AnT オペレーティングシステム

An operating system with adaptability and toughness

基本設計書

第1版　平成18年2月7日
第2版　平成23年12月22日

岡山大学 工学部 情報工学科
岡山大学 大学院自然科学研究科 電子情報システム工学専攻
岡山大学 大学院自然科学研究科 産業創成工学専攻
谷口研究室 山内研究室

目 次

1　はじめに

マイクロプロセッサや入出力ハードウェアの進歩には目覚しいものがある。また、通信路の伝送速度の向上も著しい。また、様々な場面で計算機が必要となり、提供するサービス種別も飛躍的に増大している。このような背景から、これらハードウェアの機能や性能を有効に利用でき、さらに多様なサービスの提供を支える基盤ソフトウェアが必要になっている。

そこで、

(1) ハードウェアとサービスを様々に組み合わせたシステムを管理制御できる

(2) ネットワークを高い通信効率とセキュリティで利用できる

(3) ソフトウェア開発をオープン化できる

ことを可能にする ***AnT*** オペレーティングシステム (An operating system with adaptability and toughness) を開発する。なお、***AnT*** オペレーティングシステムは、研究用ではなく実利用を目的としている。

2　設計方針

プログラムの設計においては、以下の 3 つを基本方針とする。

(1) 適応型構造

システムの環境に合わせて発展するために、環境を学習し、環境に適応できる機能やプログラム構造を有する。このため、多くのシステムの共通な機能を内コアとして用意し、システムの環境に適応した機能を外コアとして発展させることにより、1 つのコアから環境ごとに変貌し無限個のオペレーティングシステム（OS）が誕生する。

(2) ソフトウェアの公開

開発したソフトウェアの普及を促進するため、ソフトウェアは公開する。このため、第 3 者のプログラム構造の理解を支援するため、複雑な機能や構造（例えば、要求時ページング機能やハッシュ構造）は採用しない。

(3) 開発工数の削減

OS そのものの開発を重視するため、開発する言語や環境は UNIX を利用する。また、次々に登場する新しい入出力ハードウェアを速やかに利用できるようにするために、制御ソフトウェア（ドライバ）のユーザモード（プロセス）での走行を可能にし、既存ドライバの移植方法を確立する。

3　機能

3.1　機能概要

本 OS では、走行モード変更、ゼロコピー通信、および適応制御の機能を提供する。走行モード変更機能は、外コアの走行モードを動的かつ自由に変更できる機能である。ゼロコピー通信機能は、プログラム間のデータ授受をゼロコピーで行える機能である。適応制御機能は、システムの環境に合わせてソフトウェアの機能や構成を自動的に変更する機能である。

3.2　走行モード変更

プロセスの高速な実行を可能にするため走行モード変更機能がある。本機能は、他プロセス空間プログラムの走行モードを、ユーザモードからスーパバイザモードへ変更、あるいはスーパバイザモードをユーザモードへ変更する機能である。つまり、変更の指示により、プロセス空間プログラムの走行モードを変更し、変更されたプログラムと内コアの連携方式を変更する。

プロセス空間プログラムがユーザモード走行の場合、内コア呼び出しが例外を利用し、内コアからの呼び出しはプロセス割込みを利用する。これに対し、プロセス空間プログラムがスーパバイザモード走行の場合、内コア呼び出しは関数呼び出しであり、内コアからの呼び出しも同様である。したがって、プロセス空間プログラムをスーパバイザモード走行させることにより、プロセス空間プログラムと内コア間の連携オーバヘッドを大きく削減できる。

3.3　ゼロコピー通信

プロセスと内コアの間あるいはプロセス間の通信を高速化するため、ゼロコピーでのデータ授受機能がある。このためには、大きく以下の 3 つの機能がある。

(1) ページ（4 kByte）を単位とし，ｎページ分の領域の確保と解放

(2) 確保した領域（ｎページ）の実メモリ連続の保証

(3) ２仮想空間の間での領域の貼り替え

プロセスと内コアの間あるいはプロセス間で授受を行うデータについては、ページ単位で管理される領域にデータを格納して、通信を行う。この領域をコア間通信データ域 (ICA:Inter-core Communication Area) と名付ける。ICA は、内コアにより管理される。

