

## 解説

情報通信ネットワーク向き OS インタ  
フェース CTRON†

和佐野 哲 男†† 小林 吉 純††

## 1. ま え が き

ソフトウェアの生産性向上のために使用されているさまざまなアプローチは、所要の機能を果たすソフトウェアを開発する場合の開発規模当たりの作業量を減らすことに着目した「ソフトウェア開発支援環境の高度化」のアプローチと、機能を実現するためのソフトウェアの開発量そのものを減らそうとする「ソフトウェアの体系化」のアプローチとの二つに大別できる。前者の立場からは、ソフトウェア開発支援ツールの高度化や開発設備の整備などが行われているが、大幅な生産性向上を達成するためには、これら二つのアプローチを組み合わせる活用することが有効である。

特に、ソフトウェアの体系化のためには、計算機の基本ソフトウェアであるオペレーティングシステム(以降、OS と略す)のインタフェースを共通化することが必要である。

この観点から、計算機メーカー各社では、自社内の各計算機機種間に共通なソフトウェア統合環境<sup>1)</sup>を発表し、アプリケーション、通信、ヒューマンマシンの各インタフェースの共通化を進めている。

一方、標準化機関の場では、マイクロプロセッサでの OS インタフェース共通化を狙いとした MOSI<sup>2)</sup>、会話型の既存 OS インタフェースの体系化を目指した POSIX<sup>3,4)</sup> などの活動が行われている。また、ISO/IEC JTC 1/TSG-1 (コンピナ棟上昭男)では、アプリケーションのポータビリティを図るためのインタフェース IAP (Interfaces for Application Portability) 標準化に向け、作業項目抽出のための検討を開始した。

CTRON 専門委員会では、ソフトウェアのポータビ

† Operating System Interface CTRON Applicable to Information Communication Networks by Tetsuo WASANO and Yoshizumi KOBAYASHI (NTT Network Systems Development Center, Network Technology Department).

†† NTT ネットワークシステム開発センター通信網技術部

\* MOSI は、IEEE の登録商標である。

\*\* POSIX は、IEEE の登録商標である。

リティの向上に寄与でき、かつ、厳しいリアルタイム性と信頼性が要求される情報通信ネットワークサービスに適用できる OS インタフェースとして、CTR-ON<sup>5,6)-7)</sup> の設計を進めている。

## 2. CTRON の設計思想

## 2.1 TRON トータルアーキテクチャへの準拠

CTRON は、概念、用語やインタフェースの表現形式などについては、TRON トータルアーキテクチャ<sup>8)</sup> に準拠している。したがって、これらの点に関する BTRON や ITRON などの他の TRON との整合性はおのずと保証されている。さらに、仕様面に関しては、ITRON と同様に BTRON のファイル仕様に準拠するとともに、BTRON のヒューマンマシンインタフェース仕様にも準拠しており、CTRON に BTRON や ITRON を有機的に組み合わせたシステムの構築が可能となっている。

## 2.2 オープンな OS インタフェース

ソフトウェアのポータビリティを保证する基盤を提供するという目的のためには、仕様は特定の会社や個人に帰属させず、実費を負担すれば、だれもが自由に入手し使用できるパブリックドメインなものとなっていなければならない。そのため、他の TRON と同様に、CTRON 仕様には、第三者ライセンス情報は混入させず、独自のものが規定されている。そして、方言の出現にともなうソフトウェアポータビリティへの悪影響を排除するため、ユーザが仕様を変更することは禁止されている。

なお、国際標準との関係については、すでに国際標準化が進んでいる機能については、それに準拠し、国際標準の存在していないものについては、逆に CTRON 仕様を国際標準として提案していくという方針で臨んでいる。

\* CTRON の名前は以下に由来する。

TRON は、“The Realtime Operating system Nucleus” の頭字語である。また、C は、“Communication”、“Central”を表す。

### 2.3 ソフトウェアのポータビリティ

情報通信ネットワーク（以降、単にネットワークと略す）の発展につれ、サービスが複合化するとともに、サービスの追加、変更も頻繁になっている。そのサービスを効率的に実現させるためには、同様の機能を複数のネットワークノード（以降、単にノードと略す）に搭載したり、ノード間の機能分担を変更したりする必要がある。また、ハードウェア技術の進展に応じ、適切な機種を効果的に導入できる必要があり、ソフトウェアのポータビリティが重要になる。しかし、これまでの OS は、同様の機能であっても、種々のハードウェア機種や情報処理、通信処理といった適用分野対応に、異なったインタフェースを提供しており、ソフトウェアのポータビリティを保証できるものではなかった。

ネットワークでは、ノードにおけるサービス種別ごとに OS に対する要求条件は異なっており、その種別対応にインタフェースを規定しては、ノード間のソフトウェアの流通は図れない。そのため、各種サービスに共通な、すなわち適用分野に依存しないインタフェースを規定する必要がある。

また、ネットワーク内にはアーキテクチャの異なる種々のハードウェア機種が存在するため、アーキテクチャの差を隠蔽する仮想マシンを提供しなければ機種を越えたソフトウェアの流通はありえない。

さらに、ネットワーク内の各ノードは、それぞれのノードごとに資源をもつため、ノード間でのソフトウェアの効率的な配置変更を行うためには、それらノードのネットワーク内での配置に依存しない資源の確保、解放、アクセスのインタフェースの提供が必要となる。

