

トレーニングトレーサー

Training Tracer パート0 基礎知識編

Ver. 1.1
2020/06/11
RT Corporation

Contents:

| | |
|---------------------------------|-----------|
| 1. 結論 | 3 |
| 2. センサ | 3 |
| 2.1 センサの種類 | 3 |
| 2.1.1 プッシュスイッチ | 4 |
| 2.1.2 赤外線・可視光センサ | 4 |
| 2.1.2.1 発光素子と受光素子 | 4 |
| 2.1.3 エンコーダ | 5 |
| 2.1.3.1 光学式エンコーダ | 5 |
| 2.1.3.2 磁気式エンコーダ | 5 |
| 2.1.4 ジャイロセンサ | 6 |
| 3. アクチュエータ | 6 |
| 3.1 DCモータ | 6 |
| 3.2 ステッピングモータ | 8 |
| 4. バッテリ | 10 |
| 4.1 ニッケル水素電池 | 10 |
| 4.2 リチウムイオンポリマ電池 | 10 |
| 5. 電子部品 | 11 |
| 5.1 オームの法則 | 11 |
| 5.2 アナログとデジタル | 11 |
| 5.2.1 アナログ | 11 |
| 5.2.2 デジタル | 11 |
| 5.3 電子部品 | 12 |
| 5.3.1 抵抗 | 12 |
| 5.3.1.1 半固定抵抗 | 14 |
| 5.3.2 コンデンサ | 14 |
| 5.3.2.1 電解コンデンサ | 14 |
| 5.3.2.2 セラミックコンデンサ | 15 |
| 5.3.3 スイッチ | 16 |
| 5.3.4 LED(Light Emitting Diode) | 16 |
| 5.3.5 電子ブザー | 17 |
| 5.3.6 トランジスタ | 17 |
| 5.3.7 フォトトランジスタ | 18 |

| | |
|----------------------|-----------|
| 5.3.8 三端子レギュレータ | 18 |
| 6.工具 | 18 |
| 6.1 半田ごて | 18 |
| 6.2 こて台&クリーナー | 19 |
| 6.3 半田吸い取り線(半田吸い取り機) | 19 |
| 6.4 ドライバー | 20 |
| 6.5 ニッパー | 20 |
| 6.6 ワイヤストリッパー | 21 |
| 6.7 ラジオペンチ | 22 |
| 6.8 ピンセット | 22 |
| 7.ネジ | 22 |
| 7.1 ネジの種類 | 22 |
| 7.2 ドライバーの使い方 | 23 |
| 8.半田付け | 24 |
| 8.1 半田付けの仕方 | 25 |
| 9.テスター | 30 |
| 9.1 テスターの使い方 | 30 |
| 10.マイコン | 32 |
| 11.プログラミング言語 | 32 |
| 11.1 C言語 | 32 |
| 11.1.1 プリプロセッサ | 33 |
| 11.1.2 main関数 | 33 |
| 11.1.3 関数 | 33 |
| 11.1.4 型 | 35 |
| 11.1.5 演算子 | 36 |
| 11.1.6 制御文 | 36 |
| 11.1.6.1 if文 | 36 |
| 11.1.6.2 for文 | 37 |
| 11.1.6.3 while文 | 37 |
| 11.1.6.4 switch文 | 37 |
| 12.著作権について | 38 |
| 13.お問い合わせ | 38 |
| 改版履歴 | 38 |

1. 緒論

ここでは、汎用ロジックICを使った単純なものではなく、家電製品や小型のコンピュータを動かす高機能ICである[マイコン](#)を使用した、マイコン制御型のライントレーサー(英語だとLine Follower)を作る際に必要となる基礎知識について説明します。

ライントレーサーには、[センサ](#)や[アクチュエータ](#)が使われており、それらを動かすためには、それぞれを制御するマイコンやプログラミングの知識が必要になります。また、実際に組み立てるためには、電子部品の知識や工具の使い方の知識も必要になります。

このようにライントレーサーは、ロボットを作る・動かすための多くの要素をコンパクトにまとめたロボットです。ソフトウェア、ハードウェアの両方がバランスよく組み込まれたライントレーサーは、現在活躍する技術者が必要とする技術的要素を含んだ構成となっています。

