

ミニコンピュータのサポートシステム

Support System for Minicomputer

渡辺 勝*

Masaru WATANABE

ミニコンピュータのプログラム開発を大型計算機を用いて容易に行えるようにしたサポートシステムについて解説し、またこのシステムを複合計算機に応用した例を述べたものである。

1. はじめに

ミニコンピュータのプログラムの作成を大型計算機で行えるようにしたものがサポートシステムである。ミニコンピュータの本体の価格はかなり安くなってきたものの、カードリーダーやラインプリンタなどの周辺装置はそれ程でないため、ミニコンにはふつうそのような高級な周辺機器を備えず、紙テープ装置やタイプライタで間に合わせていることが多い。

最近ではミニコンのソフトウエアも整備されてきて、アセンブラやフォートランはもとより、大型機に近いオペレーティングシステムを備えたものも見られるようになった。しかし上のような事情で紙テープ装置を用いたシステムが多く、機能としてはよくても、実行の能率に問題がある。たとえばプログラムの修正を紙テープに対して行うのは、たとえエディタなどのソフトウエアの援助があったにしても厄介であり、カードの場合の能率のよさとは比較にならない。それにも増して不便なのは、リストの出力をタイプライタを用いて行うことであり、時間がかかりすぎてプログラム開発の効率をおとすことおびただしい。

これにくらべると大型計算機ではカードリーダーやラインプリンタは標準機器として備えられているので、これらを用いてミニコンピュータのプログラム作成を能率的にやらせようという考えが生れてきた。ただしミニコンと大型機では、設計上の考慮から機械語の構成が異なるため、アセンブラであればミニコンの機械語が出力されるような特別なアセンブラが必要である。これがクロスアセンブラという名で、かなり以前から使用されているものである。^{(2), (3)}

最近ではアセンブルだけでなく、アセンブルされたプログラムの実行まで大型機でやってしまうことが出来るようになり、これにプログラム作成の際、補助的に使用するユーティリティ機能を付加したサポートシステムが実現している。この解説で紹介するのは、富士通社のミニコンFACOM U200のために開発されたサポートシステムで、上位計算機としてはFACOM 230—45/55を用いている。このシステムはオペレーティ

ングシステムOS IIのもとで使用できるものであるが、これより先に230—25/35のBOSで使えるサポートシステムも完成している。

2. MFASP (クロスアセンブラ)

大型計算機によるアセンブルの方法から説明しよう。MFASP (Mini computer FASP) と称するクロスアセンブラが用意されていて、ミニコンU200用のアセンブラ言語で書かれた原プログラムをカードにパンチして入力し、アセンブルされた相対形式オブジェクトを磁気ディスク (大記憶) 上のRLIBファイルに作成する。(図1)

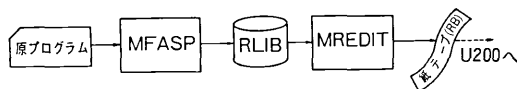


図1 MFASPとMREDIT

デバッグの段階ではMFASPだけでよく、エラーメッセージがプログラムリストとともにラインプリンタに出力されるから、これを見て必要なカードの修正を行う。デバッグが終ればオブジェクトプログラムをミニコンに移すために、紙テープに出力する。これを行うのがMREDITであり、MFASPで作成されたRLIBを入力として、これを紙テープ上に変換して出力する。この際テープを認識するための花文字などもつけ加えてパンチする。

サポートシステムにおけるアセンブルの手順は以上の通りであるが、実際に使用する際のジョブ制御文について述べておこう。まずMFASPで使用する入出力機器は図2のようにになっている。

これに対応して使用する制御文は図3のようになる。この例では引きついでMREDITも行う場合の制御文も示してある。MREDITの入力には、先行するMFASPのステップの出力RLIBファイルをSW(スイッチ)文により、切りかえて使用していることに注意されたい。

MREDITでは副制御文が必要で、これはCOIN

*東京大学生産技術研究所 第3部

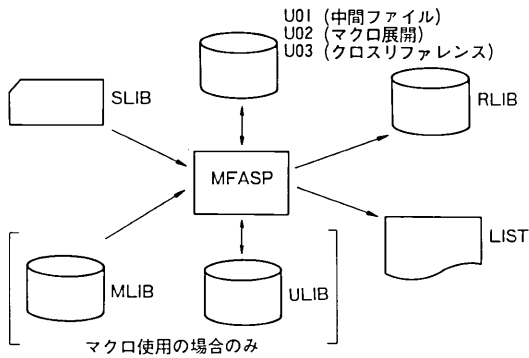


図2 MFASPの使用する入出力機器

(Control Input) 入力として与える。大記憶にあるRLIBファイルを読み出して紙テープに出力する場合の副制御文の一例を示す(図4)

