

工業高等学校における OS 学習支援環境の実践と評価

西野 洋介^{†,††} 田中 裕樹[†]
川口 貴弘[†] 早川 栄一^{†††}

本論文は OS 学習支援環境の工業高等学校における学習効果の評価の報告である。近年、工業高校では機器やネットワークの発達にともない、組み込み機器や制御系の教育を行っており、組み込みシステムを含めた OS の理解は重要な要素となっている。しかし従来のテキストなどの紙教材では OS の動作、構造のイメージがつかめず、OS 概念の理解が難しいという現状があった。これらの問題に対し、我々は OS の導入教育において、学習者が視覚的に理解することができる環境を実現した。本環境は、OS の導入段階において、OS 概念のイメージがつかめず、学習者の理解促進を妨げている点に着目し、可視化によって動作を具体化した教材を提供する。本環境は OS の概念からロボットなどの制御、組み込みまで連続的に教育支援を行うことができる。また、従来の OS 学習支援環境は学習者の理解度に応じた教材の設計が困難であり、教授者のコストが大きくなっている点に着目し、様々な学習教材を可視化できる設計を行った。さらに本環境を工業高等学校の情報系授業において教材として使用し、従来の教育方法と対比した結果、可視化および連続性の有効性を確認した。本論文では本学習環境の概要、工業高等学校における実践の結果および学習効果の評価実験において明らかとなった可視化による OS 学習支援の有効性、工業高等学校における OS 学習の考察について報告する。

Education and Evaluation of OS Education Support Environment in a Technical High School

YOSUKE NISHINO,^{†,††} YUKI TANAKA,[†] TAKAHIRO KAWAGUCHI[†]
and EIICHI HAYAKAWA^{†††}

This paper reports the evaluation of OS education support environment in a technical high school. The embedded system is taught in the industrial high school as the equipment and the network has been developed. And, the understanding of OS including the embedded system is important. However, the traditional paper-based textbook cannot help learners to grip the image of the operation of OS and the structure, and be difficult the understanding of OS concept. We develop the environment in order to alleviate difficulties which beginning-level students face, and to solve an issue of cost. In introductory courses of OS education, students often find difficulties understanding architecture and operation of OS. This environment can provide learners continuous educational support from the concept of OS to the control and building in such as robots. Also, the new environment helps to reduce the implementation burden of teachers. We administrated and evaluated this environment in a technical high school. In this report, we describe summary of this learning environment, and effectiveness and consideration of the visualized OS learning support environment.

1. はじめに

近年、初等中等教育における情報技術の理解は日本において重要な課題であり、これからの技術立国を支える重要な要素であると認識されている。特に、高等

学校においては、教科「情報」の実施に代表されるように、情報教育が推進されている。

このような状況の中で、工業高等学校（以下工業高校）では教科「情報」の導入以前より「情報技術基礎」として情報教育を実施し、教科「情報」の単位を「情報技術基礎」で代替している。工業高校は普通高校に比べ、実験・実習も多く行い、専門的な知識を持った生徒を育成している。さらに、一部の工業高校では、情報科を設置し、高度な情報教育を行っている。

また、今日では、機器や装置の計算機による制御は普通に行われていて、このような制御プログラムの開

† 拓殖大学大学院工学研究科

Graduate School of Engineering, Takushoku University

†† 東京都立府中工業高等学校

Fuchu Technical High School

††† 拓殖大学工学部

Faculty of Engineering, Takushoku University

発を行う組み込みシステム技術者の需要は高い¹⁾。この需要に対して、工業高校の卒業生は、普通高校を卒業した生徒に比べて、情報技術に関する基礎能力が高いことから、産業界や大学などから注目されている。この結果として、工業高校における情報技術教育は今後のわが国の産業界において重要な役割を担っている²⁾。

機器やネットワークなどの発達により、現在このような組み込み機器のプログラム開発においては、組み込み機器にもオペレーティングシステム（以下 OS）を用いることが多い。組み込み機器では、サイズや性能、コストに応じて、OS を選択していくことから、技術者は、OS の概念レベルの理解が最低限の知識として必須となってきた。OS の知識はソフトウェアの開発や動作に大きな影響を与えるからである。

このように情報技術、特に制御や組み込みといったシステムソフトウェア分野を学ぶ工業高校の生徒にとって、OS の理解は情報工学における重要な要素の 1 つであり、技術者の育成は重要な課題である。しかしこれまで、工業高校を含めた高等学校では OS の学習には時間的、環境的な側面からも難しいという問題があった。その原因は、主に次の 2 点がある。

- OS の構造、動作が直観的にイメージしにくく、学習者が理解しにくい。
- 学習者の理解度に応じて導入段階から実装段階まで連続的に学習支援を行える環境が少ない。

本研究では、この問題に着目し、OS 学習支援環境の開発、および工業高校における実践と学習効果の評価、分析を目的とする。これまで我々は、可視化を用いた OS 学習支援環境「港」を提案してきた³⁾。OS 学習支援環境「港」は、OS の複雑な動作を具体化し、アニメーションや配色の工夫によって OS の動作を可視化する特徴を備えている。本研究では、この「港」を学習支援環境として用いて、学習効果の分析を行った。

以下、2 章では工業高校における OS 教育の現状について述べる。3 章では OS 学習環境の要件について、4 章では既存の OS 学習支援環境とその問題点について、5 章では OS 学習支援環境「港」の概要、コースウェア、全体構成について述べる。6 章では工業高校における「港」の実践の詳細および得られた知見、「港」の学習効果の評価実験および得られたデータ、評価について述べる。7 章は本研究を通して得られた考察およびまとめである。

2. 工業高校における OS 教育の現状

工業高校では教科「情報」を代替する「情報技術基

礎」が必修であり、全生徒が履修する。特に、情報科、電子科、電気科など、電気系の専科においては「情報技術基礎」に加え、専門座学、実習全般において情報技術に関する教育を行っている。近年では組み込み機器の発達、普及や、ロボットコンテストなどが数多く開催されているように⁴⁾、制御プログラミングや組み込みシステムといった分野のニーズが高まっており、そのための OS 学習も重視した教育を行っている。

