## 最終章２　うまく動かない人へのヒント

これはどうしてもうまくいかない人へのアドバイスです。ここに記載以外でもいろいろな方法がありますので、できている人は自分なりの方法で更に進化させてください。

**１　直線コース（カーブのあるコースでは適用されません）**

直線コースで速く走らせるためには、オーバーランさせないことが大切です。その場合のセンサ配置は３つのセンサが横並び（中央センサが少し後ろ）のパターンがわかりやすく誰でもが高速に走らせることができると思います。

１　センサ配置  
このセンサ配置は左図のように、左右のセンサの間隔Cをライン幅より少し大きく、センサを３つ横並びにするときよりは少し小さくします。

また、センサの進行方向の位置Yですが、車軸より少し遠めがよいでしょう。

２　センサ感度  
　これはとても重要です。センサがうまく反応しなければコンピュータにデータが届かず、正確な指令ができないからです。  
　もう一度感度チェックをしてください。  
　センサがうまく反応しない場合は手を挙げて先生を呼んでください。

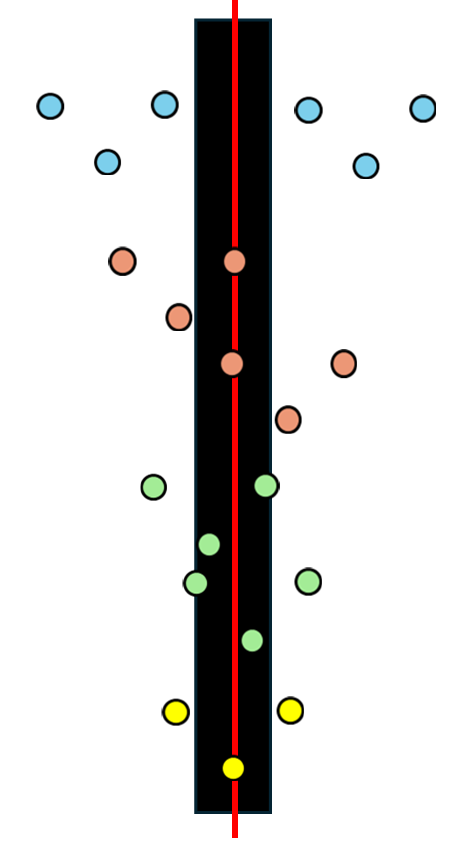
３　プログラム  
最初はスピードは遅くして、脱線せずに走ることを確かめ、徐々にスピードを上げていってください。  
センサの配置は緑色の状態、つまり、左右のセンサが黒いラインに入っても中央のセンサはまだ黒いライン上にあるように配置してください。  
  
黄色の段階：中央のセンサがライン内、左右のセンサがライン外  
　　　　　　このときはスピードMAX  
　　　　　　まっすぐ走るように左右のデューティ比を調整する。  
緑色の段階：少しだけラインからずれているだけなので、急な曲がりをせ　　  
　　　　　　ず、直線に近い緩やかな曲がりを指示します。  
赤色の段階：２つのセンサがラインからはずれているので、緑の段階で指  
　　　　　　示したデューティ比より少しだけ急な曲がりにします。