このため、CTRON では、適用分野、ハードウェアアーキテクチャおよびネットワーク内の資源配置に依存しない OS インタフェースを規定している。

### 2.4 高いリアルタイム性

交換、通信処理ノードでは、他のノードからの多数の情報を同時に交換したり、多数の通信回線を同時に制御する必要があり、数千以上の超多種処理と厳しいリアルタイム性が要求される。

情報処理ノードでは、複数ユーザの非定型処理を行うとともに、数ギガバイトにもおよぶ大規模なデータベースを効率的に操作する必要がある。ワークステーションノードでは、応答性に優れたヒューマンマシンインタフェースが要求される。したがって、これらの

ノードにおいてもリアルタイム性が重要となる。

従来、このようなリアルタイム性の要求の厳しい分野では、性能重視のあまり、各分野、各機種対応の個別 OS が中心となっており、極度に軽装化を図った OS<sup>10)</sup> も用いられている。

これに対し CTRON では、リアルタイム性に優れたタスク管理インタフェースなどを規定するとともに、機能・性能条件に応じたインタフェースの選択・使用を可能にすることにより、各分野、各機種に共通な OS インタフェースを提供している。

### 2.5 ノンストップ運転

交換、通信処理ノードの障害はネットワーク全体のサービスに大きく影響するため、厳しいフォールトトレランス性が要求される。

CTRON では、プロセッサや入出力装置の障害時の処理継続を可能とする装置多重化制御機能、ファイル、データベース、通信電文の障害復旧機能、およびサービスの中断なしに、通信回線や入出力装置の増設を可能とする機能などのフォールトトレラントインタフェースを規定している。

### 2.6 OS 製品の検定

ソフトウェア製品の検定としては、米国国防総省による Ada\* コンパイラの検定<sup>9)</sup>や ATT による UNIX\*\* の検定<sup>11)</sup>などがすでに著名であるが、CTRON においても、CTRON に基づいて開発された OS 製品が正しく CTRON に準拠しているのか否かの検定制度がある。

CTRON においては、検定対象製品における機能の過不足や CTRON 仕様の変更は認めないという方針のもとに、検定用プログラムを被検定 OS 上で動作させて検定する動的検定と、動的検定だけでは検出が難しい機能の追加や用語の使い方などを製品マニュアルを対象に検定する静的検定との2本立てで、検定精度の向上を図っていることが特徴である（図-1）。

## 3. CTRON インタフェースの分類・構造化

### 3.1 CTRON 参照モデル

OS インタフェースの共通化に当たっては、ネットワーク内の各分野で用いられる用語や概念を分野間で統一しておくとともに、OS の機能や構成に関しても分野間で意識を共通化しておくことが必要となる。これらの要求に応えるため、CTRON では仕様の設計や

\* Ada は、米国政府 Ada Joint Program Office の登録商標である。

\*\* UNIX は、米国 AT&T 社が開発し、ライセンスしている。

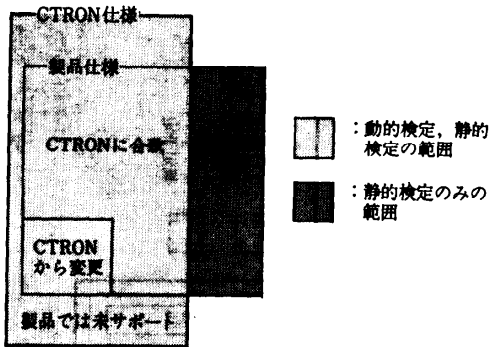


図-1 CTRON での OS 製品の検定

利用のよりどころとなるインタフェースの全体構造を規定し、これを CTRON 参照モデル<sup>12)</sup>と呼んでいる。

CTRON 参照モデルに対する要求条件は、以下のよう整理できる。

- ① ソフトウェアの生産性向上のため、アプリケーションプログラム (AP) のみならず、パッケージソフトウェア類や OS の一部も流通可能とすること。
- ② ネットワーク内の各分野で必要とされる機能を包含すること。
- ③ 用途に応じた性能の良い OS の構築を可能とするため、用途対応に CTRON で規定するインタフェースの必要部分を取捨選択的に切り出して利用できること。

### 3.2 2階層構造とインタフェース区分

CTRON 参照モデルに対する要求条件①を満足させるために、CTRON インタフェースをハードウェアアーキテクチャの差を隠蔽する基本 OS インタフェースと、アプリケーションに対するより高水準のインタフェースを提供する拡張 OS インタフェースの2階層で構成する(図-2)。これによって、拡張 OS やその上位のソフトウェアをアーキテクチャの差に依存しない形で構築することが可能となり、ソフトウェアのポータビリティ向上が図られる。

要求条件②を満足させるために、CTRON インタフェースは OS が扱う資源対応に分類することとし、これをインタフェース区分と名付けている。ハードウェアは一般にプロセッサ系資源と入出力系資源に分類できるため、基本 OS インタフェースは、プロセッサ系アーキテクチャの差を隠蔽するカーネルインタフェース<sup>13)~15)</sup>と入出力系アーキテクチャの差を隠蔽する入出力制御インタフェース<sup>16)</sup>の2種類のインタフェース区分から構成している。一方、拡張 OS インタフェー

スは、ファイル系資源、通信系資源などの論理資源の種類に対応した情報蓄積制御インタフェース、通信制御インタフェースなどの6種類のインタフェース区分から構成している。

