

DOCK/FORTRAN77によるFortranプログラムのデバック/テストについて

上妻, 真知子
九州大学大型計算機センター研究開発部

武富, 敬
九州大学大型計算機センター研究開発部

景川, 耕宇
九州大学大型計算機センター研究開発部

<https://doi.org/10.15017/1468062>

出版情報：九州大学大型計算機センター広報. 16 (1), pp.1-26, 1983-01-25. 九州大学大型計算機センター
バージョン：
権利関係：



DOCK/FORTRAN77によるFortranプログラムのデバッグ/テストについて

上妻 真知子*, 武富 敬*, 景川 耕宇*

1. はじめに

現在、センターにはFortranプログラムのデバッグ/テスト用ツールとして以下のものがある。

- i) Fortran GE, Fortran 77 処理系に備えられたデバッグ機能
- ii) Fortran 77 インタラクティブデバッグ**
- iii) Fortran プログラム テスト/デバッグ支援システム Forprex [1]

しかし, i), ii) については,

- ・デバッグ出力が過大になりやすく, 後で検査するのが大変である
- ・ソースプログラムとの対応づけが難しい

などの欠点がある。また iii) については, これらの欠点は緩和されているものの, 使用頻度調査でみ
る限り利用はさほど多くない。依然としてプログラムの要所に出力文を指定し監視するという旧態依
然の方法が採られているようである。

ここで解説する DOCK/FORTRAN77 (DOCK for Fortran 77 programming, 以後 DOCK/
FORT と略称) は, 上記ツールと比較して, フルスクリーン用ディスプレイ端末にソースプログラム
を表示し, その実行の様子を画面にスロービデオ表示することにより, 制御の流れを目で追いながら
デバッグ/テストが行えるという点に大きな特徴がある。

2 では, DOCK/FORTRAN77 概説として, その特徴および機能の概要を述べ, 3 では, 使用例を
通して DOCK/FORTRAN77 の使用法について述べる。

なお, 実行演示システムという観点だけから言えば, 九大情報処理教育センターで実働中のFortran
用演示システム FORVID [2] がある。

2. DOCK/FORTRAN77 概説

2.1 DOCK/FORTRAN77 の特徴と機能

DOCK/FORT は, Fortran プログラム (Fortran 77, Fortran IV***Fortran GE, HE) のデバ
ッグ/テストを会話的に支援するシステムである [3]。フルスクリーン機能を持つ F9525, F9
526, F6650 系のディスプレイ端末で使用できる。以下にその特徴と機能の概要を述べる。

1) メニュー方式による入力

デバッグしたい Fortran ソースプログラム, コンパイラオプション, 組込みたいライブラリなどの
初期設定に必要な情報入力, 及びデバッグ動作中にも可能である実行モードの変更や実行時デー
タセットの割当て指定などは, すべてメニュー画面の形式で提示されるので利用が容易である。図 2.1

* 九州大学大型計算機センター研究開発部

** Fortran GE インタラクティブデバッグを包含している。

*** Fortran 77 と非互換な部分を含む Fortran IV プログラムについては, コンパイラオプション
として, LANGLVL(66) を選べばよい。

に、メニュー遷移の様子を示す。すでにTSS PFDの解説[4]で、メニュー方式については詳しく述べたが同様なものと考えてよい。HELP機能(HELP画面の表示)は、PF1キーによりいつでも呼出せる。

2) 実行速度の制御

DOCK/FORTでは、プログラム単位に実行モードを指定でき、それによって各プログラム単位の実行速度や静止モード(プログラムの実行が一時的に中断された状態)への移行を制御できる。実行動作中、静止モードになった時点で、データの値の参照(後述するスナップ文のWRITE文による)やデータ値の一時変更(スナップ文の代入文による)による実行などのデバッグを行うことができる。実行モードには、以下の4種がある。

S:ステップモード……… 実行が一文ごとに中断、すなわち静止モードへ移行するモード。

L:低速モード…………… 特定の文(付録1参照)で静止モードへ移行し、それ以外は連続して実行するモード。通常は、これが省略値である。

M:中速モード…………… プログラム単位の最初と最後の実行文の実行直前で静止モードへ移行し、それ以外は連続して実行するモード。

H:高速モード…………… 連続して文を実行するモード。上の3種と異なり、実行中のソースプログラムは画面上に表示されない。

実行モードの定義および変更は、静止モードの時点で実行モード定義メニューを呼出し定義・変更するか、または実行画面(図2.2参照)の1行目MODE欄に指定して変更する。

3) プログラム実行プロセスの表示

実行モードがステップまたは低速モードの時、ディスプレイ画面上の左半分(ソース表示域、図2.2参照)に表示されたソースプログラムリストにおいて、実行中の文が高輝度(カラーディスプレイの場合は赤)で表示され、実行のプロセス、制御の流れを視覚的に捉えることができる。これによって、プログラムの論理的検証を行うことができる。

4) DOCK/FORTのデバッグ機能

DOCK/FORTのデバッグ機能は、以下の3種に大別される。

i) スナップ文

静止モードとなった時点で、スナップ文(付録2参照)と称される文を実行画面の2行目(コマンド入力域、図2.2参照)に入力することにより、データ値の一時変更、データ値の参照、制御の移行などによるデバッグを行うことができる。スナップ文としては、付録2に示す5種類がある。スナップ文の特徴を以下に列挙する。

- ・スナップ文において指定する変数名、配列名、配列要素名、部分列名および文番号は、画面に表示されているプログラム単位で定義・参照されているものに限る。
- ・スナップ文は指定された時点で直ちに実行され、その実行結果は画面の右半分(データ表示域)に表示される。
- ・スナップ文は、それが指定された時点で動的に実行状態を変更するものであり、デバッグ中のソースプログラムへの反映はまったくない。そのため、監視のための出力文を多数指定してプログラムを汚すようなことをしなくてすむ。

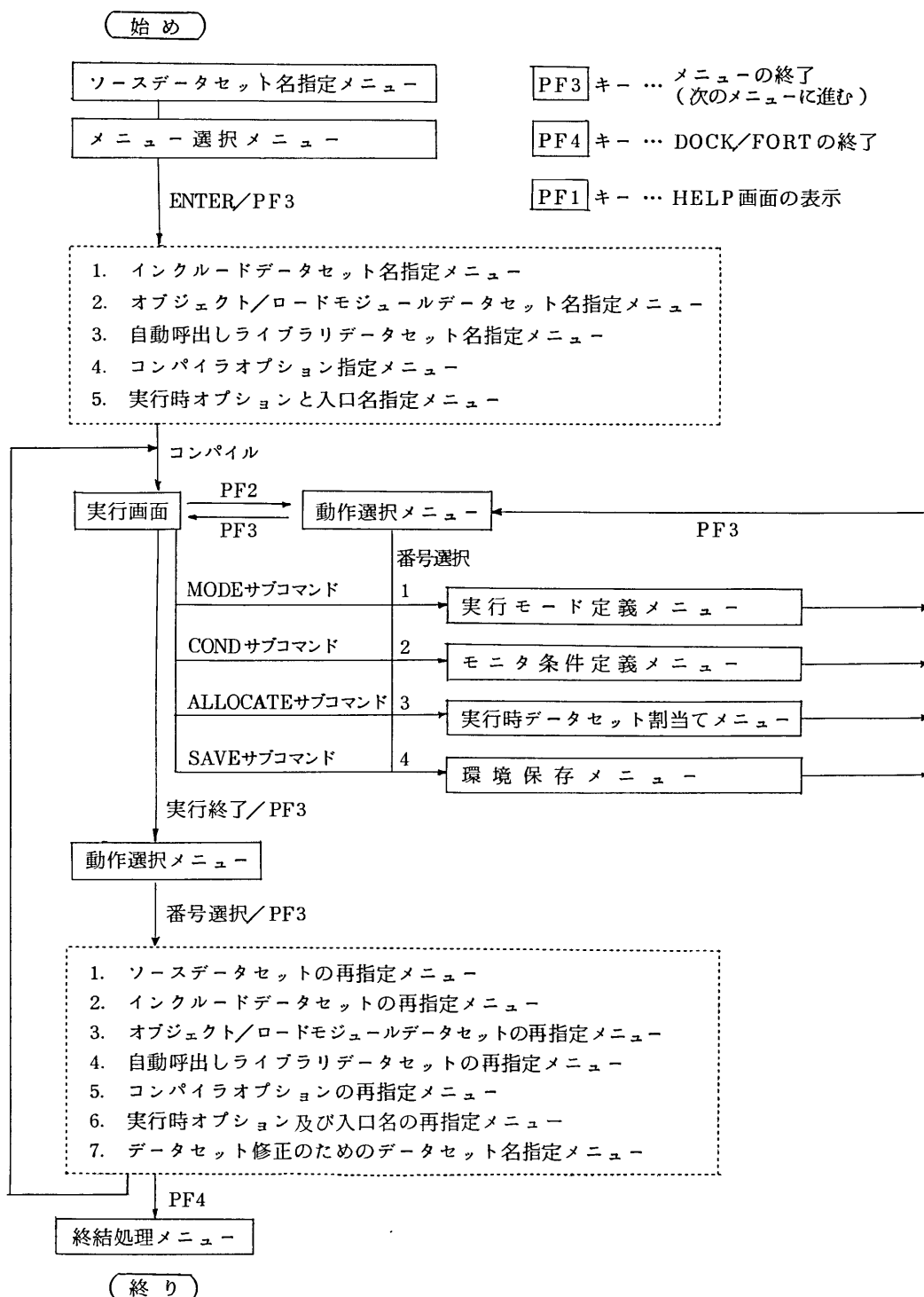


図 2.1 DOCK/FORTRAN77におけるメニュー遷移図
PF キーの機能については、付録 5 を参照のこと。

ii) データモニタ

データモニタ機能として、以下の3種がある。

a. 引数の検査ARGCHK

副プログラム引用ごとに、実引数と仮引数の個数、型、類を検査する。この検査を行わせるには、実行画面の1行目ARGCHK欄にYと指定する必要がある。

b. 配列要素・文字部分列の検査SUBCHK

配列要素・文字部分列が指定されている実行文ごとに、添字式・部分列式の値を検査する。この検査を行わせるには、実行画面の1行目SUBCHK欄にYと指定する必要がある。

c. 利用者定義のモニタ条件による監視

利用者は、静止モードの時点でモニタ条件定義メニューを呼出すことにより、プログラム単位に以下のようなデータ値を監視する条件（モニタ条件）を定義することができる。

- ・指定された変数名、配列要素名、部分列名の値が変化したか否かを追跡・監視する
- ・指定された論理式の評価結果が真となるか否かを追跡・監視する

iii) 文実行回数の表示

実行文ごとの実行回数を計測〔5〕して、実行画面におけるソースプログラムの行番号の左側に表示する。この表示を行わせるには、実行画面の1行目FREQ欄にYと指定する必要がある。

