

マイクロカーネルアーキテクチャに基づいた ユーザレベルの割り込み制御と性能評価

本田 良司 緒方 正暢

日本アイ・ビー・エム (株) 東京基礎研究所

あらまし 我々は、マイクロカーネルアーキテクチャ上で、OS サーバ及びデバイス・ドライバの性能を向上させる実現する手法について研究している。本稿では、特にその中で重要なもののひとつである、インタラプト及びエクセプション・ハンドラーの処理をユーザレベルで実現する方法と、それを DOS/V サーバ上で評価した結果について述べる。グラフィックス描画処理を行なうユーザレベルの仮想 86 モードのスレッドに I/O 特権を与えることによりエクセプションをなくし実行時間を大きく短縮できることがわかった。

User-level Interrupt Handling Mechanism in OS Servers on Micro-kernel

Ryoji Honda Masanobu Ogata

{ryoji,ogata}@trl.ibm.co.jp

IBM Research, Tokyo Research Laboratory

1623-14, Shimotsuruma, Yamato, Kanagawa, 242 JAPAN

Abstract. We have implemented a DOS/V emulation on Mach 3.0 micro-kernel. This allows multiple DOS sessions and works as a single server. To extract maximum performance from DOS applications, we have provided user-level I/O privilege support in Mach 3.0 kernel. Typical DOS applications access the graphics video buffer directly while enabling/disabling the interrupt flag. I/O privilege in the user mode gives higher performance in DOS applications.

1 はじめに

Mach 3.0 マイクロカーネル上にはいくつかのオペレーティングシステムがユーザレベルのサーバとして実現されている [1][2][3][4][5]。オペレーティングシステムが提供する機能をマイクロカーネル内で実現するか、あるいは、ユーザレベルで実現するかについていろいろな手法が提案されている。

我々は、割り込みとエクセプションをユーザレベルで制御する方法について検討している。実際に、Mach 3.0 マイクロカーネル MK58 をもとに、DOS/V を実行させるためのエミュレーションライブラリと DOS/V サーバの開発、および、マイクロカーネルの拡張を行なった [6]。現在は、Mach 3.0 MK78 上で CMU の MDOS を拡張する形で、開発を続けている。

本稿では、我々が DOS/V サーバを実装する時に行なったマイクロカーネルアーキテクチャに基づいたインタラプトの処理方法について述べる。

まず、Mach 3.0 上のユーザレベルのサーバとして最初を実現された UNIX¹ サーバについて簡単に述べ、次に、我々が開発している DOS/V サーバについて説明する。最後に、ユーザレベルでの割り込み制御の手法について検討する。

2 UNIX サーバ

BSD サーバ [1] や OSF/1² サーバ [2] では、OS サーバとエミュレーションライブラリによって 4.3BSD UNIX や OSF/1 とのバイナリレベルでの互換性を保っている。エミュレーションライブラリは、他のサーバを呼び出したり、ライブラリ自身で処理を行なう。エミュレーションライブラリは mach_init から継承され、mach_init から起動される全タスクのアドレス空間内にマップされている。

ユーザプログラム(タスク)がシステムコールを発行した時の処理手順を以下に示す(図 1

¹UNIX is a registered trademark of Unix Systems Laboratories, Inc.

²OSF/1 is a trademark of the Open Software Foundation, Inc.

参照)。

- 1) ユーザタスクがシステムコールの発行する。すると trap が発生し、マイクロカーネルに制御が移る。
- 2) マイクロカーネルは、システムコールを発行したユーザタスクのアドレス空間内にあるエミュレーションライブラリにアップコールする。
- 3) エミュレーションライブラリは trap を起こしたシステムコールを調べ、UNIX サーバに RPC を用いてシステムコールの処理を依頼する。ただし、エミュレーションライブラリ自身で処理できる時はサーバへの処理の依頼は行なわず、すぐにシステムコールを発行した次の命令へ復帰する。
- 4) UNIX サーバから処理の完了が通知される。
- 5) システムコールを発行した次の命令へ復帰する。

3 DOS/V サーバ

3.1 DOS/V

DOS/V は、漢字表示のために専用ハードウェアをいっさい使わず、ソフトウェアのみでこれを実現したものである [7]。DOS/V での日本語化は、DOS のデバイスドライバによって達成され、BIOS レベルで日本語処理が可能になっている。

