



Title	ACOSシステム (5) : S700のソフトウェア その 2 (最終回)
Author(s)	
Citation	大阪大学大型計算機センターニュース. 1978, 28, p. 63-83
Version Type	VoR
URL	https://hdl.handle.net/11094/65372
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

<https://ir.library.osaka-u.ac.jp/>

The University of Osaka

ACOS システム (5)

— S 700 のソフトウェア : その 2 —

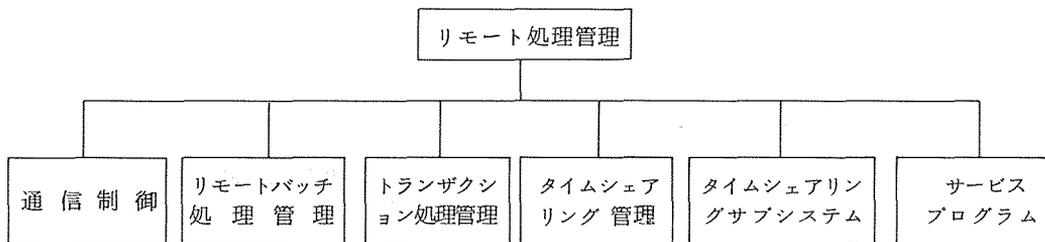
(最終回)

4.4 リモート処理

4.4.1 概要

リモート処理管理は、通信処理装置を介して入出力するデータを扱い、リモート処理形態を効果的に管理する機能です。

リモート処理管理は、次の 6 つの機能から構成されています。



(1) 特長

- ① 端末装置のユーザに多様なコンピュータシステム利用形態を提供し、豊富な機能を多様な角度から利用させることができます。
- ② リモート処理を通信処理装置と中央装置との間で機能を分担し、システムの処理能力、信頼性を高めることによって、各処理形態の同時処理を効率的に行っています。
- ③ 要求に応じたシステムを効率よく作成できます。

(2) 処理形態

S-700 では、一つ以上の FNP を接続することによって、リモート処理が可能になります。ACOS-6 のリモート処理管理は複数の FNP を制御し、リモートバッチ処理、トランザクション処理、タイムシェアリング、会話型リモートバッチ処理などの処理形態を可能にしています。さらに、メッセージ交換では中央装置上のプログラムが一つの端末と同等の意味をもって制御され、これとのデータの授受を制御できるようにしています。リモート処理管理は FNP と中央装置とのデータ授受を制御し、システム管理や各種処理形態とのインタフェースを制御し、ダイレクトアクセスインタフェース (DACI) とメッセージ交換インタフェースを可能にする通信制御機能と、その制御を利用して具体的なリモート処理を実現するリモートバッチ処理管理、トランザクション処理管理、タイムシェアリングサブシステムから成ってい

ます。

通信制御機能によって

- リモートバッチ処理
- ダイレクトプログラムアクセス

が可能になります。

トランザクション処理管理によって、

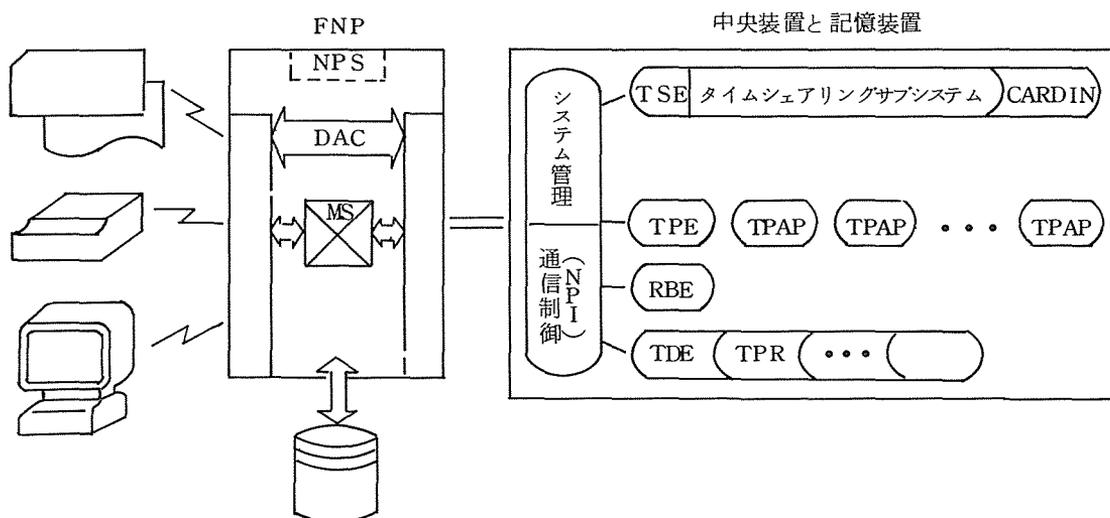
- トランザクション処理 (T D S)
- トランザクション処理 (T P S)

タイムシェアリング管理、およびタイムシェアリングサブシステムによって、

- タイムシェアリング
- 会話型リモートバッチ処理

が可能になります。

リモート処理と処理装置との関係を次図に示します。



(3) リモート処理の流れ

リモート処理の流れは次の通りです。

1) リモートバッチ

- ① 端末からの入力データをリモートバッチ処理管理プログラム (R B E) がテンポラリファイルに集める。
- ② システムスケジューラを通して入力データを S Y S I N に渡す。
- ③ S Y S I N から S Y S O U T までローカルバッチジョブと同様に処理される。
- ④ R B E が S Y S O U T から出力データを受け取り端末に出力する。

2) T S S

- ① 端末よりコネクト要求を出しサブシステムを選択する。
- ② T S Eによりサブシステムをロードする。
- ③ サブシステムを実行する。
- ④ ディスコネクトする。