工業高校における OS 教育の目的は、OS の概念を理解することにある。OS の本質である資源管理や仮想化を理解することで、コンピュータの動作原理や、ハードウェア、ソフトウェアをシステムとして総合的に理解することを目標としている。具体的には次の学習項目を設定している。

- タスクスケジューリングの学習から CPU の仮想化を学ぶ。
- メモリ割当ての概念理解から、有限の資源を有効に利用する概念を学ぶ。
- リアルタイムスケジューリングの概念理解を通して、組み込みシステムでは、時間的な制約をともなった資源管理の概念や実時間性の概念を学ぶ。

これらの概念学習は、OS の根本的な概念の習得という点で重要である。また、実時間性や有限な資源管理の概念の学習は、組み込みシステムの理解には必須である。OS が提供する資源管理、および仮想化の概念を理解しておくことで、今後、制御プログラミングや組み込みシステムなどの実習を行った際、有限な資源を意識することが容易になることが期待できる。このことから、工業高校では、OS の動作を題材として、組み込みシステムや制御などを含めたシステムソフトウェアの概念理解を行える教育への要求がある。

その一方で、工業高校では、ものづくりや産業教育の観点から、応用力をともなった、実習による実技を重視している。これに対応するには、実技指導だけではなく、理論をともなう教育が必要であり、現場においても理論から実技への連続的な教育環境の要求が高い。

このような要求に対して、工業高校を対象とした既存の学習教材や実習環境は、大きく 2 つに分けることができる。

- アニメーションなどの可視化を用いたもの

アニメーションによる可視化教材は、OS の導入教育において有用性が評価されている¹⁵⁾。しかし、アニメーションによる導入教育から、教育用 OS などを用いた実装教育を連続的に支援する環境がなく、学習者の負担や教育者の教材作成のコスト増大の要因になっ

ている。

- ワンチップボードやロボットを用いたもの

工業高校では実習や演習において、ワンチップボードやロボットによる学習支援環境を用いて、制御プログラミングなどの学習を行うことが多い。しかし、現行の制御プログラミング教材は、制御単体での学習支援のものが多く⁹⁾、OS の理解を含めた統合的なシステムソフトウェアの教材がない。この結果として、生徒は実装的な制御プログラミングと概念レベルの理論的な OS を連続的に理解することができず、OS 全体の学習効果が半減してしまっている。

これらの現状から、工業高校の OS 教育における学習環境が備えるべき要件について、次章で述べる。

3. OS 学習環境の要件

工業高校における OS 教育の中で、OS 学習環境に求められる要件を次に述べる。

(1) 視覚的、直観的に学習できる環境

学習者が、仮想化されている資源を具体的に可視化することで、直観的な理解を可能にする環境が必要となる。特に、内部動作をアニメーションによって表現できることが重要である。

従来の OS テキストによる教育環境では、OS の動作を動的に表示できるものはなく、静的に表現することしかできていない。しかし、OS は様々な要因によってつねにその状態が変化し（たとえばタスクの実行状態など）、状態の遷移を把握することは難しい。OS の導入教育においても状態遷移のイメージを理解することが大きな難関となっていた。

さらに、OS が提供する資源の仮想化にともなう、不可視性の問題を解決する必要がある。たとえば、CPU の仮想化の手段としてタスクの概念がある。タスクの概念を理解する場合、タスクの実行状態の遷移による、CPU の仮想化や資源管理の概念を学びたい。しかし、このような状態遷移は、システム内部で動作し、アプリケーションの動作やユーザインタフェースに隠れてユーザの目に見えない。その結果、学習者は、その動作や資源管理の概念をイメージすることが困難となっている。

(2) 導入から実装まで統合的に学習支援できる環境

OS 学習を概念から実装まで一貫して行える統合的な環境が必要である。OS の学習は、概念学習と実装学習の両方を繰り返し行うことで理解を確実にすることができる。しかし、従来の OS 学習を対象とした学習環境は、導入学習もしくは実装学習のどちらかをサポートして、導入から実装まで連続的に支援でき

ない。この結果、学習者は理論と実装とを、連続的にとらえることが難しくなっている。さらに、少ない授業時間内で学習環境に慣れなければならず、限られた授業時間を有効に活用することができない。また、教材を用意する教授者にとっても、可視化の粒度を変えた可視化や、スケジューリングアルゴリズムの変更、ハードウェアの追加、学習進度に応じた学習環境の構築を容易にする環境が求められている。

(3) 教材を操作できる環境

学習者が能動的に学習を行うためには、教材は、自ら教材を動かすことのできるように「操作可能」になっていることが望ましい。特に工業高校においては、座学による授業に比べ、生徒の実習科目への興味、関心の高さという点でも、実習を重要視している。また、自ら動かし問題を解決することは、ゲームに似た性質もあることから、学習者にとってゲーム感覚で学習することになり、学習効果を期待できる。

4. 既存の OS 学習支援環境と問題点

本章では、既存の OS 学習支援環境について述べる。既存の学習支援環境は大別して可視化を用いたものと、ロボットを用いたものに分けられる。

OS の構造を可視化によって支援する試みとして、SOsim¹³⁾ や VisualOS¹⁴⁾、RCOS.Java¹⁶⁾ がある。これらは、ユーザがタスクを生成しその動作をアニメーションなどで表示し、OS の概念を学習することを目的とした教材である。OS の導入教育におけるアニメーションの利用については、その有効性が報告され¹⁵⁾、学習効果が期待できる。しかし、その一方で、システムとして単体で閉じていることから、拡張性や連続性、他教材との協調、ハードウェアの可視化という点は考慮されていない。

また、工業高校の教育現場では、すでにロボット教材を用いた実習環境の構築が非常にさかんであり、多くの教材や環境が混在している。このような現状から、教授者としては、OS の概念学習とロボットをうまく連動させたいという要求が高い。しかし、SOsim¹³⁾ や VisualOS¹⁴⁾ に代表される既存の OS 学習支援環境は、環境として完結しており、導入教育から先の実践的なシステムソフトウェア教育までは支援していない。