DOS/V のアプリケーションプログラムから見た、パーソナルコンピュータの構造を図 2 に示す。DOS ならびに DOS/V は、DOS カーネルと BIOS [8] からなる。BIOS には、ハードウェアとともに装備されている ROM BIOS とデバイスドライバや TSR (Terminate and Stay Resident) のようにディスクからメモリにロードされる RAM BIOS がある。RAM BIOS には、DOS/V の日本語処理のためのデバイスドライバや、HMA (High Memory Area)、XMS (eXtended Memory Specifi-

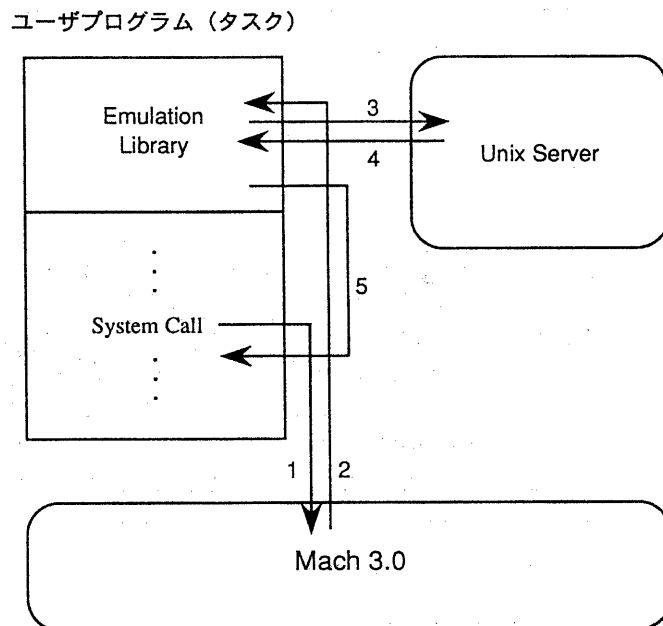


図 1: UNIX サーバにおけるエミュレーションライブラリによるシステムコールの処理

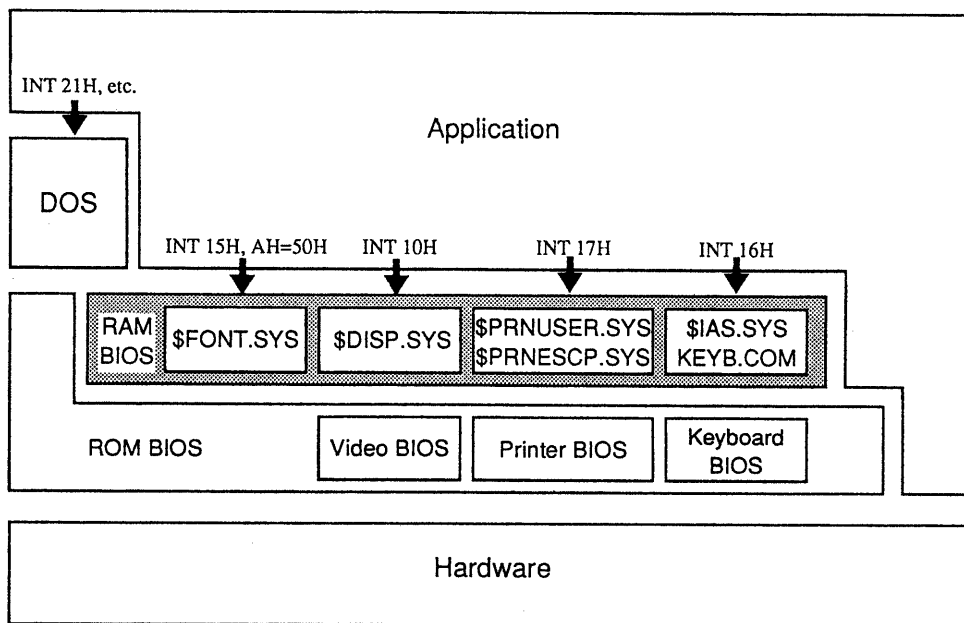


図 2: DOS/V の構造

cation)、EMS(Expanded Memory Specification)を実現するドライバなどがある。

DOS/V で提供される日本語処理のためのデバイスドライバを以下に説明する。

\$FONT.SYS ビデオサブシステムから要求のあった文字のフォントイメージを検索し、使用可能にする (INT 15H, AH=50H)。

\$DISP.SYS 漢字表示のためのビデオ BIOS(INT 10H) の拡張する。**\$FONT.SYS** を呼び出してフォントイメージを展開し、VGA の VRAM に書き込む。

