

チャレンジ 10: 多機能車



1、チャレンジ目標

- 1) **スクリプト変数 a** をマスターし、スクリプト変数と変数の違いを理解する。
- 2) **IRトランシーバー 受信コード** と **IRリモートキー コード** をマスターする。
- 3) 多機能車をプログラムで実現。リモートコントロール、ナローライトレース、ワイドライトレースと障害物回避を一つのプログラムにまとめる。

2、ガイドライン

今までのチャレンジでは、車の障害物回避、ライトレース、リモートコントロールなどの機能を実現してきましたが、今回のチャレンジでは、それらの機能を一つにまとめるプログラムを実現していきます。

多機能車は下記の機能を備えています。

*スイッチオンで RGB が 3 回点滅します。

*リモコンの1のキーが押された時、車がナローライントレースモードに入ります。

*リモコンの2のキーが押された時、車がワイドライントレースモードに入ります。

*リモコンの3のキーが押された時、車が障害物回避モードに入ります。

*リモコンの→←↑↓のどちらかのキーが押された時、車がリモートコントロールモードに入ります。矢印キーで車の動きをコントロールします。

3、組み立て

<http://www.teccubic.com/>の組立例をご参照ください。



4、プログラミング

1) スクリプト変数について

前のチャレンジでは、変数について学びました。変数はプログラミングでよく使わ

れるもので、データの入れ物のような存在です。

[変数]の カテゴリから、**スクリプト変数 a ▶** を見つけることができます。「スクリプト変数」と普通の変数の違いは、「スクリプト変数」はその「スクリプトエリア」にのみ存在することです。その利点は、複数のプログラムブロック(関数)に同じ名前の変数が存在しても、お互いに干渉しないことです。変数名を複数使うことなく、同じ変数名でプログラムを組むことができます。