要求条件③を満足させるため、CTRON では、インタフェースを選択・組合せ可能な単位(サブセット単位と呼ぶ)に分割している。

POSIX やデファクト標準の OS ではサブセットの概念はない。MOSI では、機能モジュール<sup>2)</sup>と称するサブセット単位が存在するが、その数が多く、それらを組み合わせて生成されるサブセットの種類は膨大な数になり、ソフトウェアポータビリティの観点からは問題がある。

CTRON では、サブセット単位の組合せであるサブセットの個数を限定するとともに、ソフトウェアの上方向互換を保証するサブセット系列を規定しており、ソフトウェアのポータビリティ確保を図りながら、各ノードの性能条件、機能条件に合致した OS の構築を可能としている。

## 4. 主要な機能とその特徴

### 4.1 基本 OS インタフェース

#### (1) カーネルインタフェース

カーネルインタフェースでは、プロセッサ系資源をモデル化し、個々のプロセッサアーキテクチャに依存しないインタフェースを規定している<sup>13)</sup>。具体的には、プロセッサの割当て対象である並行処理単位をタスクと定義し、タスクを実行する上での必要機能を規定しており、他の代表的な OS のカーネル相当の機能をほとんど包含したものとなっている(表-1)。以下では、カーネルインタフェースの主要な特徴を述べる。

#### (a) リアルタイム機能の強化

リアルタイム機能の実現のため、タスク制御に次の機能をもたせている(図-3)。

① タスクが事象発生と同時にスケジュールされる「プリエンプティブスケジュール」の機能が規定されている。これにより、交換ノードや通信処理ノードに特徴的な突発的な事象の発生に迅速に対処できる。

② 縮退した状態遷移をもつタスクが規定されている。一般のタスク状態遷移においては、メモリの過負荷状態を回避するために一部のタスクを強制的にディスパッチの対象外にするサスペンド状態や、入出力動作の完了などを待ち合わせるためにそのタスクをディ



表-1 カーネルと他 OS のインタフェース比較

特徴的な機能	CTRON	POSIX	MOSI	MTOS	VRTX	備 考
タスク管理機能	タスク生成/起動(終了/削除)の区別(タスクのドーマント状態)	○	—	○	—	リアルタイム性の観点からのオーバーヘッドの削減
	タスクスケジューリング禁止/解除(タスクのサスペンド状態)	○	—	○	—	○ 負荷制御スケジューリング機能の柔軟性
	タスク終了ルーチン登録	○	—	—	—	タスク終了処理の柔軟性
	タスクの周期スケジューリング	○	—	—	—	リアルタイム性の観点からのスケジューリングの多様性
	タスクのトレース機能	—	○	—	—	OS の内部制御方式を意識したインタフェース
タスク間同期通信機能	イベントフラグ	○	○	—	○	同期通信機能の利便性と高信頼化 * 応答の同期待ち機能あり 排他制御のオーバーヘッド削減
	セマフォ	○	○	○	○	
	メッセージボックス	○	○	○	○	
	ランデブ	○	—	—*	—	
	逐次再使用資源排他制御	○	—	—	—	
タイマ管理	システムタイマ	○	○	○	○	通信処理などでのタイマ制御の柔軟性とオーバーヘッド削減
	私用タイマ	○	—	○	—	
割り込み管理	割り込みハンドラの登録	○*	—	—	—	* 割り込みの仮想化を実施
	割り込みマスクの制御	○*	○	○	—	* 割り込みの仮想化を実施
例外管理	例外ハンドラの登録	○*	○	○	—	* 例外の論理的なクラス分けを実施
	例外マスクの制御	○*	○	○	—	
メモリ管理	メモリアールの生成/削除	○	—	—	○	メモリ負荷制御の柔軟性
	共用空間の確保/解放	○	○	—	—	
	仮想メモリ空間管理	○	○	—	—	メモリ負荷制御の柔軟性
	ロールイン/ロールアウト	○	—	—	—	

注) MTOS は、米国 Industrial Programming 社の登録商標である。

VRTX は、米国 Ready Systems 社の登録商標である。

スパッチの対象外にするウェイト状態が存在する。しかし、リアルタイム処理を受けもつタスクがその対象となると処理性能が保証できなくなるため、これらの状態をもたないタスクを規定している。

③ 定期的な監視動作などを行うタスクの起動の高速化を図るため、周期スケジューリング機能が規定されている。

(b) プロセッサアーキテクチャの仮想化

カーネルではプロセッサアーキテクチャに依存しないインタフェースを提供するが、すべてを無理に規定しようとする、性能の低下を招き、リアルタイム処理などへの適用が不可能となる。そこで、性能の低下を招かない範囲で、インタフェースを規定している。

規定方法をメモリ管理を例に説明する。メモリアーキテクチャは、実メモリアーキテクチャと仮想メモリアーキテクチャに大別できる。メモリに関する基本機能である、メモリの確保、解放機能、および特定の用途に使用するメモリアールの定義、削除機能などは、

両アーキテクチャで無理なく規定できるため、共通のインタフェースとしている。仮想メモリの予約機能、予約した仮想メモリへの実メモリ割当て機能などは、仮想メモリアーキテクチャ専用のインタフェースとして規定している。また、アーキテクチャによる違いの大きいメモリ保護機能については、機能名のみで規定にとどめ、ソフトウェア移植時に修正範囲を特定するための手掛かりを与えるようにしている。

