

5M-6

LLCタイプ2実装方式の検討

松尾 英普¹⁾、渡辺 晶¹⁾、今井 光雄¹⁾、大米 祥夫²⁾

1)日立電線(株)オプトロシステム研究所 2)日立コンピュータエンジニアリング(株)

1.まえがき

OSI第2層までのプロトコルを実行するLAN用通信アダプタを開発した。本アダプタは、プロセッサを内蔵するインテリジェントタイプであり、OSI第2層の上位副層としてIEEE802.2LLCクラス2を実装している。本稿は、このうち特にLLCタイプ2規格に関してn個のデータリンクコネクションをサポートする上で効率的なタイマ管理という観点から、ソフトウェアの実装方式について述べる。

2.ハードウェア構成

アダプタ機能のうち、OSI第2層の下位副層(MAC副層)以下の機能は、LAN用通信制御LSIが行う。OSI第2層の上位副層(LLC副層)の機能はアダプタに搭載したプロセッサがソフトウェアを実行することにより行う。アダプタを内蔵する上位システムとは共有メモリを介してデータの受渡しを行う。

3.ソフトウェア構成

アダプタに搭載したプロセッサは、LLC副層の機能の他に、上位システムとのインターフェース、LAN用通信制御LSIの制御等を行う。これらの機能の実行を制御するため、独自に開発したOS上で、各機能をタスクとして実現する。OSはスケジューラ、タスク管理、タイマ管理、タスク間同期通信管理、メモリ管理モジュールからなる。

4.LLCタイプ2手順の実装方式

IEEE802.2LLCクラス2として規定されたタイプ1とタイプ2手順のうち、タイプ2手順は、HDLCのABMEに相当する手順である。タイプ2手順におけるコネクション要素数(n)は、実装者に任せられており、今回は128個のコネクション要素をサポートする。また、各コネクション要素はタイプ2手順において定められたACKタイマ、Pビットタイマ、BUSYタイマ、REJタイマの他に、応答遅延タイマ、無活動監視タイマの計6個のタイマを使用する。n個のコネクション要素をサポートする方式として、以下の案が考えられる。

(案1)全コネクション要素を1タスクで実現する。
この方式はタイマの管理方法に関してさらに次の方式が考えられる。

(案1a)全タイマを、OSが管理する。(図1)
(案1b)全タイマを、タスクが管理する。(図2)
(案2)1コネクション要素毎に1タスクとする。(図3)
(案2)では各コネクション要素が使用する6個のタイマはそれぞれのコネクション要素に該当するタスクが管理する。
(案1a)では、タスクはタイマのスタート、ストップをOSのシステムコールにより指示する。これに対

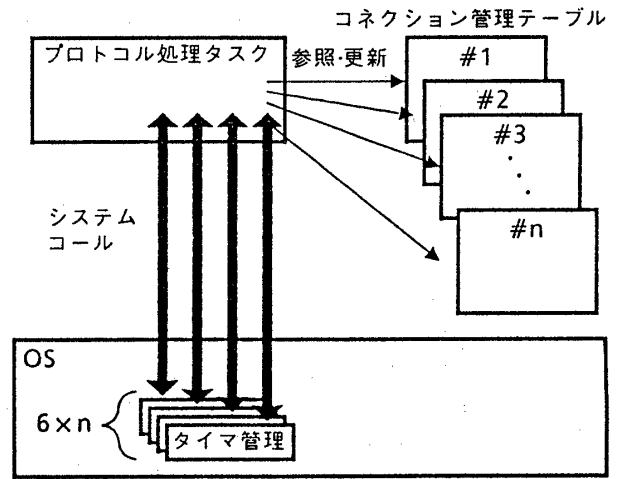


図1 案1a

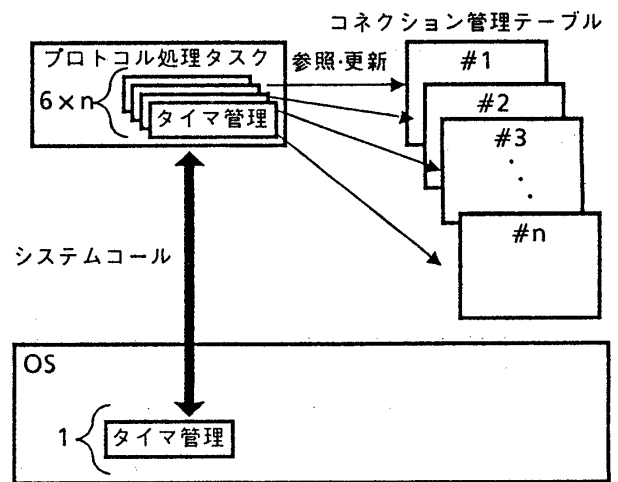


図2 案1b

A study on implementation method of LLC Type2 procedure

Hideyuki MATSUO¹⁾, Akira WATANABE¹⁾, Mistuo IMAI¹⁾, Yoshio OYONE²⁾

1)Optoelectronic System Laboratory,Hitachi Cable,Ltd. 2)Hitachi Computer Engineering,Ltd.