5) プログラムの再試行

エラーが発見されたためソースプログラムを修正したり、あるいは同じプログラムを条件を少し変えて再度実行させたい場合、再試行の機能がある。

- ・プログラム実行終了後、再度先頭から実行したい場合は、PF3キー押下またはRERUNサブコマンド入力により可能である。
- ・ソースプログラムを修正後、再実行したい場合は、実行終了後の動作選択メニューでソースプログラム再編集の項目を選択すると、FSOによる編集可能な状態となるので、それで修正後再実行できる。
- ・条件を少し変えて再実行したい場合、実行終了後の動作選択メニューで種々の再指定メニューを呼出し、データセット、オプションなどを変更した後再実行できる。

また、再試行ではないがプログラム再実行などによく用いられる行サブコマンド（ソース表示域の行番号欄などに入力可能なコマンド、付録3参照）として、以下のものがある。

- ・GO …… 指定の行の実行文へ強制的に分岐させたい時に用いる。デバッグ中に途中のパスをスキップしたい時などに便利である（実行文に文番号があれば、スナップ文のGOTO文でも同じことが可）。
- ・RUN …… 指定の行の実行文まで、実行モードに関係なく強制的に高速で実行させたい時に用いる。あるプログラム単位まで強制的に高速実行させたい時は、RUNTOサブコマンドのオペランドにそのプログラム名を指定して行う。

6) 環境の保存とリスタート機能

デバッグが度重なるような場合、初期設定に必要な多数のメニュー呼出しと情報入力の煩しさを避けるため、プログラム実行に必要な環境をシステムに保存できる機能がある。このためには、DOCK

F77 コマンド (2.2 参照) のオペランドに SAVE と指定するか、または静止モードの時点で環境保存メニューを呼出し、必要な情報を設定する。保存した環境からリスタートさせるには、DOCKF77 コマンドのオペランドに RESTART を指定し、保存した時と全く同じ情報を入力してやればよい。

7) ログ機能

ソースプログラムからの端末出力 (機番 6 など)、あるいは DOCK/FORT が出力する各種メッセージは、画面のデータ表示域に表示されると共に、システムのログデータセットにも出力される。利用者は、画面上にない以前のログ情報もスクロール操作により知ることができる。また、DOCK/FORT 終了時にこれらログ情報をラインプリンタ等にも出力させることもできる。

2.2 DOCK/FORTRAN77 の開始と終了

DOCK/FORT を開始させるには、READY のあとに DOCKF77 と入力すればよい。

READY

DOCKF77

以下に、この DOCKF77 コマンドについて述べる。

1) 入力形式

コ マ ン ド	オ ペ ラ ン ド
DOCKF77	[{ SAVE RESTART }]

2) 機能

ディスプレイ用 Fortran プログラムデバッガ DOCK/FORTRAN77 を起動する。

3) オペランドの説明

SAVE: プログラム実行に必要な環境を保存しリスタートに備える。環境の保存は、FIBジョブの形で行われる。このオペランド指定時は、プログラム実行には至らない (3.2 参照)。

RESTART: 保存した環境からリスタートし、プログラムの実行へと進む。

これらのオペランド省略時は、ソースデータセットの指定に始まり、プログラム実行など一連の処理へと進む (3.1 参照)。

DOCK/FORT の終了は、PF4 キー押下によりいつでも行える。

2.3 実行画面

初期設定終了後、あるいはリスタート処理終了後のプログラム実行開始前の実行画面の様子を図 2.2 に示す。

EXECUTION SCREEN	① ARGCHK => N	② SUBCHK => N	③ FREQ => N (Y/N)	④ MODE => L S L N H
SCROLLS ⑤: ROW(P) COL(20)	ROW(P) COL(20)	ROW(P) COL(20)	VLINE : (42)	(NEST)
00190 10 READ(5,100) N, EPS, AL, AR				
00191 100 FORMAT(12,3F10.5)				
00200 IF(N.GT.9) THEN				
00210 WRITE(6,200) N				
00220 200 FORMAT('----- N = ',I2				
00230 STOP				
00240 ENDOF				
00250 N1=N+1				
00251 N2=N*100				
00260 READ(5,110) (A(I),I=1,N1)				
00270 110 FORMAT(4E14.6)				
00280 WRITE(6,205) N, EPS, AL, AR				
00281 205 FORMAT(1H, 'N = ',I2,				
00282 1, ' AR = ',F10.5/1				
00290 CALL TRANS(N,A,ASTAR,AL,A				
00300 WRITE(6,210) N1,(ASTAR(1				
00310 210 FORMAT(1H, 'ASTAR(1)...AS				
00320 DO 20 J=1,N2				
00330 B(J)=0.0				
00340 DO 20 I=J,N2,2				

ソース表示域

データ表示域

図 2.2 実行画面の構成

- ① ARGCHK 実引数と仮引数の個数、型、類の検査、省略値はN(NO).
- ② SUBCHK 配列要素・文字部分列の添字値等の検査、省略値はN.
- ③ FREQ 文実行回数の表示、省略値はN.
- ④ MODE 現画面に表示中のプログラム単位の実行モードの表示。変更も可。
- ⑤ スナップ文(付録2参照)またはサブコマンド(付録4参照)の入力域
- ⑥ SCROLLS ソース表示域、データ表示域のスクロール量の表示。変更も可。

a ソース表示域のスクロール量, b データ表示域のスクロール量

ROW : 行方向(上下)のスクロール量。指定できる値は、1～99の数値, P(Page, 通常は20行), H(Half, 通常は10行)。

COL : 列方向(左右)のスクロール量。指定できる値は、1～99の数値。

VLINE : ソース表示域とデータ表示域の境界の縦線のコラム位置。指定できる値は、3～80の数値。

プログラムからの入力要求があった時は、ソース表示域とデータ表示域の下2行が削られて、データ入力域としてとられる。

3. DOCK/FORTRAN77 使用法

3.1 初期設定

プログラムのデバッグ／テストを開始する前に、必要な環境を整えてやらなければならない。そのために、各種のデータセット、コンパイラ／実行時オプションその他の情報を、メニューに従って指定する（図2.1参照）。

初期設定の例を以下に示す。下線部が入力した部分である。

```

-----< DOCK/FORT77 SOURCE MODULE DATA SET MENU >-----
ENTER/VERIFY PARAMETERS BELOW:
DOCK LIBRARY DATA SET:
PROJECT ==> F0040
LIBRARY ==> TEST
TYPE ==> F004077
MEMBER ==>          ==>          (BLANK FOR MEMBER SELECTION MENU)
OTHER PARTITIONED OR SEQUENTIAL DATA SET:
DATA SET NAME ==>
VOLUME SERIAL ==>          (IF NOT IN CATALOG)
DATA SET PASSWORD ==>          (IF PASSWORD PROTECTED)

PRESS ENTER TO REPEAT THIS MENU
PRESS PF3 TO GO FORWARD TO THE NEXT MENU
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

デバッグ／テストの対象となるソースデータセットを指定する。区分データセットの場合、メンバ名まで指定する。指定しなければ、以下のメンバ名選択メニューが表示される。

図 3.1 ソースデータセット名指定メニュー

⇓ PF 3

```

-----< DOCK/FORT77 MEMBER SELECTION MENU >-----
==>
SELECT DESIRED MEMBER NAME(S)
ENTER S (SELECT) OR 0 (OMIT) AS LINE COMMAND
DATA SET NAME --- F0040.TEST.FORT77
LINE | MEMBER
COMMAND | MEMBER NAME | SELECTED
-----|-----|-----
      | EX1          |
      | EX2          |
      | EX3          |
S    | TEST1         |
      | TEST2         |
*** END OF DATA ***

PRESS PF3 TO TERMINATE THE MEMBER SELECTION
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

メンバの選択は、メンバ名の左欄に行サブコマンドSを入力することにより行う。始めと終りのメンバ名の左にそれぞれSと入力して、その間の一連のメンバを選択することもできる。

図 3.2 メンバ名選択メニュー

⇓ PF 3


```

-----< DOCK/FORT77 MENU SELECTION MENU >-----
ENTER/VERIFY PARAMETERS BELOW:
SELECT MENU TO BE DISPLAYED:

```

ITEM	MENU NAME	TO BE DISPLAYED
1	INCLUDED SOURCE MODULE DATA SET MENU = YES/NO	==> NO
2	OBJECT/LOAD MODULE DATA SET MENU = YES/NO	==> NO
3	AUTOMATIC CALL LIBRARY DATA SET MENU = YES/NO	==> NO
4	COMPILER OPTIONS MENU = YES/NO	==> YES
5	EXECUTION OPTIONS AND ENTRY NAME MENU = YES/NO	==> NO

```

-----
DEFAULTS OPTIONS:
COMPILER OPTIONS(FREE,EBCDIC,NOASTER,NOINCLUDE,LANGVL(77))
ENTRY NAME(MAIN)

PRESS ENTER OR PF3 TO GO FORWARD TO THE NEXT MENU
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図 3.3 メニュー選択メニュー

↓ PF3/ENTER

