マルチプロセッサ型電子交換機のオペレーティングシステムについて

麻生忠宏、壺屋光邦、海老原進、米本誠一、中村信一、
柳原　隆、尾形初夫、小畠健治（日本電気株式会社）

１．緒言

電子交換システムは電話交換を中心に発達してきたが、近年ディジタルデータをスイッチングするデータ交換の領域が拡大してきている。特に都市間あるいは国際間のパケット交換網や、企業グループ内での国際間ネットワーク等のビジネス向けを中心とした需要が増大傾向にある。実際に、百パケット／秒程度の小規模から、数千パケット／秒の大規模までのデータ交換サービスが必要とされるが、局用大型のデータ交換機としては、コスト性能比や障害時対策の面からマルチプロセッサ化による実現が適していると言える。マルチプロセッサの形態として、いろいろな方式が発表されているが、データ交換においてはノード・ツー・ノードのリンク方式をとることが可能である。我々は、このリンク方式の採用とマルチプロセッサ機構をオペレーティングシステム(OS)に封じ込めることにより、小規模から大規模までを同一の交換ソフトウェアで実現可能とした。

以下、交換システムの特徴を明らかにし、マルチプロセッサ形態の大容量データ交換システムに適合したＯＳを取り上げて、その設計思想と特徴的機能及び実現上の主要技術について述べる。

２．交換システムの特徴

(1)交換システムの分類

音声のスイッチングを目的とした電話交換分野では、加入者交換、中継交換、国際交換、自動車交換、衛星交換などの局用のものと、私設のＰＢＸとがある。ディジタルデータのスイッチングを目的としたデータ交換分野では、音声と同様にリアルタイムにデータを転送するタイプのテレックス交換と、データを一度蓄積してから転送するタイプのパケット交換とに分類できる。表．１に分類の一覧を示す。なお今後の方向としては、サービスを統合した新しいディジタル網（ＩＳＤＮ）に適合した交換システムの実現が要求されている。

(2)交換システム共通の特徴

・高密度トラヒック

ランダムに発呼してくる大容量回線の呼（call　）を短時間でスイッチングすることが必要である。

・高信頼性

公共性があり、システムのダウンは

表.1　交換システムの分類

交換システム			
電話交換	局用		（容量の目安）
		加入者交換	500～100,000回線
		中継交換	60,000トランク
		国際交換	60,000トランク
		自動車交換	10,000回線
		衛星交換	60~3,000局
	私設		
		ＰＢＸ	100～12,000回線
データ交換			
		テレックス交換	30,000回線
		パケット交換	500～12,000回線

社会的な影響を及ぼすため高度な信頼性が要求される。

• 長期的連続運転

数十年に渡って24時間連続運転が行われるため、保守機能が万全である必要がある。

• 回線増設性

交換機は一般に、終期の回線容量は初期の数倍となるような経済的な増設性を要求される。

(3)交換用OSの条件

• オーバヘッド重視のタスク制御

毎秒のトランザクション数が極端に多く、タスクの保留時間が比較的短いという交換システムの特徴に対処できなければならない。使用頻度の高いタスク群は全てメインメモリ常駐とし、タスクの切替は無駄処理を省いて最小ステップ数で実行させる必要がある。

• 高度な障害処理能力

交換システムは、高信頼性を確保するために、CPUを含めて、メインメモリ、データチャンネル、通信制御装置など冗長構成(二重化同期運転、スタンバイ方式等)をとっている。OSの障害処理プログラム(FP)では障害の早期検出と障害装置の自動切離しを含みシステムの系再構成を行って交換処理を続行させる必要がある。

• 保守管理能力

交換処理を運転中に、その交換サービスに影響を及ぼすことなく、保守者インターフェースの機能、たとえば装置の部品交換や自動診断あるいは時計の時刻調整などを実現する必要がある。

• 増設能力

交換処理を運転中に、その交換サービスに影響を及ぼすことなく、回線装置や共通装置の増設およびプログラム上のメモリ・リソースや各種データ・テーブルの拡張を実現する必要がある。また交換プログラムの機能を変更する場合においても、交換処理のサービス中

