

## 大学学部における 組込みソフトウェア教育事例

渡辺晴美<sup>†</sup> 吉田正廣<sup>†</sup>

本稿では、大学学部における組込みソフトウェアに関する教育事例を紹介する。組込みソフトウェアは、機械、電気、ソフトウェアの境界領域であること等から、その教育は容易ではない。授業カリキュラムでは、展示会の見学、企業人による授業、学外活動にロボコン道場等により、実践面やモチベーションに配慮し様々な取り組みを行った。これらの取り組みでは工学的基礎力を重視し実施している。その結果、大学3年時では、リアルタイムカーネルが開発できる程度に成長した。

### A Case Study for Undergraduate Educations in Embedded Software Engineering

Harumi Watanabe<sup>†</sup> and Masahiro Yoshida<sup>†</sup>

The article presents a case study for undergraduate educations in embedded software engineering. As embedded software encompasses a wide range of electronics, software and mechanical engineering, the education is challenging tasks. In our education, we are emphasizing practical engineering technology and maintain strong motivation. The class curriculum includes touring Embedded Technology exhibition, practical excises, etc. Additionally, some students have challenged ET software design robot contest. Consequently, at the third grader of the university, many students acquired real-time kernel development technology.

#### 1. はじめに

組込みソフトウェアでは、技術者育成が急務であると言われるようになり、経済産業省推進の組込みスキル標準 ETSS[1]が整備され、著者らの学科も、そのような背景のもとで2008年に新設された。現在、第6セメスターまで実施されている。本カリキュラムはETSSをふまえ、各科目の実施についても、[2]に関連した教員を複数含むことから、[2][3]に関する知識体系を意識している。

組込みソフトウェアは、機械、電気、ソフトウェアの境界領域であり、各々の分野は急速に発展を遂げていることから、学べき領域は広範である。さらに、[3]でも述べられているように、情報系の教育は、産業界から、現場では役に立たないという批判も多く、実践力も求められている。一方で、近年の学力不足は顕著であり、その教育が容易でないことは周知である。

このような状況において、実践力、目的意識の効果的な育成としてPBL(Project Based Learning)が注目され、盛んに実施されている[4][5][6][7]が、PBLには限界がある。[3]らは、基本概念や技術を習得する際に、PBLで採用した開発やマネジメントのスタイルに基づく一面的な理解になりかねないと述べている。筆者らの経験からも、正確に理論を理解することは難しく、目的意識向上さえも十分効果的とは言えない[4]。

以上から、我々は、工学的基礎力と実践力を重視し、産学連携授業、展示会見学、豊富な演習・実験、科目間の連携、授業内外のPBL等、様々な側面において工夫をしてきた。その結果、大学3学年終了時には、想定以上に教育できたと自負している。例えば、授業において、リアルタイムカーネルを開発できる程度に成長し、課外活動のPBLでは、高校生向け講習会を、企画・運営から教材開発、指導にいたるまで学生のみでの活動で実現している。本稿では、組込みOSを中心に、我々の教育事例について紹介する。

以下、2章では組込みソフトウェア学科の履修モデルを中心とした学科教育全体について紹介する。3章では、組込みOSの授業紹介、4章では、組込みOSとの関連科目について記す。5章では、紹介した教育の効果について議論する。

#### 2. 履修モデルと課外活動

表1主専攻科目履修モデルを記す。本表は学科で推奨している専門科目の履修モデルであり、多くの学生が本モデルに従って履修を行っている。特徴的な内容は下記である。

<sup>†</sup> 東海大学情報通信学部組込みソフトウェア工学科  
Tokai University, Department of Embedded Technology

- (1) 組込みソフトウェア工学通論では、組込みソフトウェアに関係する技術をおムニバス形式で紹介する授業であるが、組込み総合技術展(ET)の見学も行っている。
- (2) 組込みソフトウェア特別講義 I・II：毎週、企業で活躍する方々を招待し、おムニバス形式の授業を行っている。
- (3) 企業研究：インターンシップ
- (4) 専門科目の多くで実験を導入している。
- (5) 組込みソフトウェア基本プロジェクト I・II：実践的実験の導入

