

## Mach マイクロカーネルシステムの MIPS アーキテクチャへの移植\*

光澤 敦<sup>†</sup> 林 和則<sup>†</sup> 田中 浩一<sup>†</sup>  
<sup>†</sup>NTT 情報通信研究所 <sup>†</sup>ソニー IT カンパニー

R4400 や R10000 をはじめとする最近の MIPS プロセッサは、高性能 64 ビット RISC プロセッサであり、TLB、キャッシュ、マルチプロセッササポート等の機能は、OS 研究のためのプラットフォームとして非常に興味深い。著者らは、次世代マイクロカーネル研究プロジェクト (MKng プロジェクト) のサブタスクとして、Mach3 マイクロカーネルおよび 4.4BSD Lite サーバの MIPS アーキテクチャへの移植を行なっている。本稿では、R3000, R4400, および R10000 を用いたソニー製 NEWS ワークステーションへの移植について報告する。

### Porting Mach3 Microkernel and 4.4BSD Lite Server to MIPS Architectures

Atsushi Mitsuzawa<sup>†</sup> Kazunori Hayashi<sup>†</sup> Kouichi Tanaka<sup>†</sup>  
<sup>†</sup>NTT Information and Communication Systems Laboratories  
<sup>†</sup>Information Technology Company, Sony Corporation

In Next Generation Microkernel Research Project, we are developing a new microkernel to manage system resources and provide useful services with minimal overhead on modern computer systems. Sony NEWS workstations are such computer systems based on MIPS architectures that employ 64-bit addressing, software managed TLB, large cache memories, and multiprocessors. Therefore, we are porting Mach3 microkernel and 4.4BSD Lite server to Sony NEWS to provide platforms for operating system's research. This paper describes our experience during the port to R3000 and R4400 based systems.

#### 1 はじめに

近年オペレーティングシステムの研究プラットフォームとして、PC 互換機が多く用いられている。PC 互換機の価格性能比が他の CPU を用いた計算機と比較して高いためであろう。このことは、PC 互換機に搭載されている Pentium をはじめとする i386 互換 CPU についてもあてはまる。しかしながら、PC 互換機のアーキテクチャは、そのままでは将来性に疑問が残る。

例えば、最も重要なソフトウェア動向の一つとして、プログラムやデータによって使用されるメモリ量が年に 1.5 倍から 2 倍の割合で増大している事実がある [1]。これはアドレス・ビットの消費率に換算すると、年に 1/2 ビットから 1 ビットに相当する。そのため、i386 互換 CPU 以外のほとんどの CPU アーキテクチャが 64 ビットアドレッシングを提供している。また最近の

研究によれば、キャッシュや TLB で構成されるメモリシステムが計算機の性能上のボトルネックの一つになることが予想され、マルチプロセッサ計算機ではカーネルサービスに要する時間がシングルプロセッサ計算機の場合より大きくなることが指摘されている [2]。これらの問題のうちメモリシステムの問題に関しては、i386 互換 CPU 以外の CPU アーキテクチャは対応しつつある。

そこで我々は、慶應義塾大学環境情報学部における次世代マイクロカーネル研究プロジェクト (MKng プロジェクト) [3] のサブタスクとして、MIPS アーキテクチャ [4] 上で次世代マイクロカーネルの研究を行なっている。このサブタスクの目的は、MIPS アーキテクチャの先進性を利用することにより、64 ビットアドレッシングで動作するマルチプロセッサ計算機において、効率の良い資源管理を行なうことのできる次世代マイクロカーネルを開発することである。

本研究では、その基盤環境構築を目的として、

\*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」のもとに行なわれました。

Mach3 マイクロカーネル [5] および 4.4BSD Lite サーバ [6] を、MIPS プロセッサを搭載したソニー製ワークステーションに移植する。本論文では、まず MIPS アーキテクチャにおける研究課題について述べる。次に、プラットフォームであるソニー製ワークステーションについて述べた後、移植戦略について述べる。そして、移植作業の現状および主な変更箇所について述べる。最後に今後の課題について述べ、まとめる。