(c) サブセット仕様

CTRON インタフェースは、各プロセッサ、各ノードに適用する場合、必ずインプリメントしなければならないサブセット単位である共通部とプロセッサやノードに応じて取捨選択できるサブセット単位である選択部から構成される。

サブセット単位の組合せであるサブセットが、個々の OS の仕様となる。

カーネルインタフェースは 4 種類のサブセット単位から構成される。これらから 5 種類のサブセットと、

タスク管理		タスク周期起動制御	
CRE_TSK	タスク生成	CYC_WUP	タスク周期起床
STA_TSK	タスク起動	CAN_CYC	タスク周期制御解除
CAS_TSK	タスク生成起動	WAI_TSK	タスク遅延起床
EXT_TSK	タスク終了	CAN_WUP	タスク起床解除
EXD_TSK	タスク削除	DEF_EXT	タスク終了時処理ルーチン定義
SUS_TSK	タスク停止	ABO_TSK	タスク異常終了
RSM_TSK	タスク停止解除	GET_TID	タスク識別子取得
WUP_TSK	タスク起床	REF_ABD	異常終了コード参照
SLP_TSK	タスク就寝	GET_TSI	自タスク情報取得
TER_TSK	タスク強制終了	CHG_TSI	自タスク情報変更
DEL_TSK	タスク強制削除	GET_TIF	他タスク情報取得
		CHG_TIF	他タスク情報変更

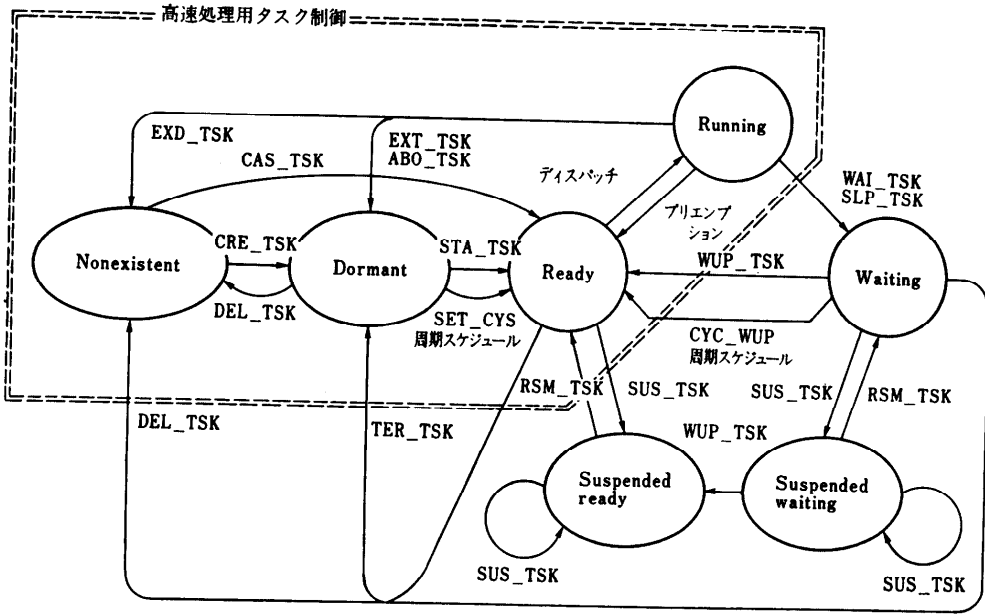


図-3 カーネルにおけるタスク制御

二つのサブセット系列を規定し、それぞれのサブセット系列で、ソフトウェアの上方向のポータビリティを保証している<sup>14)</sup> (図-4)。

(2) 入出力制御インタフェース

入出力装置に係わるインタフェースは、入出力装置へのアクセス機能と入出力装置の管理機能に分けられる。

このうち、後者は個々の入出力装置アーキテクチャの詳細に立ち入ることが少ないため、すべての入出力装置共通のインタフェースとして規定されている。一方、前者については、アーキテクチャ差の隠蔽と技術進歩への追従の二つを両立させるため、以下の2種類の論理装置インタフェースを設けている。

① BIO (Basic IO) インタフェース: 磁気ディスク

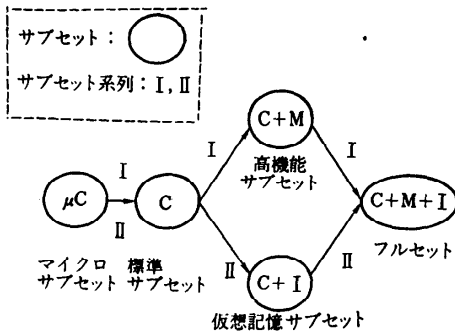
ク、磁気テープなどのように、入出力装置アーキテクチャが比較的固定化しているものに関して、入出力装置の種別対応に、アクセス機能を規定したインタフェース。

② GIO (General IO) インタフェース: 今後新たに出現する装置や、アーキテクチャが発展途上にあり類型化の困難な装置を制御可能とするため、論理装置仕様をユーザが定義できるようにしたインタフェース。