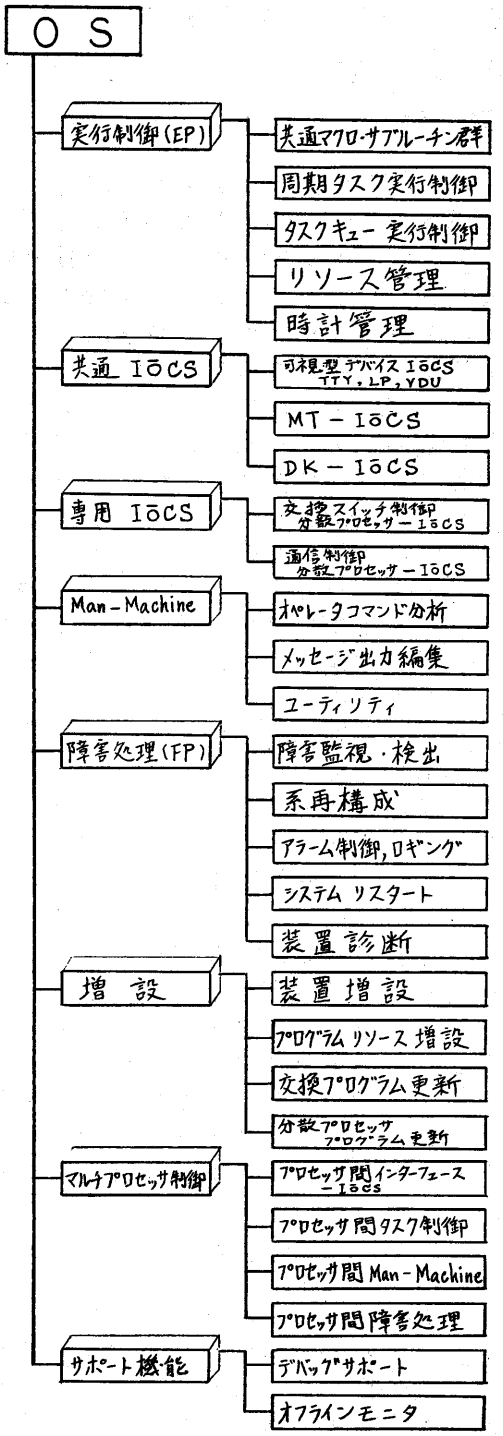

図.1 データ交換用OSの機能範囲

断を最小にするための手段を有する必要がある。

⑷交換用ＯＳの機能範囲

交換用OSは前項⑶で述べた条件を重視しているが、一方汎用コンピュータのＯＳ上には不可欠な言語処理機能（アセンブラ、コンパイラ、リンカ）などは具備していない。交換プログラムの開発は、一般的には別の汎用コンピュータを使うことが多いからである。図.1に交換用ＯＳの機能範囲を示す。

⑸交換システムの大容量化

大容量交換システムを実現する上では、最適コスト性能比を保ちつつ小規模から大規模局まで段階的にカバーできること、障害発生時の波及度を極力小さくできること等の理由によりマルチプロセッサ化が適している。マルチプロセッサの形態としては、共通メモリ方式や共通バス方式など、あるいはもっと結合度の緩いノード・ツー・ノードのリンク方式などがある。電話交換のマルチシステムでは共通メモリと共通バス方式を併用しているものがある。図２参照。電話交換では、アナログあるいはデジタルの交換スイッチを通話路として持ち、このスイッチを制御することによって交換機能を果している。起呼検出側のプロセッサは必要なリソース類のデータを、共通メモリを介してアクセスし、着側の制御は共通バスを介して相手プロセッサのタスクを制御する方法である。実際に発側と着側とのパスが交換スイッチ網に設定されてしまうと、音声データはプロセッサを介さずにそのパスを介して転送される。これに対してデータ交換のパケットシステムでは、データをプロセッサ内に引込んでメモリに蓄積する方法をとるので、プロセッサ外部に交換スイッチ網を必要としない。つまり各プロセッサは独立してノードとして