3) 会話型リモートバッチ

- ① 端末よりコネクト要求を出し、CARD INサブシステムを選択する。
- ② T S EによりCARD INサブシステムをロードする。
- ③ ジョブをユーザのパーマメントファイルからテンポラリファイルに投入する。
- ④ システムスケジューラを通して入力データをSYS INに渡す。
- ⑤ SYS INからSYS OUTまでローカルバッチジョブと同様に処理される。
- ⑥ 出力データをJOUT、SCANサブシステムにより端末に出力する。
- ⑦ ディスコネクトする。

(4) ユーザ・ジョブの利用形態

ユーザが、センターを利用する時のジョブ形態として次のものがあります。

- T S Sジョブ
- ローカル・バッチジョブ
- 会話型リモート・バッチジョブ
- リモート・バッチジョブ

4.4.2 T S S

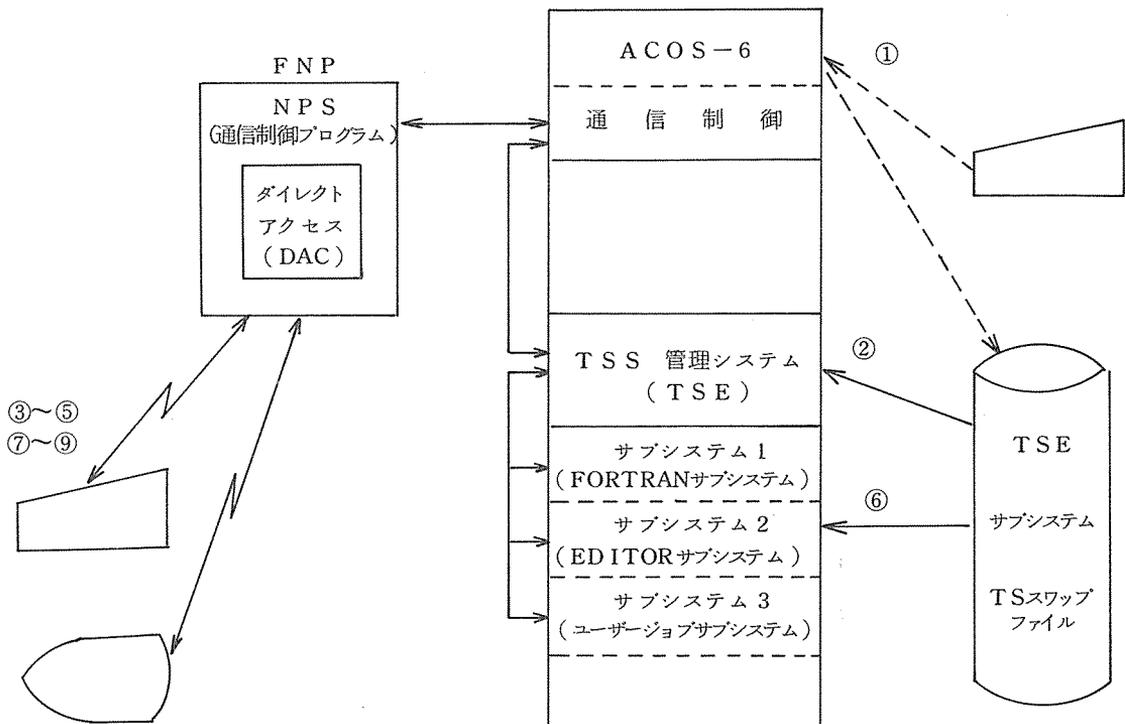
(1) 特 長

ACOS - 6 T S Sは次の特長を備えています。

- ① ターンアラウンドタイムの短いT S S利用が実現できます。
- ② バッチシステムと互換性の高いT S S利用ができます。
- ③ 多様なニーズに合うT S S利用ができます。
- ④ 使いやすいエディタが利用できます。
- ⑤ 各種端末が利用可能です。