## 2 MIPS アーキテクチャにおける研究課題

MIPS アーキテクチャにおける研究課題の一つは、64 ビットアドレッシングのサポートである。サポートしなければならない理由は二つある。一つは、プログラムやデータによって使用されるメモリ量が年に 1.5 倍から 2 倍の割合で増大している事実からであり、もう一つはデータベース処理および連続メディアデータの処理で必要だからである。

しかしながら、既存の 32 ビットアドレス空間における物理から仮想へのアドレス変換機構を用いて 64 ビットアドレス空間を管理するには二つの問題がある。一つは、相対的に TLB(Translation Lookaside Buffer) のエントリ数が減少することになるため、TLB のヒット率が減少することである。この問題に対するアーキテクチャでの解決技術として、スーパーページおよびサブブロックが存在する。これらの技術の有効性はシミュレーションを通して実証されている [7]。もう一つは、ページテーブルの参照回数増加によりプログラムの実行時間に占めるアドレス変換の時間が大きくなる、あるいはページテーブル自体の大きさが大きくなることにより有効な物理メモリ量が減少することである。最近の CPU アーキテクチャの多くは、TLB ミス時の動作すなわちページテーブルの参照をソフトウェアに依存しているため、この問題はオペレーティングシステムだけで解決できる。既にクラスタ化ページテーブル [7] とガード付きページテーブル [8] が提案され、シミュレーションを通してその有効性が実証されている。そこで、R4400 および R10000 [9] に実装されたスーパーページおよびサブブロックの機能を用いて、これらの技術を用いた新しいアドレス変換機構を Mach3 マイクロカーネルに実装し性能評価する予定である。

MIPS アーキテクチャにおけるもう一つの研究課題は、マルチプロセッサのサポートである。これには 5 つの小さい課題がある。第一に、プロセッサの同期機構およびプロセッサ間のデータのやり取りを行なうための高速なプロセッサ間通信機構の開発である。プロセッサの同期機構では、カーネルモードでの実行を任意のプロセッサで実行できるような短期間および長期間のロック機構を開発する。プロセッサ間通信では、1 対 1 および 1 対多の通信機能の両方を開発する。第二に、マルチプロセッサに対応した仮想記憶機能の開発である。MIPS アーキテ

キテクチャにおいては、TLB は一つキャッシュとして機能する。そのため、グローバルなページテーブルエントリを変更した場合には、対応する TLB のバージが必要である。マルチプロセッサシステムにおいては、すべてのプロセッサの TLB をバージしなければならない。この機能をマルチプロセッサにおける高速なプロセッサ間通信機構を用いて実現する。第三に、マルチプロセッサ対応の割り込み処理機構の開発である。デバイスドライバの割り込み処理を特定のプロセッサに割り当てて行なうか、複数のプロセッサに割り当てて行なうかを、デバイスドライバ毎にコンフィギュレーション可能にするための機構を実現する。また、コンフィギュレーションの動的な変更を可能にするための機構を実現する。第四に、スレッドのプロセッサへの効果的な割り当てを考察する。ここでは、4.4BSD Lite サーバおよび GNU Hurd の UNIX の二つの実装方式をマルチプロセッサ計算機で動作させた時の性能を実測する予定である。最後は、Real-Time Mach [10] の提供するリアルタイム機能をマルチプロセッサ環境で実現することである。

## 3 プラットフォーム

前章で述べた課題を研究するには、その基盤環境を構築することが必要である。MIPS アーキテクチャで動作するマイクロカーネルに、米カーネギーメロン大学 (CMU) が開発した Mach3 および Real-Time Mach がある。これらは R3000 を搭載した DEC 社のワークステーションである DECstation で動作する。ただし、DEC 社は MIPS プロセッサの採用を 32 ビットプロセッサである R3000 にとどめ、64 ビットプロセッサとして Alpha を採用した。そのため、R4400 や R10000 等を用いた他のプラットフォームを考える必要があった。

そこで本研究では、ソニー製 NEWS ワークステーション (以下、NEWS と呼ぶ) をプラットフォームとする。移植に用いた NEWS の種類およびそのごく簡単な仕様を示したのが表 1 である。

