

TSS

<https://doi.org/10.15017/1472528>

出版情報：九州大学大型計算機センター広報. 10 (3-特別), pp.188-195, 1977-10-15. 九州大学大型計算機センター
バージョン：
権利関係：



3. TSS

OSIV/F4においては、従来のM-VIIのCPSと比較して、ほぼ同じ機能をもつTSS(Time Sharing System)が使用できる。従来のCPSに追加された機能としては、実行時に会話的なデバッグができることと、交換回線からの利用が可能であるという2つの点があげられる。ここでは、TSSの基本的概念と新しい用語を紹介し、また、TSSのコマンド言語、プログラム言語についても簡単に取りあげることにする。TSSの利用に際しては、原則的にはマニュアル(→〔41, 42, 43, 44, 45〕)の仕様通りに使えるので、詳しくはそれらを参照していただきたい。また、OSIV/F4の端末として使えるものを、交換回線用のものも含めて付表7にのせたので、端末の種類については、そちらを参照していただきたい。

3.1 ソフトウェアの構成と処理の概要

TSSにおいて端末と計算機間のデータ伝送は、VTAM(Virtual Telecommunications Access Method)とよばれる通信アクセス法を介して行われる。端末の入出力要求は、すべてVTAMのREAD/WRITE命令に変換されて、直接VTAMによって処理される。しかし、利用者はVTAMを意識する必要がまったくない。TSSの処理概念図を図3.1に示す。

図3.1のTSSの制御プログラムについて簡単に述べる。

1) タイムシェアリング制御タスク(TSCT: Time Sharing Control Task)

TSSの初期設定、サービス終了処理を行うとともに、端末からのセッション開設要求の受け付けやセッション終了処理も行う。そのほか、システムの障害処理も受け持つ。

2) 空間制御タスク(ASCT: Address Space Control Task)

利用者のジョブのアドレス空間の初期設定、終了処理を行う。また、スワップアウトされる前の

状態の保存、スワップイン後の復旧処理やアテンション割り込み処理プログラムのスケジューリングを行う。

3) セッション制御タスク(SCT: Session Control Task)

バッチ処理における、JESリーダ、イニシエータの役割を果たす。つまり、セッション開設時に各種資源^{注3.1)}の割り当てなどの初期設定を行い、セッション終了時に、資源返却などの終了処理を行う。

4) 端末入出力制御プログラム(TSSMCP: TSS Message Control Program)

端末入出力要求の管理を行うとともに、メッセージの編集、アテンション割り込みの処理、端末属性の操作なども行う。

5) ターミナルモニタプログラム(TMP: Terminal Monitor Program)

端末とシステムとの会話を制御する。端末の利用者にコマンドの入力を促したり、要求されたコマンドに対して、コマンドプロセサのよびだしを行う。

6) コマンドプロセサ(CP: Command Processor)