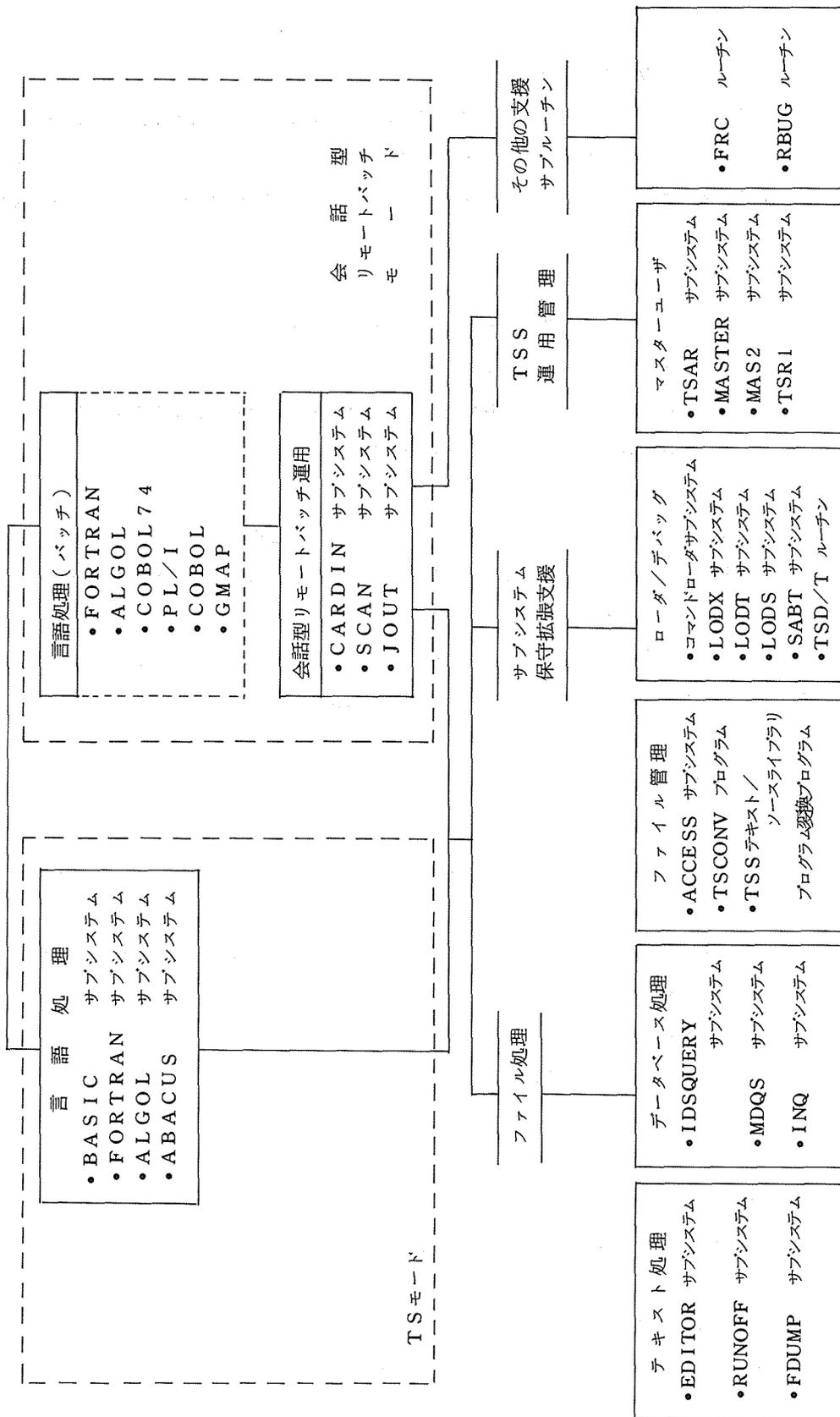
(2) タイムシェアリングの流れ

T S S管理システムが各サブシステムをT S S空間にスワップインしたり、スワップアウトして効率よく処理を行ないます。次図にその様子を示します。



- ① オペレータによる起動
- ② TSEのロード
- ③ 端末よりのコネクト要求
- ④ ユーザ識別名, パスワードの指定
- ⑤ サブシステムの指定
- ⑥ TSEによるサブシステムのロード
- ⑦ 交信(処理)
- ⑧ ディスコネクト
- ⑨ 料金計算

(3) TSSソフトウェア構成
各種処理形態よりTSSソフトウェアを利用できます。



(4) タイムシェアリングシステムコマンド

タイムシェアリングコマンドには、次のようなものがあり、機能の概要から以下に記述するように分類することができます。また、処理能力についてみても、単にサブシステムの実行を制御するだけのものから、サブシステムと同等レベルの機能を有するものまで、多種多様なものがあります。

1) 実行制御コマンド

これはタイムシェアリングの利用のため種々のセッションを開始したり、終了させたり、サブシステムの実行制御や終了をさせたりするためのコマンドです。

- BYE, NEWUSER, DONE, SYSTEM, RUN, HOLD, SEND, CRUN, DRUN, CPOS, CMOD

2) 入力制御コマンド

これはサブシステムに対するテキストデータの入力装置などの制御を行うコマンドです。

- TAPE, LUCID, AUTOMATIC, READ, PTON, PTOFF

3) 会話型リモートバッチ制御コマンド

これは会話型リモートバッチ処理サブシステムの機能を支援するためのコマンドです。

- JSTS, JABT, BPRINT, BPUNCH

4) ファイル制御コマンド

これはファイル処理サブシステムの機能を支援するためのコマンドであり、次のようなものがあります。

- ① パーマネントファイルをタイムシェアリング作業用ファイルへ写したり、また、その逆の処理を行ったりするものです。

- NEW, OLD, SAVE, RESAVE, ROLLBACK, RECOVER, PERM, LIB

- ② ファイルの内容を表示するものです。

- LIST, PRINT, LEADER, LENGTH, COUT

- ③ ファイルの内容を修正するものです。

- RESEQUENCE, DELETE, AUTOMATIC

- ④ ファイルの削除を行うものです。

- PURGE, RELEASE, ERASE

- ⑤ ファイル/カタログの登録状況を表示するものです。

- CATALOG

- ⑥ ファイルのアクセス(オープン)/ディアクセス(クローズ)を行うものです。

• GET, REMOVE

5) コード変換処理コマンド

これは、JIS, BCDファイル間のコード変換を行うコマンドです。

• JISBCD, BCDJIS, BPRINT, BPUNCH

6) その他

• HELP, LINELENGTH, STATUS, DABT, DSTS, CONNECT

(5) 端 末 操 作

次のようにTSS処理の端末操作が簡単にできます。

- ① 電源を入れる。
- ② \$\$\$CON, TSSをタイプイン(ログオン)……簡易コネクトもあります。
- ③ ユーザ識別名のタイプイン
- ④ パスワードのタイプイン
- ⑤ サブシステム名のタイプイン
- ⑥ 処理
- ⑦ BYEをタイプイン(ログオフ)

例 \$\$\$CON, TSS

ACOS-6 TSS(4.1)ON 10/29/77 AT11.030 CHANNEL 0101

USER ID - KOSYU ③

PASSWORD - XXXXXXXXXX ④

SYSTEM ?FORTRAN ⑤

OLD OR NEW - NEW

READY

* ⑥

}

*BYE ⑦

**USED RESOURCE CPU=.2SEC CON=120SEC

LINE=NCU T-ID=11

**COST: ¥21.0

(6) プログラム例

FORTRANサブシステムを使ったプログラミング例です。

```

USER ID -YAMADA ..... ユーザ名の入力
PASSWORD-
RXSEWGRBRXSS ..... パスワードの入力
SYSTEM ?COBOL ..... 誤ったコマンドの入力
009-SYSTEM UNKNOWN ..... エラー メッセージ
SYSTEM ?HELP ..... エラー メッセージの説明を求める。