ここでU01, U02は前の主制御文で定義したファイルである。/PUNCHは紙テープへのせん孔を示す。それにつづく*印は複写すべき先頭のプログラムを、つぎの*印は最後のプログラムまで続けて複写することを示す。プログラムが一個の時は*一つだけでよく、また*の代りプログラム名を書いてもよい。副制御文の形式は55機の標準と違って、/の後に空白をおかず制御文字がきていることに注意を要するが、これはBOSの形式に準じたためである。

以上の方法で作成したRBの紙テープはそのままU

```

FACOM 230-45/55 OS2      JOB CONTROL STATEMENTS LIST (SYSTEM)      V-05 L-03 DAT
-----
1      ¥ JCODE P57
2      ¥          CODE... P57...
3      ¥_TSJOB_WATANA30,30100,WATANABEKEN*,A
4      ¥          JOB_WAJANA30,ACCT=130100,WATANABEKEN*,A*,LIST=(A,
5      ¥          PRTY=2,PSW=1
6      ¥          MKEY=
7      ¥_EX_MFASP,FILE=SUPPORT, VOL=000000
8      ¥_PARA_NOMACRO,NOSEQ
9      ¥_FD_U01=DA, VOL=WORK, CYL=(2,1)
10     ¥_FD_U03=DA, VOL=WORK, CYL=(2,1)
11     ¥_FD_LIST=DA, VOL=WORK, SOUT=A, CYL=1
12     ¥_FD_RLIB=DA, VOL=WORK, CYL=1, DISP=CONT
13     ¥_FD_SLIB=*
14     ¥_EX_MREDIT, FILE=SUPPORT, VOL=000000
15     ¥_SW_U01=RLIB
16     ¥_FD_U02=PIP
17     ¥_FD_LIST=DA, VOL=WORK, SOUT=A, CYL=1
18     ¥_FD_COIN=*
19     ¥_JEND
    
```

図3 MFASPの制御文の例

```

FACOM 230-45/55      SUPPORT SYSTEM      MREDIT      V-01
-----
/PUNCH...*.*
PUNCH FILE(REVASM )
/END
    
```

図4 MREDITの副制御文

200で処理できる。すなわちRLOADによりディスクに格納したのち、PRBLによりEBプログラムに変換して一時EB領域に登録する。つづいてプログラム番号7C00として実行することができる。

3. MCS (シミュレーションによる実行)

はじめに述べたようにプログラムの実行まで大型計算機でやってしまうことは、サポートシステムの魅力であろう。ただし実行といっても、大型計算機の中にシミュレータを備えていて、ミニコンピュータの一つ一つの命令を解釈して大型計算機の命令群に変換して実行する(シミュレートする)わけである。したがって一つの命令が数十ステップを要し、U200の本来の処理時間の数十倍かかることになる。しかしこれはCPU処理の命令の場合の話である。入出力命令については、これを55機のOS IIにおける入出力装置の使用手順に変換し、FCPまたはEXCP(入出力制御プログラム)を経由してその入出力装置の使用を実行する。一般に

55機の入出力装置はU200のそれよりも性能がよく入出力速度も早いから、入出力命令の実行速度はシミュレーションによっても大差はなく、時には速いこともある。したがってプログラム全体としてのシミュレーションによる速度の低下は、一般命令と入出力命令の構成の比率によって変わってくる。

図5はMCSの実行に必要な機器構成を示す。この他
3-1 MCSの機器構成と入出力

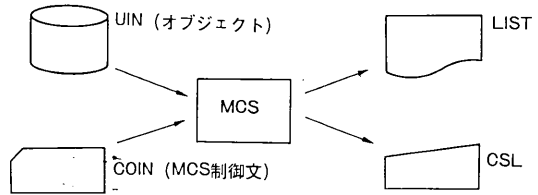


図5 MCSの必要とする基本入出力機器

```

FACOM 230-45/55 OS2      JOB CONTROL STATEMENTS LIST (SYSTEM)      V-05 L
-----
1      * JCODE=P57
2      *      CODE      P57
3      * TSJOB WATANA30,30100,WATANABEKEN*,A
4      *      JOB      WATANA30,ACCT=130100,WATANABEKEN*,A!,L
5      *      PRTY=2,PSW=#
6      *      MKEY=#
7      * EX MFASP,FILE=SUPPORT,VOL=000000
8      * PARA_NOMACRO,NOSEQ
9      * FD U01=DA,VOL=WORK,CYL=(2,1)
10     * FD U03=DA,VOL=WORK,CYL=(2,1)
11     * FD LIST=DA,VOL=WORK,SOUT=A,CYL=1
12     * FD RLJB=DA,VOL=WORK,CYL=1,DISP=CONT
13     * FD SLIB=#
14     * EX MREDIT,FILE=SUPPORT,VOL=000000
15     * SW U01=RLIB
16     * FD U02=P1P
17     * FD LIST=DA,VOL=WORK,SOUT=A,CYL=1
18     * FD COIN=#
19     * JEND
    
```

