

シングルタスクOSにおけるマルチライン制御方式

1V-4

小林 幸夫 中谷 学 井川 健史 岡本 登

富士通関西通信システム株式会社

1. はじめに

近年通信システムの普及により、端末装置は必要最小限の情報を保持し、その他の情報はホストコンピュータに所有させ、情報が必要なときには、複数の端末装置が同時にその情報を取得する分散システムが構築されているが、大型ホストコンピュータ・オフィスコンピュータの導入は、現在もコスト的に高価なものとなっている。

そこで、低価格のパーソナルコンピュータをホストコンピュータとして利用する方法を考察し、その結果として、シングルタスクOSにおけるマルチライン制御方式という概念が得られた。

本論では、シングルタスクOSにおけるマルチライン制御方式を適用した場合のソフトウェアの構成及びその有効性について述べる。

2. 従来の技術と問題点

シングルタスクOSで動作する1つのアプリケーションでは、複数の回線制御が出来なかった。その理由として、1つの回線を制御している時に他の回線に制御権を渡す事ができず、またできたとしても、制御権の切り替わるタイミングによりデータが保証されなかった。

従って複数回線制御を目的とするシステム構築は、

- (1) OSにマルチタスクOSを採用する
- (2) 複数のパーソナルコンピュータを用意し各パーソナルコンピュータにて1回線ずつ処理をさせる

以上の2通りの方法が考えられるが、(1)はタイムシェアリング処理の為、通信途中に他のアプリケーションに制御が移ることによって、データ抜けやリトライ処理が発生する。また、回線制御数分のアプリケーションがメモリに常駐する為、メモリの空き領域も少なくなる。(2)はシステムを構築するにあたりコストがかかる。また、コンピュータ間でのデータインタフェースが複雑になる等それぞれの方式においても問題がある。

3. 特徴

上記問題点を解決するマルチライン制御方式の特徴を記述する。

3.1 タスク切替

今回の制御方式は、1つのアプリケーションのソフトウェア構造を細かい機能単位(関数群)に分類し、アプリケーションのメイン処理部にて、1関数処理の終了を契機としたタスク切替を行うものである。(図1参照)

Multi Line Control Method for Single Task Operating System

Yukio Kobayashi, Manabu Nakatani,

Takeshi Ikawa, Noboru Okamoto

FUJITSU KANSAI COMMUNICATION

SYSTEMS LIMITED

Twin 21 Mid Bldg. 2-1-61, Shiromi,

Chuo-ku, Osaka, Japan

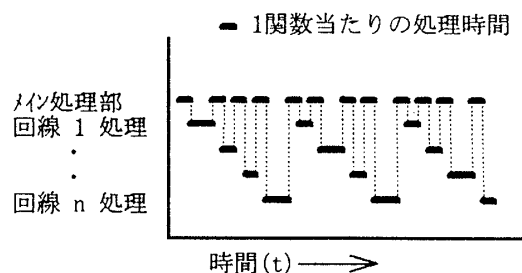


図1 各タスクの割当時間

3.2 オブジェクトコードの共有

回線数分のアプリケーションがメモリを専有するのではなく、回線数分のデータ領域と1回線分のオブジェクトコードにて、複数回線の制御を行うものである。

4. ソフトウェア構成

マルチライン制御方式を実現させる為のソフトウェア構成を図2に記し、簡単に処理概要を説明する。

ソフトウェア構成は、

- (1) 回線毎のタスク切替えを制御する メイン制御部
- (2) 各機能単位に用意された 回線制御関数群
- (3) 通信の開始から終了までの流れを関数番号にて表した シリアデータ
- (4) 各回線毎のデータを管理する 回線データの4グループに分けられる。

処理の流れとして、

- a. メイン制御部は、対象回線の回線データよりシリアファイルの何番目に格納される関数をコールするかをチェックし対象関数をコールする。

- b. コールされた関数は、回線の状態や処理結果などに応じて回線データを更新する。
- c. 上記a. b. が1回線あたりの処理であり、制御回線数分a. b. の処理を繰り返し、全ての処理が終了すれば、再び1回線目より繰り返す。

以上のようになっている。

5. 効果

この制御方法により、マルチタスクOS時に比べ、

- (1) OS部がシングルタスクOSの為、制御部の負荷（オーバーヘッド）が軽減される
- (2) タイムシェアリングによるタスク切替に比べ、1関数終了までの制御権が保証される事によりデータの抜けやリトライ処理が軽減される
- (3) オブジェクトコードの共有により、メモリ常駐量が削減される

の効果が得られる。

6. 課題

今回は、シングルタスクOSにおけるマルチライン制御方式を理論的に検証した。今後1関数あたりの処理時間と各種CPUとの組み合わせによる処理能力を評価し、最適な機能単位と制御回線数の関係を明らかにしていきたい。