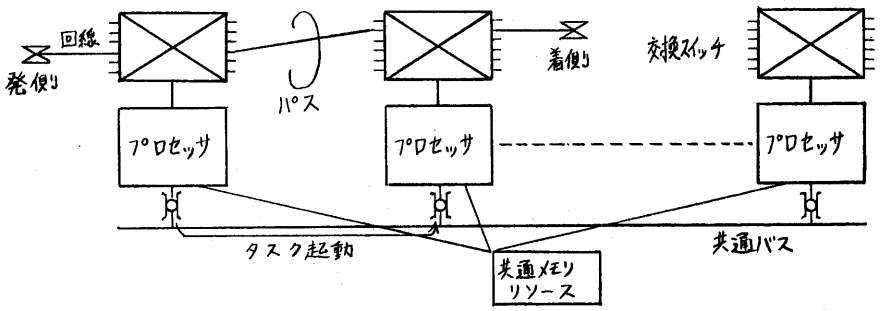

図.2 共通メモリ、共通バス併用型の例

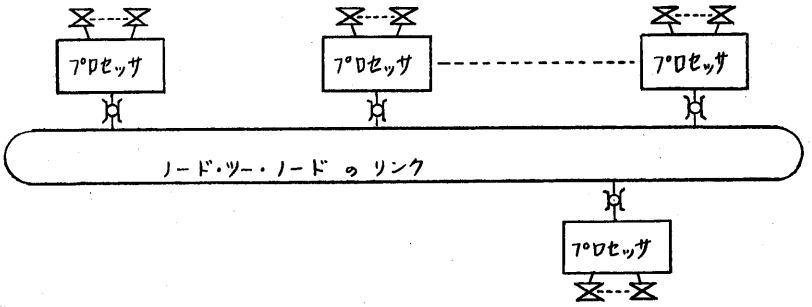

図.3 リンク結合型の例

扱うことが可能である．このことは、大容量のパケット交換局を構成する場合に、より小さな容量の交換局複数個をノード・ツー・ノードのリンク結合とし、管理上は1局として統合できるように設計することが可能であることを示している。図.3参照。

3．大容量データ系ＯＳの設計思想

パケット交換を主目的とし、ノード・ツー・ノードのリンク結合マルチプロセッサ構成を前提とした大容量データ交換システムのＯＳについてその設計思想を述べる。

(1)ＯＳの仮想化

小規模から大規模な局までカバーするために、シングルプロセッサでもマルチプロセッサでも同一の交換ソフトウェアを変更することなしに使えるようＯＳの仮想化を行うこと。

(2)マン・マシン・インターフェースの仮想化

各プロセッサはそれ自体で独立して運用できるよう保守機能を分散させ、その一方、マルチプロセッサを統合して一つの局として運用管理が出来るよう集中保守機能を実現できること。

(3)プロセッサ間プロトコルの階層化

個々のプロセッサの一時的なダウンや、プロセッサ間の通信装置障害が全システムの各交換処理動作に、大きく波及することがないようにするため、ＯＳでプロセッサ間プロトコルを階層的に定義してマルチプロセッサシステム全体の信頼性を高めること。

4．ＯＳ機能とその特徴

前述の設計思想をもとに実現したＯＳの特徴的機能を示す。

(1)タスク制御

シングル、及びマルチプロセッサの交換ソフトウェアを同一とするためには、プロセッサ内、およびプロセッサ間のタスク起動において、アプリケーションからＯＳを見た場合のインタフェースを同一とする必要がある。

この一つの方法として、タスク制御部にプロセッサ間通信機能を内在させ、通信機能を隠蔽することで可能となる。

図.4の上方のケースは、あるプロセッサi中のタスクAとタスクB、それとは異なるプロセッサj中のタスクCとタスクDとをATCHWマクロ（attach wait の略）を用いて、A→B→C→D→C→B→Aの順に起動する例である．なお、同図中のEOT（end of taskの略）は、タスク処理の終了をＯＳに通知するマクロである．

この図で、タスクBからタスクCへの起動は、異なるプロセッサ間であるが、タスク起動側（タスクB）はプロセッサ内と同一のマクロ、起動を受ける側（タスクC）から見た場合の切ロ

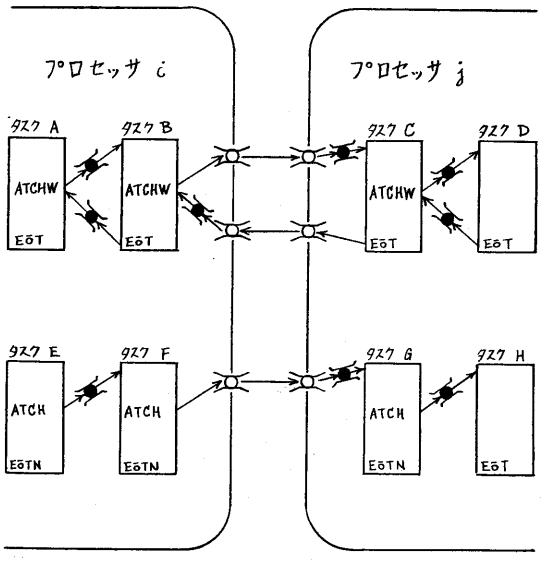

図.4　プロセッサ間タスク制御概念図

も、プロセッサ内でのタスク起動と同一である様になっている。

図.4の下方のケースは、ATCHマクロ（attachの略）の例である。

このように、タスク制御内にプロセッサ構成のシングル／マルチを封じ込めた結果、アプリケーション・プログラムは、相手のタスクが自プロセッサ内にあるか、他プロセッサ内にあるかを区別しなくてもよい様にOSが仮想化されている。