図6 MCSの制御文の例

にシミュレートすべきU200のプログラムが使用する入出力機器に対応する55機の入出力装置を用意しなければならない。原則としては、MCSの使用する入出力装置と独立に用意する必要があるが、下記のものについては兼用することができる。

(1) U200のタイプライタはコンソライブライタ (CSL) と兼用

(2) U200のラインプリンタはLISTに割付けた装置と兼用

この他のU200の入出力装置はU01~U16という論理機番をもつ55機の入出力装置と対応づけて、FD文により事前に定義しておくことが必要である。たとえばU200のディスク装置は55機の大記憶でシミュレートすることになるが、その際大記憶の一つのファイルをU200のディスク一台と見なして、つぎのFD文で対応づける。

¥ FD U01=DA, VOL=000002, FILE=U200DK, FCYL=23

詳細はサポートシステムのマニュアルを参照されたい。

つぎに基本構成の機器だけで、MCSを実行する場合のジョブ制御文を図6に示しておく。シミュレーションに先立ってMFASPによってオブジェクトRLIBを作成しておき、MCSを実行する際の入力ファイルUINは、これを前のようにSW文で切かえて使用する。

シミュレーションで必要とする55機の主記憶の大き

さは、MCSのEX文のSIZEパラメタで指定する。図7に示されている必要記憶容量のうち、U200の主記憶領域の部分はシミュレートすべきU200実装記憶容量によって決る。たとえば24KBを備えている場合の合計必要量は32+4+24=60KBである。これに少し余裕を持たせ (MCSの作業域の不足の際の予備)、前の制御文例では66KB (33ページ) にとっている。

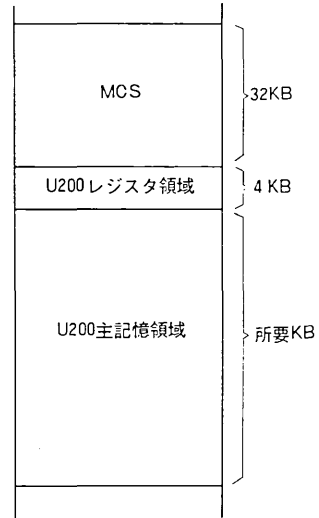


図7 シミュレーションに必要な55機の主記憶の大きさ

```

FACOM 230-45/55                                OS2/U-200 SIMULATOR                                (V-01)
-----
*OS2/U-200 SIMULATE START SIZE=(28 KB)
/READY FOR REPLY (MCS)
/LOAD REVASM,3000
  ADD=3000  SCOM=1080  COM=1088
/ASSIGN 0210/CSL,4000,KB
/ASSIGN 0218/CSL,4000,TW
/ASSIGN 0238/LIST,4000,LP
/CTOP 3034
/RUN 300C
  3228 E105 LS 05(IC)
  322A A101 LS 01(X5)
  322C 0102 LS =02
  322E 060F ANDS =0F
  3230 050F CS =0F
  3232 E201 STS 01(IC)
  3234 001E =1E
  3236 4040 40(X2)
*CTOP
/END
*OS2/U-200 SIMULATE END
    
```

図8 MCSの副制御文の例

3-2 MCSの制御文

前節のはじめに述べた入出力装置の対応づけはMCSの制御文で行う。この他にMCSで実行すべきプログラムを主記憶にロードすること、指定した番地から実行をはじめ、指定した条件で終了することなどの指定もMCS制御文で行う。それらの制御文はMCSの主制御文のCOIN入力として与えるのであって、その一例を図8に示してある。ただしここに掲げたリストは実行時のプリンタ出力の写しであって、先頭に/のついた行がMCS制御文をあらわす。その他は実行にともなうメッセージや実行時出力である。つぎに各制御文の意味を述べておく。

