

CP/Mを拡張したLAN用 マルチタスキングOSの開発

菰田元喜 池田宏明 山本博
(千葉大学 工学部)

1. まえがき

CP/Mはマイクロコンピュータ用オペレーティングシステムとして広く普及しており、CP/Mのもとで動作するソフトウェアも、エディタ、アセンブラ、コンパイラ等、良質のものが数多く市販され便利なのが多い。しかしながら、CP/Mは単一タスク単一ユーザ用として設計されているため、ファイルをプリンタに出力するだけでもシステムを独占してしまい、プログラム開発の道具として使用する場合効率が悪いところもある。現在では、外部記憶装置として、ハードディスクが低価格となっているので、CP/Mにハードディスクを使用しているユーザも多いであろう。このようなシステムでは、リソースの有効利用という点から、CP/Mシステムの外部から、このハードディスクをアクセスできると便利である。

筆者らは、LAN (Local Area Network) や GP-IB⁽¹⁾ (General Purpose Interface Bus) から、ハードディスクをアクセスしたり、プログラム開発や、ファイルのプリンタ出力が同時に行なえることを目的としたマルチタスク可能なオペレーティングシステムを開発中であるのでここに発表する。このオペレーティングシステムは、複数のCP/M 2.2^{(2),(3)}をリアルタイムモニタのもとで動作させることにより、マルチタスク化を実現しようとするもので、その実現方法や、問題点と構造、機能について述べる。

2. システム構成

本システムは、図1に示すハードウェア構成のもとで動作する。仕様は以下のようになっている。

- OS : CP/M 2.2 を改造したものの複数とリアルタイムモニタ
- CPU : Z80 A 3.2 MHz
- バス : IEEE Std. 796 (Multibus)
- メモリ : 64 kB × 5 , 共有領域 1 kB
- I/O : 5" ハードディスク 10 MB
- : 8" フロッピーディスク 1 MB × 2
- : RS-232C シリアルインターフェース 4ポート
- : GP-IB コントローラ, トーカ, リスナ
- : パラレルポート { プリンタ用
- | LANトランスポート用

- 割込み：INT0 リスタート
- 1 RS-232C 送信, 受信
 - 2 タイマ
 - 3 GP-IB
 - 4 LAN トランスポート
 - 5 A/Dコンバータ
 - 6 フロッピーディスク, ハードディスク
 - 7 コンソール入出力, プリンタ

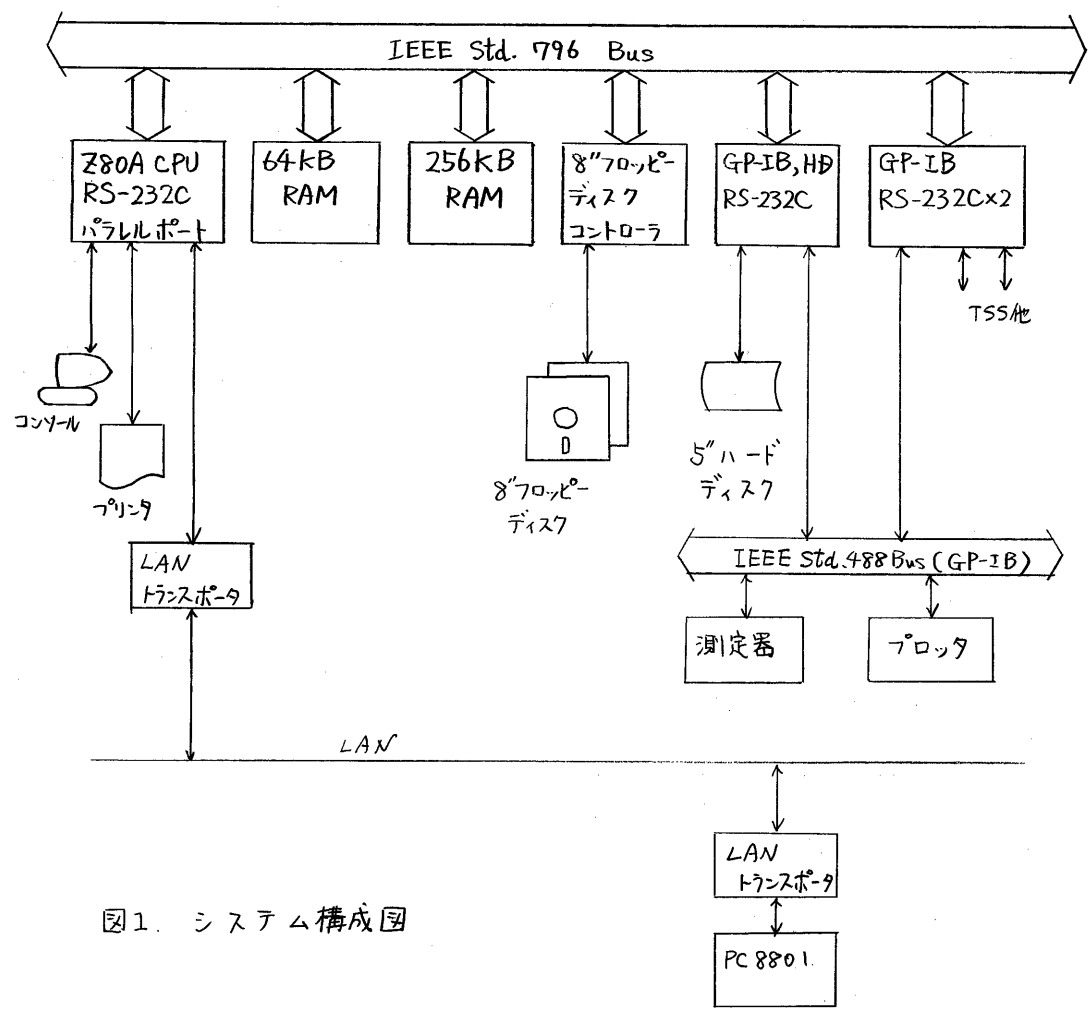
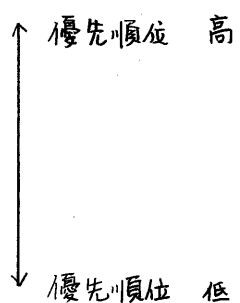


