プロッチを動かしてみよう

プロッチを接続モードで動かしてみよう

ここでは、ドリトルからプロッチを独立モードで動かす方法について解説する。独立モードでは、ドリトルで作成した制御プログラムをプロッチに転送し、プロッチ単体での動作が可能である。

ドリトルからプログラムを転送する

ドリトルで記述したプログラムは次の手順で転送する。

- ドリトルの編集画面でプログラムを記述する。
- プログラムを実行すると□protchへの転送を実行しますか?」と表示されるので「はい(Y)□を選択する。[1)]

- 「ポートを選択してください。」と表示されるのでプロッチを接続してるCOMポートを選択する。[2)]

- プログラムの転送が行われ 「転送完了」が表示されるまで待つ。[3)]

動作確認と利用方法について

ここでは、動作環境が正しく準備できたかの確認と利用方法について説明する。動作確認には、次のプログラムを使用する。

```
システム!"protch"使う。
最初に実行＝「
  プロッチ！テスト動作。
□□
プロッチ!転送。
```

上記のプログラムを転送し、下図のようにプロッチのスイッチを一度押すことで転送したプログラムが実行される。左ライトが一回点灯すればOKである。

プロッチのプログラムの基本

　ドリトルからプロッチを制御するプログラムは、次の形で記述する。先頭の行では、プロッチを使うプログラムを作成することを示している。以後、プロッチ本体を表す「プロッチ」[4]というオブジェクトを使うことができるようになる。

```
システム!"protch"使う。
最初に実行＝「
□□□□□□
□□
繰り返し実行＝「
□□□□□□
□□
プロッチ!転送。
```

最初に実行の「…」の部分には、一度だけ実行したいプログラムを書く。

繰り返し実行の「…」の部分には、何度も繰り返して実行したいプログラムを書く。

転送を実行すると、プログラムがコンパイルされ、プロッチに転送される。

LEDの利用

　プロッチに搭載しているLEDは点灯や消灯を使うことで制御できる。プロッチにはLEDが２つ搭載しているため、左側のLEDである「左ライト」、右側のLEDである「右ライト」というオブジェクトがある。

LEDの命令一覧

命令	機能
点灯	LEDを点灯する
消灯	LEDを消灯する

次のプログラムでは、左右のLEDを1秒間点灯し、消灯するプログラムである。

```
システム!"protch"使う。

最初に実行=「
  左ライト!点灯。
  右ライト!点灯。
  プロッチ!1 待つ。
  左ライト!消灯。
  右ライト!消灯。
□□
プロッチ!転送。
```

スイッチの利用

プロッチに搭載しているスイッチは、接触?を使うことでスイッチのONとOFFを検出できる。スイッチは左右に搭載しているので左側のスイッチである「左スイッチ」、右側のスイッチである「右スイッチ」というオブジェクトがある。

スイッチの命令一覧

命令	機能
読む	スイッチの入力を0か1で取得する
接触?	スイッチの入力を0か1で取得する

次のプログラムでは、左スイッチが押されている時に左ライトが点灯し、押されていないときは消灯する。プログラムは繰り返し実行に書いているため、何度の実行される。

```
システム□ "protch" 使う。
繰り返し実行＝「
    「左スイッチ！ 接触?」！なら「
        左ライト！ 点灯。
    」そうでなければ「
        左ライト！ 消灯。
    」実行。
□□
プロッチ!転送。
```

下図は、プロッチが箱に衝突し、左スイッチに接触しているので左ライトが点灯している。

光センサの利用

　プロッチに搭載している光センサは明るさ?を使うことで周辺の光量を取得できる。光センサは左右に搭載しているので左側の光センサである「左光センサ」、右側の光センサである「右光センサ」というオブジェクトがある。

光センサの命令一覧

命令	機能
読む	光量を0～255の範囲で取得する。暗い程、取得する値は大きくなる
明るさ?	光量を0-255の範囲で取得する。暗い程、取得する値は大きくなる

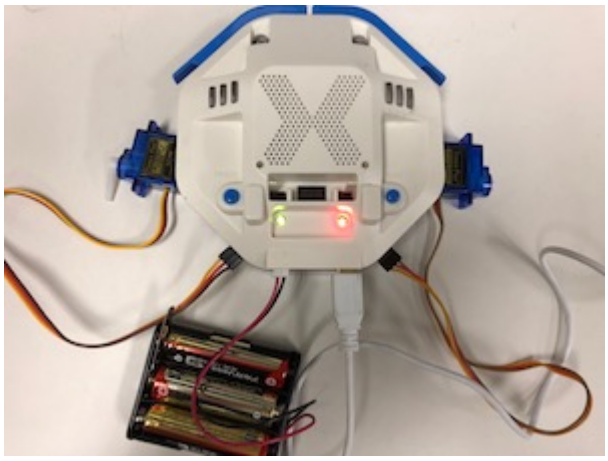
　次のプログラムでは、周辺の光量が100より大きい時に左のライトが点灯する。プログラムは繰り返し実行に書いているため、何度の実行される。

```
システム□ "protch" 使う。
繰り返し実行=「
    「(左光センサ！ 明るさ?)>100」！ なら「
        左ライト！ 点灯。
    」そうでなければ「
        左ライト！ 消灯。
    」実行。
□□
プロッチ!転送。
```

