

## MKng プロジェクトにおけるマイクロカーネル移植技法: 3 Z-8 デバッグツールを利用した PowerPC への移植†

岩崎 礼江 緒方 正暢

日本アイ・ビー・エム (株) Embedded Systems 事業

### 1 はじめに

慶應義塾大学が中心になって研究開発している MKng プロジェクト [1] では, Real-Time Mach マイクロカーネル, および, 各種サーバを SPARC[2], MIPS[3], PowerPC, Pentium プロセッサを搭載した各種コンピュータ上に実装している. 本稿では, ソフトウェアシミュレータやハードウェアデバッグといったデバッグツールを使った RT-Mach マイクロカーネルの PowerPC コンピュータ上への移植技法について述べる.

### 2 ソフトウェアシミュレータ上への移植

まず, 移植の第1段階として, ソフトウェアシミュレータである PowerPC Visual Simulator[4](以下, PVS と略す)への移植を行なった. PVS とは, 各種 PowerPC プロセッサのアーキテクチャ (命令セットおよび, マイクロアーキテクチャ) をソフトウェアだけで実現したシミュレータである (図 1). IBM AIX 上で動作し, プロセッサ内のレジスタなどの資源を X-windows/Motif を使って視覚的に操作することができる. PowerPC 601, 603/603e, 604/604e, 620, 403GA がサポートされている. ユーザモードのレジスタはもちろん, 特権モードのレジスタも完全に実現されており, アドレス変換などオペレーティングシステムを実装するための機能が利用できる.

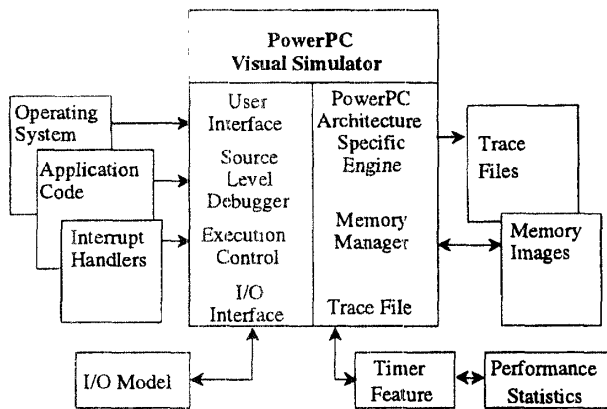


図 1: PVS の構成

実機上では, システムおよびプロセッサの初期化は ROM 内のファームウェアで行なわれる. カーネルは, ファームウェアから起動されるが, 起動の方法はシステムアーキテクチャにより異なり, また, OS のブートイメージのフォーマットもまちまちである. そこで, シミュレータ上でカーネルを起動させるために, プロセッサの初期化, 本来ブート時にシステムから渡さ

Microkernel Porting Technology in the MKng Project:  
Porting RT-Mach to the PowerPC with Debugging Tools  
Norie Iwasaki and Masanobu Ogata  
ESBU, IBM Japan, Ltd.  
1623-14, Shimotsuruma, Yamato, Kanagawa 242, Japan  
E-Mail: <norie@jp.ibm.com>

†この研究は, 情報処理振興事業協会 (IPA) が実施している創造的ソフトウェア育成事業「次世代マイクロカーネル研究プロジェクト」のもとに行われた.

れるメモリサイズなどの情報を受けとる部分もシミュレータ用の実装した. カーネルのコンパイルは, AIX 上で行ない, カーネルのオブジェクトフォーマットは XCOFF を使った. コンパイルしたカーネルは, PVS のロード機能を使って読み込まれる. このときの, カーネルのロード開始アドレスは PVS から指定した.

本来, 実機上で起動されるカーネルは, マイクロカーネル (mach\_kernel) とデフォルトページャ (bootstrap) の 2 つの実行可能形式ファイルを makeboot コマンドを使って 1 つの実行可能ファイル (mach.boot) にしたものである. bootstrap は, OS パーソナリティの初期化を行なった後, デフォルトページャとして機能する. しかし, Lites のようなサーバを動作させるためにはハードディスクなどの入出力装置が必要となる. PVS には, 入出力装置をモデル化し, メモリマップド I/O を実行する機能が備わっているが, サポートするデバイスは必要最低限のものに限定して PVS 用のスタブドライバとして実装した. キーボード入力と画面出力は, PVS の標準入出力ライブラリを使って実現したので, カーネルからの printf() によるメッセージをコンソールである X 端末に出力できるようにした. 図 3 に PVS を使ったデバッグ中の画面を示す.

