

1G-7 OSデバッグ環境構築のためのハイパOSの実現

清水正明, 早川栄一, 並木美太郎, 高橋延匡
(東京農工大学)

1. はじめに

オペレーティング・システム(以下、OS)は、ハードウェアの資源管理のために非同期処理を行うため、デバッグや再現性のあるテストが困難である。これに対して、我々の研究室では、OSの開発、デバッグ、評価のために、デバッグ機能を持つ仮想計算機モニタを使用したデバッグ環境を構築している[1][2]。

本稿では、このOSデバッグ環境の基本機能を提供する、ハイパOS「忍(SHINOBI)」の設計と実現について報告する。

2. OSデバッグ環境

我々は OS/omicon 等のOSのデバッグの経験から、OSのデバッグ環境の設計目標を次に定めた。

- (1)被デバッグOSにデバッグ用の仕掛けを入れずにデバッグ可能にすること
- (2)デバッグ・ツール、ノウハウが違うOSのデバッグに流用できること
- (3)非同期(割込み)処理を含めて動作に再現性があること

また設計方針では、設計目標を満たすために、アプリケーション層にデバッグ、OS層にデバッグOS、ハイパOS層にハイパOS「忍」という、三層構成を採った。

ハイパOS「忍」では、仮想計算機という方法を採用することで、デバッグで割込みの管理を行えるようになった。

また、ハイパOS「忍」ではデバッグ環境の基本機能のみを提供し、「忍」上に具体的なデバッグ機能を提供する層を別に設けることで、OSに依存せず、自由に拡張可能なデバッグを実現することができるようになった。

OS/omicon をデバッグするときの具体的な例を図1に示す。

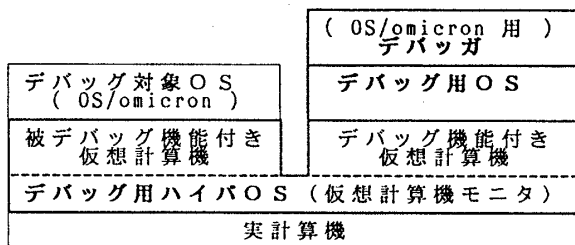


図1 OSデバッグ環境の構成

3. ハイパOS「忍」の設計

ハイパOS「忍」は、OSデバッグ用の機能を持つ仮想計算機モニタであり、仮想計算機機能、デバッグ機能の、二種類の機能を持っている。

3.1 仮想計算機機能

仮想計算機機能としては、Motorola MC68000 同等の仮想計算機環境を複数個提供する。また、入出力はハイパOSへの HyperVisorCall(HVC) によって行う。デバッグ用OSと被デバッグOSは、別の仮想計算機上で実行されるため、デバッグ時以外に互いの動作に影響を与えることはない。

3.2 OSデバッグのための機能

ハイパOS「忍」には、いくつかのOSデバッグのための機能があり、その機能を、ハイパOS上のOSである「デバッグ用OS」から HVC として使用可能である。OSのデバッグのために次の機能を用意した。

3.2.1 仮想計算機の操作

(1) 仮想計算機の状態を操作する

仮想計算機の状態には、

running : 現在アクティブな仮想計算機の状態

ready : 現在アクティブではないが、実行待ちの状態

suspended: サスペンドされた状態

stop : 仮想計算機が STOP 命令で停止した状態

halt : 二重バス障害で仮想計算機が HALT の状態

がある。デバッグ時に、デバッグ対象OSの仮想計算機を suspended 状態にすることで、デバッグ対象OSの動作を一時停止することができる。この状態遷移図を図2に示す。

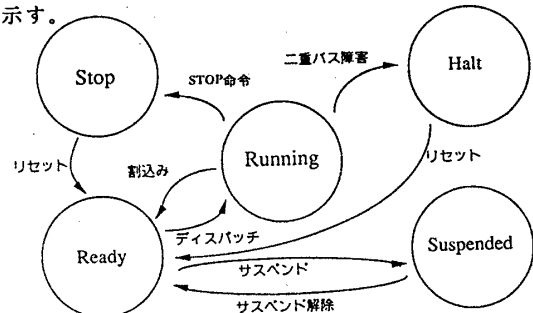


図2 仮想計算機のプロセッサ状態の遷移

(2) 仮想計算機のレジスタを操作する

デバッグ時に、デバッグ対象OSの仮想計算機のレジスタを操作をすることができる。

(3) 仮想計算機のメモリを操作する

ハイバOS上の複数の仮想計算機には、それぞれのメモリ領域が重ならないようにメモリを割り当てる。また、仮想計算機同士でメモリの操作が可能である。デバッグ時には、ハイバOSとデバッグは、被デバッグOSから破壊されないように保護される。

