

走行軌跡の提示からモータ制御までを担当する

第6章

走行制御モジュールの詳細

上村 聡文/油田 信一/飯島 純一

TIrobo01-CQは、統括制御モジュール(NetBSDサーバ)のCPUが指示を出し、ほかのサブモジュールが指示された処理を実行するという、各機能をモジュール化した機能分散型のシステム構成になっている。本章では、TIrobo01-CQにおける「走行」する機能を提供する走行制御モジュールについて、走行の指示方法から、実際にモータを回転させて移動させる方法までを解説する。(筆者)

1. 走行制御モジュールの構成

TIrobo01-CQに搭載されている走行制御モジュールは、小型移動ロボット「ビーゴ(写真1)用」に開発されたものです。今回は、これのパラメータを調整してTIrobo01-CQで使用しました。

走行制御モジュールのソフトウェアは、下記のWebサイトからダウンロードでき、またドキュメントも用意されています。

<http://www.roboken.esys.tsukuba.ac.jp/~satofumi/beego.php>

システムの概略 CPUに依存しにくい構造

走行制御モジュールは、モータが接続されるコントローラ(SH-2ボード, 秋月電子通商製)と、ライブラリの形で提供される走行コマンド(ソフトウェア)から構成されます(図1)。

コントローラ側のCPUにはSH7045Fが搭載されており、

リアルタイム性の高い制御が行えます。また、走行コマンドはライブラリのソース・コードとして提供されるため、ライブラリが動作するCPUに依存しにくい構造になっています。

走行制御モジュールのハードウェア構成

電源を含むハードウェア構成を図2に示します。

走行制御モジュールは、二つの駆動輪、つまり二つのモータを制御するハードウェアです。モータと、コントローラ側のCPUであるSH7045Fとの間には、モータ・ドライバが接続されており、モータに流す電流を制御しています。

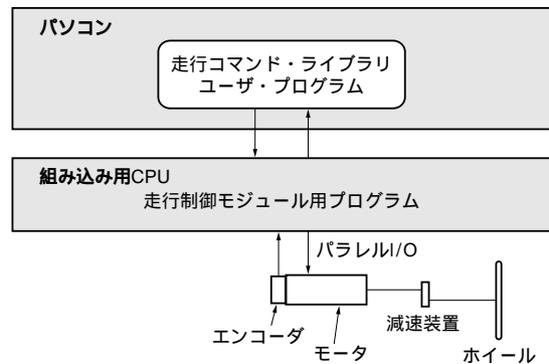


図1 コントローラとライブラリの構成図

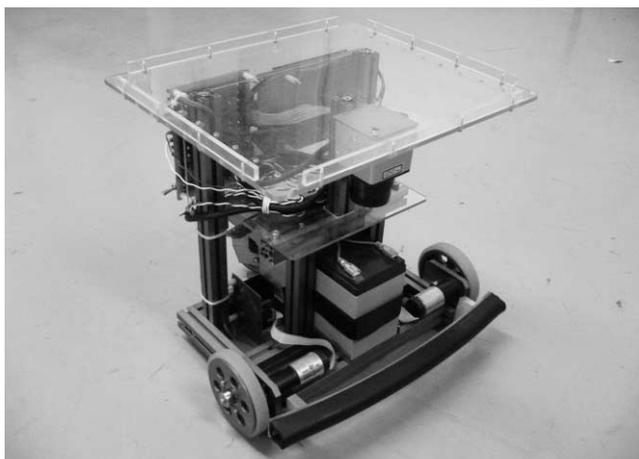


写真1 小型移動ロボット「ビーゴ」

TIrobo01-CQで使用された走行制御モジュールは、もともとはこのビーゴ用に開発されたもの

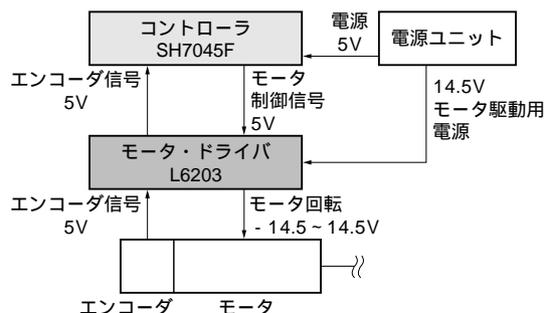


図2 ハードウェア構成

走行制御モジュールのソフトウェア構成

ソフトウェアは、コントローラ上で動作してリアルタイム制御を行うプログラムと、コントローラに指示を出すための走行コマンド・ライブラリから構成されます。コントローラ上で動作するプログラムはSH7045F依存になりますが、ライブラリ用のプログラムはPOSIX準拠のもので実装してあるため、動作環境に依存しにくくなっています。

