

Development of General Purpose Internetworking Unit

要旨

川崎製鉄では、任意のネットワーク間の接続を実現する汎用インタネットワーキング装置のプラットフォームを開発した。本プラットフォームは、複数のネットワーク・インタフェースから構成される。各インタフェース上の OS は、プロトコル処理の特性を考慮したダイナミック・タスク・スイッチング方式を採用し、プロトコル



- (1) 各種の通信インタフェースに対応できること。たとえば、IEEE 802.3, V. 24, X. 21, RS-232 C, ISDN などや、さらに今後新たに生まれるもの。
- (2) (1)の通信インタフェース上で各種のプロトコルが搭載できること。さらに、新プロトコルにも対応できること。
- (3) プロトコル処理プログラム開発の生産性を高められること。
- (4) ネットワーク管理などのアプリケーション的なシステムを搭載できること。

以上のような条件を満足するために、以下のアプローチをとった。

- (1) 機能、負荷の分散を目指し、通信インタフェースごとにプロ

めに今回開発したインターネットワーキング OS の機能構成を Fig. 2 に示す。

複数のタスクを処理するためのスケジューラをカーネル部分に持ち、そのまわりでバッファやタイムなど各種の資源を管理する OS サービスが動作する。アプリケーション・タスクはこの OS 上で動作するが、プロトコル・タスクの場合には、プロトコル生成言語インタプリタを介して、OS サービスの提供を受ける。

3.3 アプリケーション

インターネットワーキング OS 上で動作させるアプリケーションとして、次のような処理がある。

マルチプロセッシング方式。

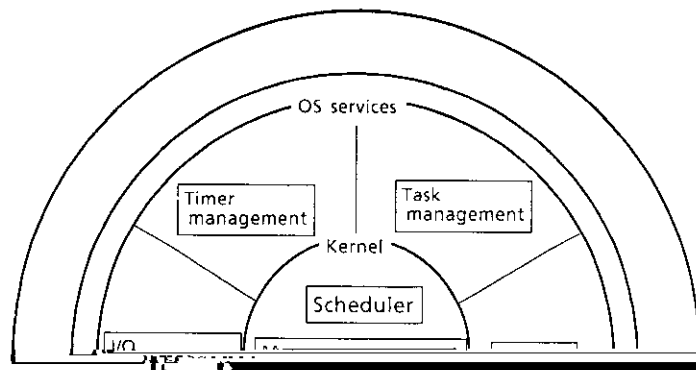
- (1) 単一プロセッサ内でマルチ・プロトコル環境を実現するた

(1) プロトコル処理

- (1) 基層モニタ

め、また、ネットワーク管理などの付加的なアプリケーション

(3) 自己診断



たるため、その解析は非常に複雑なものとなる。

TCB pointer of currently executing task

LSI にデータ受信コマンドを発行する場合。特に、LSI からの
送信完了割り込み発生後、通信の同期のため相手側の受信

6.2.1 実行状態からの状態遷移のオーバーヘッド

実行状態から実行可能状態へ遷移する場合、待機状態から実行可能状態へ遷移する場合よりもオーバーヘッドが大きい。

てくる受信確認データの受信を待つ場合。

(3) 通信エラー

外部処理実行中に、なんらかの外部条件により通信の続行が

不能となっていないかを確認する場合。

上述の(1)および(2)の場合、監視タイマをスタートさせ、LSI
からの割込みとタイマからの割込みの両方を待つことになるが、い
ずれもプロトコル処理タスク側からのコマンド発行直後はイベント

登録処理がある。プロトコル処理タスクのように、待つべきイベン
トの種類が多い場合には、それだけたくさんの登録処理が必要とな
るため、オーバーヘッドも比較的大きなものとなる。

これに対し、実行状態から実行可能状態に遷移する場合、このよ
うな処理は不要で、OS は単にタスクから実行権を取り上げるのみ
である。したがって、ダイナミック・タスク・スイッチング方式に
おけるような状態遷移よりも、実行可能状態から実行状態へ遷移する OS

7 プロトコル生成高級言語

プロトコル生成高級言語は、プロトコル処理プログラムの開発に際し、プログラマがプロトコル処理の複雑さ、プロトコル処理タスクの管理に際して意識しなくてはならないことを可能と

なツールとなる汎用インターネットワーキング装置を開発するために、そのプラットフォームとして、ハードウェア、インターネットワーキング OS およびプロトコル生成高級言語を開発した。

本装置は、任意のネットワーク間の接続とそれらの管理を目的としたもので、各種のネットワークに対応するための通信インタフェース・ボードと装置の制御のためのメイン・ボードから構成され