コマンドによる端末利用者からの要求を処理する。その時に、コマンドのオペランドの構文チェックを行い、誤ったパラメタがあれば、再入力を要求する。

以上、述べたもののほかに、ソフトウェアを構成するものとして、つぎのようなものがある。

7) サービスルーチン

端末などに対する入出力サービス、コマンドの構文チェック、データセットの割り当てを行うなどの処理ルーチンである。

8) シンタクスチェッカ

ソースプログラムの1ステートメントごとに構文の誤りをチェックする。FORTRAN, BASIC, PL/Iに対して用意されている。

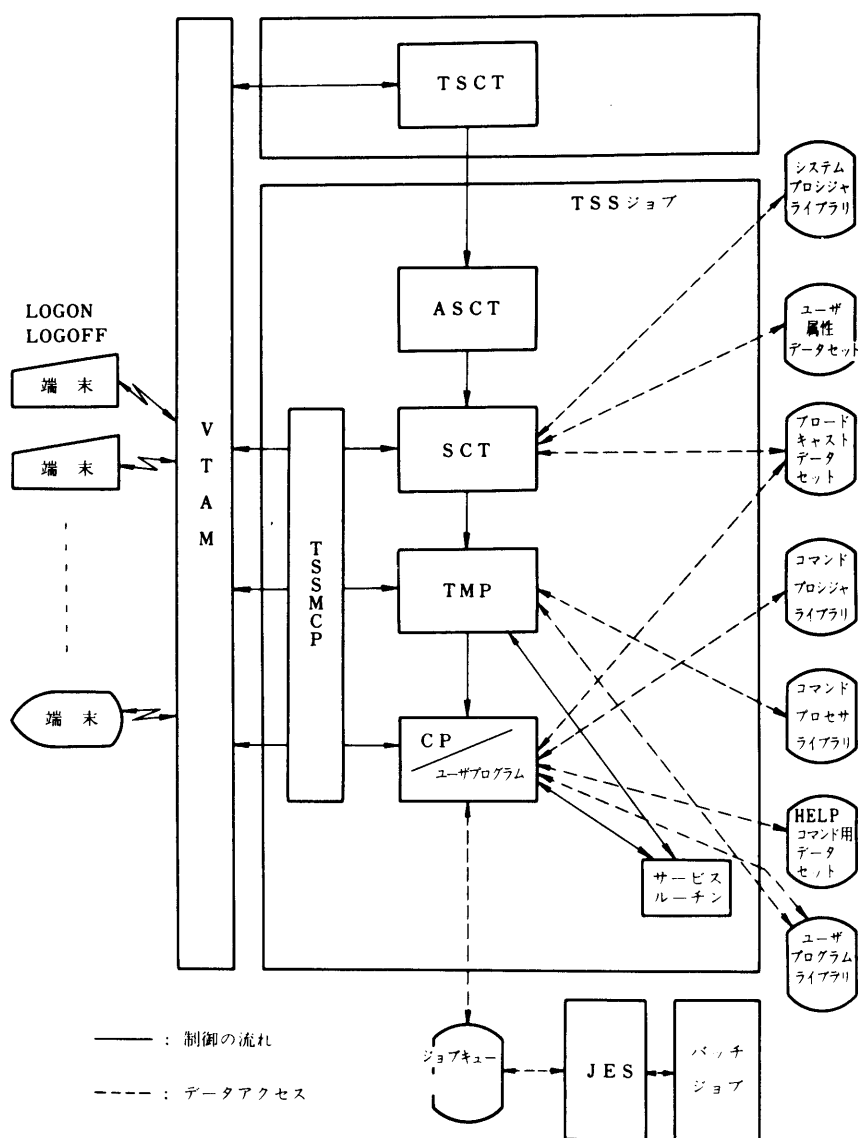


図 3.1 TSS 処理概念図

9) プロンプタ

言語処理プログラムの実行に必要な準備を行い、そのあとで、言語処理プロセサをよびだす。

10) インタラクティブデバッグ

実行時にソースプログラム中の文番号や変数名などを直接指定して、会話的にデバッグを進めるためのプログラムであり、アセンブラ、FORTRAN GE, COBOL に対して用意されている。(→ [23, 20])

つぎに、利用者の立場からみた処理の概要を示す。最初に、端末の利用者はセッション

(LOGON コマンドを入力してから、LOGOFF コマンドを入力するまでをいう) 開設のために、LOGON コマンドを入力する。セッションが開設されると、利用者が必要とする作業を、システムに指示するために、各種のコマンドを入力する。コマンドが入力されると、システムは該当するコマンドプロセサを、コマンドライブラリから呼び出して実行させる。コマンド入力時に、オペランドの省略値が設定される。また、システムで標準値が与えられないオペランドに対しては、端末にメッセージを出し、足りない

情報を入力するように促す。利用者が目的の作業を終えると、セッションの終了を行う。セッションを終了するには、LOGOFFコマンドを入力する。LOGOFFコマンドにより、システムは割り当てられていた各種資源を解放し、端末はTSSから切り離なされる。現在のセッションを打ち切って、新たなセッションを開設したときは、LOGFFコマンドを入力しないで、再びLOGONコマンドを入力してもよい。

3.2 TSSのコマンド言語

端末にコマンドを入力するには、端末がコマンドを受け付ける状態になければならない。端末がコマンドを受けとる状態にあるとき、コマンドモードにあるという。端末がコマンドモードになっ

たことは、“READY”というメッセージの出力によって示される。利用者は“READY”というメッセージが出たあとでコマンドを入力しなければならない。また、利用者が、コマンドの処理中に、その処理に必要なデータを入力したりすることがある。この状態をデータモードとよぶ。そのほかに、コマンドによっては、いくつかのサブコマンドを有するものがある。例えば、EDITコマンドやTESTコマンドなどである。このようなコマンドを入力すると、システムはそのコマンドのサブコマンドの入力を受け付ける状態になる。この状態をサブコマンドモードという。端末がサブコマンドモードになったことは、そのコマンドの名前を端末に出力することによって示される。これらの各モードの関係を図3.2に示す。

