The University of Osaka
Institutional Knowledge Archive

Title	プログラム・ノート　大阪大学大型計算機センターニュース　No.4
Author(s)	
Citation	大阪大学大型計算機センターニュース. 1970, 4, p. 20-47
Version Type	VoR
URL	https://hdl.handle.net/11094/65134
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

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

The University of Osaka

◉ プログラム・ノート

1. FORTRAN細部のリビジョン・アップと制限事項について

今回のFORTRAN コンパイラの切換えによって，これまでに見つかっている障害に対していくつかの修正が行なわれましたので，お知らせします。また，最後の《制限事項》は現在障害としてあがっていながらまだ未解決の問題点，ならびに障害とはいわないまでも使用上注意を要する点をあげてあります。

ここに述べてある内容はすべて日本電気提供の資料に基づくものです。

《修　正　点》

(1) FORMAT 文における最後の右括弧について

一つのプログラム単位内に DATA 文があり，そのプログラム単位内で使用されている FORMAT 文に最後の右括弧がなく且つ下記の例のようになってその FORMAT 文が終っている場合，正しい DATA 文に対して無関係なダイアグノステイック・メッセージ“209 UNLABELED COMMON INITIALIZATION” が出力されていましたが，出力しないように修正しました。

（例）

```
601 FORMAT (1H2, 5X, I4, 5X, :ERROR:
    DATA (A(I),  I = 1, 10)/10 * 1.0/
```

また，使用されている FORMAT 文に最後の右括弧がなく且つ下記の例のようになってその FORMAT 文が終っている場合（DATA文の有無は関係なし），次の文まで FORMAT 文とみなして処理を行なっていたため，この FORMAT 文に対して無関係なダイアグノステック・メッセージが出力されていましたが，出力しないように修正しました。

（例）

```
101 FORMAT (1H△, F10.4
    WRITE (3, 20) A, B
```

ダイアグノスティック・メッセージ“060 NON-INTEGER CONSTANT USED IN ILLEGAL CONTEXT” が出力されていました。

(2) DATA 文におけるダイアグノステイック・メッセージについて

DATA 文において，配列を変数の並びとして用いた場合，その DATA 文に誤りがあると，その配列要素の数だけダイアグノスティック・メッセージが出力されていましたが，その DATA 文に対して1つのダイアグノスティック・メッセージを出力するように修正しました。

（例）

```
COMMON   A (10)
DATA     A/10 * 0.0/
```

に対してダイアグノスティック・メッセージ“209 UNLABELED COMMON INITIALIZATION” を1行出力するように修正しました。

(3) ファイル終了状態における BACKSPACE 文の実行について

BACKSPACE 文の実行について，READ 文の END 指定（END＝n）により制御が移された後で BACKSPACE 文を実行すると動作不確定となったり，また WRITE 文に続いて ENDFILE 文を行ない，その後で BACKSPACE 文を実行すると，“UEP B” となっていましたが，この時 BACKSPACE 文を実行後のファイルの位置は，ファイル終了記録の直前に位置し，ファイル終了記録も 1 記録として処理するように修正しました。

したがって，この BACKSPACE 文を実行後に WRITE 文を実行することによりファイルを拡張することも可能です。

(4) ＤＯ文における符号付定数について

DO 文の使用において（DO n i＝m1, m2, m3），パラメタ m1, m2, m3 として符号付定数を用いた場合（例：DO 10 I＝＋1, ＋10）何のダイアグノスティック・メッセージも出力されず，あたかも正しいかの如く実行されていましたが（特に負符号を用いた実行については動作不確定でした），ダイアグノスティック・メッセージ “144 SYNTAX ERROR IN DO STATEMET” を出力する様に修正しました。

(5) ブランク・カードの処理について

MON$$ EXEQ FORTRAN カードあるいは，END 行の後にソースプログラムがなく，ブランク・カード（カラム 1 ～72がブランクのカード）だけがあった場合，そのブランクカードに対してコンパイルをしていましたが（そのブランクカードを注釈行とみなしていた），これをコンパイルは行なわず無視する様に修正しました。なお，ソースプログラムの途中のブランクカードは今までと同様に注釈行とみなしソースリスティング上にプリントされます。

(6) 不正な SUBROUTINE 文，FUNCTION 文について

SUBROUTINE 文及び FUNCTION 文の最後に右括弧以外の特殊文字が用いられた場合，“UEP B” となっていましたが，ダイアグノスティック・メッセージ “053 ILLEGAL SYNTAX IN SUBPROGRAM ARGUMENT LIST” を出力するように修正しました。

（例） SUBROUTINE SUB (A, B,

上記メッセージが出力されます。

また，SUBROUTINE SUB (A, B のように最後の右括弧を忘れた場合は，既に修正されています。

(7) 計算形 GO TO 文の不正処理について

計算形 GO TO 文の処理中，例で示す様な誤った文があるとダイアグノスティック・メッセージ “025 “name” ILLEGAL ARITHMETIC IF STATEMENT SYNTAX” が出力されたり，“UEP B” になる場合がありましたが，ダイアグノスティック・メッセージ “170 SYNTAX ERROR IN LIST OF COMPUTED GO TO STATEMENT” を出力する様に修正しました。

（例）

(i) GO TO (10, 20, 30,), ＩＩＩＩＩの場合

メッセージ“025“ＩＩＩＩ,)” ILLEGAL ARITHMETIC IF STATEMENT SYNTAX’’