(2)マン・マシン・インターフェース

運用保守機能の多様化に対応するため、各プロセッサに分散化してあるマン・マシン機能を相互に使用したり、またある機能を特定のプロセッサで集中処理することを可能としている。

以下にコマンド処理、メッセージ出力、及び障害情報管理について説明する。

コマンド実行プロセッサの選択は、次の3種類の指定がある。（図.5参照）

1)コマンド機能毎に割当てられた特定なプロセッサ。

2)コマンドを投入したプロセッサ。

3)パラメータで指定したプロセッサ。

また、コマンド転送規制レベルを設け、コマンド投入プロセッサと実行プロセッサをコマンド機能毎に制限することも可能である。

一方、メッセージ出力については、IOCS内にプロセッサ間自動転送機能を持ち、1装置への出力、及び出力カテゴリというグループ指定による複数装置への同時出力のいずれの場合でも、出力対象装置が他プロセッサのものであれば、自動転送が実行される。

このように、コマンドの投入と実行およびメッセージ出力に関し、各プロセッサは同一機能を有している。また保守業務を円滑にするため、あるプロセッサにおいて発生または検出した障害情報を特定のプロセッサへ自動転送し、アラームパネルへの表示、メッセージの出力、及び情報のロギングを集中管理することができる。

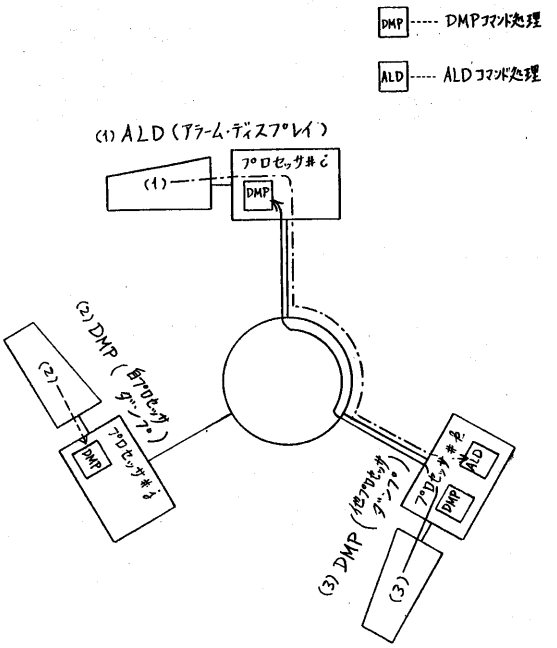

図.5 コマンド投入と処理プロセッサの関係

障害プロセッサから特定のプロセッサへの障害情報の転送はOSにより自動制御されるため、シングル・プロセッサ構成の場合と同一の保守者およびソフトウェアインターフェースで情報の登録／解除が可能である。

(3)障害・系構成

交換システムは24時間連続システムであるので、装置障害時のシステムへの影響度を最小にする必要がある。以下各プロセッサレベル及びマルチシステムとしての障害処理機能について示す。

まず各ノード毎の障害処理は次の2つに分類できる。これは従来シングルプロセッサで行われているものと同様である。

- 一般装置障害処理

障害装置の切離しおよび予備装置への切替等によりシステムの運用を続行するものであり、影響範囲は障害装置に直接関連した回線、入出力装置などに限定される。

・システム・リカバリ

プロセッサ系（CPUやメインメモリ等からなる装置群のこと）障害、または、ソフトウェア障害によりシステム全体の系再構成を行うものであり、その影響範囲により、PH0～PH3にクラス分けされる。（表.2参照）

次にマルチシステムとしての障害制御は、一部プロセッサがダウン状態になっても他のプロセッサによって正常な運転が続行できるように、各プロセッサ毎にプロセッサ管理を設け、ある特定のプロセッサの主導権のもとに、システムの系構成が管理できるような分散管理集中制御方式をとっている。