一方、制御プログラミングや組み込みシステムの導入教材として、TeC⁹⁾ や LEGO Mindstorms⁵⁾ を利用したものがある。TeC は学習者がボードを組み立てるところからはじめ、組み込み CPU 上で動作するプログラムを構築し、理論学習を行うというボトムアップスタイルの学習形式をとっている。また、legOS⁶⁾ や

brickOS⁷⁾ はオープンソースで開発されている LEGO Mindstorms 上で動作する OS である。これは小サイズで実装された組み込み OS だが、LEGO を操作するアプリケーション開発が主であり、OS そのものが教材としては作成されていないことから、コードが読みにくい。また、リモートデバッグを含めたデバッグ環境が整備されていないことから、デバッグがしにくく、初等中等教育には敷居が高い。TeC, brickOS 両者とも制御プログラミングの教育に視点をおき、OS の概念やアプリケーションまでを含めた統合的な組み込みシステム教育までは支援していない。

5. OS 学習支援環境「港」

我々が行っているプロジェクト「港」^{10)~12)} はシステムソフトウェア、特に OS の学習において、導入から実装まで連続的に学習支援を行う環境である。次に「港」の概要について示す。

5.1 特徴

OS 教育における「港」の特徴について述べる。本環境では、上述の 3 つの問題点に対し、次の方針を定め、設計を行った。

(1) 可視化による学習支援

OS の理解において動作、構造の可視化は重要な要素であるとともに理解促進の有効な手段である。そこで詳細な実装を排除し、概念を具体化することが理解を促進させると考える。

また、OS は複雑な状態遷移を持つ。これに対し、紙教材ではその状態遷移を追うことが難しい。本環境では、OS 内部の状態遷移をアニメーションとして表現することで、OS の動作を可視化する。さらに、実行状態を色表現によって可視化する。具体的には、タスクの実行状態を信号色にすることで、学習者は直観的にタスクの状態およびその変化を把握できる。さらに、システム全体で、色表現の統一で表現することによって、OS 内部の各モジュール間での協調動作を追跡しやすくしている。

本環境では、工業高校における OS 教育の現状から、タスク管理、メモリ管理についての可視化を実装した。タスク管理では、基本的な概念および、制御に用いられるものとして、次のスケジューリング機能を可視化している。

- ラウンドロビン
- 優先度ラウンドロビン
- Rate Monotonic
- Earliest Deadline First

また、OS 内部の動作として、次の 2 点を可視化対

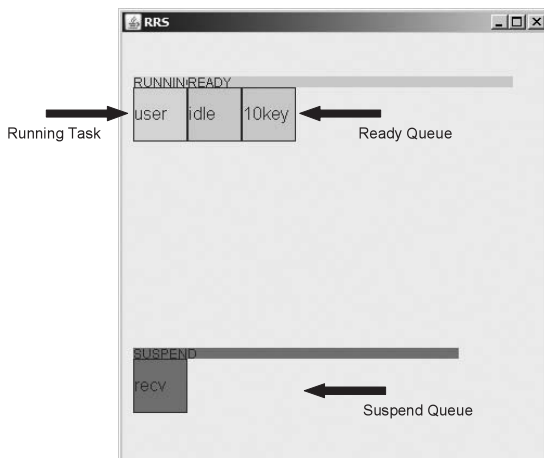


図 1 スケジューリングの可視化画面
Fig.1 Task scheduling animation.

象とした。

- タスクディスパッチ
- タスク実行状態グラフ

タスクスケジューリングの可視化では、タスクを箱で表現し、3 つの実行状態 (Running, Ready, Suspend) を信号色 (Running-緑, Ready-黄色, Suspend-赤) で表現している。ラウンドロビンスケジューリングの画面構成を図 1 に示す。

メモリ管理では、基本的な概念として、次のメモリ割当ての機能を可視化している。

- First Fit
- Best Fit
- Worst Fit

また、組み込み機器においても、メモリ管理ユニットを持った CPU が一般的になりつつあることから、次のページング機能および、ページ割当ての概念についても、可視化を行った。

- First In First Out
- Least Recently Used

(2) 連続的な学習支援

本環境では、あらかじめ複数の可視化教材をコンポーネントとして用意しておき、教授者はそれらの可視化部品を学習者の理解度に応じて組み合わせることで、学習項目や、理解度に適当な可視化環境を構築することができる。

さらに、動作のログフォーマットをシステム全体で統一し、OS やプラットフォームの違いを吸収できるスタブを準備している。これによって、可視化の変更を最小限に抑えたままで可視化対象となるコンポーネントを差し替えることを可能とする。「港」では 3 つの

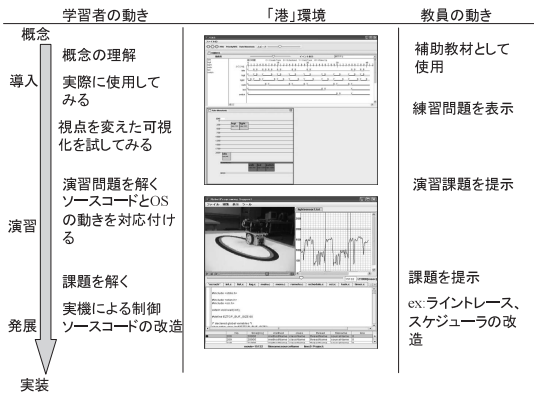


図2 「港」のコースウェア
Fig. 2 Courseware of "MINATO."

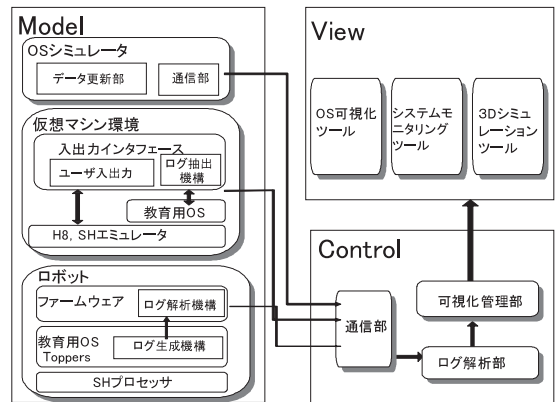


図3 全体構成
Fig. 3 Overall structure.