プロセスと内コアの間あるいはプロセス間でのデータ授受において、データの複写を避けるため、ICA に格納されたデータは、仮想空間のマッピング表を書き換えることにより

データを授受する。これをICAの貼り替えと名付ける。ICAはページ単位であるため、このような貼り替えが可能である。したがって、プロセスが内コアの機能を呼び出す際に渡すデータは、プロセスの仮想空間にあるICAを内コアの仮想空間に貼り替える。一方、内コアがプロセスに実行結果を返却する際に渡すデータは、内コアの仮想空間にあるICAをプロセスの仮想空間に貼り替える。また、プロセス間通信を行う場合、送信プロセスの仮想空間にあるICAを受信プロセスの仮想空間に張り替える。

以上に示すように、通信する2つのプログラム間でのデータ授受を2仮想空間の間でのICAの貼り替えにより実現し、ゼロコピー通信を実現する。

3.4 適応制御

新しいハードウェアやサービスに即座に対応できるために、適応制御機能を有する。この機能は、大きく以下のものからなる。

(1) 新しいハードウェアやサービスの検知
(2) 必要な外コアの起動
(3) 過去のプログラム利用履歴の管理
(4) システム適応制御
(5) 異常プログラムの管理

新しいハードウェアの組み込みを監視し検出する。また、新しいサービスの要求を受け付ける。

新しいハードウェアの検知や新しいサービスの要求に基づき、必要になるプログラムを起動する。

起動したプログラムについて、起動時刻や終了時刻を監視把握し、プログラムの利用履歴情報の獲得と管理を行う。この情報をプログラム利用履歴情報と名付ける。

プログラム利用履歴情報に基づき、頻繁に使われるプログラムについては、プログラムの常駐化、さらにスーパバイザモードへの走行モード変更を行う。一方、利用頻度が低下したプログラムについては、ユーザモードへの走行モード変更、さらにプログラムの終了指示を行う。これらの処理は、システム立ち上げの際だけではなく、定期的に行う。