さらに、プロセッサ・ダウンからの復旧は、他の正常なプロセッサの運転を中断する事なく復旧させる機能を有する。

また、プロセッサ間の通信を担う装置および伝送路は、二重化・あるいは三重化の冗長構成を持っていることを利用して、システムの信頼性をさらに高めるため、通常1つをアクト系とし、他をスタンバイ系として2つの通信ルートを設定し、スタンバイ系ではシステムの監視、アクト系の異常通知、系構成指示／応答等を行うことを特徴としている。

(4)プロセッサ間通信の階層化

マルチプロセッサ化したシステムでは、プロセッサ間通信の能力と信頼性が重要な役割を占めている。

システムとしての信頼性を保つ手段として、プロセッサ間通信のインターフェースを標準化し、目的別に階層化した以下のプロトコルを定義している。

表.2　システム・リカバリ分類

<table>
<tr><th>項目 ＼ 再開フェーズ</th><th>PH0</th><th>PH1</th><th>PH2</th><th>PH3</th></tr>
<tr><td>意味</td><td>プロセッサ系障害時の系再構成（オンライン続行）</td><td>プロセッサ系障害時の系再構成（オンライン一時中断）</td><td>PH1のバック・アップ</td><td>システム・ダウンよりの立上げ（PH2のバック・アップ）</td></tr>
<tr><td>起動条件</td><td>プロセッサ系障害割込み発生</td><td>・プロセッサ系障害割込み発生（PH0が不可の場合）
・ディスク2重障害
・ハードによる緊急処理起動要因</td><td>PH1連続15回発生</td><td>手動による緊急処理起動</td></tr>
<tr><td rowspan="2">メモリ初期設定</td><td>固定データ</td><td rowspan="5">引継ぎ</td><td colspan="2">ディスク中のバック・アップ・ファイルからロード</td><td>バック・アップ MTからロード</td></tr>
<tr><td>一時エリア</td><td>一部分初期設定</td><td colspan="2">初期設定</td></tr>
<tr><td colspan="2">システム時計</td><td colspan="3">引継ぎ</td></tr>
<tr><td colspan="2">ファイル管理状態</td><td colspan="3" rowspan="2">アプリケーション・ソフトウェアでシステム毎に決定</td></tr>
<tr><td colspan="2">課金情報</td></tr>
<tr><td colspan="2">呼</td><td>救済</td><td colspan="2"></td></tr>
</table>

（図.6 参照）

• フィジカル・レイヤー（レベル1）
プロセッサの地域分散形態、プロセッサ間の距離によって、光通信方式など、様々なハード形態が考えられるが、プロセッサ間の通信制御装置でその差異を吸収する。

• リンク・コントロール・レイヤー（レベル2）
任意の2プロセッサ間の伝送制御手順として、HDLCプロトコルを用いている。

• ネットワーク・レイヤー（レベル3）
各プロセッサのOS間のプロトコルであり、プロセッサ間通信を実現する。通信制御装置のハードウェアレベルの制御とプロセッサ間フロー制御および以下に示すレベル4とレベル5の優先制御を司さどる。

• システム・ファンクション・レイヤー（レベル4）
システム制御用に、OSが独自に使用する階層であり、一般タスク制御マクロの他に特殊なOS専用マクロにより、レベル3とのインターフェースを有する。
本階層は、システムの正常運転を制御することを目的とし、緊急レベル通信（障害処理、システム系構成）、及びシステム内監視（各プロセッサ間の時計の同期など）の機能を持つ。

• ユーザ・レイヤー　（レベル5）
アプリケーション・プログラム用の階層であると同時に、OSの中でも、4②で述べたコマンド転送機能、メッセージ出力機能及びロギング情報転送機能におけるプロセッサ間通信に利用している。