(ii) GO TO (10, 20, 30 ,), ＩＩＩＩＩＩの場合

UEP B

これまでは上記メッセージが出力されていました。

(8) EQUIVALENCE 文の不正処理について

EQUIVALENCE 文の処理中，その EQUIVALENCE 文が正しいにもかかわらず，ダイアグノスティック・メッセージ“148 EQUIVALENCE ALTERS COMMON ALLOCATION OF “name”” が出力されたりクローズドループになったりする場合がありましたが正しく処理するように修正しました。

（例）

EQUIVALENCE (A, B), (C, D), (E, F), (F, C, A) と使用した時ダイアグノスティック・メッセージ“148 EQUIVALENCE ALTERS COMMON ALLOCATION “C”” が出力されたり，EQUIVALENCE (A, B), (C, D), (E, F), (F, A, C) と使用した時クローズドループになっていました。

(9) 整数型変数（又は配列要素）に文字型データを代入した場合，そのデータによっては通常の整数型内部表現と一致しない為それらの除算の際“UEP B” となる場合がありましたが，正常に，続く演算を行なうように修正しました。ただし，このときの除算の結果は不正確な結果となっていることにご注意下さい。

(10) コンパイルが途中で打切られたときの END 行の処理について

コンパイル中，END 行が読まれる前にコンパイルが打ち切られたとき（多量のエラーが検出されるとコンパイル処理は打ち切られる），その時点で END 行まで読み飛ばしますが，END 行がない場合，クローズドループになっていましたので，正しく処理する様に修正しました。

また，END＝1.0 の様な使い方をした場合，この文は END 行とみなされ，次の文からコンパイルが行なわれていましたが，この END＝1.0 を代入文として処理する様に修正しました。

(11) 書式つき磁気テープ・ファイルにおける BACKSPACE 文の不正処理について

磁気テープを用いる書式つき記録に対して，BACKSPACE 文を実行後に WRITE 文を実行すると，出力される記録の残りの部分は空白ではなく，その BACKSPACE が行なわれた記録の内容が出力されていましたが，残りの部分は空白となるように修正しました。

《基本外部関数の精度改善と修正》

(1) 近似公式の改善について

ALOG, ATAN, DSIN, DLOG, DATAN の近似公式を作り直し，これにより，展開区間での相対誤差は次のように改善されました。

関数名	展開区間	最大相対誤差	改善前の相対誤差	改善の程度
ALOG	$15/16 \leqq X \leqq 17/16$	5.8×10^{-11} (**)	$10^{-9}\sim10^{-11}$	0.5～1桁
ATAN	$\lvert X\rvert \leqq 1/3$	1.5×10^{-11}	$10^{-9}\sim10^{-11}$	0.5～1桁
DSIN	$\lvert X\rvert \leqq \pi/2$	0.9×10^{-20}	10^{-16}	3.5～4桁
DLOG	$15/16 \leqq X \leqq 17/16$	5.0×10^{-20} (**)	$10^{-17}\sim10^{-20}$	1 ～3桁
DATAN	$\lvert X\rvert \leqq 1/8$ (*)	1.0×10^{-20}	$10^{-14}\sim10^{-17}$	3 ～6桁

(*) DATANは従来展開区間が $\lvert X\rvert \leqq 1/3$ でしたが $\lvert X\rvert \leqq 1/8$ に狭くしました。

(**) $0.99 < X < 1.01$ の範囲では $(X-1)/(X+1)$ の計算の桁落ちにより相対誤差は
ALOG(X)…… $11+\log_{10}\lvert X-1\rvert$ 桁（絶対誤差は10^{-11}）
DLOG(X)…… $21+\log_{10}\lvert X-1\rvert$ 桁（絶対誤差は10^{-20}）
となります。

(2) DEXPの修正について

$1/4\log_e 2 \leqq \lvert X\rvert \leqq 1/2\log_e 2$ での近似公式の使用法の誤りを修正しました。これにより従来この区間で $\pm1/4\log_e 2\sim\pm3/4\log_e 2$（$\pm0.173287\sim\pm0.25993$）で相対誤差$10^{-17}$，$\pm3/4\log_e 2\sim\pm1/2\log_e 2$（$\pm0.25993\sim\pm0.346573$）で相対誤差$10^{-17}\sim10^{-16}$となっていましたが，相対誤差$10^{-18}\sim10^{-19}$に改善されました。

(3) ALOG, DLOGの修正について

従来，$X=1$ の近傍での近似公式の使用法に誤りがあり，（特にDLOGでは $X=1$ で正確な0にならない）相対誤差が悪かったのを修正し，$X=1$ の近傍でALOG, DLOGの相対誤差が $(X-1)/(X+1)$ の相対誤差と殆んど同じになるようにしました。すなわち各々の相対誤差は次のごとくです。

ALOG…… $(11+\log_{10}\lvert X-1\rvert)$ 桁

DLOG…… $(21+\log_{10}\lvert X-1\rvert)$ 桁

(4) TANHの修正について

(i) 従来，$X=0$ の近傍でも $\mathrm{TANH}(X)=(e^{2X}-1)/(e^{2X}+1)$ で計算されていたため，$X\fallingdotseq0$ のとき $e^{2X}-1$ の計算で桁落が生じ，$\lvert X\rvert \leqq 0.1$ で精度が約 $(10+\log_{10}\lvert X\rvert)$ 桁程度となっていましたが，$\lvert X\rvert \leqq 1/8$ ならばTANH(X)を多項式近似で計算するようにし，$X=0$ の近傍でも9.5～10桁の精度が得られるように修正しました。