上記(1)~(3)により、産業界から求められている人物像や技術を理解させ、目的意識を向上させるようにしている。(4)に関しては、論理回路では IC トレーナー、ハードウェア記述言語では、FPGA ボード、組込みコンピュータシステム・同演習、組込み OS、計測と制御ではマイコンボードというふうに、専門科目のほとんどの科目において実験を導入している。ソフトウェア分析・モデリング、ソフトウェア設計と検証・同演習等のソフトウェア系の科目では、企業経験のある教員が、ソフトウェア開発で実際に起きやすい題材を取り上げ、モデリングやグループ討論を行ったりしている。(5)では、単に、グループでものづくりを行う PBL とはせず、各科目で学んだことを総合的に復習・応用できる内容を題材としている。

表 1 主専攻科目履修モデル

1	2	3	4	5	6	7	8								
線形代数・同演習	O4	微分積分・同演習	O4	確率統計と待ち行列・同演習	x4	データ解析・同演習	x4	入出力デバイス・同演習	x4	オブジェクト指向設計・同演習	x4	情報通信セキュリティ	x2	知的財産	x2
コンピュータリテラシ	O2	組込みソフトウェア工学通論	O2	データ構造とアルゴリズム・同演習	x4	スクリプトプログラミング	x2	ソフトウェア分析とモデリング・同演習	x4	計測と制御・同演習	x4			ユーザビリティ設計	x2
プログラミング入門	O2	電気電子回路・同演習	O4	論理回路・同演習	x4	組込みコンピュータシステム・同演習	x4	リアルタイムシステム・同演習	x4	組込みOS・同演習	x4			開発プロジェクト管理論	x2
離散数学・同演習	x4	プログラミング応用	x2	オブジェクト指向プログラミング	x2	マルチメディア実習	x2	コンピュータネットワーク	x2	ソフトウェア設計と検証・同演習	x4				
数理基礎・同演習	x4	自然科学基礎・同演習	x4			ハードウェア記述言語・同演習	x4	デジタル信号処理システム・同演習	x4	データベース・同演習	x4				
						組込みソフトウェア開発特別演習 I	x2	組込みソフトウェア開発特別演習 II	x2	組込みソフトウェア開発特別演習 III	x2				
						企業研究	x1	インターンシップ	x2						
		チャレンジプロジェクト I	x1	チャレンジプロジェクト II	x1	総合情報プロジェクト I	x1	総合情報プロジェクト II	x1						
				プレゼンテーション演習 I	O2	プレゼンテーション演習 II	O2	テクニカルコミュニケーション演習 I	O2	テクニカルコミュニケーション演習 II	O2				
						組込みソフトウェア基本プロジェクト I	O2	組込みソフトウェア基本プロジェクト II	O2	組込みソフトウェア実践プロジェクト I	O2	組込みソフトウェア実践プロジェクト II	O2		

以上のカリキュラム内の授業に加えて、ロボコン道場と呼ぶ課外活動を実施した。ロボコン道場では、様々な教員が参加し、授業の補講から授業では扱わない発展的な内容まで、産学連携により、様々な活動を行ってきた。主要な活動は下記の通りである。

- (1) 2009年10月~1月：基礎的なマイコンプログラミングから状態設計に基づいたマイコンプログラミング（企業で活躍する卒業生にも支援していただいた）
- (2) 2010年2~3月：UML、再利用開発、リアルタイム設計
- (3) 2010年3月：春合宿、他大学教員によるシリアル通信の講義と演習、ラオスでのロボコンに参加したマイコンカーの紹介
- (4) 2010年8月：関連他学部研究室との合同合宿
- (5) 2010年4~9月：ET ロボコン(9月開催)への参加準備
- (6) 2010年11月：産学連携による再利用開発に関する学内外向け講習会
- (7) 2010年10~11月：学生による高校生向け組込み講習会(11月)の準備