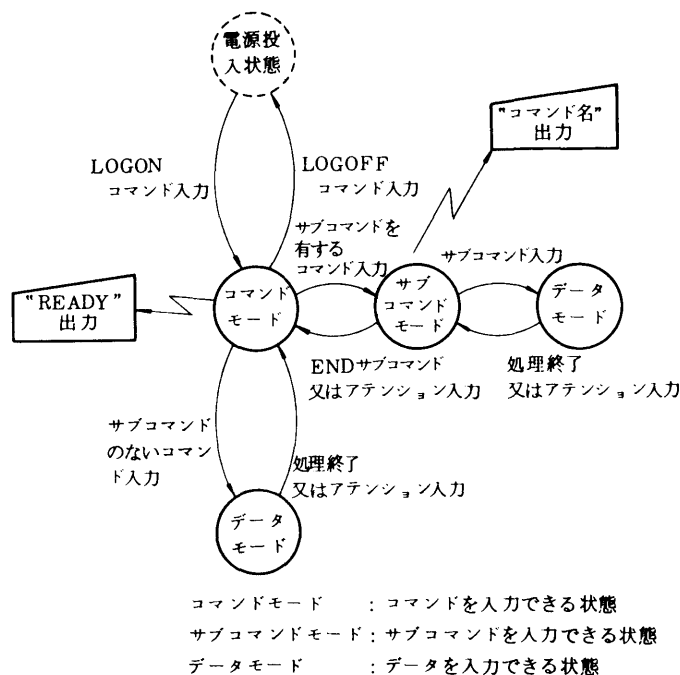


図3.2 端末のモード遷移図

種 別	コマンド名	機 能 概 略
セッション 制 御 用	LOGON	新しいセッションを開設する。
	LOGOFF	セッションを終了する。
テキスト 編 集 用	EDIT	テキスト編集を行う。
	MERGE	データセットの内容を、他のデータセット中に組み込む。
プログラム 処 理 用	RUN	ソースプログラムを翻訳・実行する。
	LOADGO	ローダにより、オブジェクトモジュールをリンクエディットして実行する。
	LINK	リンカージェディタにより、オブジェクトモジュールをリンクエディットしてロードモジュールを作成、保存する。
	CALL	ロードモジュールを実行する。
	LIBRARY	指定したデータセットをタスクライブラリとして有効にする。
	FORT	FORTRAN GEコンパイラにより、FORTRANソースプログラムを翻訳又は、翻訳・実行する。
	FORTHE	FORTRAN HEコンパイラにより、FORTRANソースプログラムを翻訳する。
	CONVERT	FORTRANソースプログラムの自由形式／標準形式の変換を行う。
	COBOL	COBOLコンパイラにより、COBOLソースプログラムを翻訳する。
	PLI	PL/Iコンパイラにより、PL/Iソースプログラムを翻訳する。
	BASIC	BASICコンパイラによりBASICソースプログラムを翻訳・実行する。
データセッ ト管理用	LISP	LISPインタプリタによりLISPソースプログラムを解釈する。
	ASM	アセンブラにより、アセンブラソースプログラムをアセンブルする。
	ALLOCATE	データセットを割り当てる。
	FREE	割り当てられたデータセットを、解放する。
	DELETE	データセットを消去する。
	ATTRIB	データセットの属性を登録する。
	RENAME	データセット名を変更する。
	LISTALC	割り当てられたデータセットの一覧を出力する。
デバグ エイド用	LISTCAT	カタログに登録されたデータセットの一覧を出力する。
	LISTDS	データセットの属性を出力する。
	TEST	アセンブラプログラムの実行時の会話的なデバグに有効な手段を与える。
端 末 制 御 用	TESTFORT	FORTRANプログラムの、実行時の会話的なデバグの手段を与える。
	TESTCOB	COBOLプログラムの、実行時の会話的なデバグ手段を与える。
	TERMINAL	端末属性を定義する。
会 話 型 リ モ ー ト バ ッ チ 用	PROFILE	ユーザの端末操作属性を定義する。
	SUBMIT	FIBジョブの処理を依頼する。
	STATUS	依頼したFIBジョブの処理状況を問い合わせる。
	CANCEL	依頼したFIBジョブを取り消す。
そ の 他	OUTPUT	FIBジョブの処理結果の出力を得る。
	SEND	端末利用者又は、センタオペレータにメッセージを送信する。
	LISTBC	センタオペレータ又は、他端末利用者から送信されたメッセージを出力する。
	TIME	LOGON時からのCPU時間、セッション時間を得る。
	HELP	コマンドの一覧や、その構文、機能、オペランドの説明を出力する。