図1. システム構成図

3 リアルタイムモニタの仕様

本リアルタイムモニタは、モトローラ社のRMS68K⁽⁴⁾を参考にして作成したのでRMSZ80と呼ぶことにする。RMSZ80の特長を次に示す。

- (1) 最大タスク数 256
- (2) 優先順位数 256
- (3) 最大セマフォ数 256
- (4) 最大事象数 256
- (5) 最大入出力装置数 256
- (6) タスクのディスパッチングは優先順位にもとづいて行なわれる。
- (7) タスクの状態としては、実行可、実行、中断、待ち、事象待ち、休眠、および遅延状態がある。
- (8) バンク管理機能をもつ

RMSZ80マクロ命令の発行形式を次に示す。

```
Cレジスタ ← 命令番号  
DELレジスタ ← パラメタ または、パラメタブロックアドレス  
RST 08H
```

命令が終了するとAレジスタにエラーコードがセットされる。

3.1 タスク管理命令

```
GTASQ ASQ (Asynchronous Queue)をタスクに割り付ける  
DEASQ タスクのASQを削除する
```

3.2 タスク状態の制御命令

```
CRTCB TCB (Task Control Block)を作成する  
START タスクを実行可にする  
STOP タスクを休眠状態にする  
RESUME タスクを中断状態から実行可状態にする  
TERM タスクを終了する  
SUSPND タスクが中断状態に移る  
WAIT タスクが待ち状態に移る  
RELINQ タスクが実行可状態に移る  
DELAY タスクが遅延状態に移る  
WTEVNT タスクが事象待ち状態に入る  
SETPRI タスクの優先順位を変更する
```

3.3 タスク事象の制御

```
QEVENT 事象を送る  
RDEVENT 事象を読み出す
```

3.4 タスクの同期命令

```
CRSEM セマフォを作成する
```

DESEM	セマフォの使用権を放棄する
WTSEM	資源の占有を要求する
SGSEM	資源を解放する

3.5 割込み処理の管理命令

CISR	ハードウェア割込みと割込みルーチンを適合する
SINT	割込みを模擬する(デバッグ用)
INT	割込みルーチンを実行する
RTS	割込みルーチンから戻る

3.6 I/O命令

INITIATE	入出力の起動
ATTACH	チャンネルの結合
DETACH	チャンネルの切離し

4. CP/Mのマルチタスク化の方法と問題点

ここでは、CP/MのDOS(BDOS, BIOS)をマルチタスク化するための方法と問題点について述べる。

4.1 マルチタスク化の方法

DOSをマルチタスク化する方法として、次の3つの方法が考えられる。

(1)リエントラント構造にする

(2)資源制御を使用する

(3)各タスクごとにDOSを持つ

(1)は、大部分のDOSが採用している方法で、主記憶装置の容量を大幅に減らすことができ、もっとも効果的である。しかし、既存のBDOS, BIOSをリエントラント構造に改造するには、かなりの作業量が必要である。