3. 2. 2 割り込みの監視、発生

(1)内部、外部割り込みの監視

OSの内部動作を詳しく知るために、仮想計算機への割り込みを監視できるようにした。監視するのは、外部(ハードウェア)割り込みと、内部(ソフトウェア)割り込みである。

(2)内部、外部割り込みの発生

OSの動作に再現性を出すために、指定仮想計算機に対して割り込みを発生できるようにした。したがって、デバッグ時に任意のシーケンスで、仮想的な割り込みを発生することができる。

これらの(1)、(2)の機能を実現するために、各仮想計算機に、図3のような割り込みキューを用意し、割り込みを管理している。デバッグ時に割り込みを起こしたいときには、被デバッグOSの割り込みキューを操作する。また、監視していた割り込みが発生したときには、被デバッグOSの割り込みキューではなく、デバッグ用OSの報告キューに割り込みデータがつながれる。割り込みの監視、発生機能によって、入出力(HVC)のエミュレーションが可能になった。

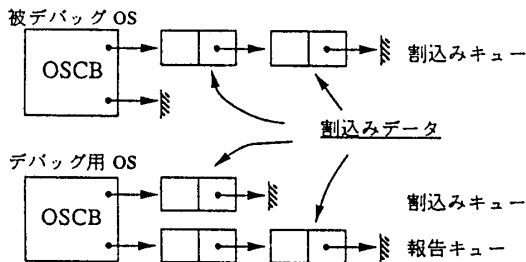


図3 割り込みキュー、報告キューの構成

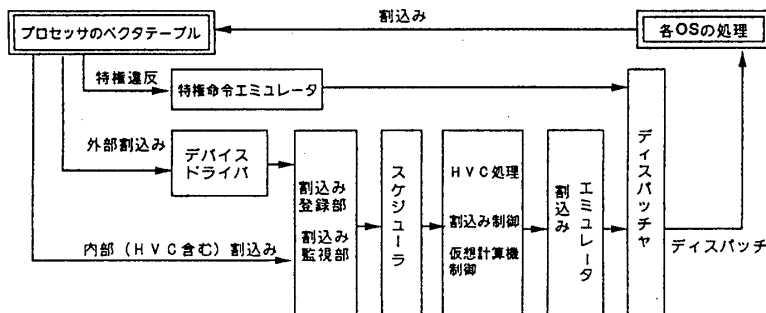


図4 ハイバOS「忍(SHINOBI)」の構成

4. ハイバOS「忍」の実現

ハイバOS「忍」は、図4の構成で実現した。デバイス・ドライバは、江戸[3]のものを利用し、カーネルは言語C及び、アセンブラで記述した。コーディング量は次の通りである。

カーネル	1000	行
デバイス・ドライバ	200	行
管理プログラム	600	行
ユーティリティ	800	行
アセンブラ	700	行
合計	21000	行
コメント	2000	行

また、デバッグ用OSに提供するデバッグ用HVCを表1に示す。

表1 デバッグ用HVC

機能番号	機能
\$F000	デバッグ用HVCの初期化
\$F001	デバッグ用HVCの終了
\$F002	デバッグ用HVCの割り込み発生
\$F003	デバッグ用HVCの割り込み発生
\$F010	デバッグ用HVCの割り込み発生
\$F011	デバッグ用HVCの割り込み発生
\$F012	デバッグ用HVCの割り込み発生
\$F013	デバッグ用HVCの割り込み発生
\$F014	デバッグ用HVCの割り込み発生
\$F020	デバッグ用HVCの割り込み発生
\$F022	デバッグ用HVCの割り込み発生
\$F023	デバッグ用HVCの割り込み発生
\$F024	デバッグ用HVCの割り込み発生
\$F030	デバッグ用HVCの割り込み発生
\$F031	デバッグ用HVCの割り込み発生
\$F040	デバッグ用HVCの割り込み発生
\$F041	デバッグ用HVCの割り込み発生

現在、ハイバOS「忍」は、日立製作所 2050/32 ワークステーション上で稼働し、OS/omicon 第二版、OMICRON V3 の実行環境を提供することができる。

5. おわりに

OSデバッグ環境が実現すると、割り込み、入出力など、かなりのOSの機能のデバッグ環境を与えることができるようになる。特に、OSを実働しながら情報を追記型光ディスクにロギングするにすれば、OSの評価に有効である。

参考文献

[1]G.R.Allred :System/370 Integrated Emulation under OS and DOS", SJCC, 1971
 [2]清水正明、他 :Hyper-OS によるOSのデバッグ環境の研究、情報処理学会第42回全国大会論文集(1991)
 [3]岡野裕之、他 :多重OS「江戸」の設計と実現、情報処理学会論文誌、Vol.30, No.8, pp.1012-1023(1989)