```

-----< DOCK/FORT77 COMPILER OPTIONS MENU >-----
ENTER/VERIFY OPTIONS FOR ALL OF SOURCE
SELECT DESIRED OPTIONS:

```

ITEM	OPTIONS (BLANK FOR DEFAULT)
1	FIXED/FREE ==> FIXED
2	EBCDIC/BCD ==> EBCDIC
3	ASTER/NOASTER ==> NOASTER
4	INCLUDE/NOINCLUDE ==> NOINCLUDE
5	ALC/NOALC ==>
6	BYNAME/NOBYNAME ==>
7	AUTOOBL(M)/AUTOOBL(NONE) ==>
8	PR(K)/NOPR (0<K<=15) ==>
9	LANGVL(L) (L:(66177)) ==> LANGVL(77)

```

-----
CHANGE SOME DATA SET'S OPTIONS ==> NO* (YES OR NO)

PRESS ENTER OR PF3 TO GO FORWARD TO THE NEXT MENU
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図 3.4 コンパイラオプション指定メニュー

↓ PF3/ENTER

実行画面へ

不必要なメニュー操作のスキップをメニュー選択メニューで指定できる。省略値はすべてNO（スキップする）なので、選択したいメニューの項目（ITEM）だけYESと指定すればよい。

この場合、コンパイラオプション指定メニューだけ選択した。

コンパイラオプション

FREEをFIXEDに変更した。なお、このメニューで*部分をYESとすることにより、データセット（順編成の時）または、メンバ（区分編成の時）単位にコンパイラオプションを指定することもできる。

メニュー選択メニューで選択できる項目について以下に述べる。

1. インクルードデータセット名の指定

デバッグを行うソースプログラムにINCLUDE行がある場合に指定する。

2. オブジェクト/ロードモジュールデータセット名の指定

デバッグの完了したプログラム単位については、実行効率の点からもオブジェクト/ロードモジュールにしておいた方がよい。それらのデータセットを指定するために用いる。

3. 自動呼出しライブラリデータセット名の指定

結合編集時に解決できない外部参照を解決するために指定するもので、通常利用者の私用ライブラリを組込むために指定する。

4. コンパイラオプションの指定

コンパイラオプションの指定を行う。

5. 実行時オプションと入口名の指定

実行時オプションと入口名（主プログラム名）の指定を行う。

3.2 環境の保存と再開

プログラム実行に必要な環境の保存と再開方法について述べる。

環境を保存するためには、DOCKF 77 コマンドのオペランドに SAVE と指定するか、実行中静止モードの時点で環境保存メニューを呼出す。前者の場合、前述した初期設定の手続きがすむと、図 3.5 に示す環境保存メニューが表示される。後者の場合、SAVE サブコマンド入力か、動作選択メニューによる指定で、図 3.5 が表示される。下線部が入力した部分である。

```

-----< DOCK/FORT77 INITIAL PROCESS KEEP DATA SET MENU>-----
ENTER/VERIFY PARAMETERS BELOW:
DATA SET FOR OBJECT KEEP :
DATA SET NAME ==> TEST.OBJ
NEW OR OLD ==> NEW
VOLUME SERIAL ==>
BLOCK SIZE ==> 3200 (BLANK FOR DEFAULT)
UNIT ==> PUB (IF NOT IN CATALOG)

SPACE : PRIMARY QUAN ==> 10 SECONDARY QUAN ==> 5 (IN TRACK UNIT)
DATA SET FOR SOURCE DISPLAY :
DATA SET NAME ==> TEST.SOURCE
NEW OR OLD ==> NEW
VOLUME SERIAL ==>
UNIT ==> PUB (IF NOT IN CATALOG)

SPACE : PRIMARY QUAN ==> 10 SECONDARY QUAN ==> 5 (IN TRACK UNIT)
DATA SET FOR TABLE KEEP :
DATA SET NAME ==> TEST.DATA
NEW OR OLD ==> NEW
VOLUME SERIAL ==>
BLOCK SIZE ==> 3072 (BLANK FOR DEFAULT)
UNIT ==> PUB (IF NOT IN CATALOG)

SPACE : PRIMARY QUAN ==> 10 SECONDARY QUAN ==> 5 (IN TRACK UNIT)
PRESS ENTER OR PF3 TO GO FORWARD TO THE SUBMITTED JOB INFORMATION MENU
PRESS PF4 TO TERMINATE THE DOCK SESSION
  
```

図 3.5 環境保存メニュー

⇓ PF3/ENTER

```

-----< DOCK/FORT77 SUBMIT JOB INFORMATION MENU >-----
ENTER/VERIFY PARAMETERS BELOW:
JOB STATEMENT INFORMATION TO SUBMIT JOB:
==> //FOO40A JOB ,CLASS=A
==>
==>
==>

SYSOUT CLASS ==> A
REGION SIZE (K BYTE) ==> 512

PRESS ENTER OR PF3 TO GO FORWARD TO THE SUBMITTED PROCESS
PRESS PF4 TO TERMINATE THE DOCK SESSION
  
```

図 3.6 サブミット指定メニュー

環境保存の指定は、以下のものについて行う。

- ・利用者プログラムのオブジェクトモジュール
- ・画面表示のためのソースモジュール
- ・各種テーブル類

データセットを新規に作成する場合は、SPACE、UNIT等の指定を行う。

環境保存は FIB ジョブの形で行われるので、JOB 文入力のためのサブミット指定メニューが表示される。これに JOB 文を指定する。

以上の様に環境保存を行ったプログラムを再開するには、DOCKF77 コマンドにオペランド RES TART を指定すればよい。すると、次の環境保存データセット名指定メニューが表示されるので、それぞれのデータセット名を指定する。下線の部分が、図 3.5 に対応する指定である。

```

-----< DOCK/FORT77 INITIAL PROCESS KEEP DATA SET MENU >-----
ENTER/VERIFY PARAMETERS BELOW:

DATA SET FOR OBJECT KEEP :
DATA SET NAME ==> TEST.OBJ
VOLUME SERIAL ==> (IF NOT IN CATALOG)
PASSWORD ==> (IF PASSWORD PROTECTED)

DATA SET FOR SOURCE DISPLAY :
DATA SET NAME ==> TEST.SOURCE
VOLUME SERIAL ==> (IF NOT IN CATALOG)
PASSWORD ==> (IF PASSWORD PROTECTED)

DATA SET FOR TABLE KEEP :
DATA SET NAME ==> TEST.DATA
VOLUME SERIAL ==> (IF NOT IN CATALOG)
PASSWORD ==> (IF PASSWORD PROTECTED)

PRESS ENTER OR PF3 TO PROCEED THE RESTART PROCESS
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図 3.7 環境保存データセット名指定メニュー

↓ PF3/ENTER
実行画面へ

3.3 実行時のメニュー指定

デバッグ動作中に実行時データセットの割当て、実行モードの変更などを行いたい場合は、静止モードとなった時点で、PF2キー押下またはサブコマンド入力(図2.1参照)により、実行時のメニュー指定ができる。実行時のメニュー指定の例を以下に示す。下線部が入力した部分である。

```

EXECUTION SCREEN ARGCHK => N SUBCHK => N FREQ => N (Y/N) MODE => L S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
-----1-----2-----3-----4-----5-----6-----7-----8-----9-----10-----
00190 10 READ(1,100) N, EPS, AL, AR | MAIN |
00200 100 FORMAT(12,3F10.5) | |
00210 IF(N.GT.9) THEN | |
00220 WRITE(6,200) N | |
00230 200 FORMAT('----- N = ',I2, | |
00240 STOP | |
00250 ENDIF | |
00260 N1=N+1 | |
00270 READ(1,110) (A(I),I=1,N1) | |
00280 110 FORMAT(4E14.6) | |
00290 WRITE(6,205) N, EPS, AL, AR, | |
00300 205 FORMAT(1H , 'N = ',I2, | |
00310 1 , AR = ',F10.5/1 | |
00320 CALL TRANS(N,A,ASTAR,AL,A | |
00330 WRITE(6,210) N1, (ASTAR(I) | |
00340 210 FORMAT(1H , 'ASTAR(1)...AS | |
00350 DO 20 J=1,N1 | |
00360 B(J)=0.0 | |
00370 DO 20 I=J,N1,2 | |
00380 B(J)=B(J)+ASTAR(I)*X(

```

図 3.8 実行画面

↓ PF2

```

-----< DOCK/FORT77 ACTION SELECTION MENU ( EXECUTION SCREEN ) >-----
OPTION ==> 1
1 - MODE DEFINITION
2 - DATA MONITOR CONDITION DEFINITION
3 - ALLOCATE OR FREE DATA SET
4 - SAVE THE DOCK INITIAL PROCESS INFORMATION

PRESS PF3 TO RETURN TO THE EXECUTION SCREEN
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図 3.9 動作選択メニュー

⇓ ENTER

```

-----< DOCK/FORT77 MODE DEFINITION MENU >-----
==>
ENTER 'S', 'L', 'M' OR 'H' TO MODE FIELD. (DEFAULT MODE : LOW)
( S : STEP L : LOW M : MEDIUM H : HIGH )

MODE | PROGRAM NAME | KIND | NEXT MODE | CURRENT MODE
-----|-----|-----|-----|-----
S | MAIN | MAIN | L | 
 | NONIAL | FUNC | L | 
 | TRANS | SUB | L | 
*** END OF DATA ***

PRESS PF3 TO TERMINATE THIS MENU
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図 3.10 実行モード定義メニュー

⇓ PF3

