{\mathbf{6}}$ 2] 」 ブロック = 「 $\overset{bc}{\mathbf{6}}$ 1] が「 $\overset{bc}{\mathbf{6}}$ 2] 以上

があります。



プログラム② (プログラム①の続き)

プログラム①を修正してセンサーの情報によって「レ」や「ミ」のいも鳴らせるようにしてみよう!できた人は (***) してみよう♪

ハードウェアを準備する



プログラム③ (プログラム②の続き)

プログラム②を修正して、センサーの情報が変わらないときに次の音を鳴らさないようにしてみよう。こうすると、より電子楽器らしくなるね♪

できた人は ►実行 してみよう♪

ハードウェアを準備する

```
実行ポタンがクリックされたとき
  もし センサー A0 Z ≤ 200
    ( ( 下) " と言う
     ピアノのドマ の音を鳴らす
      【センサー AO → ≤ 【200】
                                                ならば
      「センサー AO > (200)
                             【センサー AO ▼ ≤ 【300】
    ((レッと言う
     ピアノのレマの音を鳴らす
        (センサー AO - > 200)
                             【 センサー A0 マ ≤ 【 300】
   もし 【センサー AO > 300】
    ( 1 ) と言う
     ピアノのミラ の音を鳴らす
                         ではない
       センサー A0 - > 300
 を繰り返す
```



チャレンジ

ここまでできた人は、もっと参くの音を鳴らせるようにしてみよう♪

センサーの情報は0~1024です。センサーによっては、一番小さい情報が0よりも大きかったり、一番大きい情報が1024よりも小さかったりします。センサーの特徴に合わせて、プログラムを修定してください。

下の例では、ファの音も鳴らせるようにしています。ブロックを追加するだけではなく、一番最後のブロックも修正する必要があります。

```
( " レ " と言う
   ピアノのレマ の音を鳴らす
                                                     まで待つ
                                               ではない
       センサー AO - > 200
                        かつ センサー A0 ≥ ≤ (300)
                                               ならば
                        かつ 【センサー A0 ▼ ≦ 【400】
      センサー A0 > 300
  ( ( 2 % と言う
   ピアノのミ の音を鳴らす
                                               ではない
       センサー A0マ > (300) かつ (センサー A0マ ≤ (400)
    (センサー AO > ( 400 ) ならば
  ( マア ナ と言う
   ピアノのファマーの音を鳴らす
      センサー AO > ( 400 ) ではない
                            まで待つ
を繰り返す
```





ベストスモウルビープログラム **集**

だいひょうしゃ たかおこう じ代表者 = 高尾宏治

連絡先 = contact@smalruby.jp

公式サイト = http://smalruby.jp

この文書は、これからプログラミングを学びたい人であれば誰でも使ってもらえるように、この文書をコピーしたり、他の人に渡したりすることができます。それだけではなくて、この文書を参考にして別の文書を作り、それを販売することもできます。



©Ruby プログラミング少年団, 2014 この作品はクリエイティブ・コモンズ 表示 4.0 国際ライセンスの下に提供されています。 ライセンスの詳細は http://creativecommons.org/licenses/by/4.0/deed.ja をご確認ください。