　下図は、左光センサを手で隠すと暗くなるのでLEDが点灯している。

サーボモータの利用

プロッチにはサーボモータを接続するための端子がある。左右に接続でき、左側の端子である「左サーボモータ」、右側の端子である「右サーボモータ」というオブジェクトがある。サーボモータは角度設定を使うことで角度を指定することができる。サーボモータの接続方法は下図のとおりである。サーボモータを利用するときは、外部電源に電池ボックスを接続する必要がある。



□ サーボモータの命令一覧

命令	機能
角度設定	0度~180度の間で角度を指定する

次のプログラムでは、左右のサーボモータの角度を1秒ごとに180度と0度に動作する。

```
システム!"protch"使う。
最初に実行=「
  左サーボモータ！180　角度設定。
  右サーボモータ！180　角度設定。
  プロ!1　待つ。
  左サーボモータ！0　角度設定。
  右サーボモータ！0　角度設定。
□□
プロッチ!転送。
```

モータの利用

プロッチには左右にタイヤにつながったモータが搭載している。次のプログラムでは、1秒ごとに前進、左折、右折、後進、停止をしている。

モータの命令一覧

命令	機能
前進	左右のモータを正回転する
後進	左右のモータを逆回転する
停止	左右のモータを停止する
左折	右側のモータを正回転する
右折	左側のモータを正回転する
左回り	右側のモータが標準の速度で正回転し、左側のDCモータが標準の速度/5の速度で正回転する
右回り	左側のモータが標準の速度で正回転し、右側のDCモータが標準の速度/5の速度で正回転する

命令	機能
速度設定	DC モータの速度を0 から255 までの値で設定する。標準では100になっている

```
システム "protch" 使う。
最初に実行＝「
    モータ！ 1 前進。
    モータ！ 1 左折。
    モータ！ 1 右折。
    モータ！ 1 後進。
    モータ！ 1 停止。
」
プロッチ!転送。
```

ラインセンサを使ったライントレース

ライントレースカーは左右のDCモーターを使い、地面に描かれた線のコースに沿って動く。プロッチに搭載しているラインセンサを利用することで赤外線の反射量の値から線の上にいるかどうかを判断できる。ラインセンサは**読む**を使用することで、0〜255の値で検出できる。ラインセンサは左右に一つずつ搭載しており、左側のセンサの「左ラインセンサ」、右側のセンサの「右ラインセンサ」というオブジェクトがある。

ラインセンサの命令一覧

命令	機能
読む	赤外線の反射量を0〜255の範囲で取得する
明るさ?	赤外線の反射量を0〜255の範囲で取得する

次のプログラムでは、左右のラインセンサを比較し、線の上にいなければ右折し、線の上にいれば左折する。

```
システム!"protch" 使う。

繰り返し実行＝「
    「(左ラインセンサ!読む)<(右ラインセンサ!読む)」!なら「
        モータ！左回り。
  」そうでなければ「
        モータ！右回り。
  」実行。
」
プロッチ!転送。
```

距離センサの利用

プロッチでは距離センサをつけることができる。距離センサは**距離**?を使うことで、プロッチの正面にあるモノまでの距離をmmで取得できる。次のプログラムでは、障害物が近くにあると停止する。

距離センサの命令一覧

命令	機能
読む	距離センサの計測値をmmで取得する

命令	機能
距離?	距離センサの計測値をmmで取得する

```
システム!"protch"使う。
繰り返し実行=「
    「(距離センサ!距離?)<40」！なら「
        モータ！1 停止。
    」そうでなければ「
        モータ!前進。
    」実行。
□□
プロッチ!転送。
```

音楽演奏

プロッチに搭載しているブザーを利用して音楽の演奏が可能である。[5]次のプログラムでは、ブザーを使ってドレミの音階を鳴らしている。

ブザーの命令一覧

命令	機能
演奏	ブザーの出力を音階に合わせて順番に奏でる

```
システム□ "protch" 使う。
最初に実行＝「
  ブザー！"ドレミ" 演奏。
□□
プロ！ 転送。
```

また、ブザーを用いてメロディの演奏が可能である。[6]ここで は、譜面に相当するメロディオブジェクトに旋律を書き込み、そ れを演奏する。次のプログラムでは、「きらきらぼし」という名 前のメロディオブジェクトを生成し、「“…“」の形で旋律を追加し てから、ブザーに演奏させている。

メロディの命令一覧

命令	機能
作る	配列を作る
追加	作った配列に音階を挿入する

```
システム!"protch"使う。
最初に実行＝「
    きらきらぼし＝メロディ！ 作る。
    きらきらぼし！"ドドソソララソ～ファファミミレレド～" 追加。
    きらきらぼし！"ソソファファミミレ～ソソファファミミレ～" 追加。
    きらきらぼし！"ドドソソララソ～ファファミミレレド～" 追加。
    ブザー！（きらきらぼし） 演奏。
□□
プロ！ 転送。
```

1)
□Windowsセキュリティの重要な警告」が表示された場合は「アクセスを許可する(A)□を選択する。
2)

COM□の部分は環境によって異なる。
3)
5秒程度かかる。
4)
□protch□□プロ」も使用できる。
5)
現在、音楽演奏の機能は独立モードでのみ利用可能
6)
現在、「音符」「オクターブ」の対応を進めている

From:
https://dolittle.eplang.jp/ - プログラミング言語「ドリトル」

Permanent link:
https://dolittle.eplang.jp/ch_protch?rev=1539432477

Last update: **2018/10/13 21:07**