```

EXECUTION SCREEN ARGCHK => N SUBCHK => N FREQ => N (Y/N) MODE => S S L M H
==> ALLOC
SCROLLS : ROW( P ) COL( 20 ) ROW( P ) COL( 20 ) VLINE : ( 42 )
-----1-----2-----3-1 MAIN (NEST)
00190 10 READ(1,100) N,EPS,AL,AR
00200 100 FORMAT(12,3F10.5)
00210 IF(N.GT.9) THEN
00220 WRITE(6,200) N
00230 200 FORMAT('----- N = ',12,
00240 STOP
00250 ENDOF
00260 N1=N+1
00270 READ(1,110) (A(I),I=1,N1)
00280 110 FORMAT(4E14.6)
00290 WRITE(6,205) N,EPS,AL,AR
00300 205 FORMAT(1H 'N = ',12,
00310 1 AR = ',F10.5/1
00320 CALL TRANS(N,A,ASTAR,AL,A
00330 WRITE(6,210) N1,(ASTAR(I)
00340 210 FORMAT(1H 'ASTAR(1)...AS
00350 DO 20 J=1,N1
00360 B(J)=0.0
00370 DO 20 I=J,N1,2
00380 B(J)=B(J)+ASTAR(I)*X(

```

図 3.11 実行画面

⇓ ENTER

PF2キー押下により、動作選択メニューが表示された。

このメニューで以下の動作を選択することができる。

- ・実行モードの定義と変更
- ・モニタ条件の定義と変更
- ・実行時データセットの割当てと解除
- ・環境の保存

この場合、番号1を指定した

実行モードの指定は、プログラム名の左欄に行サブコマンドS, L, M, Hのいずれかを入力する。

この場合、MAINプログラムの実行モードをLからSに変更した。

PF3キー押下により、もとの実行画面に戻る。

実行時データセットの割当てを行うために、サブコマンドALLOCを入力する。

```

-----< DOCK/FORT77 DATA SET ALLOCATION MENU FOR PROGRAM EXECUTION >-----
OPTION ==> 1
1 - ALLOCATE TERMINALS
2 - ALLOCATE SYSOUT DATA SETS
3 - ALLOCATE OLD DATA SETS
4 - ALLOCATE NEW DATA SETS
5 - ALLOCATION INFORMATION AND FREE DATA SET

PRESS PF3 TO RETURN TO THE EXECUTION SCREEN
PRESS PF4 TO TERMINATE THE DOCK SESSION
    
```

図 3.12 データセット割当て主メニュー

↓ ENTER

```

-----< DOCK/FORT77 ALLOCATE TERMINAL MENU >-----
ENTER DD NAMES AND RECORD LENGTH:
DD NAME      ==> FT01F001
              ==>
              ==>
RECORD FORMAT ==> (BLANK FOR DEFAULT)
RECORD LENGTH ==> (BLANK FOR DEFAULT)

PRESS ENTER TO REPEAT THIS MENU
PRESS PF3 TO RETURN TO THE SELECTION MENU OR THE EXECUTION SCREEN
PRESS PF4 TO TERMINATE THE DOCK SESSION
    
```

図 3.13 端末割当てメニュー

↓ PF3

実行画面へ

動的に割当てるデータセットの種別を指定する。以下の項目を指定することができる。

- ・ 端末装置の割当て
- ・ SYSOUT の割当て
- ・ 既存データセットの割当て
- ・ 新規データセットの割当て
- ・ 割当てられたデータセットの解除

この場合、番号 1 を指定した

DD 名 FT01F001 を端末に割当て、機番 1 で端末よりデータの入出力を行うことにした。

3.4 実行時エラー等のデバッグ／テスト

実行時エラー等のデバッグ／テストの一例を以下に示す。下線部が入力した部分である。まず初めに、プログラムの概要を述べておく。

[プログラムの概要]

このプログラムは、ある多項式（ベキ級数）をチェビシェフ節約法により、元より次数の下がった多項式で近似しようとするものである [6]。

$$1) f_n(x) = \sum_{i=0}^n a_i x^i, \quad \ell \leq x \leq r$$

区間 [ℓ , r] で定義された n 次の多項式 $f_n(x)$ の次数 N ，左端 AL ，右端 AR ， $N+1$ 個の係数 A （配列），節約の手続きにおける最大許容誤差 EPS を読み込む。

$$2) f_n(z) = \sum_{i=0}^n a_i^* z^i, \quad -1 \leq z \leq 1$$

サブルーチン TRANS により、区間を $[-1, 1]$ に変更し、その際変換された係数 a_i^* を配列 ASTAR に求める。この際必要となる 2 項係数を関数 NOMIAL で計算する。

$$3) f_n(z) = \sum_{i=0}^n b_i T_i(z)$$

$f_n(z)$ をチェビシェフ多項式 $T_i(z)$ を用いて展開し、その際の係数 b_i を配列 B に求める。この変換時、係数行列として配列 X (DATA 文で定義) を用いる。

$$4) f_n(z) \simeq f_m(z) = \sum_{i=0}^m b_i T_i(z)$$

区間 $[-1, 1]$ において、それらの和が $f_n(z)$ に許容誤差 EPS 以上の寄与をしない高次の多項式 $T_i(z)$ をすべて切り捨て、次数を m に切り下げる ($m \leq n$) …… チェビシェフ節約。

$$5) f_m(z) = \sum_{i=0}^m c_i^* z^i, \quad -1 \leq z \leq 1$$

節約された $f_m(z)$ をベキ級数に展開し、その際の係数 c_i^* を配列 CSTAR に求める。この変換時、係数行列として配列 T (DATA 文で定義) を用いる。

$$6) f_m(x) = \sum_{i=0}^m c_i x^i, \quad \ell \leq x \leq r$$

サブルーチン TRANS により、2) と逆に区間を $[\ell, r]$ に戻し、節約された結果の係数 c_i を配列 C に求める。

なお、この例では 3 次の多項式 $f(x) = x^3$, $1 \leq x \leq 3$ を EPS = 3.26 でチェビシェフ節約し 1 次式 $f(x) = -14.5 + 12.75x$, $1 \leq x \leq 3$ を得ている。

[デバッグ概要]

プログラムを実行させると、エラーが発生する (図 3.17 参照)。エラーの原因が不明なので、DOCK/FORT の ARGCHK, SUBCHK により、原因を究明していく。また、FREQ により実行回数を表示させ、実行モードをステップモードとし、プログラムの流れを 1 ステップずつ追っていく。再実行させると、B の配列で SUBCHK エラーが発生する。ISUB の値をスナップ文で表示させると、ISUB の値に誤りがあることが分かる。ISUB をスナップ文の一時代入で値を修正し、モニタ条件で ISUB の値を監視する。プログラムの途中から、行サブコマンド GO により再実行する。実行を続けていくと、サブルーチン TRANS で ARGCHK エラーが発生する。これは、行番号 600 (図 3.16 参照) で、サブルーチンを CALL した時に、引数の対応がとれていないために発生した。そのために、プログラムの実行を強制的に終らせ (PF3 キー押下)、EDIT-FSO でプログラムを修正し、再実行を行う。再実行の結果を検討すると、正しいので終結処理を行い、デバッグ完了とする。

[デバッグ例]

ソース表示域行番号の下線は、現在実行中の箇所を示す。

```

EXECUTION SCREEN ARGCHK => N SUBCHK => N FREQ => N (Y/N) MODE => L S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
-----1-----2-----3-----
00010 C IMPLICIT REAL*8(A-H,O-Z)
00020 DIMENSION A(10),ASTAR(10)
00030 1 DATA X / 1.0, 0.0, 0.5,
00040 1 0.2734375, 2*0.0, 1.0, 0
00050 2 0.4921875, 2*0.0, 0.5, 0
00060 3 4*0.0, 0.25, 0.0, 0.3125
00070 4 0.125, 0.0, 0.1875, 0.0,
00080 5 0.0, 0.140625, 6*0.0, 0.
00090 6 0.03515625, 8*0.0, 0.007
00100 DATA T / 1.0, 0.0, -1.0,
00110 1 2*0.0, 1.0, 0.0, -3.0, 0
00120 2 2.0, 0.0, -8.0, 0.0, 18.
00130 3 0.0, 56.0, 0.0, -120.0,
00140 4 6*0.0, 16.0, 0.0, -112.0
00150 5 8*0.0, 64.0, 0.0, -576.0
00160 DATA ONE,ONEM / 1.0, -1.0
00170 10 READ(5,100) N,EPS,AL,AR
00180 100 FORMAT(I2,3F10.5)
00190
00200

```

図 3.14 実行画面(1)

⇓ PF5

```

EXECUTION SCREEN ARGCHK => N SUBCHK => N FREQ => N (Y/N) MODE => S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
JZV3111 PLEASE, INPUT DATA. EXECUTION MODE IS NOT HIGH
00190 100 READ(5,100) N,EPS,AL,AR IAT 190 PLEASE, INPUT DATA. EXECUTION M
00200 100 FORMAT(I2,3F10.5)
00210 IF(N.GT.9) THEN
00220 WRITE(6,200) N
00230 200 FORMAT('----- N = ',I2,
00240 STOP
00250 ENDOF
00260 M1=M+1
00270 READ(5,110) (A(I),I=1,M1)
00280 110 FORMAT(4E14.6)
00290 WRITE(6,205) N,EPS,AL,AR
00300 205 FORMAT(1H,'N = ',I2,
00310 1 AR = ',F10.5/1
00320 CALL TRANS(N,A,ASTAR,AL,A
00330 WRITE(6,210) M1,(ASTAR(I)
00340 210 FORMAT(1H,'ASTAR(1)...AS
00350 DO 20 J=1,M1
00360 B(J)=0.0
-----1-----2-----3-----4-----5-----6-----7-----
3 3.26 1.0 3.0

```

図 3.15 実行画面(2)

⇓ PF5