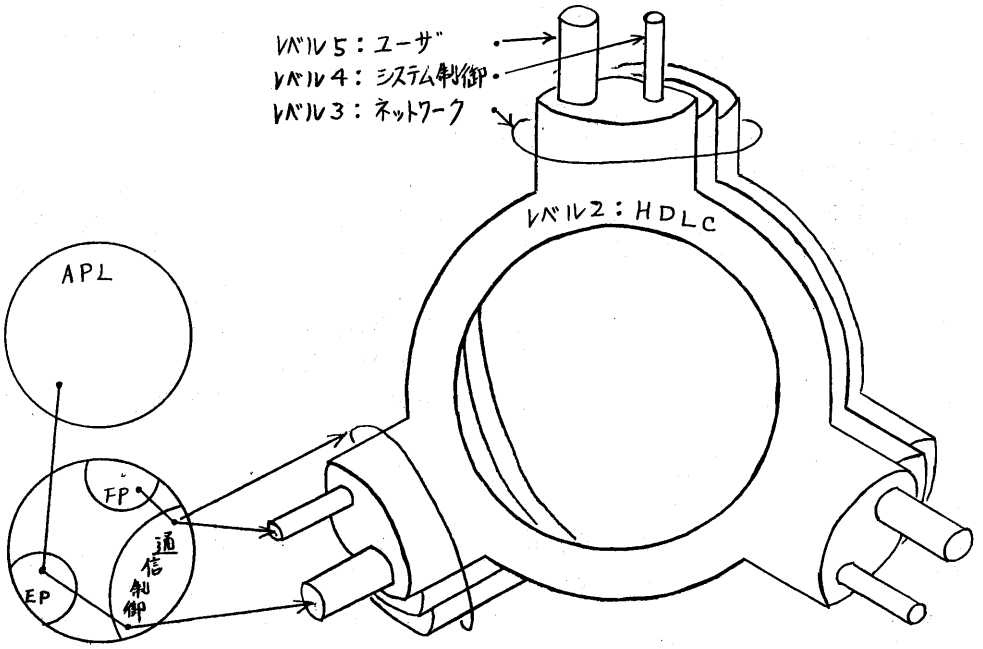

図.6　プロセッサ間通信概念図

5. マルチプロセッサOSの主要技術

以上述べたOSを前提として、ノードを構成するプロセッサ系が冗長構成を持ち、ノード間を接続する伝送路も冗長構成を持っている様なルーズカップルなマルチプロセッサシステムを構築するに際し、OSの観点から、主要技術項目を以下に示す

(1)抽象化技術

ハードウェアの冗長構成、ノードの構成等の差をOSソフトウェアロジックから出来るだけ切りはなして実現するためのソフトウェア構造抽象化技術

(2)ノード通信平準化技術

特定ノードの過負荷や異常により、特定ノードの通信リソース占有状態を防止するための通信フロー制御技術

(3)ノード間通信優先制御技術

システムの正常運転を保持するための、障害対策、系再構成等の緊急度の高いノード間通信を優先的に行う制御技術

(4)システム系構成同期化技術

ハードウェア的に各ノード間を制御する専用通信手段を持たないルーズカップリングな構成において、各ノードがお互いに同期し合いながらシステムとしての系を構成する為の同期化技術

(5)はき出し処理技術

部分的ノードのダウンや通信装置障害等により、系の構成を再確立する場合、正常系におけるデータロスを防止するための技術

(6)デッドロック防止技術

ノード間通信に使用される各ノード内リソースの浸きや、異常保留等を防止するためのデッドロック防止技術

(7)障害判定技術

冗長構成を持ち多段に接続されたシステム構成において、被疑装置から障害装置を判定するための障害判定技術

(8)部分再開技術

障害ノードや通信装置の切離し、又障害からの復帰に際し、他の正常系へ影響せぬ様、各ノードのリソース、ハードウェア、データ類を矛盾なく整合をせる為の部分再開技術

これらの諸技術を総合的に網羅する事により、交換機としての信頼性、サービス性、保守性等を満足するシステムを構築する事が出来る。

6. 結言

以上、データ交換を主体としたマルチプロセッサOSについて述べた。

本OSは、ノード・ツー・ノードのリンク結合マルチプロセッサ構成を前提とし、OS機能全般に渡り仮想化の思想を貫いた。その結果、シングル／マルチ及びプロセッサの地域分散がOS内に封じ込まれ、アプリケーション作成面では小規模から大規模へのソフト資産の継承が可能となった。また、システム運用面においても、機能分散あるいは集中化等多様な要求に対し、柔軟な対処が可能となった。

来たるべきISDN時代に向けての通信需要の拡大や技術の進歩と供に、交換システムは多様化、分散化の傾向にある。今後、その様なシステムを構築する上で、ここで述べた設計思想と主要技術は重要なポイントを示している。

最後に本OSの開発に当り、御指導御協力を頂いた関係各位に深謝致します。

〔参考文献〕

(1) H. Itoh, T. Asoh et al., "ESS Standard Operating System," NEC Res. & Develop., Dec., 1982.
(2) T. Kohashi et al., "Flexible Modular Structure of Store and Forward Oriented Data Switching System-NEDIX 510F," NEC Res. & Develop., Oct., 1979.