

# ロボットを用いた組み込みシステム学習教材の開発

田中裕樹<sup>†</sup> 大角圭吾<sup>†</sup> 川上亮太郎<sup>†</sup> 西野洋介<sup>†</sup> 早川栄一<sup>‡</sup>

## 1. はじめに

近年、携帯電話や情報家電など、組み込みシステムの需要が高まっている。しかし、組み込みシステム技術者が不足しており、その育成が重要となっている[1]。

しかし、現在の組み込みシステムの講義では効果的な成果が得られていない。これは、組み込みシステムに関する学習環境や教材が確立されていないことが挙げられる。シミュレータベースの環境では、組み込みシステムに不可欠な実装教育には不十分である。また、現在の複雑なハードウェア構成やOSを用いた場合、概念や全体構成を把握しにくく、学習者への負担が大きくなる。

一方、学習者の興味を引く教材としてロボットを利用した環境があるが、過度に簡易もしくは複雑なものしか存在せず、学習教材としては不向きである。

そこで、新たなロボットを用いた組み込みシステム教育支援環境を提案する。本環境は、組み込みシステム技術者の教育を目的として、学習者の興味を引きやすく、理解しやすいロボットハードウェアを教材として利用し、概念から実装までを単一環境下で連続的に学習可能にする。また、ロボットおよびOS内部の可視化環境を用意することによって、学習の各場面において、システムの内部動作の理解を可能にしている。

## 2. 特徴

### (1) 学習に適したハードウェア

教材として演習などで扱いやすく、動作が目に見えるロボットを用いる。ロボットの概観を図1に示す。

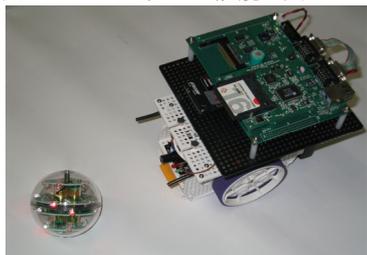


図1 ロボットの概観

ハードウェアは仕様を単純なものにすることで、学習者がデバイスの制御を把握しやすくする。また、無線LANを搭載し、CPUの性能やメモリサイズが豊富なハードウェアを用いることで、プログラミングしやすく、システム内部の状態をログとして出力でき、学習者をサポートする環境を提供できる。

### (2) 学習に適したシステムソフトウェア

OSはコードリーディングに適した構成、コード量で提供する。また、通信やデバッグ機能についてはファームウェアとして、OSと分離してコードを簡単にしている。

<sup>†</sup> 拓殖大学大学院工学研究科

<sup>‡</sup> 拓殖大学工学部

### (3) 学習時のサポート

ロボットの内部情報をログとして保存し、デバイスの動作や、OSの動作などを、ロボットの走行時のビデオと対応させ、デバッグを行えるようにする。また、実機のOSの動作ログを基に、OSの動作をアニメーション表示することで、ロボットの動作とOS、アプリケーションの動作を結びつけて学習を行うことができる。

## 3. コースウェア

プログラム開発にはC言語およびJava言語を用いるので、これらの言語を習熟しており、前提知識としてロボット工学やセンサ工学を学んだ人が対象となる。本教材は次の段階を通して学習を行う。

### (1) 制御プログラミング学習

コースを走行させるといったものを題材とし、ハードウェアの制御を学習する。

### (2) OSなどのシステムプログラムの動作を学習

ロボットの動作ログを基に、システム内部の動作を可視化し、システムソフトウェアの動作概念を学習する。次にコードリーディングを行い、最後に割込みなどのOSの機能の実装や、リアルタイムスケジューラの拡張を行う。

### (3) より高度な制御プログラミングの学習

これにはOSやネットワークの知識など、これまでの知識を活かして、システム、ネットワークの協調動作を学習する。

## 4. 設計

本システムの全体構成を図2に示す。

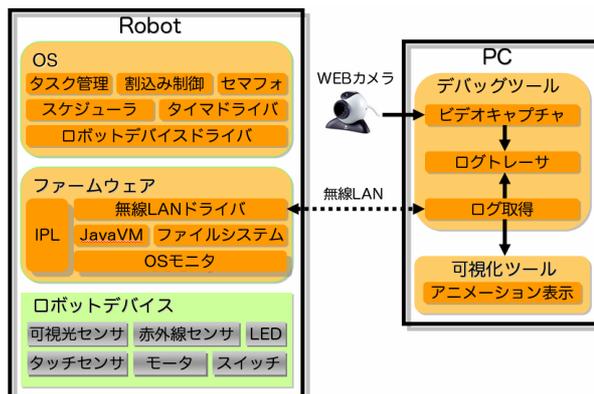


図2 全体構成

ロボットとPCは無線LANで通信を行う。OSはロボット走行時に動作ログを送信しPC上でログを作成する。学習者は作成されたログを基にデバッグ、概念学習を行う。

### 4.1 ハードウェア

組み込みボードとしてCAT709を用いる。これは、複数の実装学習やOSの学習に利用されることを考え、処理能力が高いものを選択した。また無線LANが利用できることも不可欠である。

ロボット本体としてサッカーロボ 915[2]を用いる。本教材は応用として RoboCup サッカー[3]を想定しているため、可視光センサ、赤外線センサ、タッチセンサを備えているものを選択した。