```

PLEASE ENTER MESSAGE NUMBER-009 ... メッセージ番号

```

THE REQUESTED SUBSYSTEM IS UNKNOWN TO SS OR IS NOT
INCLUDED IN THE SYSTEM FOR THIS INSTALLATION.CHECK } ... エラーメッセージ
THE NAME FOR SPELLING TOO. } ... の詳しい説明

```

```

SYSTEM ?FORT ..... FORTRANサブシステム指定
OLD OR NEW-NEW ..... 新しいプログラムの作成を指定
READY
*AUTOMATIC ..... ライン番号を自動的にプリントするように指定
*010 READ, A, B
*020 C=A*B } ..... プログラムの作成
*030 PRINT, A, B, C
*040 STOP;END
*050 CR ..... AUTOMATICの取消し指示
*LIST ..... ファイルの内容のプリント指示

```

```

010 READ, A, B
020 C=A*B
030 PRINT, A, B, C } ..... ファイルの内容のプリント
040 STOP;END

```

```

*SAVE ..... プログラムをパーマネント ファイルに保存指示
FILE NAME? ..... ファイル名指定
SAVEFL
DATA SAVED SAVEFL
*RUN ..... プログラムのコンパイルと実行
=6, 3 ..... READ A, Bに対するデータ入力
0.00000000E 01 0.30000000E 01 0.18000000E 02

```

NORMAL TERMINATION

```

*020 C=A/B } ..... プログラムの修正
*005 INTEGER A, B, C
*LIST ..... ファイルの内容のプリント指示

```

```

005 INTEGER A, B, C
010 READ, A, B
020 C=A/B } ..... ファイルの内容のプリント
030 PRINT, A, B, C
040 STOP;END

```

```

*RESAVE SAVEFL ..... SAVEFLファイルを修正済みプログラム
DATA SAVED--SAVEFL ..... で置き換える指示
*RUN
=9, 3

```

9 3 3
NORMAL TERMINATION

*BYE

(7) バッチシステムとの互換性

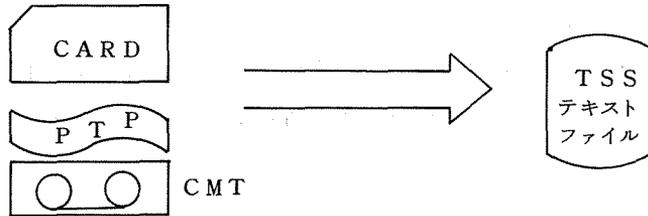
ローカルバッチ処理、リモートバッチ処理と同時動作が可能であるACOS-6 TSSは、当然バッチシステムとファイル面、プログラム面で高い互換性を有し、質の高い多次元処理を実現しています。

(8) テキストデータの一括入出力

エンドユーザTSS端末の補助入出力装置から大量データの一括処理のために次のコマンドが用意されています。

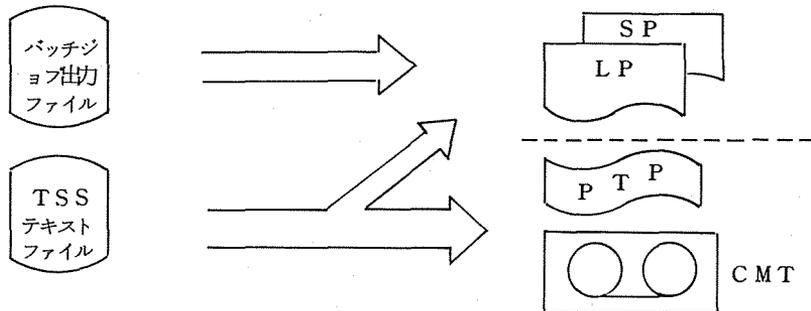
• READコマンド

端末が補助入出力装置とともに構成されているとき、指定する入力装置からファイル単位のテキストデータを一括入力できます。



• PTONコマンド

端末が補助入出力装置とともに構成されているとき、指定する出力装置へテキストデータを一括出力できます。



• PTOFFコマンド

PTONコマンドで指定した出力装置の割当てを解除します。

(9) 便利なコマンド

TSSユーザーは次のコマンドで非会話的処理が簡単にできます。

1) コマンドファイル処理

① 機能

端末利用者からのタイムシェアリングシステムへの応答をファイルから入力すること

によって代行する T S S の非会話的利用をいいます。

② 利 点

定形的なタイムシェアリングシステムの利用形態の場合、直接端末利用者が T S E とのやりとりを行わず、あらかじめ利用者が用意した一連の応答の格納されたファイルに代行させます。

2) ディファード処理

① 機 能

端末利用者があらかじめセットしておいた日時に到達すると、その時点で端末利用者の直接の介入なしに、自動的にセッションが開始され、必要なタイムシェアリング処理があらかじめ指定してあったファイルを通して行われます。

② 利 点

コマンドファイル処理の応用であり登録されていたジョブの実行開始可能な時期になったならば、そのジョブを登録した利用者識別名をもってログオンが行われ実行開始となります。

(10) 使いやすい EDITOR

T S S ユーザは、EDITORサブシステムを呼び出し、次のことが容易に行なえます。

- ・テキストファイルを作り上げる。(大量テキスト入力は一括入力コマンドを利用)
- ・既存テキストファイルに追加する。
- ・テキストファイルを追加、削除、修正して編集する。

また、ラインモード、ストリングモードにおいてきめこまかな編集ができます。

EDITORサブシステムのコマンド機能を以下に示します。

コマンド	実 行 内 容
BACKUP	サーチポイントをファイルの頭に戻す
BUILD	テキストファイルにテキストの追加を行なう
COPY	サーチポイントが指している行を含め n 行分をコピー
CUT	サーチポイントが指している行をコピーし、その行を元のファイルから削除
DELETE	サーチポイントが指している行を削除
FIND	サーチポイントを 1 行進める
INSERT	サーチポイントが指している行の後に挿入
LINE	EDITORサブシステムをストリング・モードからラインモードに戻す
NOVERIFY	VERIFYコマンドを無視する
PASTE	サーチポイントが現在指している行の後に COPY, CUT コマンドで集めたテキスト挿入
PRINT	サーチポイントが指している 1 行をプリントする
REPLACE	サーチポイントが指している行を置き換える
RUNOFF	RUNOFF サブシステムを呼び出す
STRING	EDITOR サブシステムをラインモードからストリングモードに戻す
VERIFY	EDITOR サブシステムに対するコマンドの実行を照合モードにする