```

EXECUTION SCREEN ARGCHK => N SUBCHK => N FREQ => N (Y/N) MODE => S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
-----1-----2-----3-----4-----5-----6-----7-----
00450 DO 30 I=1,N IAT 190 PLEASE, INPUT DATA. EXECUTION M
00460 ISUB=M1+1-1 I 3 3.26 1.0 3.0
00470 IF(EMAX*DABS(B(ISUB)).G IAT 270 PLEASE, INPUT DATA. EXECUTION M
00480 M1=M1-1 I 0.0 0.0
00490 EMAX=EMAX+DABS(B(ISUB)) IN = 3 EPS = 3.26000 AL = 1.
00500 30 CONTINUE I A(1)...A( 4)
00510 40 M=M1-1 I 0.0 0.0 0.0
00520 DO 50 J=1,M1 IASTAR(1)...ASTAR( 4)
00530 CSTAR(J)=0.0 I 0.8000000+01 0.1200000+02 0.600
00540 DO 50 I=J,M1,2 I B(1)...B( 4)
00550 CSTAR(J)=CSTAR(J)+B(I)* I 0.1100000+02 0.1275000+02 0.300
00560 50 CONTINUE I CSTAR(1)...CSTAR( 1)
00570 C.... CONTINUE I 0.1100000+02
00580 WRITE(6,230) M1,CSTAR(1)
00590 230 FORMAT(1H,'CSTAR(1)...CS
00600 CALL TRANS(CSTAR,C,ONEM,O
00610 WRITE(6,240) M1,C(I),I=1
00620 240 FORMAT(1H,'C(1)...C(I),I2
00630 GO TO 10
00640 END

```

図 3.16 実行画面(3)

⇓ PF5

デバッグ開始

(静止モード)

静止モードを解除して、実行を開始するためには、PF5キーを押下する。

プログラムからの入力要求

(READ文)があると、静止モードとなり、画面の24行目が入力域となる。データを入力して、PF5キーを押下すれば、静止モードは解除される。

行番号600でサブルーチンTRANSを呼出している。

```

EXECUTION SCREEN ARGCHK => N SUBCHK => N FREQ => N (Y/N) MODE => S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
JZV3161 EXECUTION ERROR OCCURRED. EXECUTION MODE IS NOT HIGH
00650 SUBROUTINE TRANS(N,COEFFI IA(1)...A( 4 )
00660 IMPLICIT REAL*8(A-H,O-Z) I 0.0 0.0 0.0
00670 REAL*8 COEFFI,COEFFT,ENDL IASTAR(1)...ASTAR( 4 )
00680 DIMENSION COEFFI(10),COEF I 0.800000D+01 0.120000D+02 0.600
00690 CON1=(ENDRI-ENL1)/(ENDRT I B(1)...B( 4 )
00700 CON2=(ENDLI-ENDRT-ENDR1=E I 0.110000D+02 0.127500D+02 0.300
00710 NP1=N+1 ICSTAR(1)...CSTAR( 1 )
00720 DO 10 I=1,NP1 I 0.110000D+02
00730 IF(CON2.NE.0.0) THEN IJZL2071-E FLOATING OVERFLOW OLD PSW=07
00740 COEFFT(I)=0.0 I INSTRUCTION=6A00D2B8 LOCAT1
00750 DO 15 J=1,NP1 I GR0 =00000000 GR1 =89A00262 GR2 =0
00760 BINOM=NOMIAL(J-1,I- I GR4 =41B00000 GR5 =00000008 GR6 =0
00770 COEFFT(I)=COEFFT(I) I GR8 =0012F570 GR9 =0010084C GR10=4
00780 15 CONTINUE I GR12=000FDE70 GR13=001310A0 GR14=6
00790 ELSE I FR0=801FFFFF7BF7C3D FR2=41
00800 COEFFT(I)=COEFFI(I) I FR4=4110000000000000 FR6=4E
00810 ENDOF I ERROR OCCURS AT TRANS LOC 0013165
00820 COEFFT(I)=COEFFT(I)*CON1 I TRANS AT LOC 00130EAB CALLED FROM
00830 10 CONTINUE I MAIN AT LOC 4012EFE2 CALLED FROM
00840 RETURN IAT 770 EXECUTION ERROR OCCURRED. EXEC

```

図 3.17 実行画面(4)

↓ PF3

```

-----< DOCK/FORT77 ACTION SELECTION MENU >-----
OPTION ==>

1 - PERFORM THE SOURCE MODULE DATA SET SPECIFICATION
2 - PERFORM THE INCLUDED SOURCE MODULE DATA SET SPECIFICATION
3 - PERFORM THE OBJECT/LOAD MODULE DATA SET SPECIFICATION
4 - PERFORM THE AUTOMATIC CALL LIBRARY DATA SET SPECIFICATION
5 - PERFORM THE COMPILER OPTIONS SPECIFICATION
6 - PERFORM THE EXECUTION OPTIONS AND ENTRY NAME SPECIFICATION
7 - PERFORM A DATA SET EDITING

```

PRESS PF3 TO RESTART THE PROGRAM EXECUTION PROCESS
PRESS PF4 TO TERMINATE THE DOCK SESSION

図 3.18 動作選択メニュー

↓ PF3

```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
-----1-----2-----3- I MAIN (NEST)
00190 10 READ(5,100) N,EPS,AL,AR I A(1)...A( 4 )
00200 100 FORMAT(I2,3F10.5) I 0.0 0.0 0.0
00210 IF(N.GT.9) THEN I IASTAR(1)...ASTAR( 4 )
00220 WRITE(6,200) N I 0.800000D+01 0.120000D+02 0.600
00230 200 FORMAT('----- N = ',I2, I B(1)...B( 4 )
00240 STOP I 0.110000D+02 0.127500D+02 0.300
00250 ENDOF ICSTAR(1)...CSTAR( 1 )
00260 N1=N+1 I 0.110000D+02
00270 READ(5,110) (A(I),I=1,N1) IJZL2071-E FLOATING OVERFLOW OLD PSW=07
00280 110 FORMAT(4E14.6) I INSTRUCTION=6A00D2B8 LOCAT1
00290 WRITE(6,205) N,EPS,AL,AR, I GR0 =00000000 GR1 =89A00262 GR2 =0
00300 205 FORMAT('H ',N = ',I2, I GR4 =41B00000 GR5 =00000008 GR6 =0
00310 1 AR = ',F10.5/I GR8 =0012F570 GR9 =0010084C GR10=4
00320 CALL TRANS(N,A,ASTAR,AL,A I GR12=000FDE70 GR13=001310A0 GR14=6
00330 WRITE(6,210) N1,(ASTAR(I) I FR0=801FFFFF7BF7C3D FR2=41
00340 210 FORMAT('H ',ASTAR(1)...AS I FR4=4110000000000000 FR6=4E
00350 DO 20 J=1,N1 I ERROR OCCURS AT TRANS LOC 0013165
00360 B(J)=0.0 I TRANS AT LOC 00130EAB CALLED FROM
00370 DO 20 I=J,N1,2 I MAIN AT LOC 4012EFE2 CALLED FROM
00380 B(J)=B(J)+ASTAR(I)*X IAT 770 EXECUTION ERROR OCCURRED. EXEC

```

図 3.19 実行画面(5)

↓ PF5

エラーが発生すると、メッセージが出力され、静止モードとなる。データ表示域の最後の行に行番号 770 でエラーが発生したことが表示されている。

プログラムを終了させるため PF3 キーを押下する。

PF3 キー押下により、動作選択メニューが表示される。プログラムの最初から再実行させたいので、PF3 キーを押下する。

再実行の最初の実行画面でデバッグのため、以下の項目の指定をする。

ARGCHK => Y

SUBCHK => Y

FREQ => Y

MODE => S


```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S L M H
=> WRITE ISUB
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
JZV2921 SUBCHK ERROR DETECTED (B)
1 00350 DO 20 J=1,N1 I GR8=0012F570 GR9=0010084C GR10=4
4 00360 B(J)=0.0 I GR12=000F0E70 GR13=001310A0 GR14=6
4 00370 DO 20 I=J,N1,2 I FR0=801FFFFF7B7C3D FR2=41
6 00380 B(J)=B(J)+ASTAR(I) I FR4=4110000000000000 FR6=4E
6 00390 20 CONTINUE I ERROR OCCURS AT TRANS LOC 0013165
00400 C.... CONTINUE I TRANS AT LOC 00130EAB CALLED FROM
1 00410 WRITE(6,220) N1,B(1), I MAIN AT LOC 4012FE2 CALLED FROM
00420 220 FORMAT(1H,'B(1)....B( I AT 770 EXECUTION ERROR OCCURRED, EXECU
1 00430 EMAX=0.0 IAT 190 PLEASE, INPUT DATA, EXECUTION M
1 00440 M1=N1 I 3 3.26 1.0 3.0
1 00450 DO 30 I=1,N IAT 270 PLEASE, INPUT DATA, EXECUTION M
1 00460 ISUB=N1+1-I I 0.0 0.0
1 00470 IF(EMAX+DABS(B(ISUB) IN = 3 EPS = 3.26000 AL = 1.
0 00480 M1=N1-I I A(1)....A( 4) 0.0 0.0
0 00490 EMAX=EMAX+DABS(B(ISU I 0.0 0.0
0 00500 30 CONTINUE IASTAR(1)....ASTAR( 4) 0.0
0 00510 40 M=M1-1 I 0.800000D+01 0.120000D+02 0.600
0 00520 DO 50 J=1,M1 I B(1)....B( 4) 0.0
0 00530 CSTAR(J)=0.0 I 0.110000D+02 0.127500D+02 0.300
0 00540 DO 50 I=J,M1,2 IAT 470 SUBCHK ERROR DETECTED

```

図 3.20 実行画面 (6)

⇓ ENTER