(ii) 従来，$709 < X < 1024$ でTANH(X)が-1となるエラーがありましたが，TANH(X) $\fallingdotseq1.0$ が得られるように修正しました。

(5) CSIN, CCOSの修正について

TANHの場合と同様に，これらの関数値の計算において，sinh(v)（Vは引数の虚数部とする）が $V\fallingdotseq0$ においても $(e^{v}-e^{-v})/2$ で計算されていたため相対誤差が大きくなっていました。これを $\lvert V\rvert \leqq 1/8$ ではsinh(v)を多項式近似で計算するように修正し，$V\fallingdotseq0$ においてもsinh(v)の精度を9.5～10桁に保つようにしました。

(6) SIN, COS の修正について

従来，基本外部関数 SIN（従って SIN を使用する基本外部関数 COS も）は，実引数X の範囲が，$|X/2\pi| < 2^{64}$ で使用されていましたが，$2^{32} \leqq |X/2\pi| < 2^{64}$ での $X - 2n\pi$ の計算において桁落ちが生じ，SIN (COS) では有効数字が殆んど残りませんでした。このため実引数Xの範囲を $|X| \leqq (2\pi)\ 2^{32}$ に狭くしました。

又，近似公式を改造し，展開区間 $|X| \leqq \pi/2$ での最大相対誤差は 1.13×10^{-10}，改善の程度は 0.5～1 桁に改善しました。

なお，$X \fallingdotseq n\pi$ ($n = \pm 1, \pm 2, \cdots\cdots$) での SIN，$X \fallingdotseq (n + 1/2)\pi$ での COS の計算では (DSIN, DCOS も同じ)，$X - n\pi$，$X - (n + 1/2)\pi$ の計算で桁落ちが生ずるため，相対誤差の増加は避けられません。

(7) 上記(1)～(6)の修正により，これらを使用する関数も精度が向上しました。

関　数　名	修　　正　　内　　容
ALOG	近似公式の改善(1)により 0.5～1 桁精度向上。X＝1 の近傍での精度が向上します(3)。
ALOG10(＊)	ALOG を使用しますので ALOG と同じ精度向上があります。
ATAN	近似公式の改善(1)により 0.5～1 桁精度向上。
ATAN2 (＊)	ATAN を使用しますので ATAN と同じ精度向上があります。
TANH CSIN CCOS	X＝0 の近傍での精度向上，及び 709＜X＜1024での誤りの修正(4)。 実引数の虚数部≒0 における sinh 計算の精度向上(5)，及び SIN, COS を使用しますので修正(6)により 0.5～1 桁精度が向上します。
CEXP(＊)	SIN, COS を使用しますので 0.5～1 桁精度が向上します。
CLOG(＊)	ALOG, ATAN を使用しますので 0.5～1 桁精度が向上します。 また(3)により $|X| \fallingdotseq 1$ で精度が向上します。
DSIN	近似公式の改善(1)により 3.5～4 桁精度が向上します。
DCOS (＊)	DSIN を使用しますので DSIN と同じ精度向上があります。
DLOG	近似公式の改善(1)により 1～3 桁精度が向上します。 また，X＝1 の近傍で精度が向上します。(3)
DLOG10(＊)	DLOG を使用しますので DLOG と同じ精度向上があります。
DATAN	近似公式の改善(1)により 3～6 桁精度が向上します。
DATAN2 (＊)	DATAN を使用しますので DATAN と同じ精度向上があります。
DEXP	$1/4 \log_e 2 \leqq |X| \leqq 1/2 \log_e 2$ で $10^{-18} \sim 10^{-19}$ に精度が向上します。(2)

(＊) これらの関数は，そのサブルーチン自体は変更されていません。

(8) 基本外部関数 CCOS, CSIN, TANH の不正計算について

基本外部関数 CCOS, CSIN の計算において引用した複素数型変数の虚数部 v が $V < -0.125$ の時，又基本外部関数 TANH の計算において引用した変数 r が $r < -0.125$ の時の不正計算を修正しました。

《制　限　事　項》

(1) 宣言文に対する文法チェック不足について

DIMENSION 文，COMMON文，EXTERNAL 文，型宣言文において英字及び配列宣言子の間を区切るコンマ（—注意事項１—）に関して下記例のように

① 余分なコンマがある場合

② 配列宣言子の最後の右括弧の次のコンマがない場合

③ コンマの代りに“＋”，“—”等が書かれている場合（—注意事項２—）

エラーメッセージが出力されず，コンパイラは上記の各々の場合に対して

①′ 余分なコンマを無視する。

②′ 配列宣言子の後にコンマがあるものとみなす。

③′ コンマが書かれているものとみなす。

としてコンパイルを続行します。

（例）

COMMON A,, B,　　→ COMMON A, B とみなします。

COMMON X(10)Y　→ COMMON X(10), Y とみなします。

REAL A＋B　　→ REAL A, B とみなします。

—注意事項１—

配列宣言子内のコンマについては正しくチェックします。

—注意事項２—

英字名に続く場合のみで，EXTERNAL 文では“＋”，“—”等は誤りとして検出されます。

(2) CALL 文で指定するサブルーチン副プログラムの名前に＄以外の特殊文字が含まれている時，正しいダイアグノスティック・メッセージが出力されず，無関係なメッセージが出力されます。

(3) 基本外部関数 DEXP について

基本外部関数 DEXP を EXTERNAL 文で宣言し，外部手続きの実引数として用いた場合，その外部手続きにおける DEXP に対応する外部手続き（例えば仮引数をＢとすれば，Ｂ）を実行した時，正しい結果が得られません。