表 1. MIPS プラットフォーム

NWS-	3150	5000	7000
CPU	R3081	R4400SC	R10000
(bit)	32	32/64	32/64
NCPU	1	1	1,2,4
MM	8M-32M	32M-512M	32M-2G
バス	-	APbus	APbus

ここで、プロセッサの性能としては既に見劣りする R3081<sup>1</sup> を搭載した NWS-3150 へも移植を行なうのは、同等のプロセッサである R3000 を搭載した DECstation で動作する Mach3 マイ

<sup>1</sup>R3081 は、R3000 とバイナリレベルで互換性があるプロセッサである。

クロカーネルがあるので、プラットフォームの相違点だけに着目して移植することができるからである。なお、相違点には次のようなものがある。

- エンディアン。MIPS プロセッサはコンフィギュレーションレジスタの設定によって、ビッグエンディアンとしてもリトルエンディアンとしても動作させることができるが、本研究ではビッグエンディアンで動作させることにしている。なぜなら、ソニー社が提供する NEWS-OS のビッグエンディアンのバイナリをそのまま利用したいからである。一方、DECstation で動作する Mach3 マイクロカーネルは DEC 社が提供する Ultrix や OSF/1 と同様リトルエンディアンを採用しているため、エンディアンが異なることになる。Mach3 マイクロカーネルはどちらのエンディアンでも動作するようにプログラムされているが、一部の MIPS アーキテクチャのコードは DECstation 用にリトルエンディアンを仮定して書かれているコードが存在する。
- バックプレーンバスおよび割り込み。これには、(1)バックプレーンバスのための構造体、(2)ROM サービスルーチンの呼び出し方法、および(3)ROM からカーネルへの引数の引渡し、を変更する必要がある。
- デバイス。NEWS-3150 が使用しているデバイスのうち、Mach3 マイクロカーネルがサポートしているのは、シリアルコントローラである Zilog Z8530 だけである<sup>2</sup>。他のデバイスドライバは新たに実装する必要がある。

NWS-5000 プラットフォームは、二つの目的に使用する。一つは、R4000 以降の MIPS プロセッサに採用された、(1)ページテーブルエントリおよび(2)TLBmiss に関する例外ベクタ構成への対応である。これらは、マイクロカーネルの実装において、物理メモリ管理および例外処理を大幅に変更しなければならないことを意味する。それで NWS-5000 における最初の移植は、R4400 を 32 ビットモードで動作させ新しい物理メモリ管理および例外処理の実装を行なう。もう一つは、研究課題の一つである 64 ビットアドレッシングにおける新しいアドレス変換機構の実装である。

NEWS-7000 は、マルチプロセッサのサポートに関する研究を行なうためのプラットフォームである。デバイスドライバをマルチプロセッサ対応にする必要がある。

移植するマイクロカーネルは、Mach3 マイクロカーネルでありバージョン MK83 (次節以降では MK と略す)である。Real-Time Mach マイクロカーネルでないのは、Mach3 の方がバグが少

<sup>2</sup>正確に言うと、NWS-3150 が採用しているシリアルコントローラは Z8530 の機能強化版である Z85230 である。また、NWS-5000 や NWS-7000 におけるシリアルボードも Z85230 を採用しているが DMA が可能になっている。これらの機能をフルに使うのであれば、デバイスドライバを大幅に書き換える必要がある。

なく安定していると考えたからである。将来的には、Real-Time Mach へ移行する。MK83 を選んだ理由は、Real-Time Mach が MK83 から派生したからである。UNIX サーバとして移植するのは、やはり DECstation をプラットフォームとして開発されたヘルシンキ工科大学の 4.4BSD Lite サーバである。バージョンは、Lites-1.1.u3(次節以降では Lites と略す)である。他の UNIX サーバに、米カーネギーメロン大学 (CMU) の UX サーバがあったが、ライセンスフリーではないので除外した。なお、MIPS 版の 4.4BSD Lite サーバには、当初非同期例外処理が実装されていなかったため、本研究で実装する。

#### 4 移植戦略

前節で述べた各プラットフォームの役割から、移植は表 2 に示す 4 つのステップに分けて行なうのが適当であると考えた。ステップを経て移植することにより、MK および Lites に対する理解が徐々に深まり、結果的に移植に要する時間を短縮できる。表 2 には、各ステップでのプラットフォーム、コンパイルに使用した計算機、コンパイラ、および実行可能形式を記した。ステ