こういう変数は「ローカル変数」と言います。

つまり、ScratchPi での「スクリプト変数」は Arduino での「ローカル変数」と同じ意味のものです。

2) IR トランシーバを効率的に使用するには

コントロールブロック(マイコン)が IR トランシーバの信号を読み取るたびに、処理時間がかかります。

この多機能車でもプログラムの一サイクルでリモコンの信号を読み取り、「→←↑↓123」の中のどのキーが押されたかを判断しなければなりません。

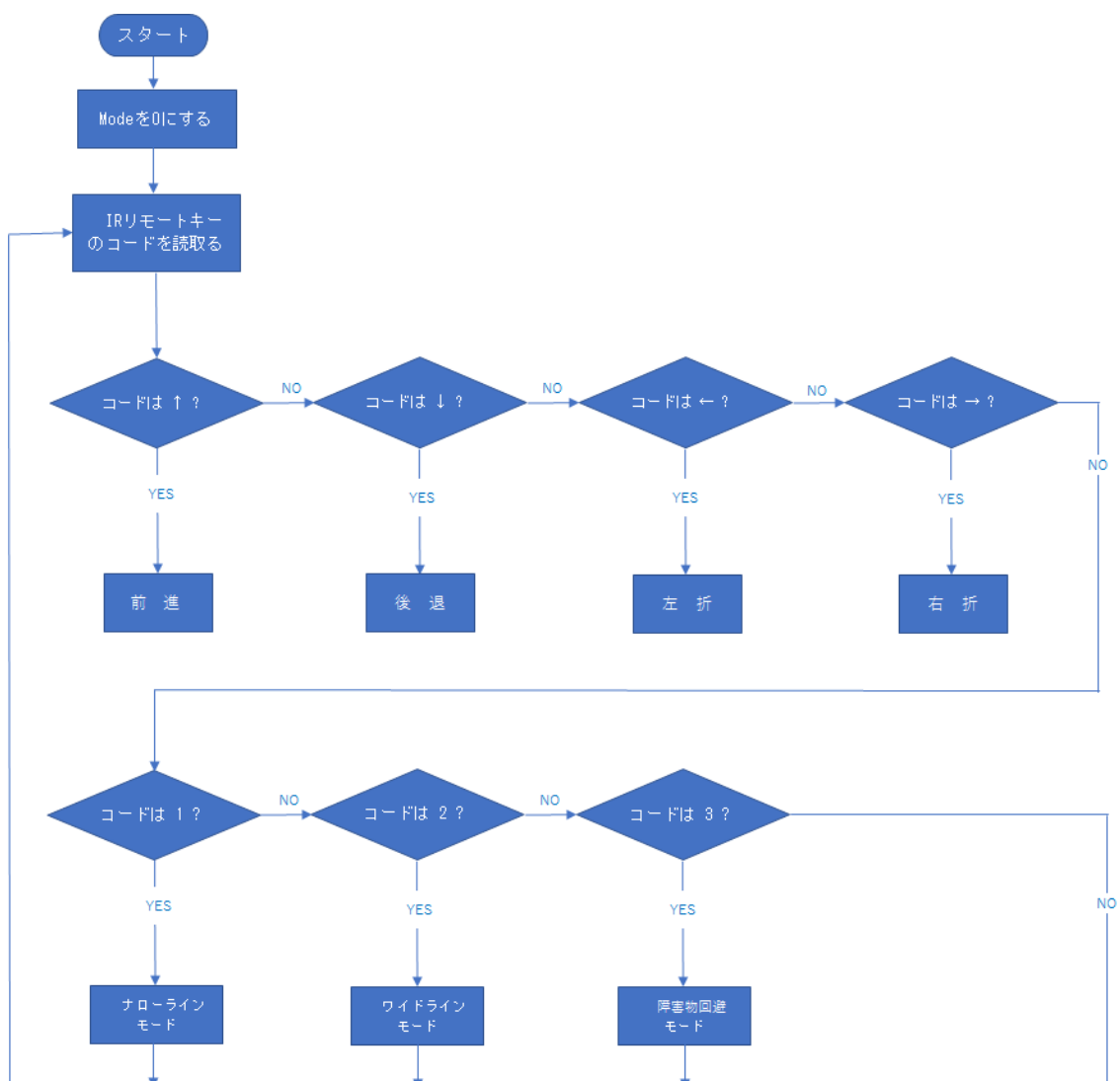
IRリモート ▼ ▼ が押されましたか? を使用する場合、プログラムの一サイクルで7回も IR トランシーバの信号を読みに行かなければならないので、車の反応が遅くなります。

特に、ライントレースの場合、ロジカルの処理に時間がかかりますので、反応速度が遅いと、車がラインを外してしまう可能性があります。

この問題を解決するため、**IRリモートキー** ▼ **コード** を使用して、“変数” と

IRリモートキー ▼ **コード** を組み合わせれば、読み取りが1回で済むようになるので、反応速度が早くなります。

3) プログラムフローチャート



4) サンプルプログラム

```
Arduinoセットアップ
mode を 0 にする
Arduinoメインループ
スクリプト変数 a
a を IRトランシーバー リモート2 受信コード にする
もし a = IRリモートキー 上 コード なら
mode を 0 にする
forward
でなければ
もし a = IRリモートキー 下 コード なら
mode を 0 にする
backward
でなければ
もし a = IRリモートキー 左 コード なら
mode を 0 にする
left
でなければ
もし a = IRリモートキー 右 コード なら
mode を 0 にする
right
でなければ
もし a = IRリモートキー 1 コード なら
mode を 1 にする
でなければ
もし a = IRリモートキー 2 コード なら
mode を 2 にする
でなければ
もし a = IRリモートキー 3 コード なら
mode を 3 にする
でなければ
もし mode = 0 なら
stop
irTrackingNarrow
irTrackingWide
ultrasonic
```

5) プログラムの実行

プログラムをコントローラブロックにアップロードして、効果を試してください。

5、 応用

パトカーの警光灯も光らせてみてください(RGB LED)。