表 3.1 コマンド一覧

アテンション入力というのは、一般には端末ごとに決められたアテンションキーを押下することである。とくに、EDITコマンドにおいては、サブコマンドモードのことを編集モード(EDITモード)、データモードのことを入力モード(INPUTモード)とよぶことがある。

TSS のコマンド言語は、用途に応じて表 3.1 のように分類される。ここでは、分類されたコマンドの中で主要なものを取り上げて解説する。

1) セッション制御用コマンド

利用者が端末からセッションを開設したいときに、LOGON コマンドを入力する。さらに、オペランドとして“TSS”を入力し、そのほか、^{注3.2)} 課題名、パスワードなどの必要な情報を入力しなければならない。ただし、同一課題番号を用いて同時に複数のセッションを開設することはできない。セッションが開設されると端末に“READY”というメッセージが出て、コマンドを入力できる状態となる。セッションを終了するには、LOGOFF コマンドを入力する。

2) テキスト編集用コマンド

ソースプログラムやデータからなるテキストのデータセットを作成。更新、追加、削除などを行う場合に、EDIT コマンドを用いる。EDIT コマンドはいくつかのサブコマンドを有していて、編集作業の形態と EDIT サブコマンドの関係は、図 3.3 のようになっている。

また、MERGE コマンドは、データセットの全体、または、データセットの中の一連のレコードを取り出し、別のデータセット、または、それ自身のデータセットの中に組み込むことができる。

ここでは、図 3.2 の関係を EDIT コマンドを例にとって説明する。図 3.2 を具体的に書き下ろすと図 3.4 のようになる。

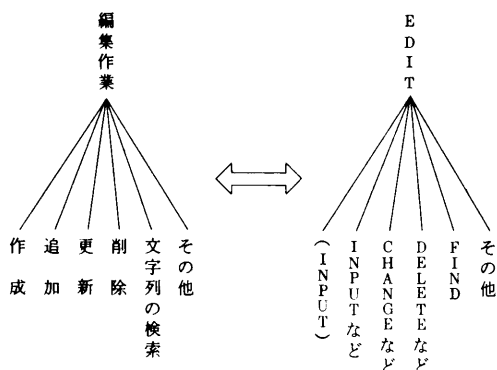


図 3.3 編集作業の形態と EDIT のサブコマンド

2.1) 入力モード (INPUT モード)

利用者が新規にデータセットを作成する場合は、EDIT コマンドのオペランドで NEW を指定する。この場合、端末は入力モードとなり、端末の左端に行番号が出力され、データの入力が促される。1 行の入力が終わると復帰改行のキーを押し、この操作をくり返す。また、INPUT サブコマンドを入力した場合も、入力モードとなる。入力モードから編集モードにするには、図 3.4 にあるように空行^{注3.3)}を入力すればよい。EDIT コマンドで作成されるデータセットは一時的なものであるから、保存したい場合は SAVE サブコマンドを用いる。

2.2) 編集モード

テキストの更新、追加、削除などを行うには、EDIT のサブコマンドを使用する。CHANGE、DELETE などのサブコマンドを用いて、行単位の更新、削除が行える。これらの編集作業は、行番号を指定して行うので行番号編集という。また、FIND サブコマンドを用いてある文字列の検索が行える。検索した後、CHANGE サブコマンドなどを用いて、その行の修正を行うことができる。これを文脈編集とよぶ。編集作業の終了を示すには、END サブコマンドを入力すればよい。