カーネルを移植する場合, プロセッサの初期設定, 例外ハンドラ, メモリ管理部分など実機上でのデバッグが困難な部分をまず行なわなければならない. 今回, シミュレータを使うことで, アセンブラで記述されたカーネルのコアの部分の移植およびデバッグが効率良く行なえた.

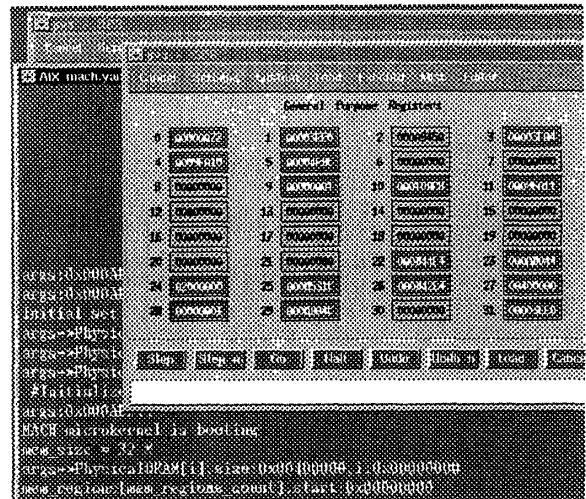


図 3: PVS を使ったデバッグ環境

### 3 ハードウェアデバッグを利用した移植

移植の第2段階として, JTAG (IEEE 1149.1) ポートを使ったハードウェアデバッグである IBM RISCWatch (図 2) を利用した. 移植ターゲットは, ThinkPad Power Series 850 と同じチップセットを搭載した PowerPC 603e 評価ボードを用いた. RISCWatch では, 実機上の PowerPC プロセッサのレジスタ

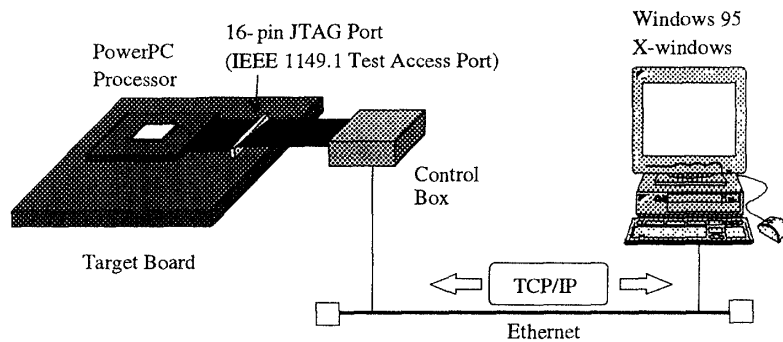


図 2: RISCWatch を使ったデバッグ環境

やメモリへのアクセス、プログラムのダウンロード、システムの起動、停止、ブレークポイントの設定ができる。

ボード上には、チップセットとして、MPC105 PowerPC-PCIブリッジチップ、Intel 82378ZB PCI-ISA バスブリッジが搭載されている。ISA バス上には、SuperI/O チップが実装されており、RS-232 シリアル、パラレルポート、フロッピディスクポートが取り付けられている。さらに、PCI 拡張スロットと ISA 拡張スロットが用意されているので各種拡張アダプタを取り付けてデバイスドライバのデバッグを行なうことができる。

この段階では、PCI や ISA バスの初期化、デバイス情報の取得、初期化部分を主にデバッグした。プログラムの開発は、AIX 上で行ない、カーネルのオブジェクトフォーマットは XCOFF を使った。カーネルの起動は、JTAG ポート経由で RISCWatch からカーネルイメージをダウンロードし、カーネルのロード開始アドレスはデバッガから指定した。入出力として、ボード上の RS-232 を使ったシリアルコンソールも実装し、カーネルメッセージはシリアルポートへ出力させた。