/ASSIGNが入出力装置の対応づけを示す制御文であり、第1パラメタはxxxx/yyyyの形をしていて、xxxxがU200の入出力装置の物理機番を、yyyyはそれに対応する55機の入出力装置の論理機番を指定する。後者にはCSL, LIST, U01~U16等を使用する。第2パラメタはU200の入出力装置の分類によって決まるコードであり、たとえば先頭桁の4は、その装置の接続される割込レベルをあらわす。第3パラメタはU200の入出力装置の機種を示す。上の例では、U200のタイプライタを55のコンソルタイプライタに、同じくラインプリンタをラインプリンタに対応させた場合である。

/LOADにより実行すべきプログラムを指定した番地に格納する。

RUNによって指定番地から実行を開始し

STOPの指定する番地にある命令を実行する直前に終了する。

/ENDはMCS制御文の終了を示す。

以上が制御文の働きである。

本例では/RUNにつづいて実行の結果がプリントされている。この例はU200の機械語をアセンブラ言語に変換する、いわゆる逆アセンブラプログラムを実行したものである。この例のように、たとえU200にラインプリンタが付いていなくても、それがあつたようにプログラムしておく、結果はシミュレートされたラインプリンタに出力される。もしプログラムをタイプライタに出力するようにしてあれば、実行の結果は55機のコンソルタイプライタに出る。このような使い方ができるのも、MCSの便利さであろう。

以上はMCSの制御文の基本的な使用方法である。MCSの制御文にはこの他にも種々の機能をもったものが用意されており、デバッグやトレースの実施、あるいは経過時間のカウントなども出来るので活用すれば便利であろう。詳細はサポートシステムのマニュアル⁽¹⁾を参照されたい。

4. 複合計算機におけるサポートシステムの応用

U200ミニコンピュータと230-55機のような大型計

算機とをチャンネル結合方式で接続した階層型の複合計算機システムの研究を当研究室で行っている。前章に述べたサポートシステムは、元来はミニコンピュータと大型計算機が独立している場合を対象にしている。しかしそれらが物理的に結合された複合計算機の場合には、このサポートシステムをより有効に使用できる可能性が生じてくる。すなわち2章で述べたクロスアセンブラMFASPの出力であるRBファイルをチャンネル結合の線を経由して、直接ミニコンピュータ側に送り出すことにより、途中でたん紙テープに出力する手間が省かれる。

このような応用を旨としたシステムの開発には、55機側に送出プログラム、ミニコン側に受取プログラムがなくてはならない。その第一段際として前者をとりあげ、MFASPによって作成されたRBファイルをユーザプログラムによって読み出す方法を考えてみよう。以下このプログラムをRBPRTと名付けることにする。

4-1 MFASPのRBファイル

MFASPによってアセンブルされて出るオブジェクトは、大記憶上のRLIBファイルに格納されているが、このファイルの構成は分割型の順編成であり、そのレコード長は80バイト、ブロック長は480バイトと規定されている。このため55機でファイルの処理に便利に使われているシステムプログラムLIBEでは扱えない。LIBEではブロック長が1040または1080の標準ファイルしか扱っていないからである。(この点はMFASPのRBファイルを変更するか、逆にLIBEの機能をより一般的にするか、いずれかの改良を行うことが望ましい)したがってMFASPのRBファイルを読み出すには、ユーザ側でアセンブラプログラムを作成しなければならない。この目的に用意されているFCP (File Control Program) のマクロ命令を用いれば比較的容易にプログラムできる。

4-2 RBファイルの印刷プログラム (RBPRT)

前節の考え方で作成した読出しプログラムRBPRTのフローチャートを図9に示す。

FCPマクロ命令を使用する際には、処理すべきファイルに関する詳細な情報が必要であり、これをまとめてファイル制御ブロック (FCB) に記述しておかなければならない。このプログラムでは、入力はMFASPで出力されたRBファイルRLIBであり、これを読出して印刷する出力ファイルLISTが必要である。それらに対応するFCB名をFCBPD, FCBOUTと名付け、各々のFCB文、ならびにプログラムの使用する作業域を下記のように定義する。

```
FCBPD  FCBPO ACNAM1=RLIB, EOFEX=
        MEOF, BUFAD1=AREA,
        BUFS=1002
FCBOUT FCBPS ACNAM1=LIST, RCDFM=
```