(4) 実数型零除算の結果について

実数型零除算の結果は，NORMAL DIVISION の場合に被除数，REVERSE DIVISION の場合に零が与えられます。

NORMAL DIVISION ： A＝B／C ←　（A ⇦　B）

REVERSE DIVISION： A＝B／(C＋D) ←（A ⇦　0.0）

(5) 整数型データに対する四則演算ならびに関係演算について

整数型データの四則演算においてオーバフロとなるような場合，正しく行なわれませんのでご注意下さい。メッセージも出ません。

（例）

I ＝34359738367

J ＝ 1

K＝ I ＋ J

この場合，I＋Jの演算によってオーバフロとなりKには－34359738368が格納されます。

また，関係式においても引き算を行ないその符号によって大小の比較判断を行ないます。したがって，引き算の結果がオーバーフローとなるような場合には正しい判断が行なわれません。

2．ATAN2, DATAN2 の結果の範囲の変更について

日本電気標準の FORTRAN では ATAN2，DATAN2 の結果は 0～2π の範囲で得られるようになっており，本学センターでもこれをそのまま使用して来ましたが，全国共同利用の他大学大型計算機センターでは－π～π の範囲で結果が得られるようになっています。

全国共同利用の意味から，互換性を考慮した上で本学センターでも他大学にそろえることとし，東北大学大型計算機センターの御好意により，東北大学の開発にかかる ATAN 2，DATAN2 のデックをご提供いただき8月1日より切換えを行ないました。

したがって，現在では先に速報でお知らせしましたように ATAN2, DATAN2 の結果は－π～π の範囲で得られるようになっています。従来より ATAN2, DATAN2 を使用されている利用者はプログラムの書換えが必要になりますから，十分にご注意下さい。

なお，この ATAN2，DATAN2 の変更は日本電気標準の FORTRAN のわくからはみ出した変更になっており，東北大学，大阪大学の大型計算機センター以外で NEAC FORTRAN をご使用になる場合には，従来通り結果の範囲は 0～2π のままですから，この点にもご注意下さい。

3．ASSEMBLER 細部のリビジョン・アップと制限事項について

8月1日のシステムおきかえ時におこなわれた ASSEMBLER の修正点，ならびにこの時点で判明している制限事項について，日本電気（株）よりの提供資料によりお知らせします。

《改　造　点》

(1) STDIO マクロについて

STDIO マクロによりスタンダード I/O ルーチンの PRINT ルーチンをダブル・バッファで使用するとき，バッファ・アドレスをとり出すコーディングが面倒でありましたが，第4オペランドを新たに設けたことにより容易にしました。

（マクロ形式）

CoL6	16	21
(label)	STDIO	PRINT, area 1 (, area 2) (, tag) (PUNCH)

(ジェネレート形式)

```
CoL6      16      21
  (label  EQU     ＊)
          MC
          B       /PRT/
                  (/PCH/ )
  (tag)   DCW     area 1
          (DC     area2)
```

以上のマクロ形式の如く，第４オペランドにタグを指定することにより，現在利用可能なバッファの左端アドレスが入っているフィールドの右端を指しますので，このタグを利用することにより容易にバッファ処理が可能となります。

(2) DTF マクロについて

ＤＴＦマクロによってファイル・テーブルを作成する際の処理がいくつか変更されました。

① ボリューム・ネームの左づめ

リリーズ１０までのアセンブラでは DTF の DSKLABEL エントリの第２オペランドに指定されたボリューム・ネームを右づめ（左に０を追加）でありましたが，リリーズ11.5よりＤＴＦ内に指定されたボリューム・ネームが６字に満たないとき左づめにし，右にブランクを追加します。

② DSKLABEL 及び LABEL エントリの第２オペランドの処理

DTF マクロの DSKLABEL エントリ及び LABEL エントリの各々の第２オペランドの区別を明確にしました。

	パラメータ		従来の処理		変更後の処理	
	DSKLABELの第２オペランド	LABELの第２オペランド	＋65フィールド	＋29フィールド	＋65フィールド	＋29フィールド
ディスクファイル	指定なし	指定なし	０００００００	００００００	０００００００	００００００
	指定なし	ｎｎｎ	０００ｎｎｎ	００００００	０００００００	００ｎｎｎ
	ｍｍｍ	指定なし	０００ｍｍｍ	００００００	ｍｍｍ△△△	００００００
	ｍｍｍ	ｎｎｎ	０００ℓℓℓ	００００００	ｍｍｍ△△△	００ｎｎｎ
磁気テープファイル	指定なし	指定なし		００００００		００００００
	指定なし	ｎｎｎ		００ｎｎｎ		００ｎｎｎ
	ｍｍｍ	指定なし		００ｍｍｍ		００００００
	ｍｍｍ	ｎｎｎ		００ℓℓℓ		００ｎｎｎ

（注１）表中「＋65フィールド」及び「＋29フィールド」はDTFによってジェネレートされたレーベル・エクステンション内の相対フィールド位置であります

＋65 フィールド…… ボリューム・ネーム
＋29 フィールド…… ファイル・シリアル・ナンバー

（注２）第２オペランドのｍｍｍ，ｎｎｎは右づめの処理の違いを明確にするため３字づつ指定したものとします。

（注３）ℓℓℓはｍｍｍとｎｎｎのうち後で処理されたものを意味します。

③ ディスクに対するボリューム・シーケンス・ナンバに関する変更

ファイル・テーブルのレーベル・エクステンション+33のフィールドは

磁気テープ・ファイルの場合

…… リール・シーケンス・ナンバ

ディスク・ファイルの場合

…… ボリューム・シーケンス・ナンバ

となります。

ディスク・ファイルに対する DTF マクロでボリューム・シーケンス・ナンバが指定されていない場合，００００をジェネレートする様に変更されました。(従来は ０００１ がジェネレートされていた)