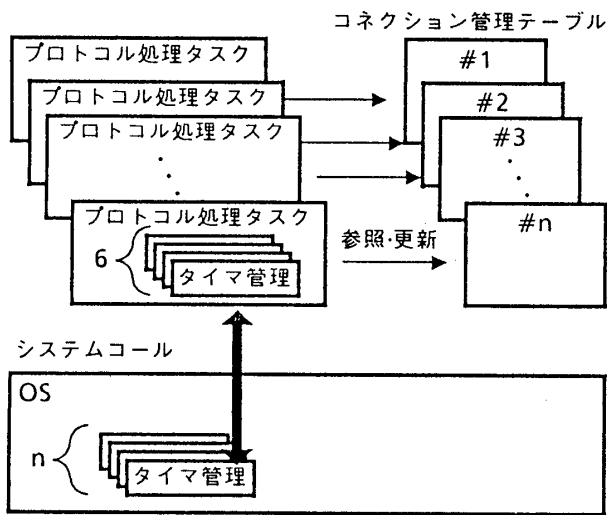


図3 案2

し(案1b)と(案2)では、タスク側で管理するタイマのなかから最も早くタイムアウトするタイマ値をOSに指示する。OSは各タスク毎にタイマを管理し、タイムアウトすると該当するタスクを起床する。この時、タスクが指定したタイムアウト時間からの経過時間を通知することによってタスク側はタイマを更新する。

又、(案2)は、タスク毎にスタックやタスク制御ブロックが必要となり、(案1a)と(案1b)と比較してメモリ容量が大きくなるという欠点がある。

次に、各方式のタイマ更新回数を比較するために以下の仮定を用いる。

- ① OSはタイマ割り込みが発生する毎にOSが管理する全タイマを更新する。
- ② タスクは起床される毎にタスクが管理する全タイマを更新する。

すると、1秒間当たりの各方式のタイマ更新回数は次のようになる。

(案1a)
$$I \sum_{i=1}^C T_i$$

(案1b)
$$I + W \sum_{i=1}^C T_i$$

(案2)
$$CI + \sum_{i=1}^C T_i W_i$$

(C : アクティブなコネクション要素数、
 T_i : コネクション要素*i*が使用しているタイマ数、
 I : 1秒間に発生するタイマ割り込み回数、
 W : (案1b)において1秒間にタスクを起床する回数、
 W_i : (案2)において1秒間にコネクション要素*i*に該当するタスクを起床する回数)

ここで、

$I = 64,$
 $T_i = 2(i = 0 \sim C),$

$W = \sum_{i=1}^C W_i = 1000$

とすると、タイマ更新回数はC=1の場合(案1a)が、C=128の場合(案2)が最も少ない。

5.考察

以上の検討の結果をまとめると表1のようになる。(案1a)はCが小さい時、タイマ更新回数が少ないものの、システムコールによるオーバーヘッドが大きく、Cが大きくなるにつれてタイマ更新回数も(案2)を上回るようになる。(案1b)は、システムコールによるオーバーヘッドは小さいものの、Cが大きくなるにつれてタイマ更新回数が極端に増える。(案2)は必要なメモリ容量が多くなるが、システムコールによるオーバーヘッドが小さく、Cが大きくなった場合のタイマ更新回数の増加が最も少ない。このため案2が最も性能的に優れていると思われる。

そこで今回は性能を重視し、(案2)を採用した。

6.おわりに

LAN用通信アダプタに搭載し、n個のコネクション要素をサポートするLLCタイプ2ソフトウェアの実装方式について検討した。今後、最適なスタックサイズや、今回採用した方式の効果を検討していきたいと考えている。

表1 各方式案の比較

	案1a	案1b	案2
システムコールによるオーバーヘッド	大 (タイマのスタート、ストップの度に必要)	小 (タイマのスタート、ストップによるシステムコールは1度)	小 (タイマのスタート、ストップによるシステムコールは1度)
メモリ容量	小	小	大
タイマ更新回数(C=1)	128	2064	2064
タイマ更新回数(C=128)	16384	256064	10192