表 2. プログラミング環境

Step	1	2	3	4
機種	3150	5000	5000	7000
ホスト	NEWS	NEWS	SGI	SGI
OS	NEWS-OS4(BSD)		IRIX6	
cc	GCC 2.7		MIPSPPro 7.0	
as	MIPS as-2.11		MIPSPPro 7.0	
ld	MIPS ld		MIPSPPro 7.0	
a.out	ECOFF(static)		ELF(static)	

プ 2 まではコンパイル環境に NEWS 上で動作する NEWS-OS4 を使用した。これは実行可能形式を DECstation と同じ ECOFF にすることにより、ローダの実装を省きたかったからである。ステップ 3 からはシリコングラフィック社 (以下、SGI 社) の IRIX6 を使用した。なぜなら、(1)64 ビットアドレッシングのコード生成、(2)MIPS-ISA (MIPS Instruction Set Architecture) III 以降のコード生成、および(3)マルチプロセッサのためのコード生成、をすべて行なう必要があったからである。なお、ステップ 3 以降の実行可能形式が ELF なのは SGI 社のコンパイラが ELF しかサポートしていないためである。

以下でそれぞれのステップについて説明する。なお、ステップ 3 およびステップ 4 は並行して作業可能である。

- ステップ 1: MK および Lites の NWS-3150 への移植。Lites がシングルユーザモードで動作し、NEWS-OS4 の ls 等が動作する、すなわち NEWS-OS4 とのバイナリ互換性を提供することが目標である。デバイスドライバは、シリアルドライバのみ移植する。

/bin/sh がシグナルハンドラを登録するので、Lites への非同期例外処理の実装はこのステップで行なう。

- ステップ 2: MK の NWS-5000 への移植。MK の R3000 から R4400 への移植である。SCSI, イーサネット, フレームバッファの実装を行なう。Lites がマルチユーザモードで動作することが目標である。
- ステップ 3: MK および Lites における 64 ビットアドレッシングのサポート。MK および Lites に ELF のローダを実装する。
- ステップ 4: MK の NWS-7000 への移植。MK を R10000 への移植しマルチプロセッサ対応にする。ただし、R4400 と R10000 の違いは、R3000 と R4400 の違いと比較すると少ないので、マルチプロセッサ対応が主である。なお、Lites の基になる 4.4BSD Lite はマルチプロセッサ対応がなされているので、Lites は大きな変更なしにマルチプロセッサ計算機で動作すると考えられる。

上記のことに加えて、Lites の移植にあたっては考慮すべき事項が他にも二つある。すなわち、

1. ディスク上のファイルシステムに、4.3BSD の FFS(Fast File System) と 4.4BSD の FFS のどちらを用いるか?
2. コマンドは、NEWS-OS4 のバイナリがそのまま動けばいいのか、それとも 4.4BSD Lite のコマンドをコンパイルする必要があるのか?

1 は二者択一の問題である。ステップ 1 では、4.3BSD のファイルシステムを使用していた。これは、開発にあたってコンパイルおよび実行の作業時間を短縮させるため、プラットフォームをディスクレス計算機として動作させたためである。というのも、NEWS-OS4 が提供する RD(Remote Disk) のプロトコルを使用したファイルサービスは、NEWS-OS4 のファイルシステム、すなわち、4.3BSD のファイルシステムを提供するためである。4.3BSD の FFS の使用は、Lites をシングルユーザモードで動作させる場合は問題がなかった。しかし、Lites はシングルユーザモードでブートする時に、ルートパーティションを書き込み禁止でマウントし、マルチユーザモードに移行する際に mount コマンドを用いて書き込み可能な状態に再マウントする。問題は、NEWS-OS4 では 4.3BSD と同様当初よりルートを書き込み可能な状態でマウントするため、NEWS-OS4 の mount コマンドには再マウントの機能が不要なく、したがって実装されていないことである。これに対する解決法には、