(注１) SYMUNIT エントリに TAPE が指定されている場合は従来通り ０００１又は－００１がディフォルト値としてジェネレートされます。

(注２) LABEL エントリ及び DSKLABEL エントリでリール・シーケンス・ナンバが共に指定されている場合は，後で処理された値が上記+33フィールドにジェネレートされます。

(3) ワーク・ファイル，マクロ・ファイル のオーバフロ時のエラー処理について

マクロ・ファイル (MW０) とワーク・ファイル (MW1, MW2, MW3) を使用中に，エンド・オブ・ファイルの状態が検出された場合，NOGO スイッチをオンにしソース・プログラムを引き続き処理するように変更しました。(従来，その状態が起きたときは NOGO スイッチをオンにし EOP としていました。)

なお，メッセージも “LAS 40I ……” “LAS41I ….” から “LAS43I WORKFILE OVERFLOW ON MWn(nn) IN ××××××××” を SPR 上へ出力する様になりました。

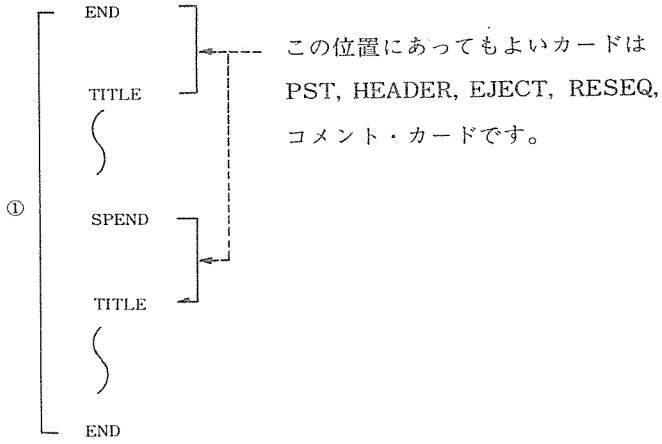
(4) H命令について

ソース・プログラムをアセンブル中に HALT 命令が検出された時，従来は NOGO スイッチをセットしないでその処理を続けていましたが，これを “LAS 01I ××××××××××× HAS FATAL ERRORS” のメッセージをコンソールおよび SPR 上に出力し，NOGO スイッチをオンにセットして処理を続けるように変更しました。

又，SPR 上にはエラー・フラッグ “R” が印字されます。

(5) TITLE カードの前のカードについて

ソース・プログラム・デック内で TITLE カードの前にあってもよいカードは PST, HEADER, EJECT, RESEQ,. コメントの各カードに限られます。



許されていないカードがあった場合は“LAS 33I INVALID TITLE CARD POSITION”がコンソールおよび SPR 上へ出力され，このサブプログラム（①で示す部分）は MGO 上に出力されず，NOGO スイッチがオンになり次のサブプログラムの処理へ移行します。

(6) DTF マクロに対するコメント・カード

DTF マクロのサブエントリ直後にカラム 6 の ＊ によって示されたコメント・カードが有る場合，従来はそのままの位置にリスティングされていたが，これを改めジェネレートされたファイル・テーブル，IORW等の後にリステングされるようにしました。

《修　正　点》

(1) 未定義アドレスのリロケーション・インディケータ

命令のオペランド欄又は，アドレス・コンスタントで未定義アドレスがあるとき，その部分に対してはアップワード・リロケーション・インディケータが割りあてられる様にしました。

(2) EQU ステートメントの X0 修飾

EQU ステートメントのオペランド欄で X0 修飾を使用してもインデックス・ビットをキャンセルできなかった件を修正しました。

(3) EQU ステートメントとインデックス修飾

EQU ステートメントにおいて下記の 2 通りの方法によってインデックス修飾を使用すると正しく処理されませんでしたが，正しく処理される様に修正しました。

```
A   EQU   B+Xn
C   EQU   Xm+A
```

```
A   EQU   Xn+B
C   EQU   D+A
E   EQU   Xm+A
```

(4) フォーマット・エラーの２進コンスタントのエラー処理について

２進コンスタント＃ｎｎＢｃｃ……ｃにおいて，ｃｃ……ｃの２進変換後の文字数がｎｎより大きい場合，下のｎｎ字分だけが取られますが，この時ｎｎが１桁で指定されているとエラー・フラッグがつきませんでしたが，エラー・フラッグをつけるように修正しました。

(5) フォーマット・エラーの８進リテラルのエラー処理について

８進リテラル＃ｎｎＣℓℓ…ℓを使用した場合，ℓℓ…ℓを格納するのに桁数ｎｎが足りない時，動作不定になる件がありましたが，上記形式の８進リテラルが使用された場合，フォーマット・エラーのフラッグ“Ｆ”をつけたｎｎ桁メモリを確保し，ℓℓ…ℓの左からｎｎ桁分だけ確保されたエリア中にセットする様にしました。