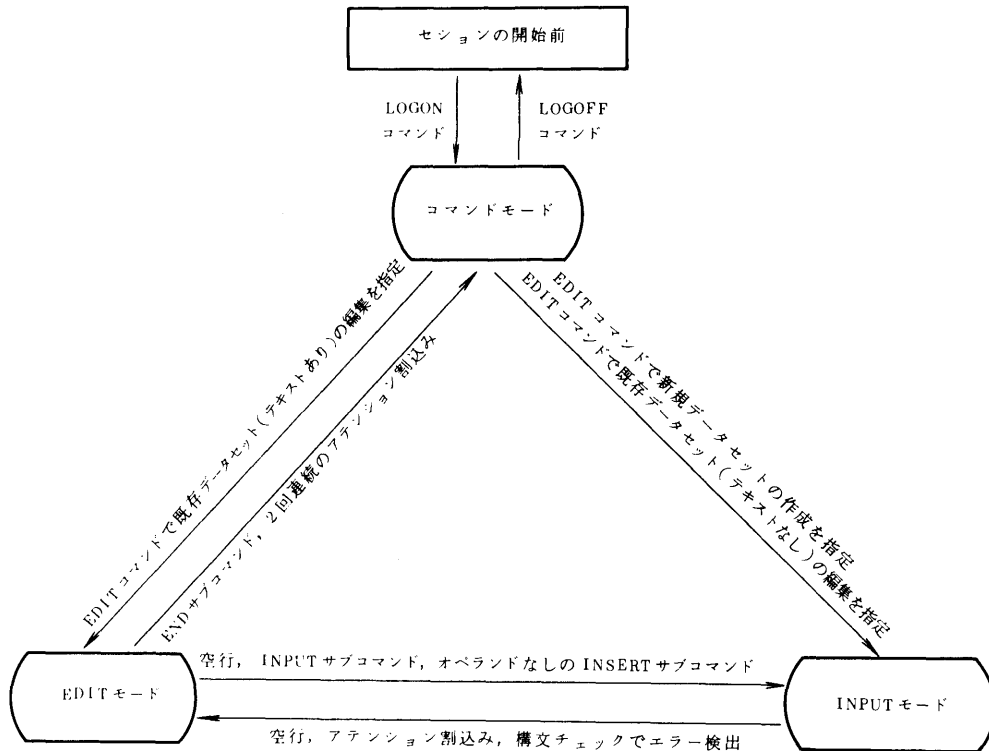


図3.4 EDITコマンドのモード遷移図

3) プログラム処理用コマンド

ソースプログラムの作成後、ソースプログラムの翻訳を行うのに、FORT コマンド、FOR THE コマンド、PLI コマンド、COBOL コマンド、ASM コマンドがあり、翻訳・実行を行うのに RUN コマンドがある。BASIC コマンド、LISP コマンドはソースプログラムの解釈を行う。オブジェクトモジュールやロードモジュールを結合編集するのに LINK コマンドがあり、結合編集・実行を行うには LOADGO コマンドがある。また、ロードモジュールの実行には CALL コマンドを用いる。翻訳から実行までの一連の処理を行うには、RUN コマンドが便利であるが、ロードモジュールは保存されない。ロードモジュールを保存したい場合は、翻訳用のコマンド（例えば FORT コマンドや PLI コマンドなど）、LINK コマンド、CALL コマンドを組み合わせ使用しな

ければならない。

4) データセット管理用コマンド

利用者が必要とするデータセットを必要になった時点で動的に確保したい場合、ALLOCATE コマンドを用いる。確保したデータセットを解放するには、FREE コマンドを用いる。データセットを消去するには、DELETE コマンドを用いる。また、確保したデータセットの一览や、データセットの属性を知りたい場合は、それぞれ LISTALC、LISTDS コマンドを用いる。

5) デバッグエイド用コマンド

プログラムの実行時のデバッグを助けるために、TEST コマンド、TESTFORT コマンド、TESTCOB コマンドがあり、それらは種々のサブコマンドを有している。TEST コマンドは、主としてアセンブラ言語で書かれたプログラムのデバッグ用である。TEST コマンドを使うた

めには、アセンブル、結合編集時にTESTオプションを指定しておく必要がある。TSSのインタラクティブデバッグ機能として、FORTRAN GE COBOLのデバッグ用にTESTFORTコマンド、TESTCOBコマンドが用意されていて、機能はTESTコマンドとはほぼ同じである。これらの機能を使うには、翻訳時にTESTオプションを指定しておく必要がある。

6) 端末制御用コマンド

端末の種類によっては、アテンションキーやバックスペースキーなど特殊なキーをもたないものがある。そのために、TERMINALコマンドやPROFILEコマンドがあり、利用者は使いやすいように端末の属性を指定することができる。

7) 会話型リモートバッチ用コマンド

TSSの利用者が、バッチの環境のもとでジョブを実行させることができるように、会話型リモートバッチ用のコマンドが用意されている。バッチに依頼するためには、図3.5のようなカードイメージのデータセットをあらかじめ作成しておく必要がある。