上記(7)において、LEGO マインドストーム NXT を使用したが、市販のビジュアルプログラミングは使用せずに、学生達は、どうすれば組込みの楽しさを伝えられるかということに熱心に議論し、高校生を2時間程度の指導で動作できるシステムを開発した。本開発では、並行性に関する技術的に難しい問題も含んでいたが、それらの問題についても、自分達で解決し、企画、広報、調達、予算管理、プロセス管理、教材開発、当日の指導も含め、完全に学生のみで行った。

### 3. 組込み OS

本章では、組込み OS のシラバスと教材システムの構成について述べる。

#### 3.1 シラバスと内容

表 2 に組込み OS のシラバスと課題概要を示す。組込み OS は、学部 3 年時秋学期に行われる 2 コマ連続の選択科目であり、56 人の在籍に対 38 人が履修している。シラバス案作成時期には、初回からマイコンを使用し、ブートから作成することを計画したが、中盤から使用することにし、アセンブラを触る部分は、授業後半にした。前半は cygwin を用いてセルフ環境で開発を行った。学生達は他授業等を通し、マイコンには十分に慣れていたが、ハードウェアへの苦手意識や、クロス環境に面倒さを感じていたためである。本授業では、特に、「並行性の問題について理解すること」、「ハードウェアマニュアルの読解力を育成すること」を重視した。前者を重視した理由は、並行性は多くの組込みソフトウェアに必要な概念であり、授業として扱わなければ、気付かない問題であるためである。後者

については、単独でマイコンを使用できるようになるための大きな障害であると考えているためである。以上から、授業前半では、セルフ環境を用い、十分に並行性について理解させた。

教科書に関して、特定の書籍は指定せず、[10][11][12][13][14][15]を参考書として、シラバスや授業で紹介した。組込み OS の書籍は、理論中心、実装中心のいずれかが重視されている。両方が充実している書籍は[14]のように、ページ数が多く、高価であるため、本学科の学部学生には適さない。

表 2 組込み OS シラバス

	内容	課題概要
1	ガイダンス, リアルタイム OS とは?	トロンプロジェクト, ITRON, $\mu$ ITRON T-Engine, T-Kernel, TOPPERS について調査を行う.
2	並行性とプログラミングの復習	構造体・関数へのポインタの演習問題
3	タスク	キュー操作
4	スケジューラ	Ready 状態と Run 状態の構築
5	タスク間通信の必要性和理論	sleep と wakeup の構築
6	タスク間通信の実現方法	セマフォの構築
7	復習	
8	中間試験	
9	開発環境とタイマ割込み	ハードウェアマニュアルでタイマの調査 1
10	アナログ出力	ハードウェアマニュアルでタイマの調査 2
11	メモリとデバッグ	デバッガでメモリ・レジスタの内容確認
12	run to completion 型スケジューラの実現	run to completion 型スケジューラの構築
13	ディスパッチャーの実現方法	ディスパッチャーの構築
14	復習	
15	期末試験	

### 3.1 教材システム

組込み OS の授業では、図 1 に示すとおり R8C29(図 1 MB-R8C29)にリアルタイム

カーネルを構築し、FT232RL(図 1 AE\_UM232R)シリアル変換モジュールを使用した。本システム構成では、シリアル変換モジュールを介し、シリアル通信が可能であり、プロセッサのモードを変更することで、ROM ライター等の装置なしに書き込みが可能である。R8C29 はアナログポートも有するため応用実験を行いやすい。さらに、コンパイラ等の開発環境は、ルネサスエレクトロニクスが配布している High-performance Embedded Workshop を使用できるため安価に構築可能である。また、本教材では、図 2 に示すブレッドボードを利用し、学生が半田付けを行う時間を軽減した。