2. センサ

周囲の環境情報をロボットやPCで扱える形に変換、収集する装置を総じてセンサと呼びます。センサが取得する情報としては、光、温度、色、圧力、磁気、速度、加速度などがあります。例としては、自動ドアのセンサは、光センサの一種である赤外線センサを使用しています。これらの情報からロボットなどは自身の速度や位置、周辺の障害物などを認識しています。ライントレーサーにも走行中に情報を取得するための複数のセンサが使われています。

2.1 センサの種類

ライントレーサーで利用されるセンサの代表的な種類と用途を表1にまとめますので、参考にしてください。

表1 主なセンサー一覧

| センサ | 用途 |
|---|-----------------------------|
| プッシュスイッチ | スタート or ストップ プログラムのモード変更 |
| 赤外線・可視光センサ | ライン読み取り用 |
| エンコーダ (フォトインタラプタ+エンコーダ板) (磁気センサ+磁気円盤) | タイヤ回転数検出 車体速度制御・位置計測 |
| ジャイロセンサ | 車体姿勢制御・方向計測 |

2.1.1 プッシュスイッチ

スイッチも分類としてはセンサに含まれます。基本的な使い方として、スタートやストップ、プログラムモードの切り替えなどに使います。また、ロボット側面に取り付けたりして、壁との衝突を検出するためにも使います。

身近なものだと、パソコンのキーボードのキーやマウスのクリックボタンも同じものが使われていたりします。

2.1.2 赤外線・可視光センサ

光を電気信号に変換する素子を使ったセンサで、周囲の光の強さを計測できます。ライントレーサーでは発光素子と組み合わせて床の光の反射を計測し、走行ラインの読み取りに使います。

2.1.2.1 発光素子と受光素子

発光素子としては[赤外線LED](#)や[可視光LED](#)が、受光素子としてはLEDに対応した波長の[フォトトランジスタ](#)などがよく使われます。センサで受光した値¹をもとに走行ラインの白黒を判別します。複数設置されたセンサを利用して、走行方向の決定や姿勢制御²を行います。

このセンサを設置する方法としては、ロボットの前方に床へ向けて複数のセンサを並べ、ラインの白と黒の反射率の違いによってラインを検出します。

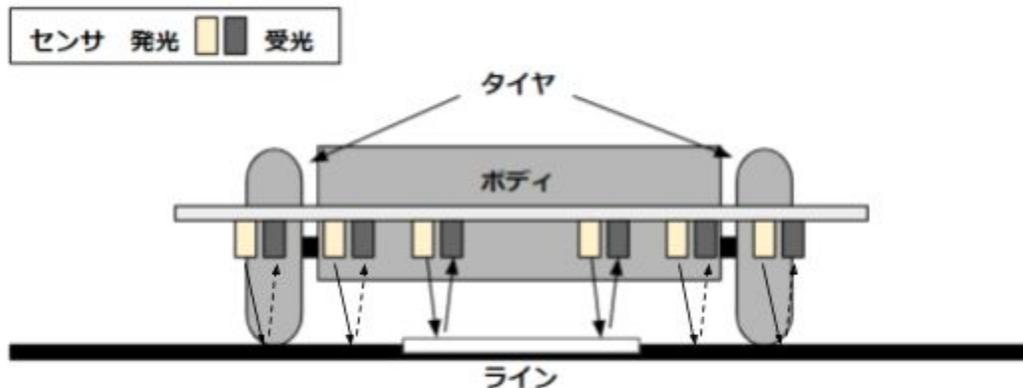


図1 ライントレーサーのセンサ部

¹ センサの種類にもよりますが、一般的にはセンサが出力する電圧

² ロボットがラインから外れないように走行させること

トレーニングトレーサー

Traning Tracer パート4 ソフトウェア解説編

Ver. 1.1

2020/06/11

RT Corporation

Contents:

| | |
|-----------------------------|----|
| 1.ソフトウェアの解説 | 2 |
| 1.1 STEP1 LEDを光らせてみよう | 3 |
| 1.2 STEP2 ブザーを鳴らそう | 9 |
| 1.3 STEP3 スイッチを使おう | 11 |
| 1.4 STEP4 モード選択をしよう | 16 |
| 1.5 STEP5 センサの値を見よう | 23 |
| 1.6 STEP6 モータを回そう | 30 |
| 1.7 STEP7 フィードバック制御をしよう-位置編 | 35 |
| 1.8 STEP8 ゴールで止まって見よう | 42 |
| 1.9 STEP9 上級者への道 | 45 |
| 1.9.1 要求仕様 1 | 45 |
| 1.9.2 要求仕様 2 | 50 |
| 1.9.3 プロジェクト | 55 |
| 2.開発のヒント | 58 |
| 3.ソフトウェアについて | 61 |
| 4.著作権について | 61 |
| 5.お問い合わせ | 61 |
| 改版履歴 | 61 |

1. ソフトウェアの解説

トレーニングトレーサーのパート2 ハードウェア製作編、パート3環境構築編でプログラムの開発環境が整いました。

ここでは、8個のサンプルプログラムの解説を行います。サンプルプログラムの解説を読みながら実際にトレーニングトレーサーを動作させて、解説通りに動くことを確認してみてください。また、各STEPごとに練習問題が用意してあります。現状の実力を見るには良い課題になっています。

STEP8までの練習課題がクリアできるようになると、次は、練習問題ではなく、さらに上を目指せるように要求仕様が与えられ、それをクリアするにどのようにする必要があるのかを紹介します。その中で、STEP8以上の性能を求めるには、Arduino IDE用にライブラリを作る必要があり、上級者クラスでないと厳しいレベルになります。そこで、STEP9として、初級者から中級者向けとして、Arduino IDE環境から上級者でも使用しているSTM32CubeIDEへ環境移行を行っていただき、Arduino IDE環境では開発が難しかった機能を追加するフローになっています。

プログラムの開発にあたり注意点があります。プログラムの入ったフォルダの名前は、なるべく全角(日本語など)を使わないでください。特に括弧"()"とスペース" "をつけると、Arduino IDEでコンパイルエラーになる場合があります。また、ファイル名も同様です。なるべくフォルダ名、ファイル名は、英語とし、スペースの代わりにアンダーバー"_"を使ってください。