EDITORの利用方法は、次に示しますように、非常に簡単です。

① ソーステキストをREPLACEコマンドでスリング置換した例

```
*AUTO
*010 5 WRITE(6,10)
*020 10 FORMAT(" ENTER X")
*030 READ(5,11,END=91X)
*040 11 F(5,96)1V)
*050 Y=SQRT(X)
*060 Z=X**2
*070 WRITE(6,30)X,Y,Z
*080 30 FORMAT(" X=",G10.5," SQRT(X)=",G10.5," X**2=",G10.5)
*090 GO TO 1
*100 9 STOP "RUN END"
*110 END
*120
*RUN
SOURCE LINE 99
<F>211 BRANCH TO NON-EXISTENT LABEL 1
*90 GO TO 5

*EDIT
-STRING
-PRINT:/G10.5/

080 30 FORMAT(" X=",G10.5," SQRT(X)=",G10.5," X**2=",G10.5)
-REPLACE:/G10.5/;3:/G13.5/
-PRINT

080 30 FORMAT(" X=",G13.5," SQRT(X)=",G13.5," X**2=",G13.5)
-DONE
```

自動的にライン番号をつけるモードに切換え (AUTOMATICコマンド)

プログラム入力

キャリッジリターンによりAUTOモードから出る
コンパイル、実行を指示 (RUNコマンド)

コンパイルエラーの表示

90 の行を入れ替え指示

EDITORサブシステム要求 (EDITコマンド)

スリング・モード指示

"G10.5"という文字列を持つ行をプリント指示

"G10.5"という文字列を3つ,"G13.5"と
結果のプリント指示 置換え指示

EDITOR終了

② 文章データをINSERTコマンドでスリング単位の挿入ができる例

```
-PRINT ;4 ..... サーチ・ポインタが指している行より4行プリント
THE TIME-SHARING SYSTEM USES A TECHNIQUE BY
WHICH PROGRAMS ARE HANDLED IN PARALLEL.
THUS, TIME-SHARING PERMITS A USER TO WORK
DIRECTLY WITH THE COMPUTER, WHETHER IT IS
-B ..... サーチ・ポインタをファイルの初めにもどす
-INSERT S:/LEL./ ..... 綴り;LELの次に挿入する
ENTER
*A
*SUPERVISORY PROGRAM ACTS AS A CONTROLLER OF
*THESE PROGRAMS, CONTROLLING "STOP" AND "GO"
*SIGNALS TO INPUTS FROM TERMINALS AND
*PREVENTING DEMANDS OF ONE TERMINAL FROM
*INTERFERING WITH DEMANDS OF OTHER TERMINALS.
*(キャリッジ・リターン)
-B ..... サーチ・ポインタをファイルの初めにもどす
-PRINT ;9 ..... 9行プリントする
THE TIME-SHARING SYSTEM USES A TECHNIQUE BY
WHICH PROGRAMS ARE HANDLED IN PARALLEL. A
SUPERVISORY PROGRAM ACTS AS A CONTROLLER OF
THESE PROGRAMS, CONTROLLING "STOP" AND "GO"
SIGNALS TO INPUTS FROM TERMINALS AND
PREVENTING DEMANDS OF ONE TERMINAL FROM
INTERFERING WITH DEMANDS OF OTHER TERMINALS.
THUS TIME-SHARING PERMITS A USER TO WORK
DIRECTLY WITH THE COMPUTER, WHETHER IT IS
-F ..... サーチ・ポインタを1行分だけ先に進める
-PRINT ..... 1行プリントする
THE PROGRAM BE CORRECTED OR CHANGED BY
-INSERT S:/RAM /:/CAN / ..... 綴り;RAMの次に綴り;CANを挿入する
-P ..... 1行プリントする
THE PROGRAM CAN BE CORRECTED OR CHANGED BY
```

4.4.3 リモートバッチ処理

(1) ジョブの構成

```
$$$CON, R BATCH
$$$LID, XX.....端末ID
$$$RCD
}
$$$OUT
$$$BYE
```

バッチジョブと同様

(2) ジョブの投入法

リモートバッチ端末から、所定の操作により端末のカードリーダーに積まれているカードが自動的に読み込まれます。

正常に読みこまれた場合には、端末のコンソールにその旨表示されます。

(3) ジョブの処理状況のチェック

リモートバッチから投入されたジョブの処理状況は、\$\$\$STSコマンドにより知ることができます。

なお、センター処理状況全般については、ステータス監視用プログラムと接続することにより表示させることができます。

(4) 処理結果の出力

リモートバッチから投入されたジョブの処理結果、ローカルバッチジョブでも\$REMOTE文のあるものおよび会話型リモートバッチのJOUTによるリモートへの出力要求などは一度シスアウトファイルに処理結果が滞りますので、これらを取り出したり制御したりするために次の3種類のコマンドがあります。

```
$$$ OUT ジョブ出力要求
$$$ OAB 出力のキャンセル
$$$ BSP バックスペース
```

4.4.4 会話型リモートバッチ処理

(1) ジョブの構成

会話型リモートバッチでは、SNUMB名がシステムにより自動的に割り当てられますので次のように構成します。

(端末でのタイプイン)

```
0010$ : JOB:KADAI BANGO
0020$ : FORTRAN:NFORM, LNO
0030$ : SELECTA:KADAI BANGO/SRCFIL
0040$ : GO
0050$ : ENDJOB
0060          (注):タブ文字
```

*SAVE JCLFIL

\$ SELECTA文により指定されたファイルの中にソースプログラムをあらかじめ格納しておきます。\$ SELECTA文を使用せずに端末よりソースプログラムをタイプインすることもできます。