(6) アセンブラの構造上のエラーについて

アセンブラがリカージョン処理を起した場合，ＵＥＰになることがありましたが，この件を修正しました。

(7) リカージョン処理について

ソース・プログラムをアセンブルする際，リカージョン処理を行なう必要がない場合でも，アセンブラが実行されているメモリ・サイズが10進６桁以上のときリカージョン処理が起る場合がありましたが，修正しました。

又，１つのサブプログラムについて１０回以上のリカージョン処理を行なう必要がある時，エンドレス・ループに入る件も修正されました。

(8) DDUMP マクロについて

ダイナミック・ダンプ・マクロ命令の DDUMP マクロが正しくジェネレートされる様にしました。

(9) エラー・メッセージについて

ソース・プログラム中で TITLE カードが誤った位置に挿入されている場合に出力されるメッセージ“LAS 33I ～”中のサブプログラム名が正しく表示されない件を修正しました。

又，上記エラー・メッセージがコンソール上にもタイプアウトされましたが，SPR上にのみ出力する様にしました。

(10) シンボル・テーブル・オーバーフローについて。

シンボル・テーブル・オーバーフローとなった時，ファイルのクローズ処理の途中でプログラム部分までクリアしてしまったため，INVALID OP CODE となる事がありました。

この件を修正しました。

《制　限　事　項》

(1) １つのサブプログラム中に TITLE ステートメントが２枚以上あっても，それに対するエラー・チェックはされず，その TITLE カードが出る度に，アドレス・アサインメトが初期値にセットされて，あたかも正常の如く処理されてしまいます。

《注　意　事　項》

(1) ソース・ステートメントのフォーマット・エラーのチェックが不十分な場合があります。

① 命令のアドレス部に＋１××……×と書いたとき，＋符号の次の文字が数字なら全体がニューメリック・リテラルとみなされますが，＋符号に続く文字のすべてが数字でない場合でもエラー・フラッグはセットされず，各文字のニューメリック・ビットのみが有効になります。

② ＤＡのサブエントリのオペランドのパラメータ内で数字でなければならないパラメータ中に英文字又は特殊文字が混在していても，下例の如くニューメリック・ビットのみが有効になり，エラー・フラッグはセットされません。

例	BUF	DA	１×20, G
	SUB1		１，１A
	SUB1		1, 1　とみなされます。

③ ＭＩＴ命令のＣ３アドレスを省略してもＭＩＴ命令自身にはエラー・フラッグがセットされずＭＩＴ命令の後にあるＤＣＷステートメントにフォーマット・エラーのフラッグ “F” がセットされます。

④ 命令又はコンスタントに対する番地のアサインが512Ｋをこえてもエラー・フラッグはセットされず，０番地に戻って番地のアサインを続行します。

(2) １つのサブプログラム中で SPEND, LTORG, END ステートメントの合計が６をこえたときのエラー処理の方法が誤っているためストール状態となります。

(3) リカージョンの回数制限を新たに設け，限度を98回とします。なお，“SYMBOL TABLE OVERFLOW”とならぬ場合でも，標準サイズ（36K）で処理可能なレーベル数は約1300個以内です。

(4) ＤＴＦマクロのサブエントリ中にカラム６の＊によって示されるコメント・カードを入れてはなりません。コメント・カードが挿入されていると，その直後までがＤＴＦマクロとみなされ，コメント・カードの後のサブエントリはエラーとみなされます。

4．ALGOL使用上の制限事項について

現在までに見つかっている ALGOL 使用上の制限事項ならびに注意事項は以下の通りです。日本電気（株）からの提供資料によりお知らせします。

《制　限　事　項》

(1) 手続の宣言内での配列の宣言を禁止します。

配列はすべて外側のブロックで宣言して下さい。サブセグメント内で宣言する配列も不可能です。

(2) 内側の PROCEDURE で外側のフォーマル・パラメータを使用することは出来ません。

フォーマル・パラメータを使用したい場合は，1つ外側の PROCEDURE で，別の変数を宣言してその変数に値を移すことによって，その変数を内側の PROCEDURE で用いるようにして下さい。

(3) 欄送りの設定続き TABULATION の使用は出来ません。

この手続きは，出力時に指定した印字位置に出力するよう欄送りを行なう手続きであるが，現在これと同効果を与える方法としては，出力の際書式によって間隔をとるようにする方法しかありません。

(4) パラメータの区切りとして)〈文字列〉：(を用いた場合，文字列にブランクを用いることは出来ません。

現在，文字列の中にブランクを用いるとコンパイル時にエラーとなってしまいます。英文字のみ使用可能です。

(5) 手続きの実パラメータとして入出力手続き名の一部（ディスク関係のもの），設定手続き名の使用は出来ません。

実パラメータとして使用不可能な入出力手続き名の一部と設定手続き名は次のものです。

PUT	WRITE1	FORMAT	VLIM	TABULATION
GET	READ 0	FORMATn	HEND	NODATA
WRITE 0	READ1	HLIM (n=0, 1…9)	VEND	BADDATA

(6) 単純行先式として（〈行先式〉）を用いることは出来ません。

例　GO TO (L1);

上の例で示すように行先式を（　）でくくることが禁止されます。行先式として文法上許されるものは，名札かスイッチ名ですから（　）を用いなくとも行先式は書けますので（　）を用いないようにして下さい。

(7) コンパイル時に出力するリストにおいて，リスト・オプションのL，I，T，C が正常に働きません。

コンパイル時出力するリストの種類のうちソース・プログラム・リストとメモリ・リザベーション・リストは正常に働きますが，レーベル・リスト，アイデンティファイア・リスト，ターゲット・リスト，コンスタント・リストの4種類は，ブロックが深くなると正常に出力されません。