サンプルプログラムは、STEPごとに以下の流れで進めていきます。

概要

大まかにそのSTEPで何を行うプログラムなのかを解説します。

プログラムソース

実際のプログラムの中からそのSTEPでの重要な所を掲載します。

動作確認と解説

プログラムを実際に書き込んでみてどのように動作するのかを解説していきます。

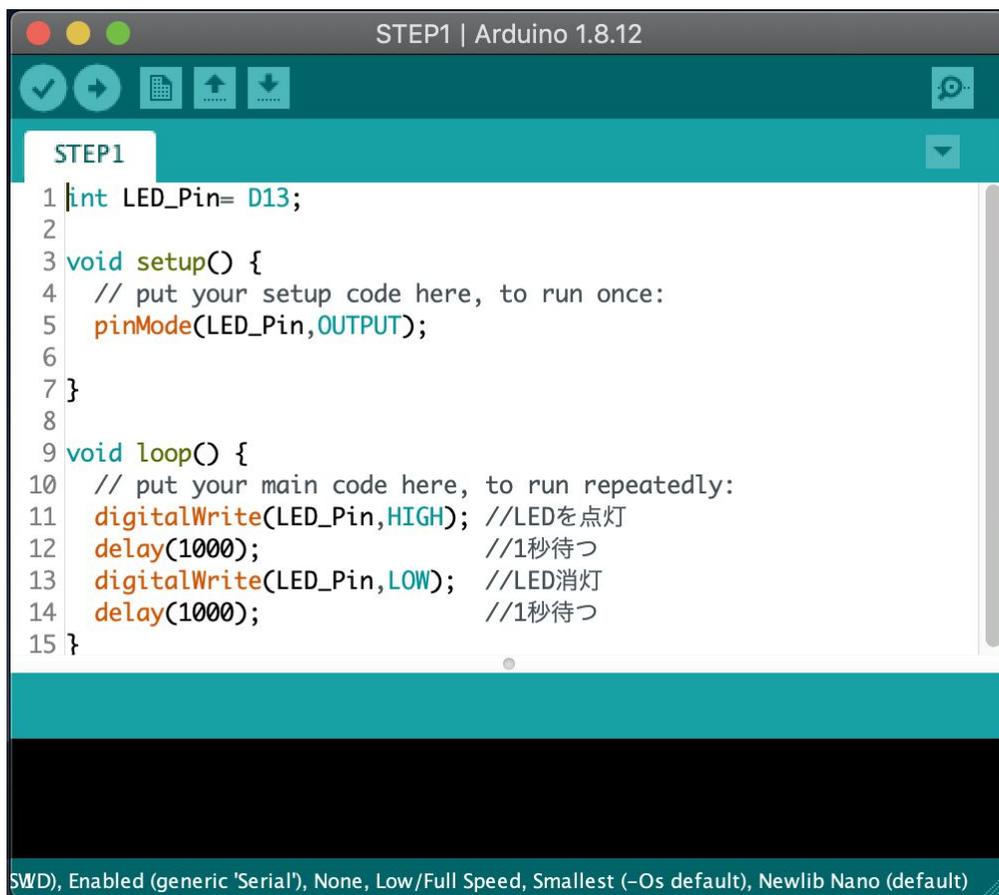
スイッチが扱える容量の関係で、スイッチのOFFだけでは、マイコン、センサ、ブザーの電源の遮断のみ行い、モータドライバーは電池が繋がったままになっています。
使用しないときは、電池を電池ボックスから抜くようにお願いします。

1.1 STEP1 LEDを光らせてみよう

概要

STEP1は、デバッグ用でもあり、モード確認用でもあるLEDを光らせてみましょう。このプログラムは、LEDを点滅させるプログラムになっています。LEDとは、日本語では、発光ダイオードと呼ばれており英語ではLight Emitting Diodeの頭文字をとってLED(エルイーディ)と呼ばれています。LEDは、ダイオードの一種で、順方向に電圧を加えたときに発光する半導体素子です。

プログラムソース



```

STEP1 | Arduino 1.8.12
STEP1
1 int LED_Pin= D13;
2
3 void setup() {
4   // put your setup code here, to run once:
5   pinMode(LED_Pin,OUTPUT);
6
7 }
8
9 void loop() {
10  // put your main code here, to run repeatedly:
11  digitalWrite(LED_Pin,HIGH); //LEDを点灯
12  delay(1000); //1秒待つ
13  digitalWrite(LED_Pin,LOW); //LED消灯
14  delay(1000); //1秒待つ
15 }
(SWD), Enabled (generic 'Serial'), None, Low/Full Speed, Smallest (-Os default), Newlib Nano (default)

```

動作確認と解説

TraningTracer¥STEP1¥STEP1.inoをダブルクリックすると、Arduino IDEが起動し、上記のスケッチ(Arduino IDEにおけるプログラムのこと)が表示されると思います。

上記のスケッチは、行数が表示されています。デバッグのしやすさと解説のしやすさのため行数の表示をしています。この行数を表示させるには、

Windowsバージョン

ファイル -> 環境設定 をクリックすると環境ファイルが現れます。



Macバージョン

Arduino -> Preferences... をクリックすると環境ファイルが現れます。



左の真ん中あたりにある、「行番号を表示する」をクリックすると「L」が追加されます。追加されたら、「OK」をクリックすると行番号が追加されます。

NUCLEO-F303K8は、Arduino Nanoとスケッチ上でのプログラムの互換性を持っており、GPIO(General Purpose Input Output:汎用入出力)のD0から D13とアナログピンの