```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S L M H
=> M1=N1
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
COUNT -----1-----2----- I MAIN COL( 20 ) VLINE : (NEST)
1 00350 DO 20 J=1,N1 I FR4=4110000000000000 FR6=4E
4 00360 B(J)=0.0 I ERROR OCCURS AT TRANS LOC 0013165
4 00370 DO 20 I=J,N1,2 I TRANS AT LOC 00130EAB CALLED FROM
6 00380 B(J)=B(J)+ASTAR(I) I MAIN AT LOC 4012FE2 CALLED FROM
6 00390 20 CONTINUE IAT 770 EXECUTION ERROR OCCURRED, EXECU
00400 C.... CONTINUE IAT 190 PLEASE, INPUT DATA, EXECUTION M
1 00410 WRITE(6,220) N1,B(1), I 3 3.26 1.0 3.0
00420 220 FORMAT(1H,'B(1)....B( IAT 270 PLEASE, INPUT DATA, EXECUTION M
1 00430 EMAX=0.0 I 0.0 0.0
1 00440 M1=N1 IN = 3 EPS = 3.26000 AL = 1.
1 00450 DO 30 I=1,N I A(1)....A( 4) 0.0 0.0
1 00460 ISUB=N1+1-I I 0.0 0.0 0.0
1 00470 IF(EMAX+DABS(B(ISUB) IASTAR(1)....ASTAR( 4) 0.0
0 00480 M1=N1-I I 0.800000D+01 0.120000D+02 0.600
0 00490 EMAX=EMAX+DABS(B(ISU I B(1)....B( 4) 0.0
0 00500 30 CONTINUE I 0.110000D+02 0.127500D+02 0.300
0 00510 40 M=M1-1 IAT 470 SUBCHK ERROR DETECTED
0 00520 DO 50 J=1,M1 I ISUB= 0
0 00530 CSTAR(J)=0.0 IAT 470 SNAP STATEMENT EXECUTED (WRITE)
0 00540 DO 50 I=J,M1,2 I <WRITE ISUB>

```

図 3.21 実行画面 (7)

⇓ ENTER

```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S L M H
=> COND =
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
COUNT -----1-----2----- I MAIN COL( 20 ) VLINE : (NEST)
1 00350 DO 20 J=1,N1 I TRANS AT LOC 00130EAB CALLED FROM
4 00360 B(J)=0.0 I MAIN AT LOC 4012FE2 CALLED FROM
4 00370 DO 20 I=J,N1,2 IAT 770 EXECUTION ERROR OCCURRED, EXECU
6 00380 B(J)=B(J)+ASTAR(I) IAT 190 PLEASE, INPUT DATA, EXECUTION M
6 00390 20 CONTINUE I 3 3.26 1.0 3.0
00400 C.... CONTINUE IAT 270 PLEASE, INPUT DATA, EXECUTION M
1 00410 WRITE(6,220) N1,B(1), I 0.0 0.0
00420 220 FORMAT(1H,'B(1)....B( IN = 3 EPS = 3.26000 AL = 1.
1 00430 EMAX=0.0 I A(1)....A( 4) 0.0 0.0
1 00440 M1=N1 I 0.0 0.0 0.0
1 00450 DO 30 I=1,N IASTAR(1)....ASTAR( 4) 0.0
1 00460 ISUB=N1+1-I I 0.800000D+01 0.120000D+02 0.600
1 00470 IF(EMAX+DABS(B(ISUB) I B(1)....B( 4) 0.0
0 00480 M1=N1-I I 0.110000D+02 0.127500D+02 0.300
0 00490 EMAX=EMAX+DABS(B(ISU I ISUB= 0
0 00500 30 CONTINUE IAT 470 SUBCHK ERROR DETECTED
0 00510 40 M=M1-1 IAT 470 SNAP STATEMENT EXECUTED (WRITE)
0 00520 DO 50 J=1,M1 I <WRITE ISUB>
0 00530 CSTAR(J)=0.0 IAT 470 SNAP STATEMENT EXECUTED (ASSIGN
0 00540 DO 50 I=J,M1,2 I <N1=N1>

```

図 3.22 実行画面 (8)

⇓ ENTER

4 行目にメッセージが出力され、配列 B で SUBCHK エラーが発生し、データ表示域のメッセージにより、行番号 470 で SUBCHK エラーが起きたことが分かる。配列 B の添字である ISUB の値をスナップ文の WRITE 文で書き出すことにする。行番号の左に実行回数が表示されている。

データ表示域に ISUB の値が書き出されている。添字の ISUB の値が 0 なので、エラーが発生したことが分かった。行番号 460 で ISUB の計算式で、N1 を NI とコーディングしているためにエラーが発生していることが分かった。スナップ文で値の一時代入を行う。

ISUB の値を監視したいので、モニタ条件の定義を行う。そのために、COND サブコマンドを入力する。オペランド * は、現在のプログラム単位に対して行うことを意味する。モニタ条件の定義を以下のメニューで行う。

```

-----< DOCK/FORT77 DATA MONITOR DEFINITION MENU >-----
===>
1-----2-----3-----4-----5-----6-----7-----
00470 IF(EMAX+DABS(B(ISUB)).GT.EPS) GOTO 40
00480 M1=N1-1
00490 EMAX=EMAX+DABS(B(ISUB))
00500 30 CONTINUE
00510 40 M=N1-1
00520 DO 50 J=1,M1
00530 CSTAR(J)=0.0
00540 DO 50 I=J,M1*2
00550 CSTAR(J)=CSTAR(J)+B(I)*T(I,J)
00560 50 CONTINUE
-----
STATUS | NAME | CONDITIONS ( STATUS : M - MONITOR, T - TRACE, S - SUSPEND )
-----
M | CHANGE | ISUB
-----
PRESS PF3 TO TERMINATE THIS MENU, PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図 3.23 モニタ条件定義メニュー

↓ PF3

```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
COUNT 1-----2-----3-----4-----5-----6-----7-----
1 00350 DO 20 J=1,M1 I MAIN
4 00360 B(J)=0.0 I TRANS AT LOC 00130EAB CALLED FROM
4 00370 DO 20 I=J,M1*2 I MAIN AT LOC 4012FE2 CALLED FROM
6 00380 B(J)=B(J)+ASTAR(I) IAT 770 EXECUTION ERROR OCCURRED. EXECU
6 00390 20 CONTINUE IAT 190 PLEASE, INPUT DATA. EXECUTION M
00400 C.... CONTINUE IAT 270 PLEASE, INPUT DATA. EXECUTION M
1 00410 WRITE(6,220) M1,B(I), I 0.0 0.0
00420 220 FORMAT(1H,'B(1)....B( I IN = 3 EPS = 3.26000 AL = 1.
1 00430 EMAX=0.0 I A(1)...A( 4) 0.0 0.0
1 00440 M1=N1 I IASTAR(1)...ASTAR( 4) 0.0 0.0
1 00450 DO 30 I=1,M I 0.800000D+01 0.120000D+02 0.600
1 00460 ISUB=N1+1-I I 0.110000D+02 0.127500D+02 0.300
1 00470 IF(EMAX+DABS(B(ISUB) IB(1)...B( 4) 0.0 0.0
0 00480 M1=N1-1 I 0.110000D+02 0.127500D+02 0.300
0 00490 EMAX=EMAX+DABS(B(ISU IAT 470 SUBCHK ERROR DETECTED
0 00500 30 CONTINUE I IISUB= 0
0 00510 40 M=N1-1 IAT 470 SNAP STATEMENT EXECUTED (WRITE)
0 00520 DO 50 J=1,M1 I <WRITE ISUB>
0 00530 CSTAR(J)=0.0 IAT 470 SNAP STATEMENT EXECUTED (ASSIGN)
0 00540 DO 50 I=J,M1*2 I <N1=N1>

```

図 3.24 実行画面(9)

↓ PF5

```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S L M H
==> TRACE CHANGE
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
JZV256I ENTER STATIONARY MODE UNDER DEFINED CONDITION
COUNT 1-----2-----3-----4-----5-----6-----7-----
1 00350 DO 20 J=1,M1 IAT 770 EXECUTION ERROR OCCURRED. EXECU
4 00360 B(J)=0.0 IAT 190 PLEASE, INPUT DATA. EXECUTION M
4 00370 DO 20 I=J,M1*2 I 3 3.26 1.0 3.0
6 00380 B(J)=B(J)+ASTAR(I) IAT 270 PLEASE, INPUT DATA. EXECUTION M
6 00390 20 CONTINUE I IN = 3 EPS = 3.26000 AL = 1.
00400 C.... CONTINUE I A(1)...A( 4) 0.0 0.0
1 00410 WRITE(6,220) M1,B(I), I 0.0 0.0
00420 220 FORMAT(1H,'B(1)....B( I 0.800000D+01 0.120000D+02 0.600
1 00430 EMAX=0.0 I IASTAR(1)...ASTAR( 4) 0.0 0.0
1 00440 M1=N1 I IB(1)...B( 4) 0.110000D+02 0.127500D+02 0.300
1 00450 DO 30 I=1,M I 0.110000D+02 0.127500D+02 0.300
1 00460 ISUB=N1+1-I I 0.110000D+02 0.127500D+02 0.300
1 00470 IF(EMAX+DABS(B(ISUB) I IISUB= 0
0 00480 M1=N1-1 IAT 470 SNAP STATEMENT EXECUTED (WRITE)
0 00490 EMAX=EMAX+DABS(B(ISU I <WRITE ISUB>
0 00500 30 CONTINUE IAT 470 SNAP STATEMENT EXECUTED (ASSIGN)
0 00510 40 M=N1-1 I I <N1=N1>
0 00520 DO 50 J=1,M1 IAT 460 CONDITION 'CHANGE' OCCURRED
0 00530 CSTAR(J)=0.0 I IISUB= 4
0 00540 DO 50 I=J,M1*2

```

図 3.25 実行画面(10)

↓ PF5

ISUB は、変数名である。

NAME 欄の ▼ CHANGE ▼ は、条件識別名である。STATUS 欄の ▼ M ▼ は、ISUB の値が変更されたら通知をして、静止モードとなる。▼ T ▼ は、静止モードとならず値の表示を行う。▼ S ▼ は、モニタ条件の定義を無効にする。