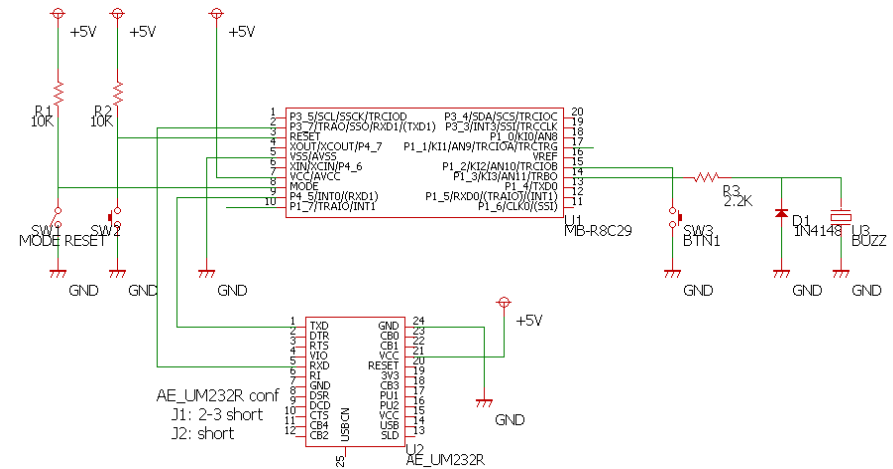


図 1 授業で使用した回路図

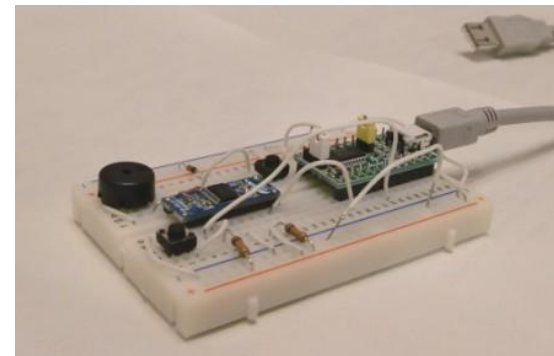


図 2 ブレッドボードの利用

#### 4. 科目型授業の充実と連携

リアルタイムカーネルを構築できる程度に、リアルタイム OS を理解するためには、プログラム、電気電子回路、コンピュータアーキテクチャの基礎力に加えて、リアルタイム OS の利用例や、並行性の必然性について、履修前に学ぶ必要がある。他の授業との連携が欠かせない。さらに、組込み OS の授業で使ったマイコンボードに、基本プロジェクト I で実習した非反転増幅回路を応用したシステムを構築することで、各科目で個別に学んだ内容が総合的に役立つことを経験する。

##### 4.1 プログラミング

プログラミングに関する授業は、プログラミング入門、プログラミング応用、オブジェクト指向プログラムの3つからなる。入門は、導入教育としての位置付けであることから、LEGO に触る等、楽しくプログラムに慣れることを目指している。プログラミング応用、オブジェクト指向プログラミングでは、実践に耐えられるプログラミング力を目指し、内容と量について以下の工夫を行っている。

内容に関しては、組込みソフトウェア学科であることから、ポインタとアドレス空間との関係を明確に理解させること、リスト構造や、関数へのポインタ等、組込み OS 開発に必要なプログラミング知識を盛り込んだ。さらに、プログラムの読解力を付けること、コーディング規約を意識した理解しやすいプログラムを記述できることに配慮している。読解力を育成することと、学習習慣を身につけさせるために、授業回ごとに、毎回 10 分間でできる小テストを行った。テストでは文法を問う問題もあるが、実行結果の推測や、プログラムの穴埋めにより、読解力育成に心がけた。

