

# プロセスとして実現したデバイスドライバにおける 仮想空間切り替えの影響 Effect of virtual address space switch in device driver running as process

野村 裕佑† 岡本 幸大†† 乃村 能成†

Yusuke Nomura Kouta Okamoto Yoshinari Nomura

谷口 秀夫† 丸山 勝巳‡  
Hideo Taniguchi Katsumi Maruyama

## 1. はじめに

計算機の進歩に伴い、次々に新たな入出力機器が登場している。オペレーティングシステム(以降、OS と略す)において、これら入出力機器はデバイスドライバと呼ばれるプログラムにより制御される。多くの OS では、デバイスドライバはカーネルの一部として動作するため、デバイスドライバの信頼性は OS 全体の信頼性に大きく影響する。

このような背景から、我々はデバイスドライバをプロセスとして実現する方式を提案した[1]。プロセスは OS との高い独立性を保って走行するため、デバイスドライバが OS に与える影響を最小限に抑えることができる。このため、OS の安全性を保証しやすい。しかしながら、デバイスドライバへの処理要求、および割り込み処理の実行の際に仮想空間切り替えが発生するため、オーバーヘッドが大きいと考えられる。ここでは、仮想空間切り替えがデバイスドライバの処理時間に及ぼす影響を評価する。

## 2. プロセスとして実現したデバイスドライバ

### 2.1 動作概要

本方式では、デバイスドライバをスーパーバイザモードで走行するプロセスとして実現している。スーパーバイザモードで走行するプロセスは、I/O 命令を発行することにより、ハードウェアを直接制御することが可能である。

プロセスとして実現したデバイスドライバの基本構造を図 1 に示す。以降では、プロセスとして実現したデバイス

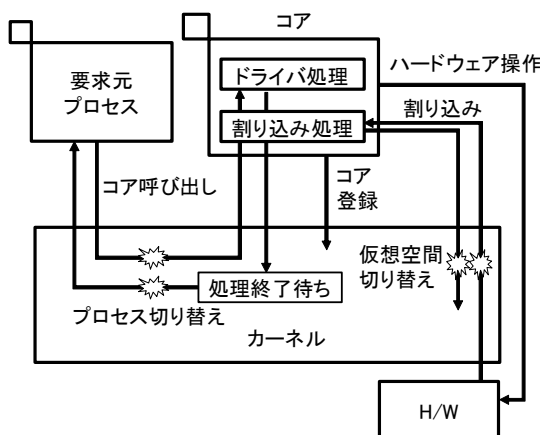


図 1 基本構造

ドライバをコアと呼ぶ。

コアは、初期化処理の最後でコア登録システムコールを発行し、デバイスの種別やドライバ処理の識別子、割り込み処理の先頭アドレスをカーネルに登録する。ユーザプロセスからコアへの要求は、コア呼び出しシステムコールによって実現される。コア呼び出しでは、プロセス切り替えによりコアのドライバ処理を呼び出す。割り込みを利用する場合は、ドライバ処理終了後に待ち状態に移行する。割り込み発生時は、コアの割り込み処理はユーザ領域に存在するため、仮想空間を割り込み処理を行うコアの仮想空間に切り替えることにより、割り込み処理を実行する。割り込み処理により処理終了待ちが解除されると、プロセス切り替えにより要求元プロセスに復帰する。

### 2.2 コアの動作による仮想空間切り替え

デバイスドライバをコアとして実現した場合、ドライバ呼び出し時に、プロセス切り替えが 1 回発生する。割り込み時は、仮想空間切り替えが 2 回発生する。ただし、割り込み時にコアが走行していた場合は発生しない。要求元プロセスへの復帰時は、割り込みにより待ち状態を解除される際に 1 回、待ち状態を解除されてから要求元プロセスに復帰する際に 1 回のプロセス切り替えが発生する。ただし、待ち状態解除時にコアが走行していた場合は、待ち状態解除前のプロセス切り替えは発生しない。

一方、カーネルに組み込まれて動作するデバイスドライバ(以降、組み込み型ドライバと呼ぶ)では、待ち状態解除前のみプロセス切り替えが 1 回発生する。ただし、待ち状態解除時に要求元プロセスが動作していた場合は、プロセス切り替えは発生しない。

プロセス切り替えは仮想空間切り替えを伴うため、ここでは、仮想空間切り替えによる処理時間の遅延について調査する。

## 3. 仮想空間切り替えの影響の評価

### 3.1 評価概要

組み込み型ドライバ内で仮想空間切り替えを発生させ、仮想空間切り替えの影響を評価する。

評価環境を表 1 に示す。Pentium 4 プロセッサでは、仮想空間切り替えの際に TLB のフラッシュが発生する。このため、仮想空間切り替えによる影響は、さらに仮想空間切り替え処理と TLB のミスヒットの影響に分類できる。

処理時間の計測には RDTSC 命令を使用し、測定結果は 10000 回の平均クロック数である。また、TLB ヒット数の計測には RDPMC 命令を使用する。

† 岡山大学大学院自然科学研究科

†† 岡山大学工学部情報工学科

‡ 国立情報学研究所

表 1 評価環境

OS	FreeBSD4.3R
CPU	Pentium 4 2.8GHz
ドライバ	RealTek 8129/8139 PCI ethernet cardドライバ

### 3.2 仮想空間切り替え処理による処理時間の増加

FreeBSD では、カーネルが使用するメモリ領域を示すページテーブルエントリは、TLB フラッシュの対象外として設定される。このため、組み込み型ドライバの割り込み処理では TLB ミスヒットは発生しない。そこで、組み込み型ドライバの割り込み処理で仮想空間切り替え処理を実行することにより、仮想空間切り替え処理自体の処理時間を測定することができる。

