

西野洋介, 早川栄一, 高橋延匡
拓殖大学大学院 工学研究科 電子情報工学専攻

1. はじめに

情報工学を学ぶ学生にとって OS の学習、理解は最重要課題の一つである。しかしながら学生が OS の概念、機構などを正しく理解しているは一概には言い切れない。その理由の一つは OS の動作の本質が実体として見えないことにある。特に OS はその機構から非同期な割込みによって動作する。そのためユーザはその動作概念がイメージできず、結果として OS の理解を妨げる原因になってしまう。そこで本報告では OS 概念、構造を抽象化し可視化を行った OS の学習支援環境の開発について述べる。

2. 本研究の目的

本研究では OS の動作をアニメーションを用いた抽象化を行うことで、従来のテキストによる OS 学習に比べ、視覚的に理解を促す事ができる学習支援環境を学生に提供することを目的とする。なおここでいう学生とは情報工学を学ぶ学生であり、特に本学習支援環境では OS についての知識がない OS 学習の初期段階の学生を対象とする。参考文献[1]のような OS の学習支援ツールが存在するが、本システムは OS 学習初期段階の学生を対象とし、可視化対象機能も OS のより基本的な機能に絞っている。

3. 設計方針

学生に OS の機能、概念を理解させるための可視化システムについての設計方針について述べる。

3. 1 OS の可視化方針

本学習支援環境の利用者は先にも述べたように OS について初期の学習段階にある学生を対象としているので、OS の基本的機能や構造など概念的な要素を学習対象とする。

Design of a Visualization Environment to Learn Operating System Course

Yosuke Nishino , Eiichi Hayakawa and Nobumasa Takahashi
Takushoku University

3. 2 可視化方針

可視化の方針として OS が管理するハードウェアリソース、ソフトウェアリソース、OS の管理構造を可視化し表示する。表示は OS とアプリケーション、各リソースの動作状況をアニメーションを用いて示す。アニメーションを用いる利点は次の通りである。

- ・状態の遷移状況を明確にできる
- ・直感的に動作の意味を捉えることができる

また学習効率の面を考慮して可視化された表示を見せるのではなく、実際に動作している OS のソースコードを表示し、現在の状況をトレースしながら動作結果と照らし合わせることで学習を進める。

ユーザインタフェースについては理解の促進を図るため必要な時点での動作停止、動作速度変更、表示項目の選択などの機能を取り入れる。

3. 3 実装方針

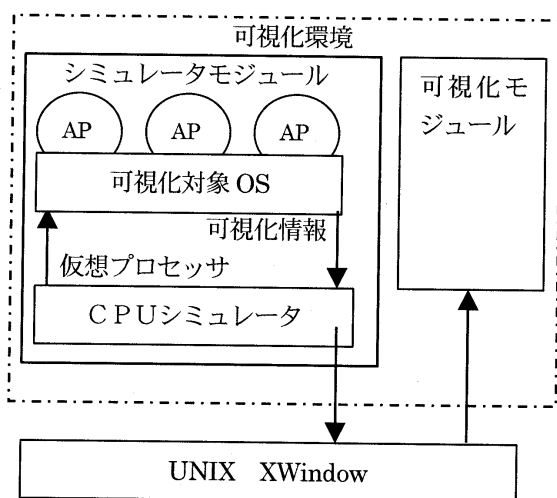
可視化する方法は学習対象 OS に独立したものを作成し、それらの動作状況を独立した可視化モジュールで示す方法を採用する。これは学習対象 OS 内に可視化に関する部分を取り除き、学習者の混乱を避けるためである。また学習対象 OS が変更されたときにシステム全体の変更を最小に抑えるためである。

4. 全体設計

先にも述べたように学習対象 OS には独立したオリジナルの OS を用いる。また学習対象 OS はシミュレートされたプロセッサ、メモリなどハードウェアリソースを制御し、それらの動作状況や OS のリソース管理状況を可視化モジュールによって表示する。全体の構成図を次項に示す (図 1)。

4. 1 プロセッサのシミュレータ設計

プロセッサには既存の RISC プロセッサ Alpha AXP アーキテクチャをベースに機能を抜粋したものを



採用する。Alpha には遅延分岐などの複雑な機能を持たないため学生が理解しやすいからである。また Alpha アーキテクチャには PAL コードと呼ばれるコンテキストスイッチ、割り込み、例外、メモリ管理などの OS の基本機能を提供している。アーキテクチャの複雑な機能を簡略化することで、学習者が余計な混乱を招く可能性を除くことができること、また実際に存在するアーキテクチャなのでクロスコンパイラを使用することで新しいコンパイラ作成する手間を省くことができるという利点がある。

4. 2 可視化対象 OS の設計

可視化対象 OS を設計するにあたり、複雑な OS の機能の中からプロセス管理、特に同期排他制御やプロセス間通信機能を実装したもの、メモリ管理、入出力管理など多機能ではなく、動作が複雑ではない OS を可視化対象として実現する。基本的な機能だけを備えた OS にすることで学習対象の的を絞ることができ、またソースコードの解析も混乱を招かずに行うことができるからである。

4. 3可視化モジュールの設計

可視化対象 OS の動作状況を表示する可視化モジュール部について次の特徴を持たせる。

(1) OS 動作のアニメーション表示

プロセス管理におけるコンテキストスイッチ、セマフォによる排他制御の動作をアニメーションを用いて表示する。

(2) ソースコードの表示

ソースコードを表示し、動作状況と対応させてどの関数を実行しているのかをトレース表示する。

(3) ハードウェアリソースの表示

ハードウェアをアイコンもしくは抽象化した形で表示することでハードウェアの動作状況を可視化、表示する。

(4) 可視化表示の操作、変更

アニメーション速度の変更や任意に情報の表示、非表示を行う。

5. プロセス管理の実現

プロセス管理の可視化については各プロセスのスケジューリングを行う様子をアニメーションを用いて表示している。また各プロセスの存在状態の遷移やコンテキストスイッチの表示を色分けして表示することで直感的にその状態を理解できるようにしている(図2)。

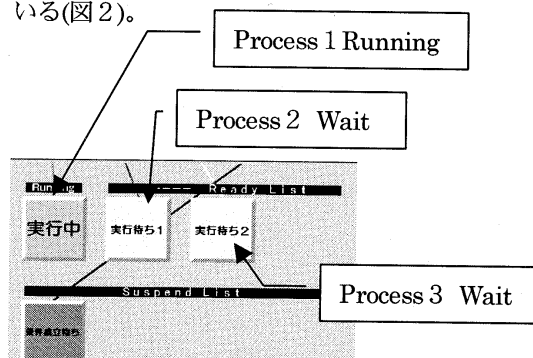


図2. プロセス管理機構の可視化画面

5. おわりに

本報告ではコンピュータサイエンスを学ぶ学生が OS の概念や動作、管理機構の学習について可視化によって支援する環境について述べた。今後は詳細な設計を行い、学習支援システムを実現していく。

参考文献

- [1]伊藤能康他：学習支援のための OS の可視化ツールの設計と実現、平成7年、東京農工大学
[2]日本デジタルイクイップメント株式会社：
Alpha AXP アーキテクチャ概要、1993年、共立出版株式会社