量に関しては、授業と無関係にプログラミングが得意になった学生がプログラミングする量を授業の中で行うことを目指している。授業回ごとに、数十行程度の課題を 7 題前後作成する。さらに、第二セメスターにあるプログラミング応用では、冬休みの課題として、逆行列の計算とバイオリズムを作成させている。第三セメスターにあるオブジェクト指向プログラミングでは、ゲームを作成させている。プログラミング応用では、300 行程度をプログラムできることを最低目標としているが、1000 行程度のプログラムを作成する学生も多い。オブジェクト指向プログラミング言語では、1000 行程度を目指しているが、5000 行以上作成する学生も複数いる。長いプログラムを作成する学生の大半は、大学ではじめてプログラミングに触れている。

尚、バイオリズムやゲームは完全に自由に作成させているのではなく、ある程度フレームワークを用意し、フレームワークに規定の内容を追加させる形式で

取っている。プログラムの構築方法を指導することで、実践的な規模や、設計モデルからシームレスにつながるプログラミング力育成をねらっている。

##### 4.2 コンピュータアーキテクチャ

履修モデルの「組込みコンピュータ・同演習」においてコンピュータアーキテクチャの指導を行っている。組込み OS の開発にとって、コンピュータアーキテクチャの知識は不可欠であるが、組込みコンピュータ・同演習の授業では、2 セメスター後に学ぶ組込み OS の必要性が理解しやすい題材を使用した課題を行っている。本課題では、マイコンボード OAKS8 を貸出し、電卓を作成している。本ボードには、図 3 に示すような回路であり、キーボードと 7 セグメント LED が同じポートを共有している。本授業では OS を使用していないため、タイマ割込みや並行性について理解し、状態設計を行わなければならない。本授業での経験は、組込み OS の必要性を理解するのに多に役立っている。

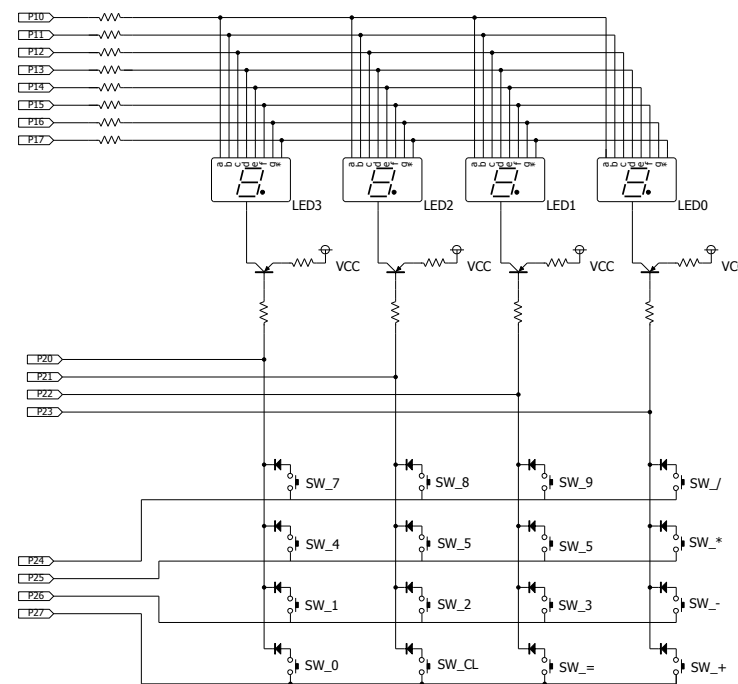


図 3 電卓の回路図

### 4.3 組み込みソフトウェアの開発

各科目の中で、様々な組み込みソフトウェアの開発課題に取り組んでいる。組み込み OS と関連した開発として、組み込みソフトウェア基本プロジェクト II がある。本授業では必須科目であり、ハードウェア記述言語、デジタル信号処理、組み込みシステム開発の 3 つのプロジェクトを全員が行う。以下、組み込み OS と関連した組み込みシステム開発プロジェクトについて述べる。本プロジェクトでは、図 1 の回路に、図 4 の赤外線距離センサを加えたシステムの開発を行う。本プロジェクトでは、センサの性質の調査、回路設計、ソフトウェア開発を行う。本システムでは、距離に応じて、音程を変化させた音を出力できる。本システムが構築できれば、部品を交換する程度で、人が近づいた時に音を鳴らしたり、ライトを点灯したりするシステムや、近づくとドアが開閉するようなシステムを構築できる。部品に応じて、ソフトウェアの実装方法や回路にも変化はあるが、モータをマイコンで制御することは他授業で行っている。