測定の処理概要を図 2 に示す。コアの割り込み処理呼び出しを想定し、図 2 の A 地点で仮想空間切り替え処理を実行した場合と実行しなかった場合の処理時間を測定する。ただし、I/O 処理 (a2-a3) は処理時間の揺れが大きいため、a1-a2 間と a3-a4 間の処理時間のみを測定する。

測定結果を表 2 に示す。表 2 より、仮想空間切り替え処理 1 回あたりの処理時間は約 380 クロックであることがわかる。

### 3.3 TLB ミスヒットによる処理時間の増加

コアはユーザ領域で動作するため、仮想空間切り替え時にコアのメモリ領域を示す TLB がフラッシュされ、TLB のミスヒットが発生する。TLB ミスヒットの影響を測定するため、組み込み型ドライバのドライバ呼び出し処理におけるユーザデータの取り込み処理を利用する。ユーザデータはユーザ領域に存在するため、データ取り込みの際に TLB のミスヒットが発生する。

測定の処理概要を図 3 に示す。プロセス切り替えの際の仮想空間切り替えを想定し、図 3 の B 地点において仮想空間切り替え処理を実行した場合と実行しなかった場合の処理時間を計測する。ここでは、TLB フラッシュの影響のみを測定するため、b2-b3 間のデータ取り込み処理の処理時間を計測する。測定に使用したデータは 4 バイトである。

測定結果を表 3 に示す。データ取り込みにより、TLB のミスヒットは 2 回発生した。このとき、TLB ミスヒット 2 回による処理時間の遅延は約 100 クロックである。つまり、TLB ミスヒット 1 回あたりの遅延は約 50 クロックであることがわかる。

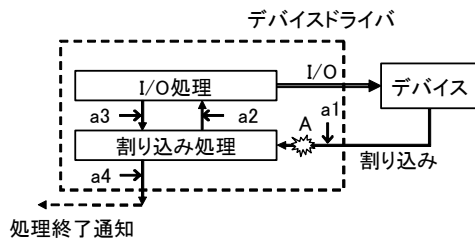


図 2 割り込み処理測定の処理概要

表 2 割り込み処理の測定結果

空間切り替え	処理時間	最大値	最小値	TLBミス
なし	13644	15435	13345	0
あり	14021	14711	13692	0
差分	377			0

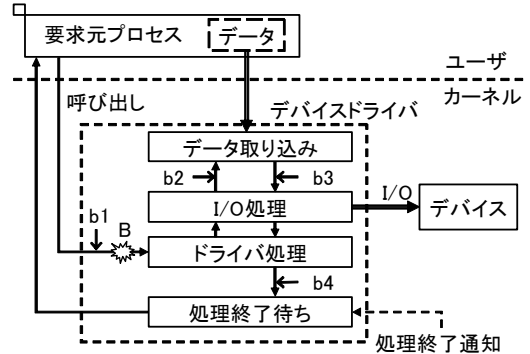


図 3 ドライバ呼び出し処理の処理概要

表 3 データ取り込みの測定結果

空間切り替え	処理時間	最大値	最小値	TLBミス
なし	416	557	388	0
あり	518	599	472	2
差分	102			2

### 3.4 考察

3.2 節と 3.3 節より、仮想空間切り替え処理 1 回あたり約 380 クロック、仮想空間切り替えにともなう TLB ミスヒット 1 回あたり約 50 クロックの処理時間の遅延が発生することがわかる。また、図 3 のドライバ呼び出し処理において、b1-b4 間の処理時間は 8866 クロックであった。これにより、デバイスドライバ内の処理では、TLB フラッシュを起こさない場合、割り込み処理において  $377 \times 2 / (14021 \times 100) \approx 5$  より約 5%、ドライバ呼び出し処理では  $377 / (8866 \times 100) \approx 5$  より約 4% の処理時間の遅延が発生することがわかる。このとき、呼び出し 1 回に対して割り込み処理が 1 回呼び出される場合の処理時間の遅延は  $377 \times 3 / (14021 + 8866) \times 100 \approx 5$  より約 5% である。このため、I/O 処理と TLB フラッシュの影響を考慮しても、デバイスドライバ内での遅延は 5% 程度になると見積もれる。

### 4. おわりに

デバイスドライバをプロセスとして実現した場合について、仮想空間切り替えによるデバイスドライバの処理性能への影響を述べた。

デバイスドライバをプロセスとして実現した場合、ドライバ処理、および割り込み処理の呼び出しの際に仮想空間の切り替えが発生する。また、仮想空間切り替えに伴い、TLB のミスヒットも発生する。これらの影響を測定した結果、仮想空間切り替え処理のオーバーヘッドは約 380 クロック、TLB ミスヒット 1 回当たりのオーバーヘッドは約 50 クロックであることがわかった。これらによる処理時間の遅延は、デバイスドライバ内の処理の 5% 程度と見積もれる。

残された課題として、応用プログラムを用いた評価がある。

本研究の一部は、科学研究費補助金 基盤研究 (B) 「適応性と頑健性を有する基盤ソフトウェアのカーネル開発」(課題番号: 18300010) による。

### 参考文献

- [1] 野村裕佑, 乃村能成, 横山和俊, 谷口秀夫, 丸山勝巳, "走行モード変更機構を利用したデバイスドライバの実現," 情処研報 2006-OS-101, pp. 25-31, Feb. 2006.