(8) 入力データがカードの 79, 80 カラムで終了することを禁止します。

入力データにおいてデータの区切として2個の空白を採用している場合，1枚のカードに書かれるデータがカラム79あるいは80で終了する場合，カードの切れ目でデータの区切りとなりませんので注意して下さい。

(9) 行先式内でエラーが有り，４０２６エラー・ナンバ出力後，行先式の終りの；の所で ２３０６，５００３エラー・ナンバ が出力される。

(10) ４０１９のエラー・ナンバを出力すべきなのに１４０１となる場合があります。

(11) １００３のエラー・ナンバを出力すべきなのに１０００となる場合があります。

(12) ４０１３のエラー・ナンバを出力すべきなのに４００９となる場合があります。

(13) ４００８のエラー・ナンバを出力すべきなのに１４０３となる場合があります。

(14) 同一のエラーに対して，出力すべきエラー・ナンバが異なって出力される時がある。

(15) MON$$ EXEQ ALGOL,,, SMカードの後，MON$$カードまたはダミー・カードを置いた場合，UEP A になります。

(16) 書式の文法チェックが完全になされていない。

(17) ６０２２のエラー・ナンバを出力すべきなのに６０２７となる場合があります。

(18) ６０２４のエラー・チェックが完全でない。

(19) ６２００のエラー・ナンバを出力すべきなのに１００１となる場合があります。

(20) ６０００のエラー・ナンバを出力後，コンパイラーの処理が正しく行なわれておらず，途中で処理が終了してしまいます。

(21) ６３１２のエラー・ナンバを出力すべきなのに６０１４となる場合があります。

(22) ６３１４のエラー・ナンバを出力すべきなのに６３１３となる場合があります。

《注　意　事　項》

(1) 実パラメータとして算術式が用いられていて，recursive call となる PROCEDURE の場合，その実パラメータに対する仮パラメータは値とりとしなければなりません。

(2) 一般に，回帰的に呼ばれる手続きに対して，次の条件を満たすと実行時ループの現象を生じます。

○ 実引数に演算子を含む式が書かれ，対応する仮引数が名前変えの場合。

○ 実引数の式に名前変えの仮引数を含む場合。

5．ダイアグノスティック・メッセージ（続）について

エラー番号	エラーメッッセージ・意味・例
１００	REPETITIŌN NUMBER NŌT AN INTEGER ◦ DATA 文の繰り返し数が整数型定数ではありません。 〔例〕 ◦ { DIMENSIŌN REP (10) DATA (REP(I), I=1, 10)/10. ＊2.0/
１０１	TWŌ ADJACENT ŌPERANDS WITH NŌ INTERVENING ŌPERATŌR IN CŌNSTANT LIST ◦ DATA 文の定数の並びの中で定数間に区切記号がありません。 〔例〕 ◦ { LŌGICAL L DATA L/3. TRUE. /

エラー番号	エラーメッセージ・意味・例
1 0 2	ŌPERAND IN CŌNSTANT LIST NŌT A CŌNSTANT ∘ DATA 文の定数の並びに定数以外のものがあります。 〔例〕 ∘ DATA B1, B2/ A1, A2/
1 0 3	SYNTAX ERRŌR IN NUMERIC CŌNSTANT ∘ 数値型定数に文法違反があります。 〔例〕 ∘ DATA B/23 F 0.2/
1 0 4	SYNTAX ERRŌR IN CŌMPLEX CŌNSTANT ∘ 複素数型定数に文法違反があります。
1 0 5	REPETITIŌN NUMBER GREATER THAN 100, 000 ∘ DATA 文の繰り返し数が 100, 000 を越えています。 〔例〕 ∘ { DIMENSIŌN A (100001) DATA (A(I), I=1, 100001) / 100001＊1.0 /
1 0 6	CŌNSTANT USED AS FUNCTIŌN ŌR SUBRŌUTINE NAME ∘ 定数が関数名またはサブルーチン名として使われています。 〔例〕 ∘ B= 2 1 2(I)＋3 1 8(J) ∘ CALL 601 (I, R, D, C, L, S)
1 0 7	ILLEGAL BRANCH LIST IN ARITHMETIC IF ∘ 算術 IF文の文番号の並びが不正です。 〔例〕 ∘ IF(I) 1, 2 ∘ IF(R) 1, 2, 3, 4
1 0 8	ILLEGAL EXPRESSIŌN SYNTAX ∘ 式に文法違反があります。 〔例〕 ∘ L＝A. NŌT. B ∘ X＝A＊(B＋R)Y
1 0 9	SUPERFLUŌUS＝SIGN IN EXPRESSIŌN ∘ 式の中に不用な“＝” 記号があります。 〔例〕 ∘ IF(A＝B) GŌ TŌ 109
1 1 0	TŌŌ MANY RIGHT PARENTHESES IN EXPRESSIŌN ∘ 式の中で右かっこが多すぎます。 (左カッコが足りません) 〔例〕 ∘ A＝B＊(C＋D))
1 1 1	DATA EXCEEDS 262K ŌF STŌRAGE ∘ データが 262K を越えています。