\$PRNUSER.SYS, \$PRNESC.P.SYS 文字コードまたはユーザフォントのイメージをプリンタに送出する (INT 17H)。

\$IAS.SYS, KEYB.COM

キーボード BIOS(INT 16H) を拡張する。キーボード割り込みを処理し、コードページ処理、漢字キーや変換キーなどの日本語処理用のキー入力、ローマ字仮名変換などを行なう。

3.2 DOS/V サーバ

Mach 3.0 上で DOS/V の機能をエミュレーションするのが、DOS/V サーバである。DOS/V サーバでは、PC-DOS、MS-DOS、DR-DOS、IBM DOS/V、MS-DOS/V、DR-DOS/V を実行することができる。DOS ならびに DOS/V のプログラムを動かすには、i386/i486 の仮想 86 モードを使用している。Mach 3.0 MK73 からはカーネル内に CMU で開発された DOS サーバをユーザレベルで実現するための以下の機能が標準で含まれている。

- 仮想 86 モードで動作するスレッド
- ハードウェア仮想 86 Faults である、STI/CLI, PUSHF/POPF, IRET のエミュレーション
- 仮想 86 スレッド内の外部割り込みのエミュレーション

DOS/V サーバでエミュレートするものは、以下の三つに分類される。

INT (ソフトウェア・インタラプト) DOS や ROM BIOS/RAM BIOS (デバイスドライバ) の呼び出しは、すべてこの INT 命令を介する。

ハードウェア割り込み キーボード、マウス、タイマなどのハードウェア割り込みベクタは、アプリケーションによって上書きされる可能性があるため、これもエミュレーションして動かす必要がある。

IN/OUT 命令 ビデオ・バッファなどのアクセスは、必ずこの入出力命令をとまなう。インタラプトを引き起こすか否かは、ポートごとに設定できる。

エミュレーションには、以下の 2 つの方法がある。

A) 図 3

DOS アプリケーションの実行中に発生したインタラプトは、マイクロカーネル内部のトラップハンドラに先ず制御を移す。トラップハンドラは、トラップを引き起こしたタスクのエミュレーションライブラリにアップコールし、然るべきエミュレーションが実行される。その後、インタラプト直後の命令から再開される。エミュレーションライブラリは Mach の IPC や Mach のデバイスドライバを使ったり、UNIX サーバの機能を利用して BIOS のエミュレーションを行なうことができる。

B) 図 4

DOS、BIOS またはアプリケーションのインタラプトハンドラそのものをそのまま実行させる。アプリケーションの中には、直接ハードウェアをアクセスするものがあるので、必要に応じてハードウェアを仮想化する。