コントローラとライブラリ間は、RS-232-Cでデータ通信を行います。また、ソフトウェアはレイヤ(階層)で構成されています(図3)。

レイヤは、プログラムの機能単位でファイルに分割されています。走行コマンド・ライブラリと走行制御コントローラでのプログラム構成は、以下のようになっています。実際には、レイヤに含まれないプログラムである、main()処理、数学関数、SH7045F操作のプログラムがあります。

▶ 走行コマンド・ライブラリ

【アプリケーション層】

● 走行コマンド

- run_ctrl.c : ライブラリ初期化, ほか
- run_utils.c : 補助関数, ほか
- parameter_ctrl.c : パラメータ管理
- move_ctrl.c : 移動コマンド
- direct_ctrl.c : デバイスの直接制御コマンド
- judgement_ctrl.c : 判定コマンド

● 座標系管理

- coordinate_ctrl.c : 座標系コマンド

▶ コントローラ

【制御マネージャ層】

- modeCtrl.c : 制御モードの管理

【制御層】

- velocityCtrl.c : 制御モードごとの速度制御

【インターフェース層】

- bodyCtrl.c : 筐体制御
- bodyPosition.c : 推定自己位置の計算
- coordinateCtrl_target.c : 座標系の処理
- velocityInfo.c : 速度情報の管理
- wheelCtrl.c : 車輪の回転制御
- directWheelCtrl.c : 車輪の直接制御

【デバイス層】

- motorCtrl.c : モータ制御
- encCtrl.c : エンコーダ信号の読み出し
- directDeviceCtrl.c : デバイスの直接制御

2. 走行機能の実現

車輪型の移動ロボットは、モータを回転させれば移動します。さらに、指示どおりの経路を走行させるためには、どのように走行経路の指示を行うのかを決める必要があります。

この走行制御モジュールでは、座標系を用いて経路を指示する走行コマンドを提供し、それを実現するコントローラを実装します。

経路の指示方法

どのような処理が必要になるかを示す例として、ロボットの向いている方向を90°だけ回転してから直進する、というタスク(図4)を考えます。また、このタスクを実現するためのプログラムをリスト1に示します。

このプログラムが実行されると、統括コントローラからRS-

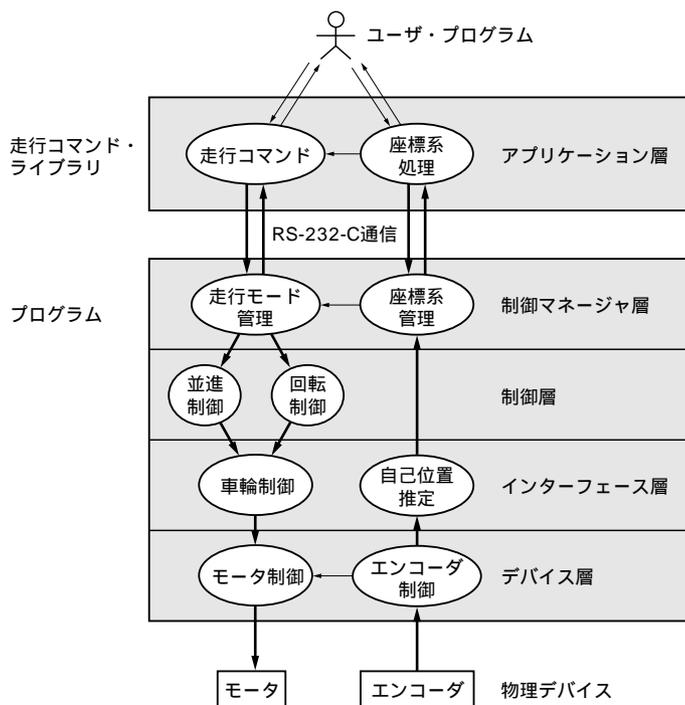


図3 ソフトウェアの階層構成

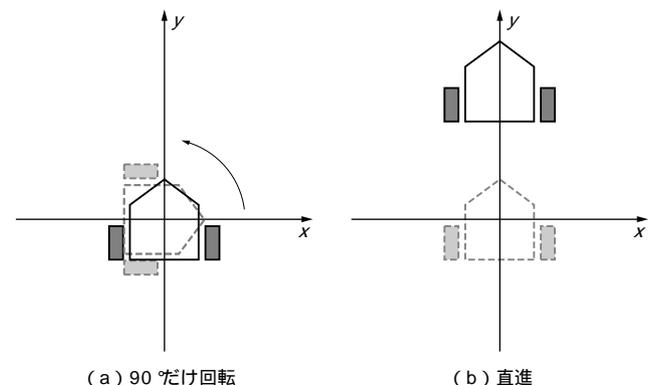


図4 90°回転してから直進