※まっすぐ走る、左右大きなカーブで走るなど、モータドライバテストプログラムでちょうどよいデューティ比を確かめてください。

４　プログラム例（●：高速、◎：中速、〇：低速）

色分けしてある部分は、全ページのそれぞれの色の段階と一致しています。











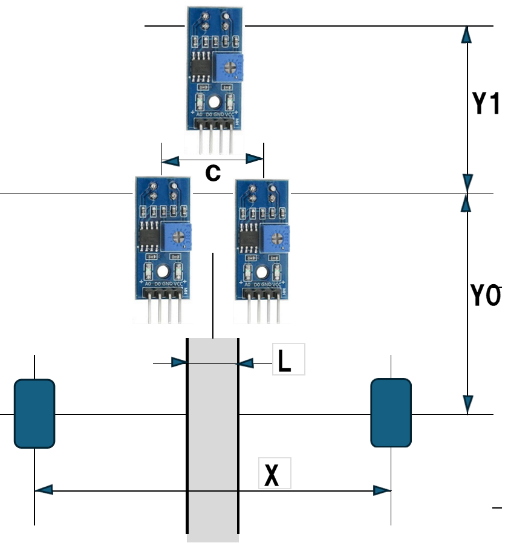


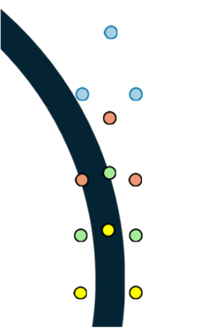






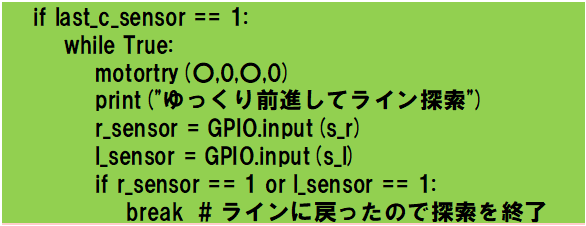
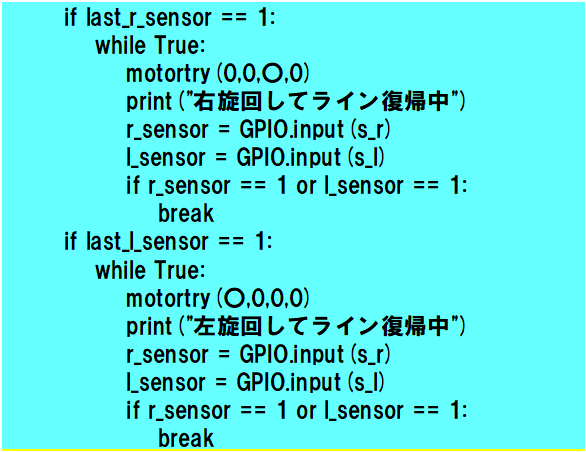
**２　直線とカーブが混在するコース**

直線とカーブが混在するコースでは、スピードを落として、カーブのデューティ比さえ問題なければラインに沿って動くはずです。しかし、ここでは、できるだけ速く動かしたいのですから、オーバーランすることは当然としてプログラムを組む必要があります。  
なぜなら、ルンバのような家電製品や工場で使用される自動掃除機などで使用するモータはステッピングモータやサーボモータ（ほとんど惰性（だせい）で動くことはなく、とまれと指示すればその場ですぐに止まるモータ）を使用していますが、私たちが使用しているモータはラジコンのおもちゃなどで使用しているDCモータだからです。  
だから、直線とカーブが混在するコースでは、スピードが遅ければ問題ないのですが、速く走ろうとする場合はオーバーランは当然起こるものとしてプログラムを組む必要があります。そのためここでは中央センサを前にするセンサ配置で進めることします。  
１　センサ配置  
無難に進めるとすれば、この配置が良いでしょう。  
Cの間隔は直線コースのときと同等か、または少しだけ広めにします。  
Y０は直線コースのときと変えなくてもよいでしょう。Y１は４cm前後かなと思います。  
これらの値は何回か繰り返す中で、最適なセンサ配置を決定してください。  
  
  
  
  
  
２　センサ感度  
この場合もセンサ感度は重要です。特に中央のセンサを前に取り付けることにより、少し下がり気味にあるようなら、なおさら注意する必要があります。  
また、振動でのがたつきが生じないように、しっかりと固定してください。

３　プログラム  
センサ配置は左図のように、半径２５cmのラインに対して、緑色の状態、つまり、左右センサがラインを踏むよりも先に中央センサがラインから出てしまう状態（緑色になる状態）になるように３つのセンサを配置してください。

黄色の状態：中央センサはライン上、左右センサはライン外  
　　　　　　この時はＭＡＸ前進  
緑色の状態：中央センサがライン外に出る、左右センサはライン外  
　　　　　　この時はスピードダウンしてゆっくり前進  
赤色の状態：左右のセンサのうち、どちらかのセンサが黒いライン  
　　　　　　内に入った状態  
　　　　　　　　　　　　　　　　この時は左右どちらかに曲がる指示をだすが、その  
　　　　　　　　　　　　　　　　時の曲がる指示は半径２５cmで曲がるよりは少しだ  
　　　　　　　　　　　　　　　　け小さな半径で曲がるようにデューティ比を調整して  
　　　　　　　　　　　　　　　　指示する。  
青色の状態：３つのセンサすべてがライン外にある場合  
　　　　　　スピードが大きいと赤色の状態で曲がるように指示しても、惰性があるので  
　　　　　　すぐに青色の状態になってしまう。  
　　　　　　この時は、直前の状態を確認して、直前にラインの中にあった方向に曲がる 　　　　　  
　　　　　　指示をすればよい。  
　　　　　　このときの曲がる半径は最小半径、つまり、片輪を止めて曲がる指示をした  
　　　　　　方がよい。そうしないとラインに復帰しなくなる場合も生じてしまう。

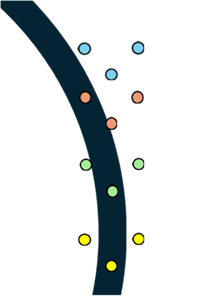
４　プログラム例（●：高速、◎：中速、〇：低速）

左のプログラムの色は前頁のセンサの色の状態でのプログラムを表している。  
緑色の部分と青色の部分については、一般的には右にあるような書き方をしますが、  
while文を使うことで、無駄なく動くことがかえって惰性力を生み、逆の方向にオーバーランしてしまうことが繰り返されてしまう。そのため、while文は使わない。