4.2 拡張 OS インタフェース

4.2.1 情報蓄積制御インタフェース

情報蓄積制御は、ファイルやデータベースなどの情報のアクセス、管理を行うためのインタフェースである。



サブセット単位	機能	システムコール数
μC (マイクロ)	タスク管理 タスク間同期通信 (イベントフラグ, メッセージボックス) 割込み管理 時計管理 メモリ管理	44
C (共通部)	タスク間同期通信 (セマフォ, シリアルリユーザブル) 例外管理 統計情報管理	84
M (高機能選択部)	高速処理用タスク制御 タスク周期起動制御 メッセージ選択受信制御 ランデブ制御 専用タイマ制御	19
I (仮想記憶選択部)	仮想記憶制御 ロールイン/ロールアウト制御 基本プログラム管理	22

図-4 カーネルのサブセット体系

(1) 一般ファイル管理インタフェース

ファイルの作成・削除、アクセスの効率化を目指して、一般ファイル管理インタフェース<sup>17)</sup>を規定している。

ファイルの操作に当たっては、ファイルの所在管理機能とファイルへのアクセス機能が必要である。

CTRONでは、多数のユーザのファイルやファイル群を階層的に管理するために、木構造形式のディレクトリをもった所在管理を規定している。

一方、ファイルアクセスのための編成法として、以下の2種類を規定している。

① 複数ブロック単位での高速入出力を含め、種々の単位での入出力を行える基本編成ファイル

② キーに従って、レコードの検索、挿入、削除などを行える索引順編成ファイル

(2) データベース管理インタフェース

一般に、データベースは、その論理的な構造やデータの操作(参照, 更新)の観点から、①網構造(木構造は網構造の特殊な場合)に基づくネットワークデータベースと②表構造に基づくリレーショナルデータベースの二つに分類されるが、CTRONでは、操作性の良さと標準仕様の普及度合から、リレーショナルデータベースを対象としている。具体的には、ISO標準であるSQL<sup>18)</sup>インタフェースを採用し、そこでインプリメント定義扱いになっている事項のうち、ソフトウェアポータビリティ上定義が必要なデータ型の精度などの項目を規定している。

(3) 実身/仮身ファイル管理インタフェース

CTRONでは、BTRONの実身/仮身ファイル管理インタフェース<sup>19)</sup>を採用している。これにより、BTRONノード上で作成したファイルをCTRONノード上で利用し、さらにその結果をBTRONノード上に戻して編集、加工できるようになる。

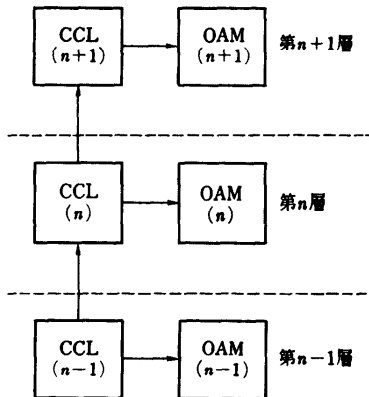
4.2.2 通信制御インタフェース

通信制御インタフェース<sup>20)</sup>は、OSI(Open Systems Interconnection)<sup>21)</sup>の7階層モデルに準拠し、さらにCCITT勧告の通信プロトコル体系とも整合をとっている。第1層は、ハードウェアで実現されるのが一般的であるため、OSインタフェースとしては規定しない。第2層から第5層までは、各層対応にOSインタフェースを規定することにより、性能のよい通信制御プログラムの実現を可能とするとともに、通信制御用装置での層のサポートレベルに応じた適用性を確保している。第6, 7層については、FTAM<sup>22)</sup>, MOTIS<sup>23)</sup>, ISDNユーザ制御<sup>24)-26)</sup>などの応用分野に対応する単位でOSインタフェースを規定している。また、各層間の電文の授受を行う場合に必要となるバッファ管理インタフェースなどを各層共通のものとして規定している。

この結果、ISDN対応のインタフェースを他の通信機能と同じOSインタフェース体系で規定できており、情報処理技術と交換処理技術の融合が図られている。

通信制御では、各層が以下の機能モジュールから構成される(図-5)。

① CCL(Communication Controller: 通信プロトコル処理部): OSI規定のサービスで、コネクションの確立と解放、データの転送などの通信にかかわるサービスを実現するとともに、OSI資源の活性/非活



CCL: Communication Controller (通信プロトコル処理部)

OAM: Operation Administration and Maintenance (通信資源管理部)

図-5 通信制御各層のインタフェースの分類

性化、監視、試験などの管理のための機能を提供する機能モジュール

② OAM (Operation Administration and Maintenance: 通信資源管理部): CCL の機能を用いて、OSI 資源の管理を行う機能モジュール

通信制御インタフェースとしては、下位層 CCL が上位層 CCL に提供するインタフェースおよび層内で CCL が OAM に提供するインタフェースを規定している。ネットワーク管理構築の自由度を考慮し、OAM が提供するインタフェースは規定していない。

通信制御では、各層内で共通部、選択部のサブセット単位、およびその組合せであるサブセットを規定している。また、層間では回線交換、パケット交換、LAN などのネットワーク形態対応に各層のサブセットの組合せを規定している。

#### 4.2.3 実行制御インタフェース

実行制御インタフェース<sup>27)</sup>は、各種情報通信処理を効率的に実現する上で必要な資源の管理機能、実行対象の制御機能を規定するものである。実行制御インタフェースは、各処理形態に共通な一般プログラム管理、プロセス管理と、各処理形態独自のバッチ制御、対話制御、トランザクション制御などのサービス制御から構成される。