プログラムの異常により終了した（させられた）プログラムについては、その情報を収集し、プログラム利用履歴情報に登録する。

4　適応型構造

4.1　基本構造

プログラムは、OS とサービスからなる。OS は、内コアとプロセスとして動作する外コアからなる。サービスは、プロセスからなる。この様子を図 1 に示す。

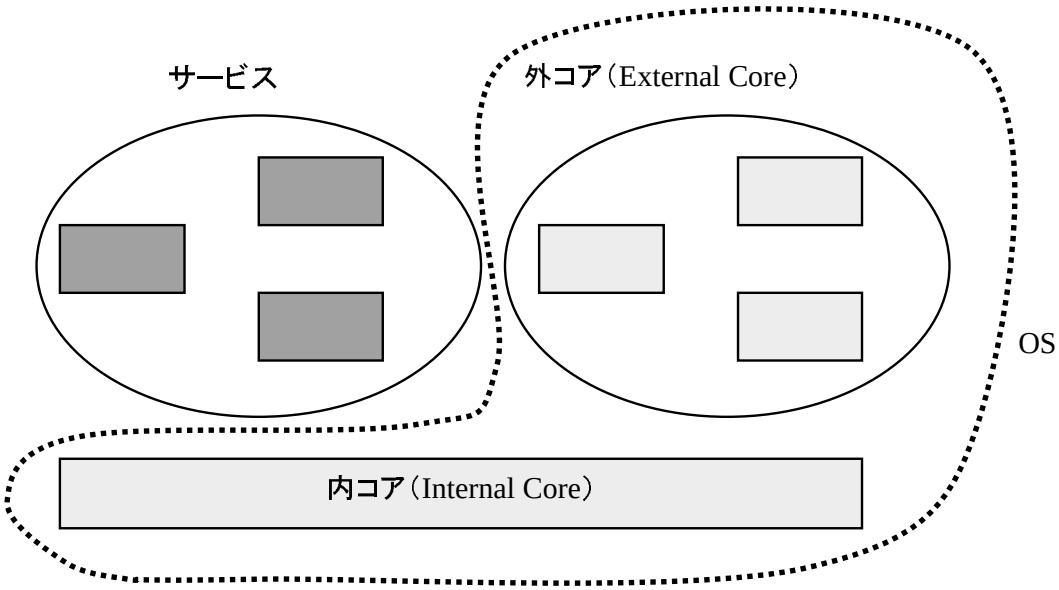

図 1　基本構造

内コアは、最小のシステムの動作を保証するプログラム部分である．外コアは、適応したシステムに必須なプログラム部分であり、動的に再構成な構造を有する。

サービスは、サービスを提供するプログラム部分である。

4.2　信頼性の確保

プログラム動作の信頼性を確保するため、記憶空間の分離と後作成プログラムの外コア実行を行う。この様子を図 2 に示す。

内コアはカーネル空間で実行し、サービスと外コアはプロセス空間で実行する。これにより、内コアを保護する。また、サービスと外コアの間および外コア間を保護するため、プロセス空間は多重仮想記憶とする。

ドライバなどのようなプログラムは、新たな入出力ハードウェアの登場と共に順次開発される。このため、OS の開発当初ではなく、後から開発されたプログラムを後作成プログ

ラムとして区別する。多くの場合、ソフトウェアの開発においては、開発初期に開発されたプログラムに比べ、後作成プログラムは低品質な場合が多い。また、開発者によりソフトウェアの品質が大きく異なることは言うまでもない。したがって、開発時期の違いやドライバ開発元の違いにより、その信頼性（ソフトウェア品質）には大きな差が生じてしまう。このため、異なる信頼性を有するプログラムの共存制御が必要である。そこで、ドライバなどのようなプログラムは外コアとして実行し、多重仮想記憶による相互保護を行う。

