

カード端末装置における入出力管理方式

7N-8

藤本 和生、 伊藤 守
 松下電子部品(株) 電子部品研究所

1 はじめに

カードサービスの多様化に伴い、多種多様な高速の入出力装置を備えたカード端末が要求されている。しかし従来のドライバー一体型のOSでは、デバイスの追加や変更が困難であるという問題点があり、さらにこれらの端末には高速な多重処理が要求される。

そこで、デバイスの変更、追加が容易で、高速なI/O処理が可能な端末の入出力管理方式を検討したので、以下に報告する。

2 基本方針

(1)入出力要求は、デバイス毎にシステムコールを構成することにより、高速処理を実現する。

(2)従来の入出力ドライバを、直接ハードウェアの制御を行うデバイスドライバ部(以下ドライバと記す)と、デバイスの機能を拡張するデバイスハンドラ部(以下ハンドラと記す)に分割し、必要に応じてデバイス管理テーブル(以下テーブルと記す)に登録する機能を設けることにより、デバイスの変更、追加を容易にする。

3 入出力管理方式

ブロック転送処理を例にとり、今回検討した入出力管理方式の機能を説明する。図1に本方式の構成を示す。まずタスク側よりコマンド解析部に対して、入出力システムコールを発行し、コマンド解析部は使用デバイスに対応するドライバを獲得する。割り込み入力が発生すると、該当ドライバが処理を開始し、バイト単位のデータ受信処理を行う。一方ハンドラは、指定バイト数の入力処理を行い、処理が終了するとメールボックスを用いて、タスクに処理結果を通知する。

図2にデバイスの関する情報を登録するためのテーブルの構成を示す。デバイス情報は全てテーブルに登録され、実行時にはコマンド解析部が、該当デバイスの検索を行い、指定ハンドラとデバイスの獲得状態を知る。またデバイスの追加や変更は、テーブルの内容を書き換えることで実現できる。高速処理が必要とされるものは検索時間が短くなるように配置することが望ましい。

タスク部とシステムコール実行部は独立に動作可能であるので、システムコール発行後は、タスク側では自タスクの実行を続けることができる。また必要に応じて他のデバイスに対応するシステムコールを発行することもできる。さらにコマンド解析部は、デバイス

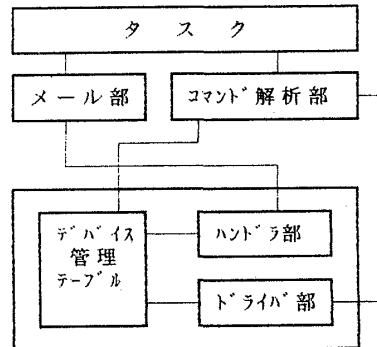


図1 入出力管理方式の構成

ドライバ名	ハンドラ名	獲得状態
0 1	A	1
0 1	B	0
0 2	C	1
0 3	D	0

図2 デバイス管理テーブルの内容

管理テーブルを利用し、デバイスの競合状態を回避している。

4 インタプリタの導入

しかしながら一般にマルチタスクのプログラム開発は困難であり、並行処理機能を有するインタプリタの導入が望まれる。インタプリタは、デバイスに対する指令を記述したテキスト文を主記憶上にダウンロードし、そのテキスト文を解釈して、該当する各タスクの起動、消去、優先順位等の設定等を行う。そのためユーザは並行処理を意識せずに、マルチタスクプログラムの開発が可能になる。

5 むすび

高速かつ変更、追加が可能な入出力管理方式と、ユーザに対して並行処理を簡便に実現するインタプリタの導入を検討した。現在本稿に基づいた端末を開発しており、性能面の評価を行う予定である。