

日曜研究室

技術的な観点から日常を綴ります

[xv6 #22] Chapter 2 – Traps, interrupts, and drivers – Systems calls, exceptions, and interrupts

テキストの31～32ページ

本文

ユーザプログラムはシステムコールを使って、第1章の最後で見たように、OSのサービスに対して問い合わせることが出来る。

例外（exception）という用語は、割り込みを発生させるプログラムの異常な動きの事である。プログラムの異常な動きの例としては、ゼロ除算、無効なPTEが指すメモリにアクセスしようとする事、などが挙げられる。

割り込み（interrupt）という用語は、ハードウェアから生成された信号の事であり、OSに対して注意が必要だということを指し示す。

例えば、カーネルが時分割の機能を実装する事を可能にするために、クロックチップは100ミリ秒毎に割り込みを発生させる。

他には、ディスクが一つのブロックを読み終わったとき、そのブロックのデータが取得可能だということをOSに知らせるために、ディスクが割り込みが発生させるという例がある。

それぞれのプロセスが割り込みを制御するのではなく、カーネルが割り込みを制御する。

なぜなら、ほとんどの場合、必要な権限と状態をカーネルだけが持つからである。

例えば、クロック割り込み応答によって、複数のプロセス間で時分割させるために、非協力なプロセスにプロセッサを強制的に手放させさえすれば、カーネルは関係しなければならない。

（ちょっと意味不明ですが、カーネルがプロセスの切替を行うので割り込みもカーネルで受け取ったほうが都合がいいということかなと）

以上のような場合では、OSのデザインはその後に起きる事を解決しなければならない。

システムは、後で全く同じように復元するためにプロセッサのレジスタを保存しなければならない。

システムは、カーネルを実行するためにセットアップしなければならない。

システムは、カーネルが実行を開始する位置を選ばなければならない。

カーネルは、そのイベントに関する情報、例えばシステムコールの引数、を復元できなければならない。

それらは安全確実に完了しなければならない。

システムは、ユーザのプロセスとカーネルの独立性を維持しなければならない。

このゴールを達成するために、OSはハードウェアがどうやってシステムコールや例外や割り込みを制御するかの詳細を承知しておかなければならない。

多くのプロセッサのその3つのイベントは、一つのハードウェア機構によって制御される。

例えば、x86では、int命令を使って生成された割り込みによってプログラムはシステムコールを呼び出す。

同様に、例外も割り込みを発生させる。

このように、もしOSが割り込みの制御に関する計画を持っているなら、OSはシステムコールと例外を制御することが出来る。

その計画の基本的なものは以下のとおりである。

割り込みが通常のプロセッサループを止め、割り込みハンドラ（interrupt handler）と呼ばれる処理の実行を開始する。

割り込みハンドラの実行を開始する前に、プロセッサはそのレジスタを保存する。

それによってOSは割り込みから戻ってきたときにレジスタを復元することが出来る。

この移り変わり（割り込みハンドラへの処理の移行と割り込みハンドラから処理がもどること）における挑戦は、プロセッサがユーザモードからカーネルモードへ、またその逆へ切り替わらなければならないという事である。

専門用語について。

正式なx86用語は割り込み（interrupt）とは言え、x86はそれらの全てをトラップ（traps）として指す。

PDP11/40で使われてた用語であり、ゆえにUnixの伝統的な用語であるというのが主な理由である。

この章では、トラップと割り込みという用語を置き換え可能なものとして使う。

しかし、トラップはプロセッサ上で実行中の現在のプロセスに引き起こされる（例えば、プロセスがシステムコールを使うと結果的にトラップが生成される）のに対し、割り込みは現在のプロセスに関係のないデバイスによって引き起こされるという違いにはしっかり注意しなさい。

例えば、ディスクが、あるプロセスのために一つのブロックを読み込み終わったとき、割り込みが派生するが、その割り込みが発生した瞬間は別のプロセスを実行中だろう。

割り込みのこの特性は、トラップについて考えるより割り込みに着いて考えることを難しくしている。

なぜなら、割り込みは他の活動状態と並列に発生するからである。

しかし、両方とも、ユーザーモードとカーネルモード間で安全確実に制御を移すために同じハードウェア機構に依存している。

それについては次に説明する。

感想

例外でも割り込みが発生するし、システムコールでも発生するし、デバイスからも発生するということみたいです。

int命令について。

[http://en.wikipedia.org/wiki/INT_\(x86_instruction\)](http://en.wikipedia.org/wiki/INT_(x86_instruction))

INT is an assembly language instruction for x86 processors that generates a software interrupt.
ということみたいです。

「x86はそれらの全てをトラップ (traps) として指す」の部分はxv6の間違いじゃないかという気がします。

まあとにかくinterruptがx86の正式な用語だけどPDP11の用語であるtrap (ゆえに昔ながらのUnix用語でもある) を使うってことですかね。

これみたら確かにPDP11では割り込みをtrapと呼んでるみたいです。

http://en.wikipedia.org/wiki/PDP-11_architecture

カテゴリー: 技術 | タグ: xv6 | 投稿日: 2012/2/27 月曜日 [<http://peta.okechan.net/blog/archives/1345>] |