(2)は、BDOSコール, BIOSコールの前に資源の占有要求を出し、戻り時に資源を解放する方法である。この方法によれば多重に呼び出されることはないが、BDOS, BIOS内の作業領域についても資源制御を行なう必要があり、(1)と同様、かなりBDOS, BIOSの変更が必要となる。

(3)は、(1)とは逆に主記憶装置の使用量が増加する欠点があるがBDOS, BIOSが多重に呼び出されることがなく、また、作業領域を各タスクごとに管理するので、各タスク間で共有するデータの資源制御をする必要がある以外にBDOS, BIOSの変更が少なくすむという利点がある。

本研究では、(i)BDOSの変更はできるだけ少なくしたい (ii)IEEE Std. 796バスを使用しており、主記憶装置の大容量化が容易である (iii)ディスク装置を使用するタスクの数がそれほど多く必要ないなどの理由により(3)の各タスクごとにDOSを持たせる方法により、マルチタスク化を実現することにする。

4.2 マルチタスク化の実現方法

上で述べたように、一つのタスクに対し一つのDOSを持たせることによってマルチタスク化を実現する。すなわち、ユーザプログラムとDOSが一体となって一つのタスクを形成し、そのタスクがRMS 80の管理の下で動作することになる。各バンクごとにCCP, BDOS, BIOSを持っておりTPA内のプログラムやCCPは、BDOSコール, BIOSコールによってコントロールやディスクのアクセスを行なう。BDOSは、BIOSエントリをコールし、また、BIOSの入出ルーチンはRMS 80に対してマクロ命令を発行するだけである。つまり、ハードウェアに依存するE/Oに対するアクセスはRMS 80内のE/Oハンドラが行なうことになる。

このように、一つのシステム内に複数のCP/MをもったOSをMCP/M (Multiple CP/M)と呼ぶことにする。

4.3 共有資源の相互排斥

ここでいう共有資源とは、コンソール、プリンタ、ディスク、GP-IB、LANトランスポート等のE/Oをさす。これらの共有資源は、各タスク間で排他的に使用しなければならぬ。その手段としてRMS 80に用意した同期命令W TSEM, S GSEMを用いる。BIOSは、E/Oをアクセスする時にW TSEMマクロにより資源を占有した後I M I T E A T Eマクロにより、入出力を起動し、W T E V N Tマクロにより入出力が完了するのを待つ。入出力が完了するとR D E V N Tマクロにより、データやステータスを受けとった後S G S E Mマクロによって資源を解放する。

ディスクに関しては、さらに相互排斥が必要となる。タスクがディスクに対し書き込み動作をする際に、ディスクのどの物理アドレスを使用するかを決めるのはBDOSである。したがってBDOSレベルでの相互排斥が必要となる。BDOSは、ディスクに対し新しい領域を割当てるとき、BIOS内のディスクアロケーションマップを参照し更新する。つまり、ディスクアロケーションマップをBDOS間で共有し、排他的に更新しなければならぬ。そのために①ディスクアロケーションマップを共有バンク領域(1KB)に配置し、さらに②BDOS内のディスクアロケーションマップ更新ルーチンで、同期命令を用いる、よう変更する必要がある。その他のBDOS内の変更点は、DPH (Disk Parameter Header)内のBDOSで使用されるスクラッチパッド領域とCSVに位置づけられた領域に対する相互排斥を行なうためのものである。