コンポーネント (OS シミュレータ, 仮想マシンと教育用 OS, ロボットと教育用 OS) を用意している. これらを同じ可視化プラットフォームで実行できることから, 導入教育から実装教育までの連続的な支援が可能である.

(3) 操作可能な学習支援

本環境は OS シミュレータやロボット, 教育用 OS を提供する. これらは学習者が実際に改良や操作を行うことができる.

OS シミュレータでは, 粒度を変えた可視化やアニメーションの変更など学習者が操作することができる. また, システムに統合した改造可能なロボットを提供することによって, 学習者が組み込み機器そのものについて, 積極的に興味を維持し, 理解が促進することを可能にしている. また, 仮想マシン上で教育用 OS を動作させることで, 安全な動作および, ロボットがない在宅での学習環境を提供している.

5.2 利用対象者とコースウェア

本教育支援環境では次の 2 通りの利用形態を想定している.

- 講義支援ツールとして使用

OS の講義を行う教授者を対象とし, 講義を補う補助教材として使用する.

- OS の独習ツールとして使用

学習者を対象とし, 独習を行う際や自宅での自習に使用する.

本環境を使用した講義計画の一例を図 2 に示す. 本環境は, 概念からソースコードに触れる実践学習まで一連して支援することができる. 概念学習から実装学習へとトップダウンに学習を進めていくことで, 概念から実践へと理解していく形態をとっている.

(1) 導入段階

導入段階において OS の概念を学習する. OS の動作, 構造を具体化し, OS の主要機能であるタスク管理, メモリ管理をアニメーションによって可視化することで, テキストなどでは理解しにくい, 状態遷移のイメージを理解する. これにより, OS の仮想化や資源管理の概念の理解を支援する.

(2) 演習段階

演習段階においては仮想マシン上で実際に動作する OS のログを可視化する. OS のソースコードを学習者が改造や機能の追加を行い, その結果を可視化することで, ソースコードと OS の動きを連続的にイメージできるようになる.

(3) 発展段階

発展段階では, ロボットを用いて, OS および開発環境のサービスを用いた高度な制御とともに OS を学ぶ. 具体的には, ライトトレースや, 簡単なコース走行などを題材とする. また, ロボットプログラムを支える OS などのシステムプログラムがどのように動作しているのかを学習することができる. 特に, ロボット内部の OS の処理の様子を可視化することで, ロボットの挙動と OS の動作を連続的に理解することが可能となる. また, ロボット OS はタスクスケジューラの差換えが可能である. これらのスケジューリングアルゴリズムの違いによるロボットの挙動と OS 内部の様子を連動して学習することができる.

5.3 全体構成

本環境の全体構成を図 3 に示し, 各部の設計について述べる.

本学習支援環境は, 学習の各フェーズでの利用を可能にするために, システム全体を, MVC モデルをもととした 3 部分のコンポーネントとして構成してい

る．この全体構成において，可視化の対象となる 3 つの Model について述べる．

● OS シミュレータ

OS シミュレータは OS の動作，機能を擬似的に再現するものである．OS の動作を具体化し，OS 初学者の混乱要素である複雑な動作を排除し，概念学習に必要なイメージの理解を促進することができる．

● 仮想マシン (H8 および SH3) と教育用 OS

発展学習段階の教材として，H8 仮想マシン，SH3 仮想マシンと学習用に開発した OS を提供する．学習者は仮想マシン上で教育用 OS のソースコードの改良や，仮想マシンの動作と OS の可視化を連携させ，ソースコードと動作を対比することができる．

● 自走ロボットとロボット用教育 OS

実装学習段階の教材として自走ロボットとロボット用教育 OS を提供する．自走ロボットはタッチセンサ，可視光センサなどを備えており，無線 LAN によってコントロールできる．また，ロボット用教育 OS は，ロボットの制御に必要なハードウェア依存部分をファームウェアとして分離し，OS 学習の要素を独立している．自走ロボットの動作をロギングし，内部状態を可視化することで挙動，ソースコード，内部動作を連続的に理解することができる．さらに，実践的な環境を提供するために，組み込み OS である Toppers も移植可能となっている．

これらの教材は，学習者の理解度に応じて差換えが可能であり，同じ可視化プラットフォームで連続的に利用することができる．従来の環境に比べて，可視化のユーザインタフェースや，その他共通部分への修正の手間をかけずに，可視化対象のモジュールの差換えが可能になっている．これによりユーザインタフェースの違いによる学習者の混乱を低減させるとともに，新たな教材作成のコストを削減することが可能となる．これらの Model の差換え可能部分が，既存の OS 学習支援環境との大きな違いである．

また，可視化はより直観的な理解を得られるよう，OS の状態遷移を配色によって表現する．特に，信号色を用いることで，学習者は直観的に理解することができるようになる．

図 4 に Rate Monotonic タスクスケジューリングのアニメーション，図 5 にタスク実行状態のグラフのスクリーンショットを示す．図 4 は各タスクを具体化し，信号色によって実行中，実行待ち，停止中を表現している．また，アニメーションによってタスクスケジューリングの様子を可視化し，スケジューリングアルゴリズムの違いを表現している．図 4 は周期によ

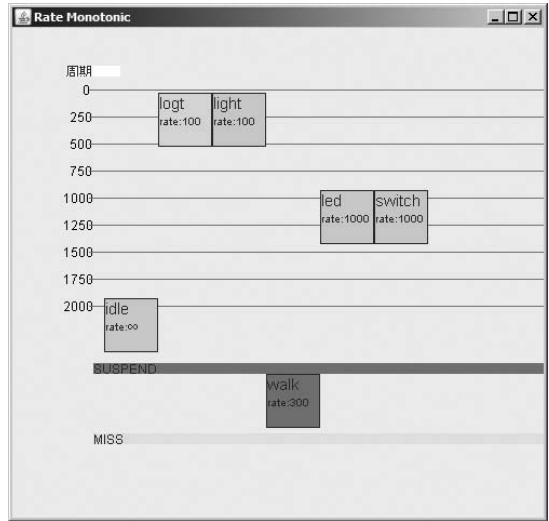


図 4 タスクスケジューリングのアニメーション
Fig. 4 Task scheduling animation.