行サブコマンド GO により、行番号 450 より再実行する。

モニタ条件が成立すると、その旨が通知され、静止モードとなる。TRACE サブコマンドを入力して、これ以降モニタ条件が成立しても、静止モードとならず、値の表示を行うだけにする。

```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
J2V2911 ARGCHK ERROR DETECTED.
00650 SUBROUTINE TRANS(N,CDE IB(1)...B( 4 )
00660 IMPLICIT REAL*8(A-H,O- I 0.110000D+02 0.127500D+02 0.300
00670 REAL*8 COEFF1,COEFF2,E IAT 470 SUBCHK ERROR DETECTED
00680 DIMENSION COEFF1(10),C IISUB= 0
1 00690 CON1=(ENDRI-ENL1)/(CEN IAT 470 SNAP STATEMENT EXECUTED (WRITE)
1 00700 CON2=(ENDL1-ENDRI-ENDR I<WRITE IISUB>
1 00710 NP1=N+1 IAT 470 SNAP STATEMENT EXECUTED (ASSIGN
1 00720 DO 10 I=1,NP1 I<N1=N1>
4 00730 IF(CON2.NE.0.0) THEN IAT 460 CONDITION 'CHANGE' OCCURRED
4 00740 COEFF1(I)=0.0 IISUB= 4
4 00750 DO 15 J=1,NP1 IAT 460 CONDITION 'CHANGE' OCCURRED
10 00760 BINOM=NOMIAL(J-1 IISUB= 3
10 00770 COEFF1(I)=COEFF1 IAT 460 CONDITION 'CHANGE' OCCURRED
10 00780 15 CONTINUE IISUB= 2
0 00790 ELSE ICSTAR(1)...CSTAR( 2)
0 00800 COEFF1(I)=COEFF1(I I 0.110000D+02 0.127500D+02
4 00810 ENDF IAT 650 ARGUMENT NUMBER NOT COINCIDED
4 00820 COEFF1(I)=COEFF1(I)*C0 IAT 650 ARGUMENT CLASS NOT COINCIDED
4 00830 10 CONTINUE IAT 650 ARGUMENT TYPE NOT COINCIDED
1 00840 RETURN IAT 650 ARGUMENT CLASS NOT COINCIDED

```

図 3.26 実行画面(11)

⇓ PF3

```

-----< DOCK/FORT77 ACTION SELECTION MENU >-----
OPTION ==> 7

1 - PERFORM THE SOURCE MODULE DATA SET SPECIFICATION
2 - PERFORM THE INCLUDED SOURCE MODULE DATA SET SPECIFICATION
3 - PERFORM THE OBJECT/LOAD MODULE DATA SET SPECIFICATION
4 - PERFORM THE AUTOMATIC CALL LIBRARY DATA SET SPECIFICATION
5 - PERFORM THE COMPILER OPTIONS SPECIFICATION
6 - PERFORM THE EXECUTION OPTIONS AND ENTRY NAME SPECIFICATION
7 - PERFORM A DATA SET EDITING

PRESS PF3 TO RESTART THE PROGRAM EXECUTION PROCESS
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図 3.27 動作選択メニュー

⇓ ENTER

```

-----< DOCK/FORT77 EDIT DATA SET MENU >-----
ENTER/VERIFY PARAMETERS BELOW:

DOCK LIBRARY DATA SET:
PROJECT ==> F0040
LIBRARY ==> TEST
TYPE ==> FORT77
MEMBER ==> EX2

OTHER PERTITIONED OR SEQUENTIAL DATA SET:
DATA SET NAME ==>

DATA SET PASSWORD ==> (IF PASSWORD PROTECTED)
DATA SET TYPE (FIXED/FREE) ==> FIXED
NEW/OLD ==> OLD

PRESS ENTER TO GO FORWARD TO THE EDIT PROCESS
PRESS PF3 TO RETURN TO THE ACTION SELECTION MENU
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図 3.28 エディットデータセット指定メニュー

⇓

ISUBの値の変化がデータ表示域に表示されている。

行番号650でARGCHKエラーが発生したことが、4行目とデータ表示域に通知されている。行番号600(図3.16参照)で、引数の指定に誤りがあることが分かった。PF3キーの押下により、プログラムを終了させる。

プログラムに誤りがあることが分かったので、それを修正するため、項目番号7を選択する。

修正すべきプログラムを指定する。ENTERを押下すると、EDIT-FSOの画面となるので、サブルーチンの引数と、以前のISUBの計算式の修正を行う。

```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
-----1-----2-----3-----4-----5-----6-----7-----8-----9-----10-----11-----12-----13-----14-----15-----16-----17-----18-----19-----20-----21-----22-----23-----24-----25-----26-----27-----28-----29-----30-----31-----32-----33-----34-----35-----36-----37-----38-----39-----40-----41-----42-----
00190 10 READ(5,100) N, EPS, AL, AR, I B(1)...B( 4) (NEST)
00200 100 FORMAT(12,3F10.5) IAT 0.110000D+02 0.127500D+02 0.300
00210 IF(N.GT.9) THEN IAT 470 SUBCHK ERROR DETECTED
00220 WRITE(6,200) N IISUB= 0
00230 200 FORMAT('----- N = ',12, IAT 470 SNAP STATEMENT EXECUTED (WRITE)
00240 STOP I <WRITE IISUB>
00250 ENDOF IAT 470 SNAP STATEMENT EXECUTED (ASSIGN
00260 N1=N+1 I <N1=N1>
00270 READ(5,110) (A(I),I=1,N1) IAT 460 CONDITION 'CHANGE' OCCURRED
00280 110 FORMAT(4E14.6) IISUB= 4
00290 WRITE(6,205) N, EPS, AL, AR, I IAT 460 CONDITION 'CHANGE' OCCURRED
00300 205 FORMAT(1H, 'N = ',12, IISUB= 3
00310 1 ' AR = ',F10.5/1 IAT 460 CONDITION 'CHANGE' OCCURRED
00320 CALL TRANS(N,A,ASTAR,AL,A IISUB= 2
00330 WRITE(6,210) N1,(ASTAR(I) ICSTAR(1)...ICSTAR( 2)
00340 210 FORMAT(1H, 'ASTAR(1)...AS I 0.110000D+02 0.127500D+02
00350 DO 20 J=1,N1 IAT 650 ARGUMENT NUMBER NOT COINCIDED
00360 B(J)=0.0 IAT 650 ARGUMENT CLASS NOT COINCIDED
00370 DO 20 J=1,N1,2 IAT 650 ARGUMENT TYPE NOT COINCIDED
00380 B(J)=B(J)+ASTAR(1)*X IAT 650 ARGUMENT CLASS NOT COINCIDED

```

図3.29 実行画面(12)

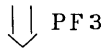


```

EXECUTION SCREEN ARGCHK => Y SUBCHK => Y FREQ => Y (Y/N) MODE => S L M H
==>
SCROLLS : ROW( P ) COL( 20 ) , ROW( P ) COL( 20 ) VLINE : ( 42 )
JZV2531 USER PROGRAM TERMINATED NORMALLY. RETURN CODE= 0
2 00190 10 READ(5,100) N, EPS, AL, A IAT 650 ARGUMENT CLASS NOT COINCIDED
00200 100 FORMAT(12,3F10.5) IAT 190 PLEASE, INPUT DATA, EXECUTION M
2 00210 IF(N.GT.9) THEN I 3 3.26 1.0 3.0
1 00220 100220 WRITE(6,200) N IAT 270 PLEASE, INPUT DATA, EXECUTION M
00230 200 FORMAT('----- N = ', 10.0 0.0
1 00240 STOP IN = 3 EPS = 3.26000 AL = 1.
1 00250 ENDOF I A(1)...A( 4)
1 00260 N1=N+1 I 0.0 0.0 0.0
1 00270 READ(5,110) (A(I),I=1, IASTAR(1)...ASTAR( 4)
00280 110 FORMAT(4E14.6) I 0.800000D+01 0.120000D+02 0.600
1 00290 WRITE(6,205) N, EPS, AL, I B(1)...B( 4)
00300 205 FORMAT(1H, 'N = ',12, I 0.110000D+02 0.127500D+02 0.300
00310 1 ' AR = ',F10. ICSTAR(1)...ICSTAR( 2)
1 00320 CALL TRANS(N,A,ASTAR,A I 0.110000D+02 0.127500D+02
1 00330 WRITE(6,210) N1,(ASTAR(I) IC(1)...C( 2)
00340 210 FORMAT(1H, 'ASTAR(1)... I -0.145000D+02 0.127500D+02
1 00350 DO 20 J=1,N1 IAT 190 PLEASE, INPUT DATA, EXECUTION M
4 00360 B(J)=0.0 I10
4 00370 DO 20 J=1,N1,2 I----- N = 10 PROGRAM STOP -----
6 00380 B(J)=B(J)+ASTAR(1) IJZV2531 USER PROGRAM TERMINATED NORMAL

```

図3.30 実行画面(13)



```

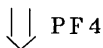
-----< DOCK/FORT77 ACTION SELECTION MENU >-----
OPTION ==>

1 - PERFORM THE SOURCE MODULE DATA SET SPECIFICATION
2 - PERFORM THE INCLUDED SOURCE MODULE DATA SET SPECIFICATION
3 - PERFORM THE OBJECT/LOAD MODULE DATA SET SPECIFICATION
4 - PERFORM THE AUTOMATIC CALL LIBRARY DATA SET SPECIFICATION
5 - PERFORM THE COMPILER OPTIONS SPECIFICATION
6 - PERFORM THE EXECUTION OPTIONS AND ENTRY NAME SPECIFICATION
7 - PERFORM A DATA SET EDITING

PRESS PF3 TO RESTART THE PROGRAM EXECUTION PROCESS
PRESS PF4 TO TERMINATE THE DOCK SESSION

