

# ART Linux を用いた知能移動ロボットのシステム構成の紹介

岡山大学 永谷 圭司, 五福 明夫, 田中 豊

## Introduction to System Integration for Autonomous Mobile Robot based on ART Linux

Keiji Nagatani, Akio Gofuku, Yutaka Tanaka, Okayama University

**Abstract:** The authors had developed four mobile robots as research platforms to research motion planning for mobile robots. Each robot's controller consists of a standard PC and some ISA/PCI boards to control motors and sensors. To supervise them, ART Linux is adopted for operating system. In this paper, an overview of the robot's control system using ART Linux is introduced.

### 1 はじめに

筆者らの研究室では、知能移動ロボットの動作研究を行うため、これまでに4台の自律型移動ロボットを構築した。各ロボットのコントローラは、制御用PCを中心としたシステムで構成されており、ISAバスおよびPCIバスに接続されたボードを用いて、センサやアクチュエータを制御している。これを統括するOSには、ART Linuxを採用している。本稿では、ART Linuxの適用事例として、このOSを用いたロボットシステムの構成と特徴について紹介する。

### 2 ハードウェア構成

筆者らの研究室で構築した4台の移動ロボットは、自律移動ロボットのナビゲーションや移動マニピュレータの動作計画の研究を行うために開発されたものである (Fig.1)。このため、これらのロボットには、複数のDCモータ、各種センサおよび、これらを統括するコントローラが搭載されている。本節では、これら4台のロボットが有するハードウェアの構成要素について紹介する。

#### 2.1 アクチュエータ

走行系は、メカナムホイールを有する全方向車椅子をベースとしたロボット (Fig.1 - ) 以外は、独立操舵輪を採用している。そのため、各ロボットは、最低2個のモータを有する。なお、Fig.1 - のロボットについては、車椅子付属のジョイスティックの信号を制御することで走行制御を実現している。

各ロボットは、を除き、マニピュレータを搭載している。このうち、双腕マニピュレータを搭載したタイプでは、腕だけで11個のモータを有する。

これらのモータの位置/速度制御を行うために、各モータは、各々の回転数を計数する必要がある。そのため、全

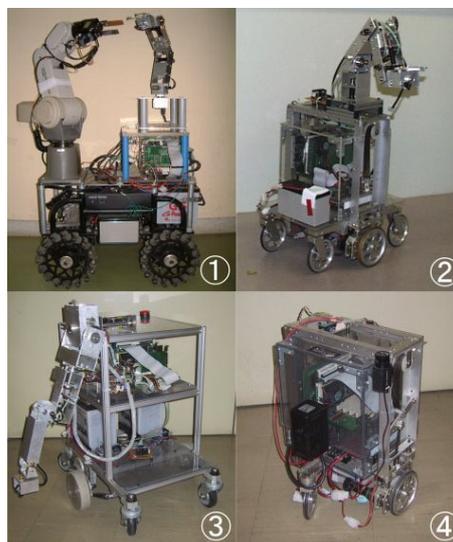


Fig. 1: Autonomous mobile robots

てのモータには、エンコーダが搭載されている。これらのモータ制御のため、ロボットには、以下に示すインターフェースボード (全てISAバス) を搭載している。

1. 4Ch D/A ボード ( に搭載)
2. 8Ch Up/Down カウンタボード ( に搭載)
3. Ritech Board IF-0139-1 (岡崎産業) ( に搭載)  
(Up/Down カウンタ, PWM ジェネレータ, A/D コンバータ, D/A コンバータを搭載)
4. CPLD を用いた自作コントローラボード ( に搭載)  
(Up/Down カウンタ, PWM ジェネレータを搭載)

#### 2.2 センサ

各ロボットは、研究内容に応じて、外界センサとして、視覚センサ、力覚センサ、距離センサを搭載している。こ

これらのセンサと統括コントローラ（2.3 節参照）とのインタフェースは、下記に示すとおり、PCIバスまたはシリアル接続となっている。

1. 視覚センサ：日立 IP5000 (PCIバス)
2. 力覚センサ：ニッタ レシーバボード (PCIバス)
3. 距離センサ：北陽電機 PB-9 (シリアル接続)
4. 距離センサ：シャープ GP2D12 (AD 変換器接続)

### 2.3 統括コントローラ

上述のハードウェアを統括するためのコントローラには、制御用 PC を利用している。このコントローラは、PCIバスおよび ISAバスを利用可能としつつ、小型化を図るために PISAバスを採用している。このロボットを例とした統括コントローラの構成図を Fig.2 に示す。