黄色のデューティ比：MAX前進なので、まっすぐ走る最大スピードのデューティ比を与え  
　　　　　　　　　　る。  
緑色のデューティ比：中央センサがカーブラインがあることを確認したので、最小のスピ  
　　　　　　　　　　ードでまっすぐ進むデューティ比を与える。  
赤色のデューティ比：中央センサがライン上にある時は直線コースの中にあると考え、直  
　　　　　　　　　　線用カーブで、それ以外はスピードダウンし、半径２５cmより少  
　　　　　　　　　　し小さな半径を描くようなデューティ比を与える。  
青色のデューティ比：オーバーランした状態なので、ライン復帰するために最小半径を描  
　　　　　　　　　　く（一方の車輪のデューティ比を０にする）ようなデューティ比を  
　　　　　　　　　　与える。

５　直線用センサ配置でのプログラム例（●：高速、◎：中速、〇：低速）

プログラムの色は上記センサの状態での色と一致させている。

このセンサ配置は直線コース向きであるが、オーバーランを前提として考えるのであるならば、直線＋カーブ用としても利用できる。  
調整をうまく行えば、中央センサを前に配置するセンサ配置よりスピードアップができるかもしれない。  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
黄色のデューティ比：MAX前進なの  
　　で、まっすぐ走る最大スピードの  
　　デューティ比を与える。  
緑色のデューティ比：左右のセンサが  
　　少しはずれただけなので、直線コ  
　　ース上にあると判断し、直線用カ  
　　ーブで走らせる。  
赤色のデューティ比：２つのセンサが  
　　ラインをはずれているので、半径  
　　２５cmより少し小さな半径（緑  
　　色のときより更に小さ目な円弧を  
　　描くデューティ比を与える。  
青色のデューティ比：オーバーランし  
　　た状態なので、ライン復帰するた  
　　めに最小半径を描く（一方の車輪  
　　のデューティ比を０にする）よう  
　　なデューティ比を与える。

**３　デューティ比を決めよう**

直線コースの考え方やカーブコースの考え方が理解できたら、デューティ比チェックプログラムを用いて、自分のマシンに合ったデューティ比を決めよう！

|  |  |
| --- | --- |
| try:  while True:  moji = input()  if moji == ("a"):  motortry(100,0,100,0)  print("テスト１")  elif moji == ("b"):  motortry(100,0,90,0)  print("テスト２")  elif moji == ("c"):  motortry(90,0,100,0)  print("テスト３")  elif moji == ("d"):  motortry(50,0,50,0)  print("テスト４")  elif moji == ("e"):  motortry(50,0,40,0)  print("テスト５")  elif moji == ("f"):  motortry(40,0,50,0)  print("テスト６")  elif moji == (""):  motortry(0,0,0,0)  print("停止")  sleep(0.01) except Exception as err: ・・・・ | 左のデューティ比チェックプログラムにはデューティ比が仮で入れてありますので、下記のデューティ比が得られるよう、テスト数値を６種類くらい入れて、よさそうな値を決めてください。  確認するデューティ比 １　まっすぐ走る （１）ＭＡＸスピード 　　motortry（　　　，０，　　　，０） （２）微速前進 　　motortry（　　　，０，　　　，０） ２　右にカーブする  （１）半径１ｍ以上(少しだけ右に曲がる) 　　motortry（　　　，０，　　　，０） （２）（１）よりもう少しだけ曲がる 　　motortry（　　　，０，　　　，０） （３）半径２５cmより少し小さく曲がる 　　motortry（　　　，０，　　　，０） （４）（３）よりもう少しだけ曲がる 　　motortry（　　　，０，　　　，０） （５）最少半径で低速で曲がる 　　motortry（０，０，　　　，０） ３　左にカーブする  （１）半径１ｍ以上(少しだけ左に曲がる) 　　motortry（　　　，０，　　　，０） （２）（１）よりもう少しだけ曲がる 　　motortry（　　　，０，　　　，０） （３）半径２５cmより少し小さく曲がる 　　motortry（　　　，０，　　　，０） （４）（３）よりもう少しだけ曲がる 　　motortry（　　　，０，　　　，０） （５）最少半径で低速で曲がる 　　motortry（　　　，０，０，０） |

この表が完成したら、このデューティ比をプログラムに入れれば間違いなくライントレースすると思います。  
もし、これでうまくいかない場合はフォトセンサの位置の再確認とフォトセンサの感度を再調整してください。