FC, RCDS=144, EOFEX=
LEOF, BLKS=1440, BUFS
=1440, BUFAD1=LPBUF

SBOUND 3

AREA DA 501 SUPPORT·RLIB=480B,
DIRECTORY=1000B

LBUF DA 720 1440B

LPARA DA 72 144B

FINDM DC S'0' RETURN CODE
DC C'REVASM ' MEMBER NAME
DC S'0' TTR FOR MEMBER

プログラムのフローに戻って説明しよう。RBファイルは分割型順編成であるため、そのメンバ名によって

読み出すべきファイルのある場所(先頭アドレスTTR)を求めなければならない。この働きするのがFINDマクロである。つづいてREADマクロによりブロック単位で読み出す。READマクロはいわゆる“おいてきぼり”方式であり、READコマンドを出したまま完了を待たず使用者プログラムに制御を渡すので、つづいてCHECKマクロによりその完了を確認する。

そのあとはバッファAREAに読み出された1ブロックのデータをラインプリンタに整頓して印刷するためのプログラムである。前述のようにRBファイルのブロック長は480バイトであるから、これを2バイト1語にまとめて、1ブロックを20語×12行に印字することができる。(図10) 1ブロックの印刷が終わったら、つぎの

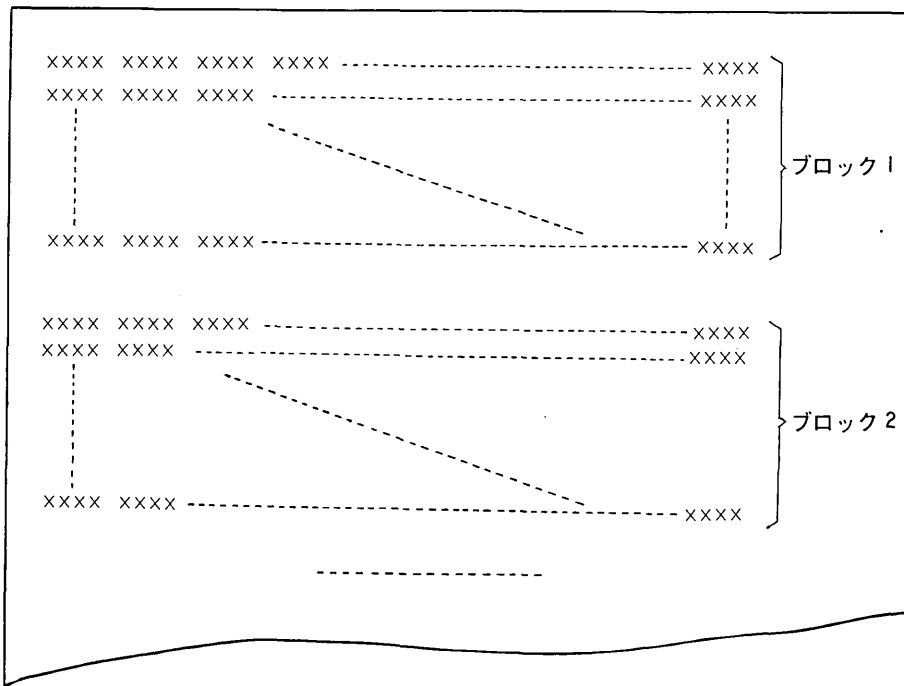


図10 RBPRTの印刷形式 (1ブロック=20語×12行)

ブロックの読出しのためREADコマンドに戻る。これを反復してファイルの読出しが終了するとEOF(エンドオブファイル)を検出し、各ファイルをクローズしてプログラムを終了する。なおTRUNCマクロはPUTマクロでバッファに書き出した残りのブロックを出力するために必要となる。

このようにして印刷されたRBファイルの内容の詳細はU200FASP文法篇 § 7.3「RBの構造」の項に記述されているので参照して頂きたい。

実際に複合計算機システムに応用する場合には、プリントの部分に代えて、チャンネル結合装置(CCA)に対する転送コマンドをEXCPで作成する必要があるが、

これについては別の機会に譲りたい。

5. おわりに

ミニコンピュータのプログラム開発を大型計算機の設備を用いて、より効果的に行えるようにしたサポートシステムと、これを複合計算機に応用する方法について述べた。読者の参考となれば幸せである。このような便利なシステムも、大型計算機のターンアラウンドが早くて円滑な運営が行われていなければ、宝の持ちぐされになってしまいかねない。大型計算機の運営技術の向上を期待して結びとしたい。

御討論いただいた「多重計算機連携システムの研究」グループ（東大生産技術研究所内，共同研究班）に感謝する。
(1975年9月30日受理)

参 考 文 献

- (1) "FACOM U-200 HICS SUPPORT-OS II/VS" 1974,

富士通社

(2) 浜田 喬：“FORTRANで書いたNovaミニコンのクロスアセンブラ”，未発表

(3) 安達 治美他：“FACOM270-30によるFACOM270-10およびU-200のクロスアセンブラ”，未発表