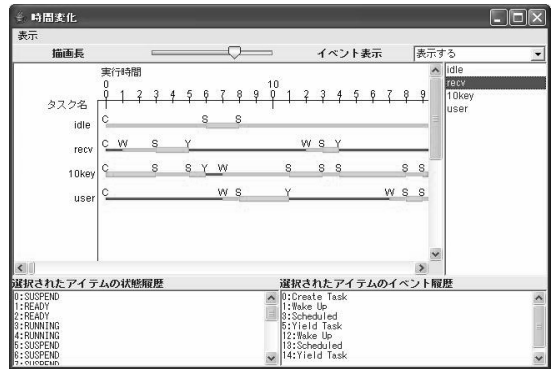


図 5 タスク実行状態のグラフ
Fig. 5 Graphs of task states.

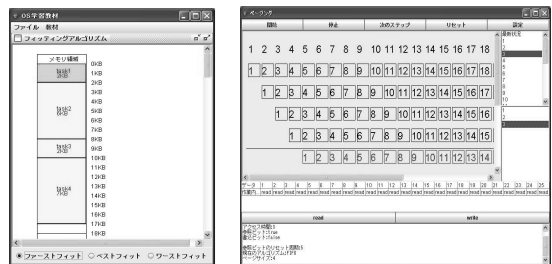


図 6 メモリ管理のアニメーション
Fig. 6 Memory management animation.

てタスクの配置の高さがアニメーションによって変化し，次に実行されるタスクを直観的に理解することが可能である．

図 5 ではスケジューリングアニメーションと連動して，すべてのタスクの実行状態を，時間軸をもとにグラフ表現することで，より視覚的，直観的に理解する

ことができる。

図6にメモリフィッティング、ページングのスクリーンショットを示す。タスクと同様にアニメーション、信号色によって直観的に可視化している。さらに、タスクスケジューリングの可視化と連動することによって、タスクとメモリの対応の概念を理解することが可能である。

6. 工業高校での実践

OS学習において、本環境のOS学習導入段階における学習効果の評価を得るため、実験授業を工業高校において行った。実験授業の対象は1学年および2学年の情報技術科、電気科の生徒である。

実験授業の詳細は次のとおりである。

- 対象校：東京都立府中工業高等学校
- 科目名：情報技術基礎（1学年）および課題研究（2学年）
- 対象人数：4クラス（1学年3クラス、2学年1クラス）計140名
- 期間、回数：4コマ（1コマ50分）
- 使用テキスト：東京電機大学出版「情報技術基礎」、オーム社「基本情報技術者標準テキスト」
- 生徒の前提知識：OSの構造、機能に関する知識はないが、OSの利用方法は理解している。

実験授業では、システムソフトウェアの単元においてOSの概念を理解させることを目標とした。4クラスとも表1に示すカリキュラムで実践した。

実験授業のねらいは、資源管理や仮想化の理解である。この概念の理解によって、実装学習（制御プログラミングや組み込みシステムなど）を行った際に、少ない資源（PCに比べて貧弱なハードウェアなど）や時間的な制約（リアルタイム性）などの制限の中で、資源管理を意識した効率的なソフトウェアの設計につなげることを意図している。また、スケジューリングやメモリ割当ての理解を通して資源管理や仮想化の概念を、リアルタイムスケジューリングの理解を通して、時間の制約に対する効率的な資源管理の概念を意識させる。

実験授業は、2つの群に分けて行った。2つの群は出席番号の前半、後半で分けており、学力差は均等である。次に実験群と対照群の詳細を示す。

● 実験群（グループA）

実験群は本環境を使用した群である。実験群はテキスト、板書による本環境の簡単な説明を行った後、本環境による自主的な学習を行い、教員は机間巡視を行いながら適時質問に答える。おおむねの生徒が納得し

表1 実践カリキュラム

Table 1 Curriculum of experimental lesson.

授業回数	内容
1回目	タスク管理とは ラウンドロビンスケジューリング 優先度付きラウンドロビン
2回目	リアルタイムスケジューリングとは レートモニタリング
3回目	メモリ管理とは メモリフィッティングの違い
4回目	ページングとは ページングアルゴリズムの違い



図7 授業の様子

Fig. 7 Screenshot of lesson.

たようであれば次の学習項目へ進むといった授業展開を行った。

● 対照群（グループB）

対照群は従来のテキストおよび板書によって講義を行った。

今回行った実験授業では時間の制約上、4.2節で述べた「港」のコースウェアにおける、(1) 導入段階におけるOSの可視化、(2) 演習段階におけるスケジューリングアルゴリズムの違いの可視化、(3) 発展段階におけるロボット内部のリアルタイム動作の可視化を実践した。本授業では、実際のロボットは教材の数の都合もあり利用していないが、可視化時に利用しているデータ類は、すべて実機の動作結果を用いている。ロボットの動作ログや、教育用OSのスケジューリングアルゴリズムの差換えによる挙動の違い、また、ロボットによるライントレースの動作ログにおいて、リアルタイム処理のあり/なしによる動作の違いなどの例を可視化することで、従来の可視化環境に比べて、概念から実装への連続的な可視化を単一環境で行っている。

授業風景を図7に示す。

6.1 実験授業から得た知見

実験授業における「港」の運用から得た知見を次に述べる。利点は次のとおりである。

● アニメーションによる効果

「港」を利用した授業形態はテキストと板書を使用した授業形態に比べて生徒の興味や関心が高く、普段

は意欲のない生徒も能動的に授業を受けるようになる。

● 操作できることによる効果

「港」は講義の補助教材としてだけではなく、生徒が自ら操作して学ぶことができるので、ゲーム感覚のように集中して理解しようとする意欲が見られた。たとえば、タスクの初期状態や、デッドロック、リアルタイム処理が破綻する例などを実験的に設定することで、問題を自ら解決していく意欲が見られた。