##### (1) 一般プログラム管理インタフェース

一般プログラム管理は、プログラムの格納されたファイルのサーチ機能 (ライブラリ管理)、ファイルからメモリへのプログラムのロード機能、プログラムの

起動機能から構成される。ここでは、リアルタイム性を保証するためのプログラム起動高速化の考え方について述べる。

プログラムを起動するためには、その前処理として、対象となるプログラムが格納されているファイルを特定し、そのファイルの内容をメモリにロードする必要がある。プログラム起動時の高速化を達成するため、プログラムロードまでの一連の処理をプログラム起動と分離し、プログラム起動以前に行うことを可能としている。

なお、対話型システムのユーザ向けとして、プログラムロードからプログラムの起動までの一連の処理を一つのシステムコールで行える機能も規定している。

##### (2) プロセス管理インタフェース

カーネルでは、並行処理単位としてタスクが定義されているが、タスクへの資源の割当ては利用者に任されている。一方、実行制御では複数のタスク (1個でも可) にメモリ、プログラム、ファイルなどの資源を割り当てたものをプロセスと定義し、これを並行処理の単位として利用者に提供している。プロセスを定義し、その実行制御機能を規定することにより、タスクに比べより高度な資源管理をサポートすることが可能となり、利用者側プログラムの開発が容易となる。また、プロセス内のタスクは、ユーザの要求に応じ、最適な構成をとることができ、たとえば、同一プロセス内のタスク間でのメモリ、ファイル資源の共用が容易になるなどの利点がある。

プロセス管理インタフェースとしては、資源の割付・解放、プロセスの起動・停止、プロセス間同期通信などの各機能を規定している。

#### 4.2.4 その他の拡張 OS インタフェース

##### (1) 通話路制御インタフェース

交換システムは、回線交換とパケット交換に大別される。前者は、入側・出側の回線を通話路と呼ばれるハードウェアにより、スイッチングするものである。後者は蓄積交換であり、ユーザデータを直接メモリ内に取り込み、フロー制御など各種の操作を加える方式である。パケット交換は、通信制御インタフェースとして規定されており、通話路制御<sup>7)</sup>では、回線交換方式の交換システムに適用できるインタフェースを規定している。

具体的には、通話路構成を、集線段のみ、集線段+分配段、分配段のみの3種類とし、これに接続される信号装置を組み合わせる形でパターン化し、通話パスの補足や接続などのインタフェースを検討している。



表-2 CTRON インプリメント状況/予定

項目	適用分野/サービス	使用プロセッサ	最初の提供時期	CTRON 実現範囲/使用サブセット	特 徴
A	情報通信処理 交換処理	Micro シリーズ (TRON 仕様チップ)	'89	カーネル (C+M+I) 入出力制御 通信制御など	<ul style="list-style-type: none"> <li>高性能</li> <li>拡張性</li> <li>デバッグ/運用両モードのサポートなど</li> </ul>
B	通信処理 (当初は通信ネットワークの ゲートウェイ処理サービス)	B 社仕様プロセッサ (汎用機)	'90.3	カーネル (μC) 入出力制御 (BIO(DK, FD, PR, CDV), GIO) 通信制御	<ul style="list-style-type: none"> <li>通信ネットワーク高速化、多様化に十分対応できる性能、機能</li> <li>拡張性を十分に考慮</li> </ul>
C	ISDN 端末	I 80186	'88.12	カーネル (μC)	<ul style="list-style-type: none"> <li>MS-DOS との協調動作</li> <li>他 CPU への移植しやすい構造</li> </ul>
D	通信処理	M68020 I 80386	'89.4 '90.4	カーネル (μC) カーネル (C) 入出力制御 (BIO(DSP, FD, HD), GIO)	<ul style="list-style-type: none"> <li>簡易プロセッサ間通信機能</li> <li>簡易プロセッサ間通信機能</li> <li>マルチ CPU/マルチ OS</li> </ul>
E	交換機などの組込みシステム	E 社仕様プロセッサ (汎用機)	'89.9	カーネル (C+M)	<ul style="list-style-type: none"> <li>多重仮想空間を利用した組込みシステム用 OS</li> <li>ランデブ機能の高速実現</li> <li>システムジエネレーション機能の強化</li> </ul>
F	通信処理 交換処理 通信処理	MC 68000 系 F 社仕様プロセッサ (汎用機)	'90.3 '89.3	カーネル (C) 入出力制御 (BIO(DK, FD), GIO) カーネル (C+I) ただし、一部未サポート 入出力制御 (BIO(DK, FD, MT, PR, DSP), GIO) 一般ファイル管理 (B) 一般プログラム管理 通信制御 (セッション層)	<ul style="list-style-type: none"> <li>通信処理指向のリアルタイム OS</li> <li>CTRON 準拠 OS 上で UNIX 環境を提供</li> <li>今後、ISDN への対応、リレーショナル DB サポートを予定</li> </ul>
G	交換処理 通信処理 性能、機能評価用のプロトタイプ機	F 社仕様プロセッサ (交換通信処理用) MC68000 系列 TRON 仕様チップ	'88.4 '88.8 (MC68020 スタンダード タイプ)	カーネル (C+M) 入出力制御 (BIO(DK, FD, MT, PR), GIO) カーネル (C) (C+M+I) も開発中 入出力制御 (BIO(DK, FD, DSP, PR, CDV), GIO) 一般ファイル管理 (B+E) 装置管理を開発中 DB管理/通信制御を予定	<ul style="list-style-type: none"> <li>高性能</li> <li>CHILL サポート</li> <li>トストスタンバイの二重化プロセッサ方式によるフォールトトレラント環境の設計 (スタンダードタイプ)</li> <li>並列処理方式による高性能化 (ハイパフォーマンススタイル)</li> <li>入出力処理をメインプロセッサから分離した IOP 方式の採用</li> <li>シリコン化された基本 OS</li> <li>マルチタスクデバッグによるセルフデバッグ</li> <li>CTRON 準拠 OS 上で UNIX 環境を提供</li> </ul>
H	端末	I 80386 I 80386	'89 '90.3	カーネル (C+M+I) 入出力制御 (BIO(DK, FD, シリアルポート), GIO) カーネル (C+I) 入出力制御 (BIO(DK, FD), GIO)	<ul style="list-style-type: none"> <li>オンラインデバッグ機能</li> <li>SG 機能によるカーネルの切り出し</li> <li>C 言語によるカーネル記述で移植性確保</li> <li>多重仮想空間およびオンディマンドページング機構を実現</li> <li>CTRON 準拠 OS 上で UNIX 環境を提供</li> </ul>