実際に、キーボード、マウス、タイマなどは、ハードウェア割り込みの処理が上書き

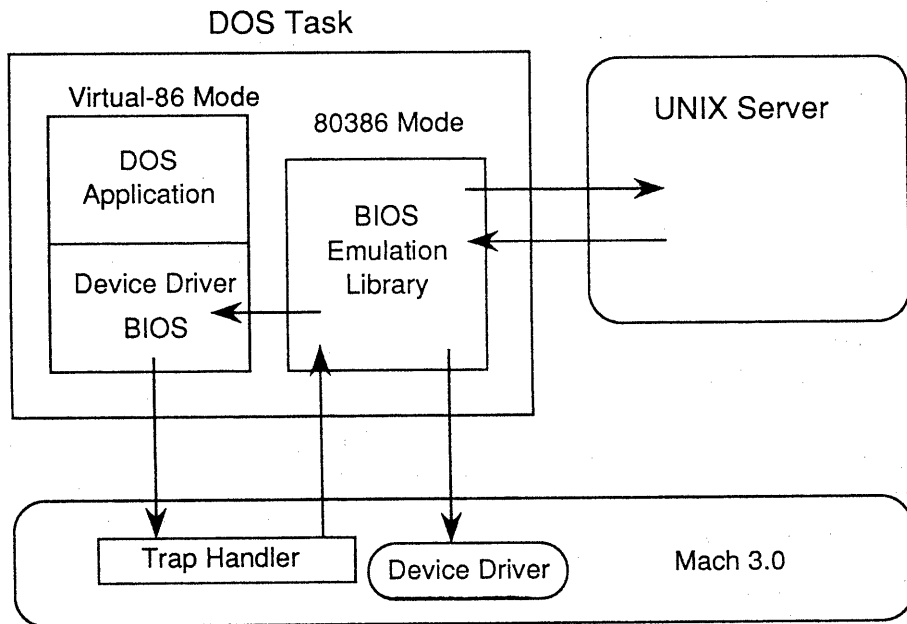


図 3: BIOS エミュレーションライブラリを用いた DOS サーバの構成

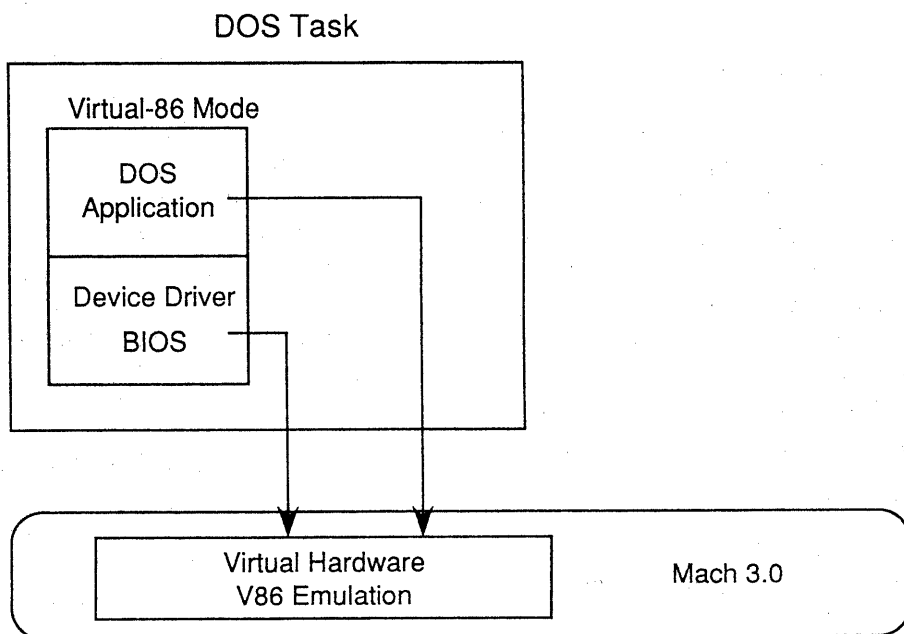


図 4: カーネル内で v86 をエミュレーションする DOS サーバの構成

されることがあるので、そのまま動かしてプログラマブル・インタラプト・コントローラ (PIC) を仮想化して見せる必要がある。ここでいう仮想化とは、実際にはすでに PIC の EOI などが済んでいるのに、いまそれをしているかのように見せたり、現在のステータスも仮想的なステータスとして返すということである。

CMU で開発された DOS サーバ MDOS[3] では、A の手法を主に用いている。多くの BIOS エミュレーション、EMS 3.2 ドライバ、XMS 2.0 ドライバなどが実現されている。また、BSD サーバと共存して UNIX のファイルシステムを DOS のネットワークドライブとしてアクセスできるようなサービスも提供されている。

我々の作成した DOS/V サーバは、その多くが B のケースを多用している。なぜなら、DOS/V の日本語処理のために拡張された BIOS はデバイスドライバとして実現されており、ユーザが自由に新しいドライバを導入して機能拡張を行なうことを許す必要があるからである。つまり、これらの BIOS 機能はエミュレーションライブラリによってエミュレーションするのではなく、BIOS のコードをそのまま実行させなければならない。さらに、IBM-DOS、MS-DOS、DR-DOS といった異なる DOS を動作させたかったからである。

3.3 描画処理における性能評価

DOS や BIOS で管理されていない VGA のレジスタなどの資源はそれらの資源を使っている間に他のプログラムが動作しないように割り込みを禁止している。また、DOS アプリケーションプログラムの中にはビデオバッファを直接アクセスしているものもある。特に、グラフィックス・モードの場合図 5 のようにプログラムされていることが多い。

pushf はステータスレジスタのセーブ、cli はインタラプトをディセーブルにして、グラフィックスのアクセス中に割り込まれないようにしている。popf は、ステータスレジスタの復帰である。

```
pushf
cli

Access to Video Buffer

popf
```