- NEWS-OS4 の mount コマンドは、4.3BSD の mount コマンドであるため、それを変更して、再マウントできるようにする。
- 4.4BSD の mount コマンドをコンパイルして使う。ただし、これには 4.4BSD のファイルシステムを使う必要がある。

の二つの方法があるが、この問題が生じた時には SCSI ドライバ実装後であったので、後者の方

法を取ることにした。そのため、ステップ 2 の途中からは、ディスクレスブートによる開発を断念した。

2 の解は、ファイルシステムにおける議論で明らかであろう。4.4BSD Lite のコマンドをコンパイルする必要がある。例えば、ifconfig は、これが呼び出す ioctl システムコールで使用するリクエストの種類やデータ構造が 4.3BSD と 4.4BSD で異なるため、NEWS-OS4 の ifconfig をそのまま使うことはできない。これにより、少なくとも mount コマンドおよび ifconfig コマンドは 4.4BSD のコマンドを用いる必要がある。おそらく、4.4BSD Lite において/sbin 以下にあるコマンドはコンパイルした方が良いであろう。Lites を、NEWS-OS4 とのバイナリ互換性を提供すると同時に、4.4BSD Lite のコマンドも実行できるようにするための実装については、5.5 節で述べる。

最後に、図 1 に我々の開発環境およびブート環境の概略を示す。

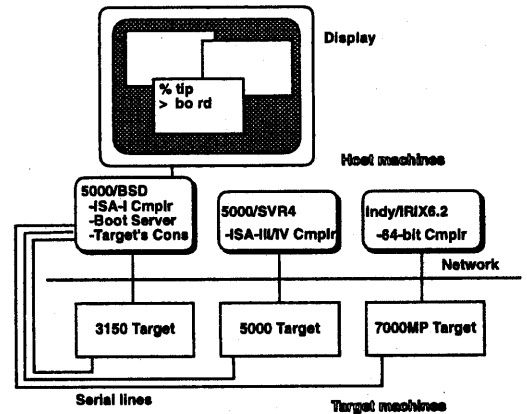


図 1. 開発およびブート環境の概略図

## 5 現状および主な変更箇所

現状では、NWS-5000 上で MK および Lites がマルチユーザモードで動作し、ファイルシステムおよびネットワークも含めて動作中である。すなわち、ステップ 2 をほぼ終えている。デバイスとしては、(1)Zilog Z85230 ESCC, (2)HP SPIFI3 SCSI[11], および (3)SONIC イーサネットが動作中であり、フレームバッファを実装中である。

本節では、これまでの経験から重要と思われる変更箇所について述べる。まず 5.1 節から 5.3 節で MK の移植について述べ、次に 5.4 節および 5.5 節で Lites の移植について述べる。

### 5.1 makeboot の移植

MK の移植で最初にやらなければならないのは、MK を作成するのに必要なツールを移植する

ことである。これらのツールのうち、移植で最も問題になるのは makeboot というコマンドである。Mach では、マイクロカーネルは /mach.boot であるが、これはリンカから生成された実行可能形式のファイルではなく、マイクロカーネル(ファイル名は mach\_kernel) とデフォルトページャ(ファイル名は bootstrap) の二つの実行可能形式のファイルを makeboot によって一つの実行可能形式にしたものである。今回の場合、NEWS-OS4 のブートプログラム (/boot) がこの /mach.boot というファイルを通常の COFF ファイルとしてブートすることになる。

しかしながら、DECstation のブートプログラム用に作成された makboot プログラムが作成した /mach.boot ファイルは、本来あるべきセクションヘッダを持たないため、NEWS-OS4 のブートプログラムがこれをブートすることはできなかった。そのため、生成した COFF ファイルにセクションヘッダを持たせるように、makeboot を変更した。変更後の makeboot の機能を図 2 に示す。