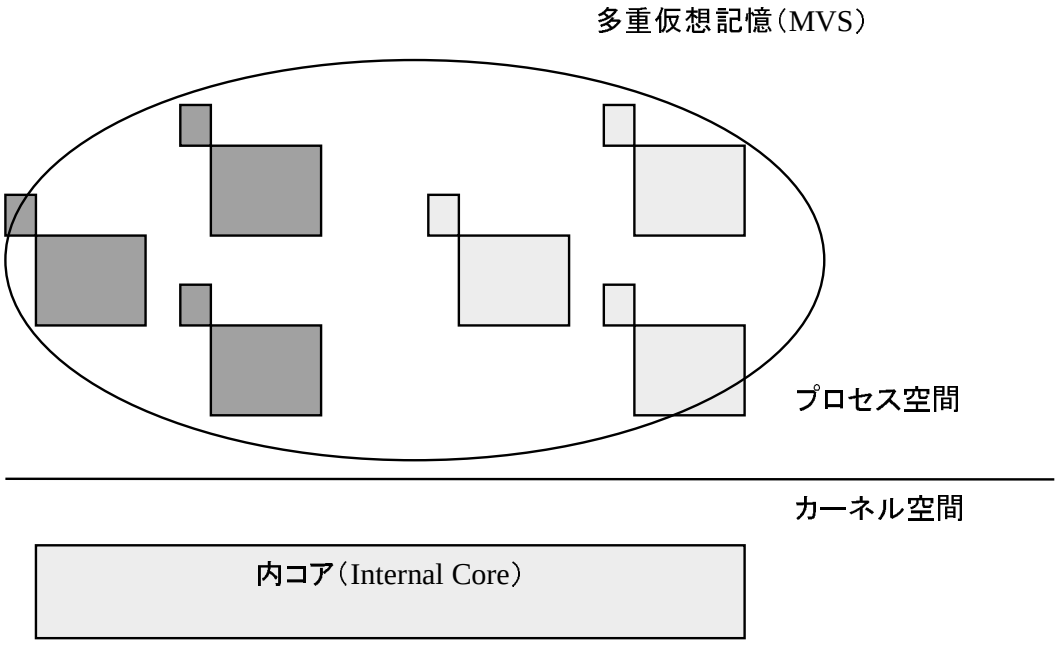

図 2 信頼性の確保

4.3 高性能化

プログラム実行におけるオーバヘッドを削減し、プログラムの処理性能を向上させるため、走行モードの変更とゼロコピーによるプログラム間通信を行う。

信頼性が高くかつ頻繁利用される外コアのプログラムは、スーパバイザモードで実行する。スーパバイザモードで実行することにより、外コアのプロセスからの内コア呼び出しを直接行える。つまり、例外を発生させることなく、関数呼び出しのようにジャンプ命令のように呼び出すことができる。これにより、外コアと内コアの連携を高速化でき、ドライバ処理の高性能化を実現する。

サービス提供を適切に行うためには、サービスを提供するプロセスとドライバプロセスの高速な連携が必要である。具体的には、両プロセス間での高速なデータ授受（高速 IPC）が必要である。さらに、プロセスと内コア間の高速なデータ授受も必要である。そこで、プログラム間のデータ授受をメモリマッピングで提供する。これにより、データ授受できるデータの基本単位は、MMU ハードウェアのページに制限されるものの、高速な通信が可能になる。この様子を図 3 に示す。

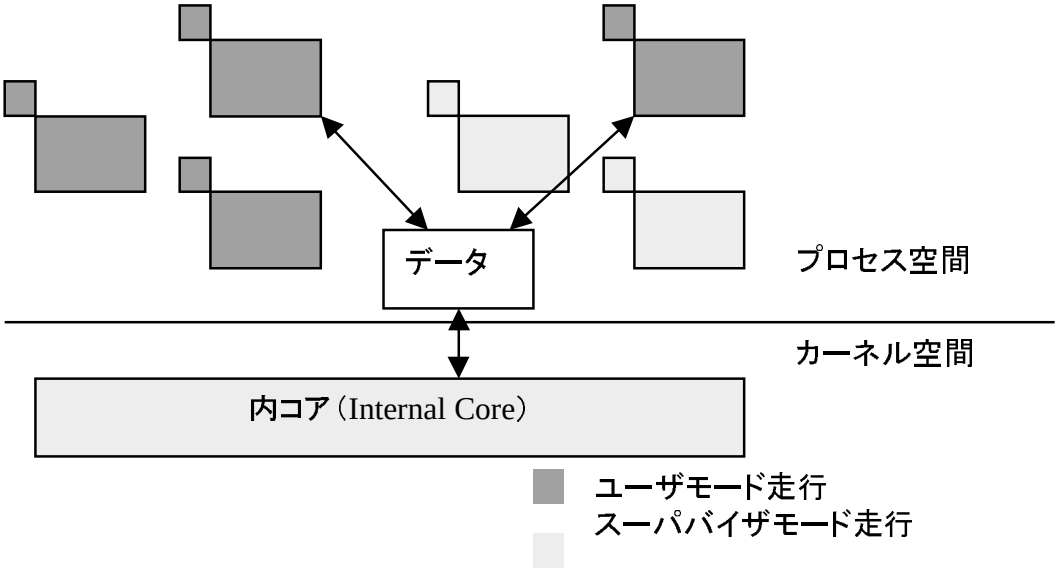

図 3 ゼロコピーによるプログラム間通信

5 おわりに

AnT オペレーティングシステム (An operating system with adaptability and toughness) について、設計方針、機能、および適応型構造について述べた。設計方針として、適応型構造、ソフトウェアの公開、および開発工数の削減がある。機能として、走行モード変更、ゼロコピー通信、および適応制御の機能を提供する。適応型構造として、内コア、外コア、およびサービスの 3 層構造を有し、信頼性の確保と高性能化を実現する。なお、***AnT*** オペレーティングシステムは、研究用ではなく実利用を目的としている。