図 5: 一般的なビデオ・バッファのアクセス方法

これらインタラプトフラグの制御命令は、I/O 特権がない場合、エクセプションを必ず引き起こす。これらエクセプションは描画の度に頻繁に起こり、描画速度を無視できないほど遅くする要因になっている。そこで、I/O 特権がない時と比べ I/O 特権を持たせた時にどのくらい描画速度が向上するか実際に測定してみた。実験では、DOS アプリケーションを実行する仮想 86 モードのスレッドに I/O 特権を持たせた。測定には、IBM ThinkPad 700C (C52) を用いた。実験結果を表 1 に示す。

表 1: Lotus 1-2-3 R2.3J の VIEW_3.WJ2 の描画時間の比較

仮想 86 モード、I/O 特権なし	30 seconds
仮想 86 モード、I/O 特権あり	11 seconds

ハードウェア：IBM ThinkPad 700C

(i486SLC 25MHz)

ソフトウェア：IBM DOS バージョン J5.02/V

I/O 特権を与え、エクセプションを起こさないようにするとパフォーマンスが極だつて向上することがわかる。つまり、DOS エミュレーションの性能向上のためにはできるだけインタラプトの発生を抑えることが重要である。

4 インタラプトの仮想化およびユーザーレベル制御の有用性に関する考察

性能向上のため、仮想 86 モードのプログラムに I/O 特権を与える方法を提案する。ただし、無条件に I/O 特権を与えてしまうとシステムの信頼性を損なってしまう。

- 信頼できるソフトウェアの実行中に限ってユーザーレベルの仮想 86 モードのスレッドに一時的に I/O 特権を与えて、エクセプションを起こさないようにする。
- ページ単位で、I/O 特権を与える。信頼できるソフトウェアが入っているページ内のプログラム実行時に、一時的に I/O 特権が与えられる。例えば、DOS/V では、RAM BIOS として多くのデバイスドライバが提供されている。これらのプログラムは十分に信頼できると考えられるので、それらを含むページの実行は I/O 特権を与えることができる。

5 おわりに

本稿では、DOS/V サーバーというある意味で特種なサーバーにおける、インタラプトの制御をユーザー・レベルで行なう方法について述べた。要約すれば、次のことをユーザー・レベルで実行したことになる。

- 割り込み制御ハードウェア (PIC) の仮想化
- パフォーマンス向上のための一時的 I/O 特権の獲得

本結果は仮想 86 モードという仮想マシン上で行なったことであるが、これを他の OS サーバーにも基本的に適用可能である。

参考文献

- [1] D. Golub, R. Dean, A. Forin and R. Rashid: "Unix as an Application Pro-

gram," Proc of the USENIX Summer Conference, pp. 87-95 (1990).

- [2] "Architecture and Benefits of a Multi-threaded OSF/1 Server," OSF Research Institute Memorandum, Open Software Foundation (1991).
- [3] G. Malan, R. Rashid, D. Golub and R. Baran: "DOS as a Mach 3.0 Application," Proc of the Second USENIX Mach Symposium, pp. 27-40 (1991).
- [4] D. Golub, R. Manikundalam and F. L. Rawson III: "MVM - An Environment for Running Multiple Dos, Windows and DPMI Programs on the Microkernel," Proc of the Third USENIX Mach Symposium, pp. 173-190 (1993).
- [5] J. M. Phelan, J. W. Arendt and G. R. Ormsby: "An OS/2 Personality on Mach," Proc of the Third USENIX Mach Symposium, pp. 191-201 (1993).
- [6] 本田: "Mach 3.0 マイクロカーネル上のマルチプル DOS サーバー," 情報処理学会第 45 回 (平成 4 年後期) 全国大会予稿集, 1P-10 (1992).
- [7] "IBM DOS バージョン J5.0/V BIOS インタフェース技術解説書," 日本アイ・ビー・エム株式会社, 営業資料番号 N:SC18-2489 (1991).
- [8] "Personal System/2 and Personal Computer BIOS Interface Technical Reference," IBM, S04G-3283 (1991).