TSS 処理と共通のソースファイルを使用する場合は、通常上記の方法によります。

(2) ジョブの投入方法

RUNコマンドにより SNUMB # が割りあてられバッチ処理が開始されます。

```
* RUN
      SNUMB # 2288T
      CARD FORMAT,DISPOSITION?
      NORM, JOUT(,WAIT)
```

*

NORM指定は標準タブを使用することを意味し、JOUTは処理結果をシスアウトファイルに滞めることを指示します。

JOUT指定がない場合には、無条件でセンターのプリンタに出力されます。

(3) ジョブの処理状況のチェック

JSTSコマンドにより、自分のジョブがどのように処理されているか知ることができます。

WAITの指定があれば処理終了後、端末に制御がもどります。

また、RBUGルーチンがプログラムに組込まれていれば、リアルタイムにバッチジョブのデバッグをおこなえます。

処理が終わったことがわかった場合には、つぎの(4)の処理により処理結果をとりだします。

(4) 処理結果の出力

JOUTサブシステムの利用によりシスアウトファイルに滞められている処理結果を検索し、必要があれば、更にセンターのプリンタに出力指示をおこなうことができます。

(5) 会話型リモートバッチ処理の例

次のようにCARDINサブシステムの中からRUNコマンドを入力することにより簡単に
リモートバッチジョブを実行できます。

```
ACOS-6 TSS(R302) ON 10/07/76 AT 11.419 CHANNEL 0200

USER ID -KADAIBANGO
PASSWORD--XXXXXXXXXXXX
SYSTEM ?CARDIN QLD FORTRAN ..... CARDIN サブシステムの呼び出し
READY
*LIST FORTRAN

0010$:JOB:KADAIBANGOSPASSUORD
0020$:FORTRAN:NFORM,INLNO
0030$:SELECTA:KADAIBANGO/FILEFORT } バッチジョブの制御カード
0040$:GO
0050$:ENDJOB

READY

*LIST FILEFORT

0010C-FREE FORMAT
0020D0 10 I=1,10
0030K=I**2
0040 10 WRITE(6,11)I,K
0050 11 FORMAT(1H ,I3,'**2=',I3)
0060STOP
0070END

READY

*RUN FORTRAN ..... RUN コマンドで実行要求
  SNUMB # 2288T
  CARD FORMAT,DISPOSITION: ?
  NORM,JOUT ..... 投入端末へ実行結果を出力する要求
  *JOUT 2288T ..... JOUTサブシステムでジョブ2288Tの出力要求
  2288T -02 TERMINATING ..... ジョブ2288Tは実行終了処理中
  *JOUT 2288T
  FUNCTION?PRINT 74

2288T 01 10-07-76 11.447 FORMAT LABEL -FREE FORTRAN! R3.0 PAGE 1
  1 C-FREE FORMAT 00000010
  2 D0 10 I=1,10 00000020
  3 K=I**2 00000030
  4 10 WRITE(6,11)I,K 00000040
  5 11 FORMAT(1H ,I3,'**2=',I3) 00000050
  6 STOP 00000060
  7 END 00000070

THERE WERE NO DIAGNOSTICS IN ABOVE COMPILATION
27K WORDS WERE USED FOR THIS COMPILATION

END OF 74
FUNCTION?DIRECT ONL
*DONE
SYSTEM ↑BYE
**USED RESOURCE.....CPU= .1 SEC CON= 1 5 2 SEC LINT=PRIVATE
  T-ID= 2 0
**COST : ¥ 2 0.5
```

4.5 プログラム管理

4.5.1 概要

ACOS-6では、FORTRAN, PL/1, ALGOL, BASIC, COBOL, JOVIAL, LISP, アセンブラ(GMAP)が言語処理プログラムとして使用できます。

これらの特長は次のように要約できます。

- COBOLは1974年に改訂された米国規格を包含しており、世界でも最新かつ最高水準のものです。
- FORTRANは最高水準の高速コンパイラです。
- ALGOLはJIS ALGOL水準5060に準拠し、優れた機能をもっています。
- バッチ処理, タイムシェアリングの両方で使用できます。

プログラムを作る場合に使用する処理形態として、代表的なものには、バッチ処理形態, タイムシェアリング(TS)の二つがあります。

TSSでプログラム作成のために、

FORTRAN, ALGOL, BASIC, JOVIAL

の4種の言語が使用できます。バッチ処理, リモートバッチ処理, 会話型リモートバッチ処理では

FORTRAN, PL/1, ALGOL, COBOL, JOVIAL, GMAP,

の6種の言語が使用できます。

言語処理プログラム \ 処理形態	ローカルバッチ処理 リモートバッチ処理 会話型リモートバッチ処理	タイムシェアリング システム
FORTRAN	○	○
PL/1	○	×
ALGOL	○	○
BASIC	×	○
COBOL	○	×
JOVIAL	○	○
GMAP	○	×

○ 使用できる × 使用できない

また各言語処理プログラムの規格は次の通りです。

言 語	規 格
FORTRAN	JIS FORTRAN 水準7000の拡張
PL/1	ECMA & ANSI PL/1 標準案(BASIS/1)
ALGOL	JIS ALGOL 水準5060
BASIC	—
COBOL(JIS & BCB)	JIS COBOLの最高水準
COBOL74	改訂米国規格COBOL
JOVIAL	—
GMAP	—

各言語処理プログラムの翻訳時と実行時の必要メモリサイズは次の通りです。

コンパイラ/アセンブラ	翻訳時必要メモリ(標準)	実行時必要メモリ(標準プログラム)
FORTRAN	30 KW	オブジェクトサイズ+10 KW
PL/1	90 KW	オブジェクトサイズ+33 KW
ALGOL	30 KW	オブジェクトサイズ+13 KW
COBOL(JIS)	32 KW	オブジェクトサイズ+ 7 KW
COBOL74	48 KW	オブジェクトサイズ+ 8 KW
GMAP	24 KW	4 KW~