● 連続的な学習支援による効果

OS シミュレータの動作ログも、ロボットの動作ログも、同一環境で可視化を行っているため、実際のロボットを用いた場合にも、ユーザインタフェースの違いから起こる混乱を最低限に抑えることができる。また、同一環境で可視化することで、ロボットで動作する OS も、普段利用している PC の OS も資源管理の原理は同じであることがイメージできた。

このような利点が観察できる一方で、次の問題も発生した。

● 確実な理解、習得

板書による授業ではないため生徒はノートにメモをとることなく授業に参加していることから、操作そのものに集中してしまい、その場限りの理解に終わってしまう可能性がある。これについては教員の授業展開によって問題を解決することが可能である。

● 授業の時間的な制約

高校の授業では、その単元に多くの時間を費やすことができないことから、授業内容に時間の制約が大きく関わってくる。より深く学習しようとしている意欲を見せる生徒にとっては、時間的制約により授業時間の中で教えることができないため、学習効率が悪くなってしまうことがある。これについては実習科目などの連続的な授業時間を利用することで問題を解決することができる。

6.2 評価テスト

実験授業において評価テストを行った。講義内容に対する理解目標は、資源管理や仮想化の概念的な理解、アルゴリズムの違いによる挙動の理解である。確認テストを通してこれらの達成度を測る。これらの理解は前述した工業高校における OS の教育目標であり、OS を学ぶうえでの重要な概念である。

テストは計 4 クラス、140 名のデータを得た。実験群のグループ A および対照群のグループ B 両者と同じテストを行った。テストの問題を付録に示す。

テストの内容は、3 つのタスクスケジューリングアルゴリズム、

- Round Robin (RR)

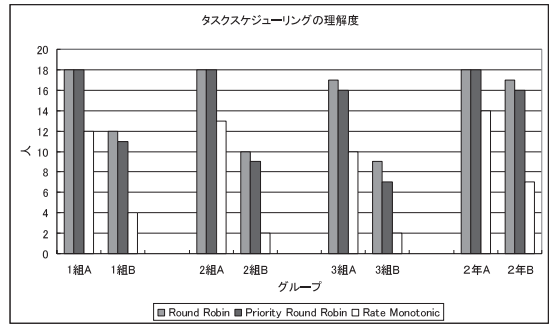


図 8 タスクスケジューリングのテスト結果
Fig. 8 Task scheduling result.

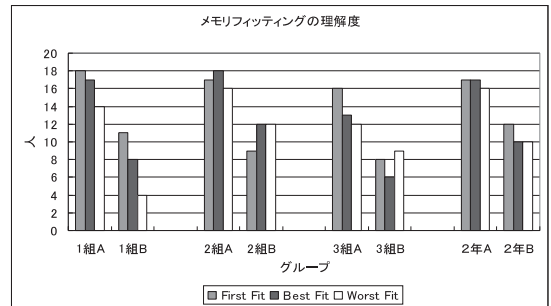


図 9 メモリフィッティングのテスト結果
Fig. 9 Memory fitting result.

- Priority Round Robin (PRR)
 - Rate Monotonic (RM)
- 3 つのメモリフィッティングアルゴリズム、
- First Fit (FF)
 - Best Fit (BF)
 - Worst Fit (WF)
- 2 つのページングアルゴリズム
- First In First Out (FIFO)
 - Least Recently Used (LRU)

についての概念に関する理解の確認問題である (図 11 では上記括弧内の略語で表記している)。

各学習項目におけるグループ別のテスト結果を図 8 , 図 9 , 図 10 に示す。なお、縦軸は正解人数である。各学習項目別の結果を図 11 に示す。

テスト結果から実験授業の評価を行った。テストの結果から、各学習項目において本環境を使用したグループ A のほうが高得点であることが読み取れる。それに対して、すべての項目において、グループ B の正答率は低い。これはその他の意見や授業中の実践においての知見から得られたように、PC を利用して自ら動かすことのできる環境を用いたことにより、能動的に学習でき、結果として、従来のテキストによる授業に比べて生徒の興味や集中度が上がっていることが

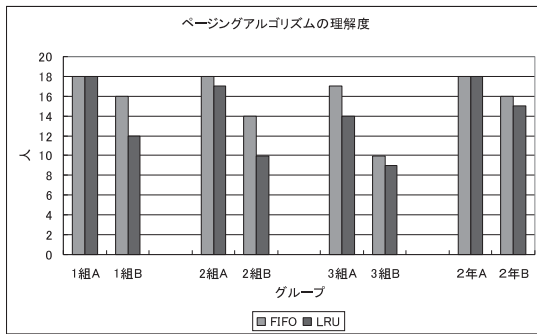


図 10 ページングのテスト結果
Fig. 10 Paging result.

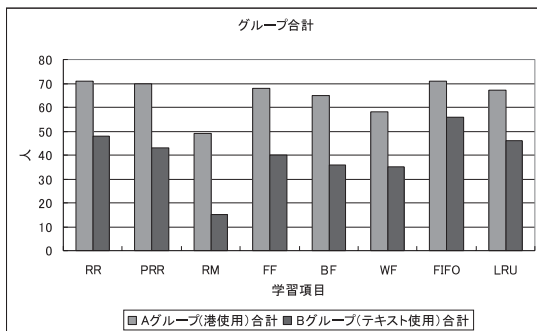


図 11 学習項目ごとのグループ別結果
Fig. 11 Result of education elements.

表 2 アンケート項目
Table 2 Item of a questionnaire.

質問番号	内容
A	テキストに比べて分かりやすいか
B	アニメーションは分かりやすいか
C	配色による状態表現は分かりやすいか
D	レイアウトは分かりやすいか

回答結果
1 2 3 4 5
悪い 少し悪い 普通 少し良い 良い

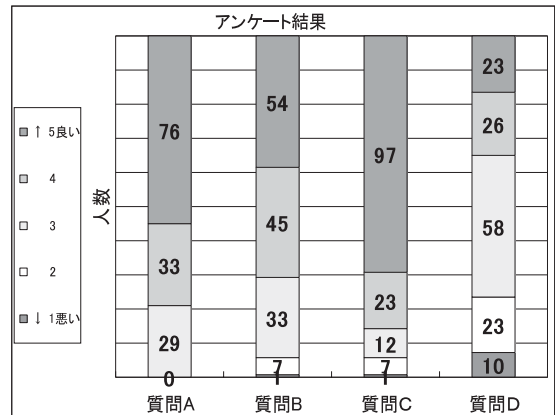


図 12 アンケート結果
Fig. 12 Questionnaire result.