データセットの作成は、バッチで行ってもよいし、TSSのEDITコマンドを用いてもよい。

SUBMITコマンドのオペランドで、作成したデータセットの名前を指定すると、ジョブはバックグラウンドキュー（会話型リモートバッチ用のキュー）につがれ、そのあとで、バッチの入力キューにつながれる。このようにしてバッチに依頼されたジョブのことをFIB(Foreground Initiated Background)ジョブという。出力結果の取り出しには、OUTPUTコマンド、FIBジョブの取り消しには、CANCELコマンドを用いる。

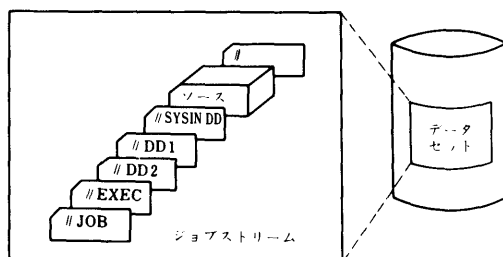


図3.5 FIB用のジョブストリーム

8) そのほかのコマンド

端末やCPUの使用時間を知るために、TIMEコマンドが用意されている。また、HELPコマンドを用いて、コマンドの意味やオペランドの文法を知ることができる。

3.3 TSSのプログラム言語

端末でソースプログラムを作成し、翻訳、結合編集、実行を行うことのできる言語にFORTRAN COBOL, PL/I, アセンブラがある。また、ソースプログラムの翻訳・実行、解釈を行うことのできる言語に、それぞれBASIC, LISPがある。FORTRAN GE, COBOL, アセンブラには、会話的にプログラムのデバッグを行える機能があり、BASICやLISPにも同じような機能がある。

ここでは、FORTRAN(→5)を例にとり、FORTRANプログラムの操作手順を示す。(→図3.6)

FORTRANのソースプログラムの作成は、バッチ処理で行うか、または、TSSのEDITコマンドを用いて行う。EDITコマンドで作成するときは、SCANサブコマンドを用いてシンタクスチェックを行うことができる。FORTRAN GEでは、自由形式でもソースプログラムを作成できる。翻訳を行うにはFORT, FORTHEコマンド、結合編集にはLINKコマンド、実行にはCALLコマンドを用いる。RUNコマンドを用いれば、翻訳

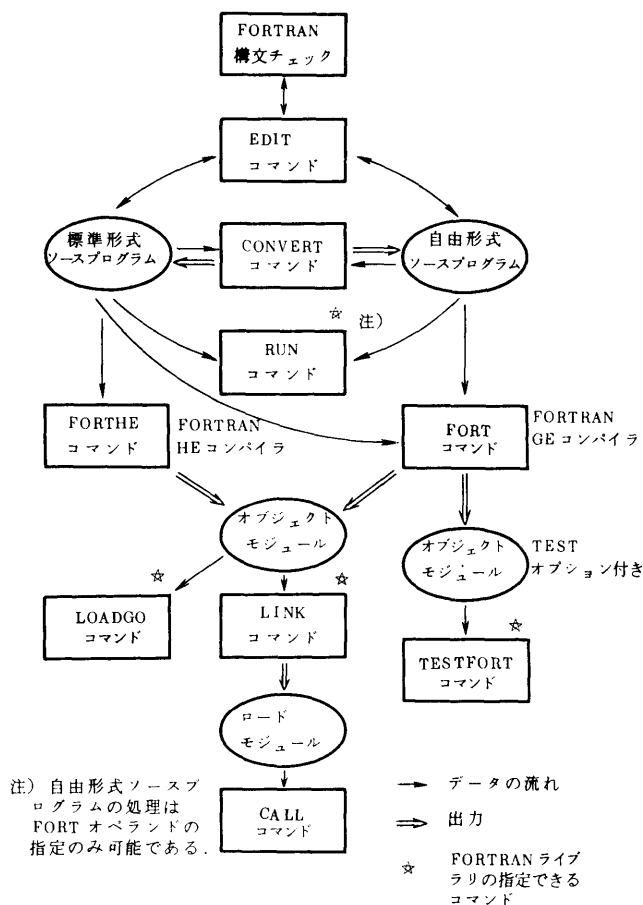


図 3.6 FORTRAN プログラムの操作手順

・結合編集・実行が行える。インタラクティブデバッグを行うには、TESTFORT コマンドを用いる。ただし、翻訳時に TEST オプションが指定されていなければならない。

注31) 注 2.2) 参照

帰改行のキーを押下すること。

注32) F n n n n のことである。n n n n は課題番号。

注34) J I S で定められた記入欄の規約に従わないソースプログラムを意味する。

注33) 入力行の内で、テキストなしに、直接、復