5. MCP/Mの構造

5.1 Xメモリマップ

MCP/Mシステムのメモリマップを図2に示す。0FC00H~0FFFFHは、各バンク間に共通な領域である。

システム共有データは、ディスクアロケーションマップ等の各タスク間で共有しなければならないデータである。

割込みベクタは、割込みコントローラ8259Aが発生するINT0~7までの割込みルーチンのエントリポイントである。

割込みハンドラは、各割込みに共通のエントリであり RMSZ80 に対し INT マクロを発行する。

X RMSZ80 は、RMSZ80 の一部でありバンク切換えを伴うルーチンを保持している。

ベースページ(0~0FFH)は、8H 番地に、X RMSZ80 内のマクロ命令呼び出しエントリへのジャンプベクタを追加した以外は、CP/M2.2 と同じである。

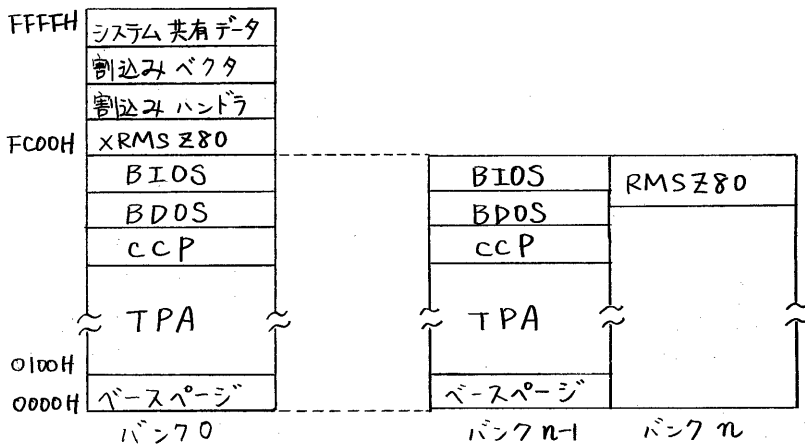


図2. MCP/M のメモリマップ

5.2 BIOS の機能

CP/M2.2 の BIOS とほぼ同じである。WBOOT エントリは、システムの再ロードを行なわない。

5.3 BDOS の機能

CP/M2.2 の BDOS と同じである。

5.4 CCP の機能

MCP/M では、ウォームブート時にシステムの再ロードを行なわないため、TPA 領域は CCP の下までしか使用できない。TPA 領域をできるだけ大きくするためにビルトインコマンドは SAVE だけとした。CP/M2.2 では、6,7 番地の内容が BDOS のアドレスであり、ユーザ領域の上限でもある。MCP/M では、CCP を破壊しないように 6,7 番地の内容を CCP-3 とし、CCP-3 番地には、BDOS へのジャンプ命令を置く。プロンプトとしてユーザ番号も表示する。

6. MCP/M の機能と特徴

- ・ファイル管理やメモリ配置が CP/M と同じため、従来のソフトウェアがそ

のまま使える。

- タスク間で、ファイルの共用ができるのは読み出し時だけである。書き込み時にファイルの保全本をはかるために、ファイル書き込みタスクは、タスク間で異なるユーザ番号を使用する必要がある。
- 一つのバンク内で使用できるタスクの数は一つに限られる。ただし、BDOSコールを行わずに直接RMS 280に対しマクロ命令を発行する場合はその限りではない。
- ウォームブートでシステムを再ロードしない。
- ビルトインコマンドはSAVEだけである。
- プロンプトとして $\emptyset A \rangle$ のようにユーザ番号とドライブ名を表示する。
- トランジェントコマンドがログインされているドライブ、ユーザにない場合は、Aドライブ、ユーザ \emptyset を捜すようにし、ユーザごとにファイルを重複してもつ必要がない。
- ^Cにより実行中のユーザプログラムが終了する。
- ^Dにより実行可能で次に優先順位の高いタスクにコンソールを接続する。
- LANの上位プロトコルをサポートすることにより、外部からの要求によりディスクのアクセスが可能になる。

あとがき

複数のCP/M 2.2をリアルタイムモニタの管理下で動作させることによるマルチタスク化の実現方法について述べた。このOSは現在開発中であり、今のところRMS 280の動作確認が終了したところである。またCP/M 2.2のCCP、BDOSの解析もほぼ完了しているのであとはCCP、BDOSの改造を残すのみである。なお、RMS 280はマクロアセンブラM80で記述されており約1200ステップ、約2900バイトの大きさである。

- 1) IEC Standard : 『An interface system for programmable measuring instruments (byte serial, bit parallel) 』, Publication 625-1 (1979) ; Publication 625-2 (1980)
- 2) Digital Research社 : 『CP/M 2 Interface Guide 』, (1979)
- 3) Digital Research社 : 『CP/M 2 Alternation Guide 』, (1979)
- 4) 日本モトローラ(株) : 『M68000 リアルタイムマルチタスキングソフトウェアユーザズ・ガイド 』, CQ出版, (1982)
- 5) David K. Hsiao : 『Systems Programming 』, (1975)
斎藤信男訳 『システムプログラミング 』, 日本コンピュータ協会, (1980)
- 6) 江村潤郎 : 『オペレーティングシステムへの構造的アプローチ(下巻) 』, 日本コンピュータ協会, (1977)