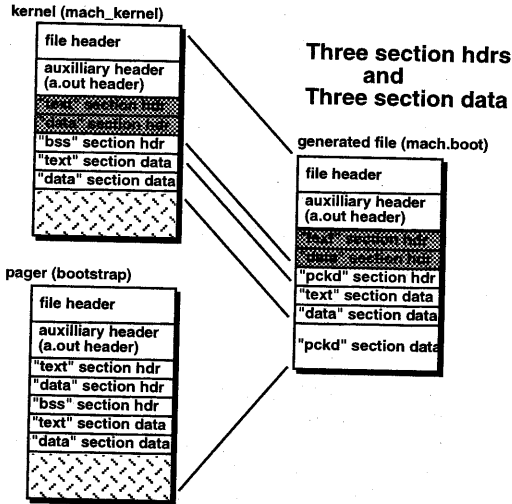


図 2. 変更後の makeboot の機能

## 5.2 仮想/物理メモリ空間の構成

図 3 に、仮想および物理空間の構成を示す。二つの変更点がある。一つは、NEWS の ROM ルーチンが、KSEG2(R3000) あるいは KSEG3(R4400) を使用しているらしく、MK のブート時に TLB の内容をフラッシュすると ROM ルーチンを利用できなくなるので、そこに 1M バイト程度の仮想空間を ROM ルーチンのために割り当てたことである。これにより、初期のデバッグを ROM 内の printf 関数を呼び出すことにより行なえるようになった。もう一つは、デバッグのために物理メモリを確保するようにしたことで、例外処理のトレースおよびプログラ

ムのチェックポイントのために使用している。

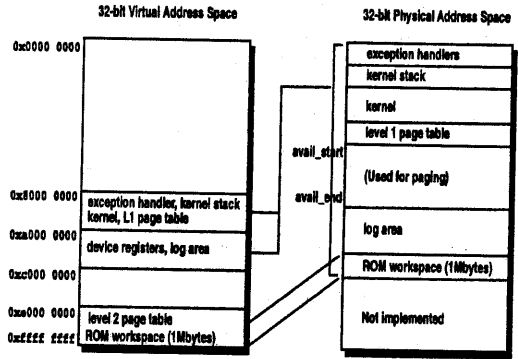


図 3. 仮想/物理メモリ空間の構成

## 5.3 ROM ルーチンを用いたディスクドライバ

SCSI ドライバの実装は時間がかかると思われたので、ROM コールを用いた仮想ディスクドライバを実装した。このドライバを使うと、ブートプログラムおよび MK が格納されていたブートデバイスにアクセスすることができる。すなわち、一番目の SCSI ディスクの a パーティションである scsi(0,0,0) からブートした場合は、このドライバは通常の SCSI ディスクのドライバとして機能する。また、Remote Disk (RD) あるいは NFS のプロトコルを用いて、リモート計算機からネットワークブートした場合は、このドライバに対する read 要求は ROM ルーチンによって RD または NFS プロトコルに変換されリモート計算機へ送られる。このドライバを用いることにより、ターゲットプラットフォームをディスクなしでブートすることが可能になり、開発時間の短縮となる。

## 5.4 非同期例外処理

Lites で特徴的なことの一つが非同期例外処理である。Lites では非同期例外処理を通常の例外処理と別のスレッドで行なう。ここで、例外処理を行なうのは、エミュレータであり、Lites ではエミュレータのコードを実行するスレッドが 1 つのプロセスにつき 2 つ存在することになる。2 つのスレッドのうち一つはプロセス生成時に作成され、もう一つのスレッドはプログラムが sigvec() システムコールを用いて、ハンドラの登録を行なった時にはじめて生成される。今回の実装では、プロセスの実装を図 4 に示すように行なった。

## 5.5 システムコール

前述したように、Lites は NEWS-OS4 のバイナリと 4.4BSD のコマンドの両方を実行できなければならない。これは、Lites に両方のシステムコールを実装しなければならないことを意味する。ここで、NEWS-OS のシステムコールと