#### 4.2 システムソフトウェア

システムソフトウェアは、組込み OS およびファームウェアの 2 層から構成する。組込み OS の実装学習としては、オープンソースなどで用いられるコードリーディング[4]を想定し、2000 行以内に必要な機能を実装する。また、コメントおよびドキュメントを充実させて、学習者が読みやすいような記述を行う。

ファームウェア層では、ネットワーク通信やファイルシステム、JavaVM など、制御に直接関係ない部分をコンポーネント化して提供する。これによって、OS 自身のコードや構成が単純になる。

本システムでは、必要となる次の機能を実現する。

##### (1)組込み OS

組込み OS で必須であるリアルタイムスケジューラについては、基本的な優先度付きスケジューリング、RM、EDF スケジューリングをサポートする。また、ロボット関連のドライバ群の提供も行う。さらに、4.4 で示す可視化環境で用いる OS の内部データを取得する機能も提供する。

##### (2)ファームウェア

ファームウェアは割り込み処理、デバッグ用モニタ等から構成される。割り込み処理は、ハードウェアの割り込みベクタを仮想化した形で提供する。これは、OS 自身の移植性を向上させるとともに、ネットワーク装置などのデバイスの割り込み処理を、OS のコードから隠すために用いる。

#### 4.3 デバッグツール

本システムではログ解析アプリケーション[5]を用いてログ解析を支援する手段を学習者へ提供する。

本デバッグ環境では、動作中の内部状態をログとして記録し、デバッグやテスト時の参考データとして学習者へ提供する。ログから特定の時間におけるプログラム内で実行中の関数名、スレッド ID、ロボットに搭載されたセンサの値が取得できる。この情報は組込み OS 自体にも適用できるので、OS 内部までのログトレースが可能である。さらに、実行時の動作の動画をログと同期して学習者へ提供する。学習者は再現性の低いバグを何度でも見ることが可能になり、プログラムと動作との関係を理解することができる。この手法は、従来の方と組み合わせることで、さらに強力なデバッグ、テスト環境を提供することができる。

実行画面を図 3 に示す。ロボット動作のビデオと連動して、センサ入力の値がグラフ化される。同時に、現在実行しているプログラムが示される。これによって、センサがどのスレッドのどの部分で動作したか、それによってロボットがどのような動作をしたかを学習者は確認することができる。

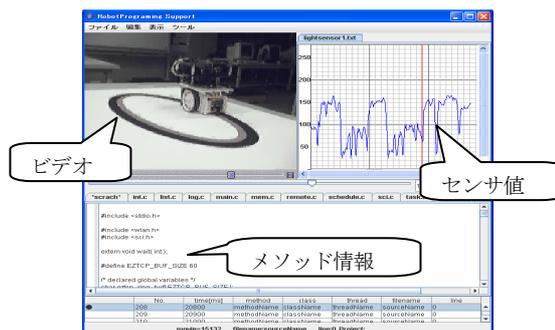


図 3 ログトレース画面

#### 4.4 可視化ツール

OS のタスク情報のログを基にタスクスケジューリング、タスク実行時間、割り込み処理、I/O処理を動的に可視化する。これにより、ロボットが実際に走行しているときのタスクの動作を目で見て確認できるので、学習者はロボットの動作とシステムの動作を結び付けることができ、組込みシステムの導入教育が可能になる。

さらに、実装教育において OS の改良や高度な制御プログラム実行時のデバッグツール、タスク実行時間グラフによる性能評価を可能とする。

アニメーション画面を図 4 に示す。ここではリアルタイムスケジューリングの状態遷移アニメーションおよび、グラフ表示を行っている。

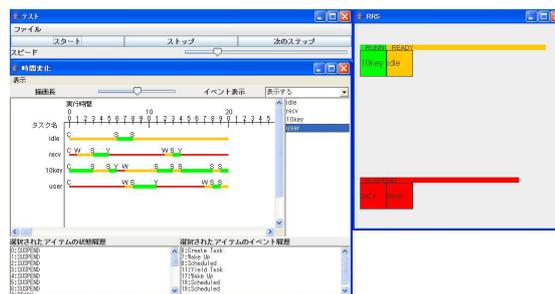


図 4 アニメーション表示画面

#### 5. おわりに

本稿では、ロボットを用いた組込みシステム学習教材の開発について述べた。現在、試作したロボットおよび、OS、デバッグ環境、可視化環境が動作しており、本システム一つの環境で組込みシステムの基礎部分から、OS の概念学習および、実装学習を行える環境を構築した。

今後の課題は、本システムの有効性の評価を行うことと、Web ベースでの動作を行うことである。

#### 参考文献

[1] 大原茂之, “日本の組込みシステム産業と技術者育成の課題,”情報処理, Vol. 46, No.2, pp.161-168, feb, 2005  
 [2] エレキット株式会社, <http://www.elekit.co.jp/index.php>  
 [3] Robocup, <http://www.robocup.or.jp/>  
 [4] Diomidis Spinellis, “Code Reading: The Open Source Perspective (Effective Software Development)”, mar., 2003  
 [5] 吉田恵美, 山本茂樹, 西野洋介, 早川栄一, “ロボットプログラミング学習支援環境の開発”, FIT2003 N-026