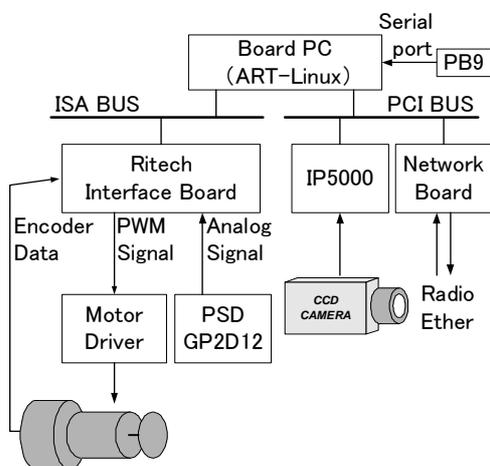


Fig. 2: Controller for our mobile robot

### 3 ソフトウェア構成

ソフトウェアは、システムプログラムとユーザプログラムの2階層に分かれている。この模式図を Fig.3 に示す。

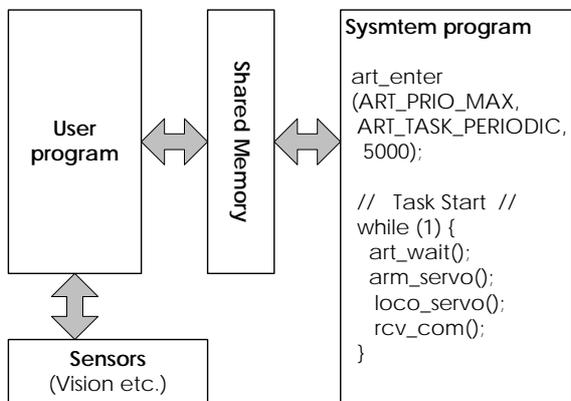


Fig. 3: Software structure

システムプログラムは、ロボットのハードウェアを制御するプログラムであり、主に走行制御やマニピュレータの制御を行う。制御のループには、リアルタイム性が要求されるため、このプログラムを実行するプロセスは、ART Linux<sup>1)</sup> のリアルタイムプロセスで実行される。Fig.3 の System program の枠中に、ART Linux のシステムコールを用いた「5 ミリ秒のサーボループを構築するためのプログラム」の記述を示す。ここで、arm\_servo, loco\_servo は、サーボループ用の関数であり、エンコーダ情報とモータの回転角度 / 角速度を基に、PID 制御によりモータの目標電圧を決定する。また、rcv\_com 中には、shared memory にアクセスし、制御モードや変数を獲得する関数が含まれる。

一方、ユーザプログラムは、ロボットの行動を記述するものであり、現段階の実装では、各センサを制御するコードも、この中に含まれる。なお、このプログラムを実行するプロセスは、通常のプロセスで実行される。

これら2つのプログラムは、shared memory を介して通信を行う。具体的には、ユーザプログラムから、マニピュレータの目標関節角や走行制御コマンドを shared memory 上に記述することで、システムプログラム上で実行しているモータの制御手法やパラメータの変更を行う。一方、システムプログラムは、shared memory 上に格納したロボットの位置・姿勢などの情報を随時更新する。これにより、ユーザプログラムは、非同期に shared memory にアクセスすることで、これらの情報を獲得することができる。

### 4 おわりに

本稿では、現在筆者らの研究室で開発を進めている移動ロボットのハードウェアとソフトウェアの構成について紹介した。本稿のまとめにかえて、以下に、筆者らの研究室で ART Linux を採用した主な理由について述べる。

1 つ目の理由は、デバイスドライバの互換性である。筆者らの研究室では、日立製の画像処理ボード (IP5000) や無線 LAN ボードを導入する際、提供された Linux のデバイスドライバが動作しないといった問題が発生していたが、ART Linux を導入することで、この問題が解消された。

2 つ目の理由は、実装の簡易さである。大学の研究室では、毎年学生が入れ替わるため、ソフトウェア教育は、重要な項目の一つとなる。ART Linux は、シンプルなシステムコールを用いてリアルタイムプロセスを記述することができ、さらにプロセス間通信には、通常の Linux のシステムコールが利用可能である。これにより、通常の Linux を用いたソフトウェア教育の延長として、ロボットシステムに関する教育が行えるようになった。

### 参考文献

- [1] 石綿陽一, 特集リアルタイム OS としての Linux 活用法, Interface, CQ 出版社, 1999 年 11 月号