Contents:

| | |
|--------------------------------|-----------|
| 1. ロボトレース競技とは | 2 |
| 1.1 ロボトレースの競技内容 | 3 |
| 2. ロボトレーサーの基本的なハードウェア構成 | 4 |
| 2.1 CPU ユニット | 4 |
| 2.2 センサユニット | 5 |
| 2.3 モータユニット | 5 |
| 2.4 電源ユニット | 5 |
| 3. トレーニングトレーサーの回路 | 6 |
| 3.1 CPUユニット | 6 |
| 3.2 センサユニット | 7 |
| 3.3 モータユニット | 8 |
| 3.4 電源ユニット | 9 |
| 4. 参考 | 10 |
| 5. 著作権について | 10 |
| 6. お問い合わせ | 10 |
| 改版履歴 | 10 |

Contents:

| | |
|--------------------|-----------|
| 1.安全について | 2 |
| 1.1.安全にご使用いただくために | 2 |
| 2.ハードウェアの製作 | 3 |
| 2.1.事前準備編 | 3 |
| 2.1.1. パーツ一覧 | 3 |
| 2.1.2.必要用具一覧 | 6 |
| 2.2.製作編 | 8 |
| 2.2.1やすり掛け | 8 |
| 2.2.2.ケーブル加工 | 10 |
| 2.2.3.半田付け | 14 |
| 2.2.3.1.マーカーセンサ基板 | 15 |
| 2.2.3.2.メイン基板 | 22 |
| 2.2.4.組み立て | 43 |
| 3.各部位の説明 | 51 |
| 4. 著作権について | 52 |
| 5.お問い合わせ | 52 |
| 改版履歴 | 52 |

Contents:

| | |
|----------------------|-----------|
| 1.序論 | 2 |
| 2.開発環境の整備 | 3 |
| 2.1 Arduino環境をインストール | 3 |
| 2.2 Nucleoの環境整備 | 8 |
| 2.3 環境の確認 | 13 |
| 3.ソフトウェアについて | 33 |
| 4.著作権について | 33 |
| 5.お問い合わせ | 33 |
| 改版履歴 | 33 |

Contents:

| | |
|-------------------------|-----------|
| 1.序論 | 2 |
| 2.開発環境の整備 | 2 |
| 3.Mbedのサンプルプログラム | 11 |
| 3.1 STEP2のサンプルプログラム | 11 |
| 3.2 STEP3のサンプルプログラム | 12 |
| 3.3 STEP4のサンプルプログラム | 12 |
| 3.4 STEP5のサンプルプログラム | 13 |
| 3.5 STEP6のサンプルプログラム | 13 |
| 3.6 STEP7のサンプルプログラム | 14 |
| 3.7 STEP8のサンプルプログラム | 17 |
| 4.オフラインの開発環境の整備 | 19 |
| 4.1 windows環境 | 20 |
| 4.2 Mac環境 | 25 |
| 5.著作権について | 27 |
| 6.お問い合わせ | 27 |
| 改版履歴 | 27 |

Contents:

| | |
|----------------------------------|-----------|
| 1.序論 | 2 |
| 2.開発環境の整備 | 2 |
| 2.1 STM32CubeIDEをインストール | 2 |
| 2.2 トレーニングトレーサーのピンの設定 | 18 |
| 2.2.1 LEDピンの設定 | 18 |
| 2.2.2 ブザーピンの設定 | 18 |
| 2.2.3 スイッチのピンの設定 | 19 |
| 2.2.4 アナログピンの設定 | 20 |
| 2.2.5 モータドライバのピンの設定 | 21 |
| 2.2.6 9軸センサのピン(I2C)の設定 | 24 |
| 2.2.7 エンコーダーの設定 | 24 |
| 2.3 機能の細かい設定 | 26 |
| 2.4 Printfについて | 36 |
| 2.5 環境確認 | 37 |
| 3. STM32CubeIDEのサンプルプログラム | 39 |
| 3.1 STEP2のサンプルプログラム | 41 |
| 3.2 STEP3のサンプルプログラム | 42 |
| 3.3 STEP4のサンプルプログラム | 42 |
| 3.4 STEP5のサンプルプログラム | 42 |
| 3.5 STEP6のサンプルプログラム | 43 |
| 3.6 STEP7のサンプルプログラム | 45 |
| 3.7 STEP8のサンプルプログラム | 46 |
| 4.著作権について | 47 |
| 5.お問い合わせ | 47 |
| 改版履歴 | 47 |