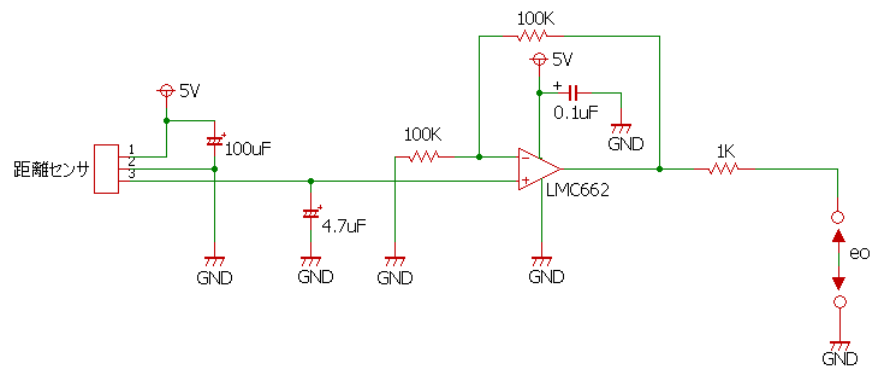


図 4 非反転増幅回路を利用した距離センサの回路

## 5. 議論

本章では、我々のアプローチによる教育効果について議論する。直観的ではあるが、下記から、想定以上に成功したと感じている。

- (1) リアルタイムカーネル等の複数の技術が必要なソフトウェアが構築できる。
- (2) 一般に頻繁に利用されている小規模な組み込みシステムを構築できる。
- (3) 講習会等のイベントを学生達のみで開催することができる。

- (4) 難しい課題に対する正答率、高い品質の成果物を提出した人数が、想定よりも多い。

成功した要因は、展示会への参加、企業の方々の講演等を通し、早い時期から目的意識を持たせたこと、豊富な演習、実習課題により学習習慣が身に着いたことが挙げられる。ET ロボコン等の本格的な PBL に参加した学生の人間力が向上したことは述べるまでもないが、授業の範囲で行った演習・実験、PBL に関しても、数学や理論的に難度のある内容や、他科目と関連した課題は非常に効果的であった。これらの課題に関しては、自然とグループ学習や議論の場が形成され、学習する雰囲気作りができた。

これまで、本稿で具体的に紹介してきた事例は、比較的小規模なシステムである。産業界では、大規模開発に耐えられる人材が求められている。本問題については、ソフトウェア分析とモデリング、ソフトウェア設計とモデリング等で、大規模システムに関する品質、プロセス、マネジメントを考慮した講義を行っている。これらの講義では、実践的なグループ討論、仕様書作成、レビュー等を行っている。さらに、課外活動であるロボコン道場では、ET ロボコンへの参加、再利用開発に精通した企業人による実習等も行ってきた。

本稿で具体的に紹介したリアルタイムカーネルを構築できる程度に、組み込み OS を理解するためには、ソフトウェアとハードウェアの両方について精通していなければならない。本授業で扱う並行性は、大規模開発の際、重要な理論であり、発見することが困難な誤りを産み出してしまう性質でもある。従って、本授業で開発するプログラムの規模は 1000 行前後と小さいが、大規模開発にとって重要な基礎力となると言える。

組み込み OS の授業は、著者にとって初回であり、教材開発と授業が同時並行であったため、改善の余地が多い。本授業の後半で、OS を設計するためのモデリングや、テスト仕様書等の演習を行うことができれば、より実践的な内容で実施できたのではないかなと思う。