```

図3.31 動作選択メニュー



最初からプログラムの再実行を行う。図3.19と同様に、以下の項目の指定を行う。

ARGCHK => Y
SUBCHK => Y
FREQ => Y
MODE => S

今度は正常に動作した。

プログラムを終了させるために、Nに10を入力すると、STOP文の前で静止モードとなる。PF5キーを押下すると、リターンコードが表示される。

終結処理を行いたいのので、PF4キーを押下する。

```

-----< DOCK/FORT77 TERMINATION MENU >-----
ENTER/VERIFY DISPOSITION OF LOG DATA SET FOR THIS SESSION:
+-----+
| PARAMETERS | LOG DATA SET |
+-----+
| PROCESS OPTION | ==> 下 |
| SYSOUT CLASS | ==> |
+-----+

VALID PROCESS OPTION:
J - SUBMIT JOB TO PRINT (AND DELETE)
L - LISTING TO PRINTER (AND DELETE)
D - DELETE DATA SET WITHOUT PRINTING

JOB STATEMENT INFORMATION TO SUBMIT JOB
==> //FOO40X JOB ██████,CLASS=A
==>
==>

PRESS ENTER OR PF3 TO GO FORWARD TO THE LOG DATA SET PROCESS
    
```

ログ情報を出力させるために
指定を行う。

図 3.32 終結処理メニュー

参考文献

1. 牛島, 河村 Fortran プログラム テスト/デバッグ支援のためのシステム — Forprex の使用について — , 九大大型計算機センター広報, 11, 4, 1978, 270—283.
2. 宇都宮, 樽美, 荒牧, 牛島 TSS 環境におけるプログラム実行演示システムの作成とその検討, 情報処理学会論文誌, 23, 1, 1982, 25—34.
3. 計算機マニュアル FACOM OS IV/F4 DOCK/FORTTRAN77使用手引書(64SP-3680—2), 富士通㈱
4. 南, 武富, 遠矢, 景川 TSS PFDの使用について, 九大大型計算機センター広報, 15, 1, 1982, 21—80.
5. 牛島, 藤村 M-190 OS IV/F4 FORDAPシステム, 九大大型計算機センター広報, 11, 1, 1978, 29—33.
6. 藤田他訳, 「計算機による数値計算法」, 日本コンピュータ協会, 1982.

付録 1. 静止モードへ移行する文〔富士通マニュアル〔 3 〕より転載〕

大 分 類	小 分 類	必ず静止 する	文番号があれ ば静止する	大 分 類	小 分 類	必ず静止 する	文番号があれ ば静止する
代 入 文	算術代入文		○	入 出 力 文	リスト指示WRITE文		○
	論理代入文		○		ネームリストREAD文**		○
	ASSIGN文		○		ネームリストWRITE文		○
	文字代入文		○		非同期READ文		○
	多重代入文		○		非同期WRITE文		○
	16進代入文		○		WAIT文		○
	互換用文字代入文		○		内部データセットREAD文		○
制 御 文	単純GOTO文		○		内部データセットWRITE文		○
	計算形GOTO文		○		書式付直接READ文		○
	割当て形GOTO文		○		書式付き直接WRITE文		○
	算術IF文		○		書式無し直接READ文		○
	論理IF文		○		書式無し直接WRITE文		○
	ブロックIF文	○			FIND文		○
	ELSE IF文	○			OPEN文		○
	ELSE文				CLOSE文		○
	END IF文				INQUIRE文		○
	DO文	○			BACKSPACE文		○
	CONTINUE文		○		ENDFILE文		○
	PAUSE文	○			REWIND文		○
	STOP文*	○			ENCODE文		○
	END文	○			DECODE文		○
	CALL文		○		PRINT文		○
	RETURN文*	○			PUNCH文		○
入 出 力 文	書式付き順次READ文**		○	拡張された	DO WHILE文	○	
	書式付き順次WRITE文		○	DO文	DO UNTIL文	○	
	書式無し順次READ文		○	最初の実行文		○	
	書式無し順次WRITE文		○	最後の実行文		○	
	リスト指示READ文**		○	DOの端末文		○	

* 論理IF文のトレーラ部に書かれている場合には静止しない。

** データセット識別番号が端末に割り当てられている場合は、必ず静止する。

付録2. ス ナ ッ プ 文

種 類	形 式	機 能	使 用 例
代 入 文	$y = e$	e の値を y に与える (データ値の一時変更, Fortran の代入文に準ず る) y : 変数名 配列要素名 部分列名 e : 式	$A(J, K) = 3.0$ 配列要素 A(J, K) に 一時的に値 3.0 を入れ て再実行する。
論理 IF 文	IF(e)S	論理式 e が真の場合, ス ナ ッ プ 文 S を実行する。 e : 論理式 S : 論理 IF 文を除 くスナ ッ プ 文	IF(I.LT.J) WRITE X,Y I が J より小さい場合, X と Y の値を出力する。
GOTO 文	GOTO n	文番号 n へ強制的に分岐 する。 n : 実行プログラム 単位内の文番号	GOTO 100 途中スキップして文番 号 100 から再実行。
STOP 文	STOP	実行中のプログラム単位 を強制的に終了させる。	
WRITE 文	WRITE olist	出力項目 olist に指定し たデータ値を出力する (データ値の参照) olist : 変数名 配列要素名 配列名 の並び 部分列名 文字定数	WRITE A,B,C 実行プログラム単位内 の変数 A, B, C の現 在の値を出力する。

付録 3. 行サブコマンド一覧 [富士通マニュアル [3] より転載]

行サブコマンド	機 能	有 効 な 画 面
M	一文ごとのモニタ条件監視 (値のトレースと 静止モードへの移行)	モニタ条件定義メニュー
S	モニタ条件監視の一時停止	
T	一文ごとのモニタ条件監視 (値のトレース)	
S	ステップモード指定	実行モード指定メニュー
L	低速モード指定	
M	中速モード指定	
H	高速モード指定	
SS	ステップモードの範囲指定 (二つ一組で使 用)	
LL	低速モードの範囲指定 (二つ一組で使用)	
MM	中速モードの範囲指定 (二つ一組で使用)	
HH	高速モードの範囲指定 (二つ一組で使用)	
O	データセット又はメンバ削除	データセット名再指定メニュー メンバ名選択メニュー
OO	範囲指定によるデータセット又はメンバの削 除 (二つ一組で使用)	
S	メンバの選択	データセット名再指定メニュー メンバ名選択メニュー
SS	範囲指定によるメンバの選択 (二つ一組で使 用)	
GO	指定の行番号への制御移行	実行画面
RUN	指定の行番号までの高速実行	
F	データセットの解放	データセット割当て解除メニュー
FF	範囲指定によるデータセットの解放	

付録 4. サブコマンド一覧 [富士通マニュアル [3] より転載]

サブコマンド名	機 能	有 効 な 画 面
ADD	データセットの追加	データセット名再指定メニュー
OMIT	データセット又はメンバの削除	データセット名再指定メニュー
LOCATE (LOC)	指示項目の表示域の先頭への表示	メンバ名選択メニュー 実行モード指定メニュー
MAX	スクロールキーとの併用で上下左右方向への最大スクロール	モニタ条件定義メニュー プログラム単位名選択メニュー 実行時データセット表示メニュー データセット名再指定メニュー コンパイラオプション再指定メニュー 実行画面 エラーメッセージ表示画面*
UP	上方向スクロール	実行画面 (データ表示域) モニタ条件定義メニュー (条件定義域)
DOWN	下方向スクロール	実行画面 (データ表示域) モニタ条件定義メニュー (条件定義域)
LEFT RIGHT	左方向スクロール 右方向スクロール	実行画面 (データ表示域)
BACK (B)	前のページへ	HELP 画面
NEXT (N)	次のページへ	
SKIP (S)	次の選択項目へ	
TOP (T)	先頭の選択項目へ	
UP (U)	前の選択項目へ	
ALLOCATE (ALLOC)	実行時データセット指示メニューへの移行	実行画面
COND	モニタ条件定義メニュー又はプログラム単位名選択メニューへの移行	
MODE	実行モード指定メニューへの移行	

サブコマンド名	機 能	有 効 な 画 面
DISPLAY	指定のプログラム単位の原始プログラム表示又は元の画面（ソース表示域）の再表示	実行画面
LISTLOG (LISTL)	ログ情報のリスト出力	
MONITOR (MON)	一文ごとのモニタ条件監視 (値のトレースと静止モードへの移行)	
SUSPEND (SUS)	モニタ条件監視の一時停止	
TRACE (TR)	一文ごとのモニタ条件監視 (値のトレース)	
RERUN	利用者プログラムの再実行 (プログラム単位の先頭から)	
RUNTO	指定のプログラム単位までの高速実行	
SAVE	再試行のための各種情報の格納	

* エラーメッセージ表示画面ではMAXのみ有効である。

付録5. P F キー機能一覧〔富士通マニュアル〔3〕より転載〕

機 能 キー番号	P F キ ー の 機 能*	
	メ ニ ュ ー 画 面	実 行 画 面
PF1	HELP 画面の表示	HELP 画面の表示
PF2		動作選択メニューの表示
PF3	メニュー画面への入力の終了	利用者プログラムの終了通知
PF4	セッションの終了通知	セッションの終了通知
PF5		静止モードの解除
PF6		D O ループの高速実行指示
PF7**	上方向スクロール	上方向スクロール
PF8	下方向スクロール	下方向スクロール
PF9		
PF10	左方向スクロール	左方向スクロール
PF11	右方向スクロール	右方向スクロール
PF12***	カーソル移動	カーソル移動
PF13		
PF14		
PF15		
PF16		
PF17		
PF18**		
PF19	上方向スクロール	上方向スクロール
PF20	下方向スクロール	下方向スクロール
PF21		
PF22	左方向スクロール	左方向スクロール
PF23	右方向スクロール	右方向スクロール
PF24		

* 説明のないPFキーは、DOCK/FORTでは意味を持たないことを示す。

** 画面上でスクロール可能な表示域が、2箇所ある（上下又は左右に画面が分割される）場合はPF7以下の四つのキーは、上側又は左側のスクロールに使用でき、PF19以下の四つのキーは、下側又は右側のスクロールに使用できる。

1箇所だけの場合は、PF7以下の四つのキーだけが使用できる。

*** 画面上の最初の入力可能な欄の先頭へ移動する。ただし、実行画面の場合は、2行目のサブコマンド入力欄の先頭へ移動する。