## 4 実機への移植

### 4.1 ターゲットシステム

PowerPC プロセッサを搭載しているコンピュータのアーキテクチャは、主に以下の 5 種類があるが、我々が移植の対象とするのは、PReP (PowerPC Reference Platform)[5]、PCI Power Macintosh (以下、PowerMac と略す) と CHRP[6] の 3 つである。PReP システムは、PowerPC パソコン用の規格として登場した。この規格に準拠した製品が、IBM や Motorola から出ている。今回、ターゲットとしている ThinkPad 850 もこの規格に準拠している。また、多くの組み込みシステムもこの規格に準拠している。PCI PowerMac は、現在最も、普及している PowerPC を搭載したコンピュータである。PowerBook シリーズでも PCI アーキテクチャを採用している。我々がターゲットとして予定している PowerBook 2400 はこの規格に準拠している。CHRP は、PReP、PowerMac 規格を統合する形で規定された最新の規格である。1 台のコンピュータでビッグエンディアンの Mac OS やリトルエンディアンの Windows NT が選択的にブートできるように設計されている。

これらのシステムでは、いずれも PCI バスを採用しており、利用されるデバイスは Intel x86 系のパソコンと共通のものが多い。Open Firmware (以下、OF と略す)[7] を搭載したコンピュータでは、デバイスの構成、デバイス情報の取得は、OS が行なうのではなく OF が行ない、その情報を起動時に取得する。

### 4.2 カーネルデバッガを使った実機上のデバッグ

基本的な OS の部分のデバッグが完了した後のデバッグでは、アセンブラレベルでのステップ毎の実行ではとても対処できない。やはり、ソースコードに対応したシンボリックデバッガが必

要である。これを行なうには、GDB[8] などのリモートデバッグ機能を利用する。もともと Mach マイクロカーネルにもデバッガが装備されているのでこれを利用することにした。PowerPC 組み込みシステムや、PowerPC 拡張ボードのような実機でも JTAG ポートを備えているものがあり、これらのターゲットシステムは、デバッガ自身の動作が安定しないことも多い。このような場合には、ハードウェアデバッガとシンボリックデバッガを組み合わせたデバッグを行なうことができる [9]。

## 5 おわりに

本稿では、マイクロカーネルの移植を効率良く行なうためにソフトウェアシミュレータやハードウェアデバッガを利用した技法について述べた。

## 参考文献

- [1] 徳田 他: “MKng: 次世代マイクロカーネル研究プロジェクトの概要,” 第 55 回情処全大論文集, 1Z-2 (1997).
- [2] 船渡 他: “MKng プロジェクトにおけるマイクロカーネル移植技法: SPARC 用マイクロカーネルの実装と性能評価,” 第 55 回情処全大論文集, 3Z-6 (1997).
- [3] 光澤 他: “MKng プロジェクトにおけるマイクロカーネル移植技法: MIPS 用マイクロカーネルおよび UNIX サーバの性能評価,” 第 55 回情処全大論文集, 3Z-7 (1997).
- [4] Armstead, M., et al.: “PowerPC Visual Simulator: Peeking under the Hood of the PowerPC,” IEEE Intl Conf on Computer Design: VLSI in Computers and Processors, pp. 413-418 (1994).
- [5] IBM: PowerPC Reference Platform Specification Version 1.10 (1994).
- [6] Apple Computer, IBM, Motorola: PowerPC Microprocessor Common Hardware Reference Platform: A System Architecture, Morgan Kaufmann Publishers, Inc. (1995).
- [7] IEEE Std 1275-1994, IEEE Standard for Boot (Initialization Configuration) Firmware, Core Requirements and Practices (1994).
- [8] Stallman, R. M.: “Debugging with GDB version 4.16,” Free Software Foundation (1994).
- [9] 河内谷, 森山: “PowerPC を用いたハードウェアシステム用の GDB サーバ,” 第 55 回情処全大論文集, 6Z-4 (1997).