## (2) HMI 制御インタフェース

ヒューマンマシンインタフェース (HMI) 制御に関しては、BTRON の HMI 制御インタフェースをベースとし、これに CGI<sup>28)</sup> などの国際標準化の成果を取り込む方針で検討している。

## (3) 保守運転管理インタフェース

保守運転管理は、システムの保守・運転時に必要となる装置管理、運転制御、課金制御、障害処理、試験制御などを行う機能である。これらの機能は、システムの運用・サービス条件に依存する事項が多いので、CTRON では基本的にこれらのインタフェースは規定しない。ただし、他のインタフェース単位との関係の深い装置管理インタフェースについては最小限の規定を行っている。

## 5. インプリメントと評価

## 5.1 CTRON 準拠 OS のインプリメント状況

CTRON 仕様に基づく OS のインプリメントが各社で実施されており、すでに製品の出荷が始まったもの、実サービスに適用されているものもある。1990年 には、汎用マイクロプロセッサや各社仕様プロセッサ上での製品が出揃う予定である (表-2)。第 1 期製品は、交換・通信処理分野を対象とした基本 OS 製品が

中心であるが、拡張 OS についても順次製品化が進展している。

開発が完了している製品には、高性能な基本 OS を実現し、ISDN サービスの交換システムや通信システムに適用している例<sup>29)</sup>や、ホットスタンバイの二重化プロセッサによってフォールトトレランス性を追求した例<sup>30)</sup>などがある。

## 5.2 インプリメントによる評価

CTRON 仕様の改善に役立てることを目的に、開発の進んでいる基本 OS に関して、インプリメントからの意見を収集した (表-3)。

これらの意見から現 CTRON 仕様を評価すると、以下のことがいえる。

① 情報通信ネットワークサービスへの適用性に関しては、機能、性能的にはほぼ十分であるが、フォールトトレランス性や分散処理機能に関して、さらに仕様の充実が必要である。

② ソフトウェア流通性に関しては、代表的なマイクロプロセッサ対応の規定が必要である。

③ サブセット規定に関しては、カーネルサブセットの見直しを実施したにもかかわらず、まだ改善要求が存在しており、サービスへの適用性とソフトウェアの流通性を両立させることの難しさを示している。

表-3 インプリメントによる評価 (基本 OS に関して)

大項目	小項目	評価, 改善点 (規定の追加を必要とする項目: ☆)
情報通信ネットワークサービスへの適用性	機能	●機能的には、ほぼ十分 ☆トランザクション処理制御などの規定
	リアルタイム性	●他のリアルタイム OS を上回る性能が実現可 ●排他制御は、性能向上の余地あり ●拡張 OS で呼とカーネル資源の対応付けを実現すると、交換システムではスループットが低下
	フォールトトレランス性	●完全二重化システムのための障害検出、復旧機能の検討が必要
	分散処理	☆ネームサービス機能、プロセッサ間同期通信機能の規定やセキュリティチェック機能の充実
ソフトウェア流通性向上のための追加規定	ハードウェアアーキテクチャ依存事項	●カーネルでのプロセッサ系アーキテクチャ差の隠蔽効果は高い ☆代表的なマイクロプロセッサ対応にアーキテクチャ依存仕様の規定 ●入出力制御は、インプリメント依存部分が多く、隠蔽効果は不十分
	初期化, SG	☆初期化やシステムジェネレーション (SG) など、インプリメント依存となっている事項の規定
	シンボル定義	☆エラー番号のシンボル定義 (プログラム理解性とソフトウェアポータビリティの向上のため)
サブセット規定	バージョン規定	☆サブセット仕様とバージョンとの関連の規定
	カーネルサブセット	☆マイクロサブセット ( $\mu C$ ) よりさらに小さな、小規模通信機器向けサブセット仕様 ☆MMU 付きハードウェア用サブセット仕様 ●高速処理用タスク制御の共通部化

## 6. 今後の展望

CTRON はオープンな組織である CTRON 専門委員会にて検討しており、検討の成果であるインタフェース仕様書は、英語、日本語で出版され、パブリックドメインとして広く世界に公開されている。CTRON のうち、カーネル、入出力制御、情報蓄積制御（一般ファイル管理、データベース管理）、通信制御、実行制御（一般プログラム管理、プロセス管理）、保守運転管理（装置管理）の各インタフェース仕様書は、すでに出版されている。さらに他のインタフェースについても、1989年度末までに一通りの設計を完了し、順次出版・公開される。