原因であると分析できる。

次に項目ごとの評価の分析を次に述べる。

- タスクスケジューリングに関しては、ラウンドロビン、優先度ラウンドロビンともにグループ B の正解率約 60% に対して、グループ A の正解率は 95% 以上である。また、Rate Monotonic に対してもグループ B に対して大幅に正解率を上げていることが読み取れる。これはリアルタイムスケジューリングの概念は、テキストでは表現しにくく、アニメーションによって直観的に理解できたからであると考えられる。つまりタスクスケジューリングの状態遷移を、アニメーション化することが理解を促進する大きな要因であると分析できる。
- メモリ管理に関しても同様に、動作の様子をアニメーションによって表現しているため、アルゴリズムの違いによる動作の様子を生徒が直観的に理解していると分析できる。

6.3 アンケート結果

テストとともに、アンケートによる調査を行った。アンケートの項目を表 2 に示す。

これらのアンケート調査の結果を図 12 に示す。

また、その他として次の意見が得られた。

- アニメーションによる動作がよかった。
- 黒板の説明より分かりやすかった。
- 状態の色分けが分かりやすかった。
- 自分で操作できたので、いつもより集中した。
- 様々な例を試すことができてよかった。
- ロボットの中の処理の様子が分かった。
- 実際にロボットを使うイメージがわいた。
- 決まった時間で処理ができないと大変なことになるイメージが分かった。
- OS の働きが分かったので、ほかのコンピュータの授業内容とつながりが分かった。
- 今行っている実習（ワンチップボードによる制御プログラミング）と連動して使えるようにしてほしい。
- もう少しレイアウトを使いやすくしてほしい。

次に、アンケート結果から実験授業の評価を分析する。

アンケート結果から質問 A, B, C に関して 5: 良い, 4: 少し良い, の割合が 70% 以上を占めていることが分かる。これは前述した本環境の設計方針の有効性を実証していることと分析できる。しかし、質問 D のレイアウトに関しては、3: どちらともいえないが

特に多く、その他の意見からも分析できるように、画面構成やレイアウト、ユーザインタフェースの再設計を検討する必要がある。

また、その他の意見から得られた結果を見ても、実際のロボットの動作ログを可視化することで、OS の資源管理における実時間の意識の重要性を意識付けることができた。また、アニメーションや色の表現による可視化によって従来のテキストに比べより直観的に理解できるようになっていることが読み取れる。また、従来までは、情報系の他の教科や分野をそれぞれ個別に理解、認識していた生徒が、OS の概念を理解することで連続的に考えることができるようになったと考えられる。

7. 考察とまとめ

本研究では、可視化による OS 教育支援環境を工業高校において運用し、実験授業を通して評価、考察を行った。評価の分析によって次の知見を次に示す。

- アニメーションの有効性を確認

工業高校において OS を主としたシステムプログラミング教育において、アニメーションによる可視化の有効性を確認した。特に、従来のテキストを主とした授業に比べ、アニメーションによる状態遷移の可視化が有効であることを確認した。

- 操作ができる学習環境の有効性を確認

生徒が自ら学習支援環境を使用することで、従来の講義形態に比べ、生徒の授業に対する興味関心が向上し、能動的な学習を行えることを確認した。これらの結果は特に学力の低い生徒や、学年が低い生徒に対しては大きな効果を得ることができた。

- 連続的な学習支援の有効性を確認

タスクスケジューリングの概念からスケジューリングアルゴリズムの違い、ロボットの動作とリアルタイム処理の例までを、実際の事例データを用いて、単一環境で可視化することで、本学習支援環境の特徴である連続的な学習支援の実践を行った。特に、すべての実験データに実際に走行したロボットのログデータを用いることで、ロボットの内部動作のイメージをつかむことができた。その結果、得られた知見として、単一環境による連続的な学習支援を行うことで、学習者は学習環境の違いによる混乱を起こすことなく、概念学習から実装学習へとスムーズに移行することができることを確認した。従来、工業高校の教育現場において、システムソフトウェアを連続的に支援する環境がなく、教員側も様々な環境を使いこなす必要があり、大きな課題であったが、得られた結果からも、単一環

境による連続的な学習支援は有効であると考えられる。

- 工業高校における OS 教育の重要性を確認

OS の概念を理解することで、並行して行っているハードウェア教育やソフトウェア教育、実習など、従来まで個別のものとして理解していたものが、連続的に、コンピュータシステム全体の理解につながり、大きな効果を得ることを確認した。

- 可視化教材の問題点を確認

配色やレイアウトによって学習効果が大きく変わることを確認した。特に可視化においては直観性が重要であり、レイアウトや配色についての改良の必要性を確認した。

本実験を通して、可視化による本教育支援環境は、従来のテキストを用いた場合よりも、直観的に理解を進めることが明らかとなった。さらに従来の OS 学習支援環境にはなかった連続的な学習支援を実現し、工業高校の OS 学習において効果を得ることができた。

今後は、アンケート結果によって明らかとなった、レイアウトの再設計および、今回の授業で得られた成果をもとに工業高校だけでなく普通教科「情報」で活用可能なコースウェアの作成、環境の整備を行う予定である。

謝辞 本研究の実験にご協力いただいた東京都立府中工業高等学校の皆様へ感謝いたします。

参考文献

- 1) 経済産業省商務情報製作局：2005 年度版組込みソフトウェア産業実態調査報告。
- 2) 情報処理学会情報処理教育委員会：日本の情報教育・情報処理教育に関する提言 (2005).
<http://www.ipsj.or.jp/12kyoiku/proposal-20051029.html>
- 3) 大角圭吾, 小久保政樹, 西野洋介, 早川栄一：システムソフトウェア教育支援環境「港」における実装レベルの OS 学習支援システム, 情報処理学会コンピュータと教育研究会, CE-078, pp.9-14 (2005).
- 4) Lund, H.H.: Robot Soccer in Education (1999). <http://legolab.daimi.au.dk/Publications/pdf/SocEdu.pdf>
- 5) LEGO Mindstorms:
<http://mindstorms.lego.com/japan/>
- 6) legOS: <http://www.noga.de/legOS/>
- 7) brickOS: <http://brickos.sourceforge.net/index.html>
- 8) 文部科学省：高等学校学習指導要領：工業 (2002).
- 9) 重村哲至, 守川和夫, 力規晃, 新田貴之：教育用マイコンボードを用いた HDL 演習環境の実

現, 情報処理学会研究報告 2005-CE-78, pp.43-48 (2005).