4.5.2 FORTRAN

ACOS-6 FORTRAN はJIS水準7000を上まわる言語仕様、オブジェクトプログラムの最適化機能を持ち、バッチ処理でもタイムシェアリングでも利用することができます。

主な特長は次のとおりです。

- ① JIS FORTRAN の水準7000を満すとともに有効で使いやすい種々の拡張機能を含む言語仕様を持っています。
- ② イン・コア方式による高速コンパイラです。
- ③ 局部的および全体的な2つのレベルの最適化機能により、高速なオブジェクトプログラムを生成します。
- ④ コンパイラはローカルバッチ処理、リモートバッチ処理、会話型リモートバッチ処理およびタイムシェアリングのいずれでも利用することができ、ソースプログラムの完全な互

換性が得られます。

- ⑤ さらにオブジェクトプログラムでもバッチ、タイムシェアリング両処理形態での互換性があります。
- ⑥ バッチ、タイムシェアリングの両処理形態で、行番号あり、行番号なしの自由形式のコーディングが可能です。
- ⑦ 豊富なコンパイラオプションがあります。
- ⑧ ソースプログラムの変更なしにデバッグが可能です。

4.5.3 PL/1

ACOS-6 PL/1は、現在国際的に標準化が進められているPL/1の言語仕様(BASIS/1)に準拠しています。

次の特長を持っています。

- ① PL/1に含まれている各種データを、正確に表現します。
10進データは10進のまま演算します。UNALIGNED属性が指定されると、2進データはビット単位でパックされます。パックされたデータのアドレスをロケータ値として使用できます。
- ② AUTOMATIC, CONTROLLED, BASED変数を使用することによって、変数(スカラー、配列、構造体)を動的に割り付けることができます。
- ③ コンディション処理機能によって、プログラム実行中に生じる各種のエラー処理を行うことができます。
- ④ ロケータ変数と変数の動的割り付けの機能を利用して、広範囲のリスト処理プログラムを作ることができます。

4.5.4 ALGOL

ACOS-6 ALGOLは、JIS ALGOL水準5060に準拠したもので、豊富な機能を備えています。ACOS-6 ALGOLは、バッチ、TSどちらの処理形態でも使用できます。

次の特長もっています。

- ① 文字はJISコードを使用し、カナ文字が取り扱えます。
- ② データの型には、整数型、倍精度整数型、実数型、倍精度実数型、論理型、および文字型があります。
- ③ 式には、算術式、論理式、行先式、および文字式があります。

- ④ 標準関数が豊富にあります。
- ⑤ 単純変数および配列の宣言文に INITIAL を指定することによって初期値設定が可能です。
- ⑥ EXTERNAL によって、外部手続きの宣言が可能です。
- ⑦ 手続き単位の翻訳ができ、オーバーレイが行えます。
- ⑧ 入出力手続きが豊富にあります。

4.5.5. BASIC

BASIC は、TSで広く用いられている数値計算用会話型言語です。BASIC は簡単に習得でき、しかも、デバックが会話形式で容易に行えるのが特長であり、比較的簡単な数値計算問題を解くには最適の言語です。

4.5.6 COBOL

ACOS-6 COBOLは1974年に改訂された米国規格 COBOL を基にしており、その全仕様を満たしているばかりでなく、さらに独自の機能を追加しています。

主な機能と特長は次のとおりです。

- ① 磁気ディスク装置におけるファイルに編成が扱えます。

ファイル編成に順編成、索引編成、相対編成の三つがあり、データ処理の目的に応じて選択できます。また COBOL はデータベース処理言語と一体化しているため、データベース処理に COBOL の豊富なデータ処理機能を利用することができます。

- ② 通信機能が使えます。

SEND, RECEIVE 等のように、簡単な命令を記述するだけで端末装置との通信ができます。

- ③ 使いやすいデバッグ機能があります。

ソースプログラムにデバッグ用指定を入れることによって、プログラムの実行時にプログラムの流れを追跡したり、データ項目、ファイル領域、通信領域の参照を監視したりすることができます。デバック完了時には、プログラムを翻訳し直すことなしにデバッグ機能を排除できます。

- ④ 制御部

制御部指定を行うことによって、ソースプログラムの特定の文字列を置き換えて翻訳することができます。この機能は他機種からのプログラム変換に活用できます。

また、個々のデータ記述を書き換えることなしに、数字項目の符号を重ね符号から分離符

号に変えることができます。

⑤ CALL/CANCEL 命令

CALL 命令によって呼び出したサブルーチンを CANCEL 命令によって無効にし、サブルーチンが占める記憶領域を解放することができます。

⑥ コードセット文字の大小順序

COBOL では ACOS II コードの磁気テープだけでなく、BCD コードや IBM コードの磁気テープも変換なしに直接入出力することができます。さらに、プログラム内で文字の大小順序を決めるためのコードセットも、上記のいずれかを選択することによって可能です。

4.5.7 JOVIAL

JOVIAL (Jules'Own Version of the International Algebraic Language) は、数値計算機能に加えて文字処理機能などのシステムレベルのプログラム作成用の多くの機能を含んだ汎用言語です。