CTRON 仕様については、CTRON 専門委員会参加メンバーの拡大、仕様書の出版、インプリメントの進展などを通じて、さらにリファインされる予定である。

また、今後、より多くのコンピュータメーカ、通信機器メーカが CTRON 仕様に基づく性能の良い OS ソフトウェアを開発することにより、CTRON が今後の新しい OS インタフェースとして広く普及し、ソフトウェアポータビリティの向上に寄与することを期待する。

**謝辞** CTRON 開発に当たって、終始ご指導をいただいた東京大学坂村健博士、NTT 石野福弥博士ならびに CTRON 専門委員会の富士通(株)、(株)日立製作所、松下通信工業(株)、三菱電機(株)、日本電気(株)、日本電信電話(株)、ノーザンテレコムジャパン(株)、沖電気工業(株)、(株)東芝の関係各位に深謝いたします。

## 参 考 文 献

- 1) Wheeler, E.F. and Ganek, A.G.: Introduction to Systems Application Architecture, IBM Systems Journal, Vol. 27, No. 3, pp. 250-263 (1988).
- 2) Jackson, D.L. (chm.): IEEE Trial-Use Standard Specification for Microprocessor Operating Systems Interfaces, Institute of Electrical and Electronics Engineers, Inc., New York (1985).
- 3) Isaak, J. (chm.): IEEE Trial-Use Standard Portable Operating System Environment P1003/D6, Computer Society of the IEEE, Washington, D.C. (1985).
- 4) Cragum, D.W.: Portable Operating System Environment, Proc. 5th Annual Phoenix Conf. on Computers and Communications, IEEE, New York (1986).
- 5) Wasano, T., Ohminami, M., Kobayashi, Y., Ohkubo, T. and Sakamura, K.: Design Principles and Configuration of CTRON, Proc. FJCC '87, Dallas, pp. 159-166 (1987).
- 6) Wasano, T., Ohminami, M., Kobayashi, Y., Ohkubo, T. and Sakamura, K.: Design of CTRON, TRON Project 1987, Springer-Verlag, pp. 157-172 (1987).
- 7) 坂村 健監修, (社)トロン協会編: CTRON 概説, オーム社 (1988).
- 8) Sakamura, K.: The TRON Project, IEEE Micro, pp. 8-14 (1987).
- 9) The Ada Compiler Validation Procedures and Guidelines, Ada Joint Program Office (AJPO) (1987).
- 10) MTOS-68K User's Guide, Industrial Programming Inc. (1982).
- 11) System V Interface Definition Issue 2, AT &T (1986).
- 12) Wasano, T., Kobayashi, Y. and Sakamura, K.: CTRON Reference Model, TRON Project 1988, Springer-Verlag, pp. 145-155 (1988).
- 13) Ohkubo, T., Wasano, T. and Kogiku, I.: Configuration of the CTRON Kernel, IEEE Micro, pp. 33-44 (1987).
- 14) Kogiku, I., Ohkubo, T. and Matsushita, M.: Enhancement of the CTRON Kernel Interface, TRON Project 1988, Springer-Verlag, pp. 189-211 (1988).
- 15) 坂村 健監修, (社)トロン協会編, カーネルインタフェース, オーム社 (1988).
- 16) Narimatsu, S.: Design of CTRON Input-output Control Interface, TRON Project 1987, Springer-Verlag, pp. 183-196 (1987).
- 17) Kumazaki, K.: Design of the CTRON File Management, TRON Project 1987, Springer-Verlag, pp. 173-182 (1987).
- 18) ISO 9075: 1987 Database Language SQL.
- 19) Sakamura, K.: BTRON: The Business-oriented Operating System, IEEE Micro, pp. 53-65 (1987).
- 20) Shimizu, Y.: Design of CTRON Communication Control Interface, TRON Project 1988, Springer-Verlag, pp. 157-166 (1988).
- 21) ISO 7498: 1984 Information Processing Systems-Open Systems Interconnection-Basic Reference Model.
- 22) ISO 8571-1: 1987 (E) Information Processing Systems-Open Systems Interconnection-File Transfer, Access and Management-Part1: General Introduction.
- 23) ISO 10021-1 Message Handling: System and Service and Overview.

- 24) CCITT 勧告 Q. 931/I. 451 (1988).
- 25) CCITT 勧告 Q. 76 X (1988).
- 26) CCITT 勧告 Q. 77 X (1988).
- 27) Baba, Y., Ohminami, M., Kusumoto, H. and Kosugi, H. : Design of CTRON Execution Control Interface, TRON Project 1988, Springer-Verlag, pp. 167-187 (1988).
- 28) ISO/DP9636(CGI) : 1988—Interim Draft "Information Processing Systems-Computer Graphics-Interfacing Techniques for Dialogues with Graphical Devices-Functional Specification".
- 29) 大類 駿, 須田宏一, 大久保利一 : CTRON の交換システムへの応用, トロン技術研究会, pp. 73-79 (1988).
- 30) Ishizuka, M., Ikeda, Y., Fukuyoshi, M., Ogawa, T., Sakuma, T. and Matsushita, M. : An Implementation of the CTRON Basic OS, TRON Project 1988, Springer-Verlag, pp. 213-234 (1988).

(平成元年1月23日受付)