- 10) Nishino, Y. and Hayakawa, E.: Development of an OS Visualization System for Learning Systems Programming, *HCI2003 International Conference on Human Computer Interaction* (2003).
- 11) 川上亮太郎, 田中裕樹, 大角圭吾, 西野洋介, 川口貴弘, 早川栄一: システムソフトウェア教育支援環境「港」における教育用組込み OS の開発, 情報処理学会 OS 研究会 2006-OS-101, pp.99-104 (2006).
- 12) 田中裕樹, 川口貴弘, 西野洋介, 早川栄一: ロボットを用いた組込みシステム学習環境における学習支援環境の開発, *SWEST8*, pp.59-66 (2006).
- 13) Maia, L.P., Machado, F.B. and Pacheco, A.C.: A constructivist framework for operating systems education: A pedagogic proposal using the Sosim, *ACM SIGCSE Bulletin, Proc. 10th annual SIGCSE conference on Innovation and technology in computer science education ITiCSE'05*, Vol.37, Issue 3 (2005).
- 14) Hill, J.H. and Gokhale, A.S.: Education: Visual OS: Design and implementation of a visual framework for learning operating system concepts, *Proc. 43rd annual southeast regional conference*, Vol.1, ACM-SE 43 (2005).
- 15) English, B.M. and Rainwater, S.B.: The effectiveness of animations in an undergraduate operating systems course, *Journal of Computing Sciences in Colleges*, Vol.21, Issue 5 (2006).
- 16) Jones, D. and Newman, A.: RCOS.java: A simulated operating system with animations, *Proc. Computer-Based Learning in Science Conference*, Rep. Tchecha (2001).

付 録

付録として評価実験で行った評価テストの問題の一部を示す.

1. タスクスケジューリング
あるスケジューラではスケジューリングは1ミリ秒ごとに発生し、(1)~(3)のアルゴリズムを利用できるものとする。このとき、表1のようなパラメータを持ったタスクがあると、1~3ミリ秒後における実行中のタスクをそれぞれのアルゴリズムについて答えよ。

表1 タスクのパラメータ

タスク名	IDLE	LED	MOTOR
初期状態	RUNNING	READY	RRADY
優先度(注1)	1	3	6
周期	∞	5	8
SUSPENDに併行する 実行の頻度(注2)	∞	2	3
他のタスク	LED	MOTOR	IDLE

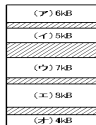
(注1)優先度の値が大きいほど優先度が高い。
(注2)ここで指定されている時間だけ実行されると、タスクはSUSPEND(入方向)になる。

解答欄

	(a)ラウンドロビン	(b)優先度付きラウンドロビン	(c)Rate Monotonic
1ミリ秒後			
2ミリ秒後			
3ミリ秒後			

2. メモリ管理(フィッティングアルゴリズム)

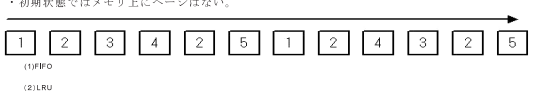
メモリに図1のような空き領域があるとき、(1)~(3)の順番に空き領域を必要とするタスクが実行された。(a)~(c)のアルゴリズムにおいて、それぞれのタスクにはどの領域が割り当てられるか答えよ。



	(a)ファーストフィット	(b)ワーストフィット	(c)ベストフィット
(1)1KB	(M)IDLE		
(2)2KB	(M)LED		
(3)3KB	(M)MOTOR		

3. メモリ管理(ページングアルゴリズム)

図2のような順番でメモリ上にページが呼び出される時、それぞれのアルゴリズムにおいて物理メモリにはどのページが置かれているかを、途中経過を含めて答えよ。ただし、次の条件に従った解答をすること。
・ページサイズ=4
・初期状態ではメモリ上にページはない。



(平成 18 年 12 月 4 日受付)
(平成 19 年 5 月 9 日採録)



西野 洋介 (学生会員)

1976 年生. 1999 年拓殖大学工学部情報工学科卒業. 2001 年同大学院工学研究科博士前期課程修了. 2003 年東京都立高等学校工業科教諭として着任. 現在, 拓殖大学大学院工学研究科博士後期課程電子情報工学専攻在籍中. 教育工学に興味を持つ.



田中 裕樹

1982 年生. 2005 年拓殖大学工学部情報工学科卒業. 現在, 同大学院工学研究科博士前期課程電子情報工学専攻在籍. 組み込みシステム, 教育支援, ロボット開発に興味を持つ.



川口 貴弘

1982 年生. 2006 年拓殖大学工学部情報工学科卒業. 現在, 同大学院工学研究科博士前期課程電子情報工学専攻在籍. 組み込みシステムとその教育支援に興味を持つ.



早川 栄一（正会員）

1964 年生．1989 年東京農工大学
工学部数理情報工学科卒業．1991
年同大学大学院博士前期課程修了．
1994 年同大学院博士後期課程単位
取得退学．1994 年同大学電子情報
工学科助手．1998 年拓殖大学工学部情報工学科助手，
1999 年同専任講師，2003 年同助教授，2007 年同准教
授．主にオペレーティングシステム，組み込みシステ
ムを中心とするシステムソフトウェアの研究開発に従
事．博士（工学）．