ACOS-6 JOVIAL は、バッチ、タイムシェアリングシステムのいずれの処理形態でも使用できます。

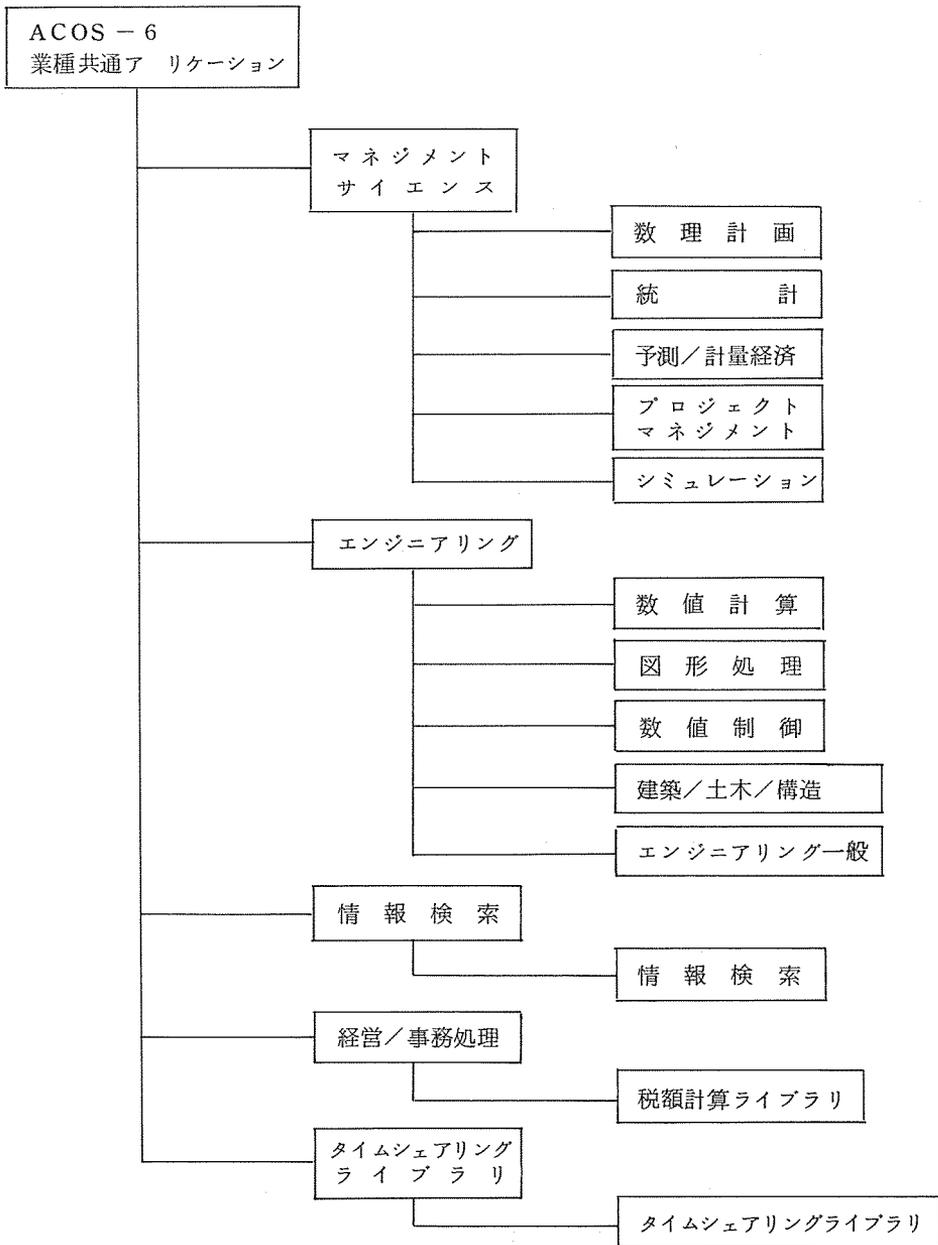
4.5.8 GMAP (アセンブラ)

GMAP は、S700 の豊富な命令セット、アドレス修飾機能を使えるように、全機械命令のほか、多数の疑似命令が用意されている機能の高いアセンブラであり、どのような問題に対しても最適なプログラムを作ることができます。入出力用、デバッグ用、ソート/マージ用マクロは、GMAP 自身に組み込まれていますが、さらに、各種のマクロ記述用機能を利用してエンドユーザ用のマクロを定義できます。

4.5.9 アプリケーション

ACOS-6 業種共通アプリケーションの体系は、次図に示してあるように、マネジメントサイエンス、エンジニアリング、情報検索、経営/事務処理、TSS ライブラリの 5 つのカテゴリに分類されており、大型コンピュータのユーザに不可欠な製品が準備されています。

ACOS-6 は多次元処理形態をとることが可能なシステムです。従って、タイムシェアリングモードでは TSS 専用のパッケージおよびライブラリが TSS ライブラリとして準備されており、マネジメントサイエンス、エンジニアリングおよび情報検索の各分野のアプリケーションはローカルバッチモード、リモートバッチモードのいずれの処理状態でもサービスが可能といったように、オペレーティングシステムの運用形態に準じた体系が組まれています。



大分類	小分類	項目	名称
マネジメントサイエンス	数理計画	数理計画システム	MPS-6
	統計	統計解析システム	STATPAC-6
		クラスタ分析	CLUSTER-6
	予測/計量経済	予測/計量経済システム	FORES-6
		タイムシェアリング用予測/計量経済システム	FORES/TSS-6
	シミュレーション	離散型シミュレーション言語	GPSS/V-6
		システムダイナミックス言語	DYNAMO/F-6
連続型シミュレーション言語		CSPL-6	
エンジニアリング	数値計算	数値計算ライブラリ	MATHLIB-6
	図形処理	グラフィックディスプレイライブラリ	GDSP-6
		デジタルプロッタライブラリ	PGL-6
	数値制御	数値制御システム	APT-6
	建築/土木/構造	建築/土木/構造 パッケージライブラリ	ICEP-6
		複合構造解析システム	ISAP-6
	電気	電子回路網解析システム	ECAP-6
		電子回路解析システム	NECTAR-6
情報検索	情報検索	情報検索システム	IRS-6
		タイムシェアリング用 情報検索システム	IRS/TSS-6
タイムシェアリングライブラリ	タイムシェアリングライブラリ	タイムシェアリングライブラリ	TSS/LIB-6
その他 (テスト中)	マイクロコンピュータ	クロスアセンブラ言語システム	MCOM/A-6
		クロスコンパイラ言語システム	PL/N-6