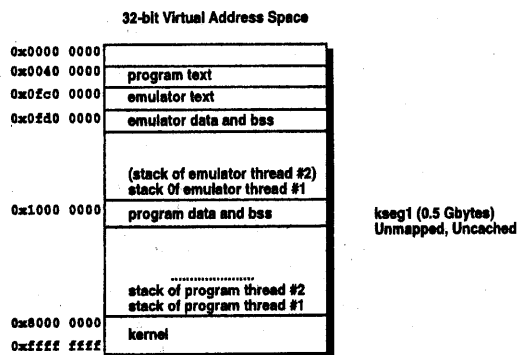


図 4. プロセスの仮想メモリ空間

4.4BSD のシステムコールを区別する方法として、(1) 実行可能形式あるいはマジック番号で異なるシステムコールエントリテーブルをひく方法、および (2) システムコール番号を一意にする方法、がある。今回の実装では、(2)の方法を採用した。なぜなら、NEWSでは、SVR4のNEWS-OS6上でBSD系のNEWS-OS4のバイナリを実行可能にするために、システムコール番号をNEWS-OS6で4000からNEWS-OS4で1000からはじめているからである。MKおよび4.4BSDを含めた最終的なシステムコール番号一覧は表3のようになる。

表 3. システムコール番号一覧。MK は UNIX のシステム番号と重ならないように-1以下の整数を利用している。このうち、-25はMIG(mach\_msg)で使用されている。

MK	-1以下
4.4BSD	0 - 204
NEWS-OS4	1000 - 1177
NEWS-OS6	4000 - 4204

## 6 おわりに

本稿では、MIPSプロセッサを搭載したソニー製NEWSワークステーションへのMach3マイクロカーネルおよび4.4BSD Liteサーバの移植について述べた。現在、R4400プロセッサを用いたNEWS-5000上へのこれらの移植とシリアル、SCSI、イーサネットの各ドライバの実装を終えており、マルチユーザモードで動作中である。

今後の課題として、MPTS[12]によるマイクロカーネルの性能評価がある。また、NEWS-OSとの性能比較を行なうことにより、UNIXシステムとしての評価も行なう予定である。

## 7 謝辞

本研究を行なうにあたり御協力頂いた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」の皆様へ感謝致します。

## 参考文献

- [1] John L Hennessy and David A Patterson, editors. *Computer Architecture A Quantitative Approach*. Morgan Kaufmann Publishers Inc., 340 Pine Street, Sixth Floor, San Francisco, CA 94104-3205 USA, second edition, 1996. ISBN 1-55860-329-8.
- [2] Mendel Rosenblum, Edouard Bugnion, and Stephen Alan Herrod. The Impact of Architectural Trends on Operating System Performance. In *Proceedings of the 15th ACM Symposium on Operating System Principles*, 1995.
- [3] 徳田 英幸, 追川 修一, 西尾 信彦, 萩野 達也, and 斎藤 信男. MKng: 次世代マイクロカーネル研究プロジェクト. In 第53回情報処全国大会論文集 (5B-4), 1996.
- [4] Gerry Kane and Joe Heinrich, editors. *MIPS RISC Architecture*. Prentice Hall, 1992.
- [5] Mike Accetta, Robert Baron, William Bolosky, David Golub, Richard Rashid, Avadis Tevanian, and Michael Young. Mach: A New Kernel Foundation For UNIX Development.. In *USENIX 1986 Summer Conference Proceedings*. USENIX Association, June 1986.
- [6] Johannes Helander. Unix under mach - the LITES server. Master's thesis, Department of Computer Science, Faculty of Information Technology, Helsinki University of Technology, December 1994.
- [7] Madhusudhan Talluri, Mark D Hill, and Yousef A Khalidi. A New Page Table for 64-bit Address Spaces. In *Proceedings of the 15th ACM Symposium on Operating System Principles*, 1995.
- [8] Jochen Liedtke. Address Space Sparsity and Fine Granularity. *Operating Systems Review*, 29(1), January 1995.
- [9] MIPS Technologies, Inc. *R10000 Microprocessor User's Manual*. MIPS Technologies, Inc., second edition, January 1996. Version 1.1.
- [10] Hideyuki Tokuda, Tatsuo Nakajima, and Prithvi Rao. Real-Time Mach: Towards a Predictable Real-Time System. In *Proceedings of Mach Workshop*. USENIX Association, October 1990.
- [11] Hewlett Packard. *SPIFI3-SE SCSI-II Protocol IC*. Hewlett Packard, October 1990.
- [12] François Barbou des Places and Philippe Bernadat and G.N.Madhusudan and Yves Paindaveine and Nick Stephen. *OSF Microkernel Performance Test Suite MPTS Release 1.1*. OSF Research Institute, April 1996.