## 6. おわりに

本稿では、筆者らの学科における組み込みソフトウェア教育事例について、組み込み OS を中心に紹介した。組み込みソフトウェアは学ぶべき領域が広く、教育が難しい分野であるが、大学 3 年終了時において、技術力、人間力のいずれについても、想定以上に効果的であった。システム開発の上流から下流、ソフトウェア、ハードウェアに関する知識を実践的に育成できたと自負している。本成果は、展示会への参加、企業の方々の講演等を通し、早い時期から網的意識を持たせたこと、

豊富な演習，実習課題により学習習慣が身に着いたことに起因していると推測する。今後，教材を充実させ，科目間の連携を強化し，より効果的な教育を行っていききたい。

Embedded System Education at Carnegie Mellon, ACM Transactions on Embedded Computing Systems, Vol. 4, No. 3, Pages 500-528, August (2005).

## 参考文献

- 1) 情報処理推進機構ソフトウェア・エンジニアリング・センター：組込みスキル標準 ETSS2008(2008). <http://sec.ipa.go.jp/ETSS/index.htm>
- 2) 大原茂之：特集 情報専門科目カリキュラム標準 J07：コンピュータエンジニアリング領域.1
- 3) 阿草清滋，西康晴，沢田篤史，鷲崎弘宜：特集 情報専門科目カリキュラム標準 J07：ソフトウェアエンジニアリング領域 (J07-SE)，情報処理，Vol. 49, No. 7, pp. 743-749(2008).
- 4) 小倉信彦，渡辺晴美：ロボットコンテストを利用した組込み教育の実践，情報処理学会論文誌，Vol. 49, No. 10, pp. 3531-3540 (2008).
- 5) 沢田篤史，小林隆志，金子伸幸，中道上，大久保弘崇，山本晋一郎：飛行船制御を題材としたプロジェクト型ソフトウェア開発実習，情報処理学会論文誌 Vol.50, No.11, pp.2677-2689, Nov. (2009).
- 6) 井上明，金田重郎：実システム開発を通じた社会連携型 PBL の提案と評価，情報処理学会論文誌，Vol. 49, No. 2, pp. 930-943(2008).
- 7) 松澤芳昭，大岩元：“産学協同の Project based Learning によるソフトウェア技術者教育の試みと成果，” 情報処理学会論文誌，Vol. 48, No. 8, pp. 2767.2780, (2007).
- 8) A. Abran, J. W. Moore, P. Bourque and R. Depuis (eds.): Guide to the Software Engineering Education Body of Knowledge (SWEBOK), 2004 Version, IEEE Computer Society, (2004).
- 9) The Joint Task Force on Computing Curricula: Software Engineering 2004 –Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, A Volume of the Computing Curricula Series, IEEE Computer Society and Association for Computing Machinery, (2004).
- 10) 坂井 弘亮：12 ステップで作る組込み OS 自作入門，カットシステム(2010).
- 11) 白川 洋充，竹垣 盛一：リアルタイムシステムとその応用 (システム制御情報ライブラリー)，システム制御情報学会，朝倉書店，(2001).
- 12) Qing Li: リアルタイム組込み OS 基礎講座，翔泳社，(2005)
- 13) Andrew S. Tanenbaum (著)，水野 忠則 (翻訳): モダン オペレーティング システム，ピアソン・エデュケーション・ジャパン，(2004).
- 14) Andrew S. Tanenbaum (著)，吉澤康文，木村信二，永見明久，峯博史 (翻訳): オペレーティングシステム，ピアソン・エデュケーション・ジャパン，(2007).
- 15) T - Engine フォーラム (著)，坂村 健 (監修): T - Kernel 標準ハンドブック，パーソナルメディア，(2003).
- 16) P. Koopman, H. Choset, R. Gandhi, B. Krogh, D. Marculescu, P. Narasimhan, J. M. Paul, R. Rajkumar, D. Siewiorek, A. Smailagic, P. Steenkiste D. E. Thomas, and C. Wang: Undergraduate