エラー番号	エラーメッセージ・意味・例
1 1 2	EQUIVALENCE ALTERS THE BASE OF COMMON ∘ EQUIVALENCE 文で，共通ブロックの拡張によって共通ブロックの開始番地を越えています。 〔例〕 ∘ COMMON X, Y DIMENSION Z(4) EQUIVALENCE (Y, Z(3))
1 1 3	IMCOMPLETE EXPRESSION ∘ 式が不完全です。 〔例〕 ∘ A＝B＊ ∘ A＝
1 1 4	EQUIVALENCE JOINS TWO COMMON BLOCKS VIA "$" ∘ EQUIVALENCE 文で2 つの共通ブロックを結んでいます。 〔例〕 ∘ COMMON /BLK1/A, B, C, D COMMON/ BLK2 /X, Y, Z EQUIVALENCE (A, X)
1 1 5	WRONG TYPE OF INTRINSIC FUNCTION ARGUMENT ∘ 組み込み関数の引用で実引数のデータ型が正しくありません。 〔例〕 ∘ INTEGER I R＝SIN(I) ∘ REAL R I＝IABS(R)
1 1 6	WRONG NUMBER OF ARGUMENTS FOR INTRINSIC FUNCTION ∘ 組み込み関数の引用で実引数の個数が正しくありません。 〔例〕 ∘ A＝SIN (X, Y) ∘ I＝IOR (J)
1 1 7	AT LEAST TWO ARGUMENTS REQUIRED FOR FUNCTION "$" ∘ この関数には少なくとも２つの実引数が必要です。 〔例〕 ∘ I＝MAX 0 (J) ∘ I＝MIN 0 (J)
1 1 8	"$" UNDEFINED SUBSCRIPT NAME ∘ 未定義な添字があります。 〔例〕 ∘ DIMENSION A (20 , 20) DATA ((A(I, J), I＝1, 20), JJ＝ 1, 20) / 400＊1.0 /

エラー番号	エラーメッセージ・意味・例
1 1 9	DATA STATEMENT INITIALIZES DUMMY ARRAY “$” ∘ DATA 文で仮引数の配列を使用しています。 〔例〕 ∘ FUNCTIŌN FUN (A, B) DIMENSIŌN A(10) DATA A/10＊10.0/
1 2 1	SYNTAX ERRŌR IN VARIABLE LIST ∘ DATA 文の変数の並びに文法違反があります。 〔例〕 ∘ DATA A＋B / 2.0, 1.0 /
1 2 2	WARNING: ARRAY ELEMENT ŌF “$” USED IN STATEMENT FUNCTIŌN EXPRESSIŌN ... ACCEPTED ∘ 文関数の式の中に配列要素があります。 〔例〕 ∘ FUN(X, Y, Z)＝X＊Y＊Z＊DIM(4)
1 2 3	SYNTAX ERRŌR IN STATEMENT FUNCTIŌN ARGUMENT LIST ∘ 文関数の定義で，仮引数の並びに文法違反があります。 〔例〕 ∘ F(A, B)＊C＝A＊B＊C＊
1 2 5	TŌŌ MANY LEFT PARENTHESES IN EXPRESSIŌN ∘ 式の中で左括弧が多すぎます。 （右括弧が足りません） 〔例〕 ∘ A＝B＊((X＋Y)
1 2 6	CŌMMA FŌUND EMBEDDED IN EXPRESSIŌN ∘ 式の中に“,” 記号があります。
1 2 7	CŌNSTANT USED ŌN LEFT SIDE ŌF ASSIGNMENT STATEMENT ∘ 定数が代入文の左辺に使用されています。 〔例〕 ∘ 1.0 ＝ A
1 2 8	WARNING: REDUNDANT IMPLICIT TYPE DECLARATIŌN ∘ IMPLICIT 文に2重の宣言があります。 〔例〕 ∘ IMPLICIT REAL (A) IMPLICIT REAL (A)
1 2 9	WARNING: REDUNDANT TYPE DECLARATIŌN FŌR “$” ∘ 型宣言文に2重の宣言があります。 〔例〕 ∘ CŌMPLEX C CŌMPLEX C

エラー番号	エラーメッセージ・意味・例
1 3 0	ILLEGAL PUNCTUATION OR OPERAND "$" IN TRACE STATEMENT ∘ TRACE 文に文法違反があります。 〔例〕 ∘ BEGIN TRACE 4 ∘ BEGIN TRACE NN(4)
1 3 1	CONSTANT EXCEEDS 70 DIGITS ∘ 定数の長さが 70 桁を越えています。 〔例〕 ∘ AREA＝ $\underbrace{1111\cdots\cdots1111}_{71}$
1 3 2	CHARACTER STRING OR OCTAL STRING USED IN ILLEGAL CONTEXT ∘ 文字定数あるいは 8 進定数が不正な所で使用されています。（代入文，CALL 文，理論IF 文，DATA 文，STOP 文，PAUSE 文，FORMAT 文以外の所で使用されています） 〔例〕 ∘ REWIND :1:
1 3 3	MORE VARIABLES THAN CONSTANTS ∘ DATA 文で定数の数よりも変数の数が多すぎます。 （定数の数が足りません） 〔例〕 ∘ DATA A1, A2, A3 / 1.0 , 2.0 /
1 3 4	MORE CONSTANTS THAN VARIABLES ∘ DATA 文で変数の数より定数の数が多すぎます。 （変数の数が足りません） 〔例〕 ∘ DATA A4, A5 / 4.0, 5.0, 6.0 /
1 3 5	INVALID CONTROL PARAMETER IN IMPLIED DO ∘ DO 形並びの制御変数が正しくありません。 〔例〕 ∘ DIMENSION A(10) DATA (A(I), I=1, 1A)/ 10 ＊ 1. 1/
1 3 6	ILLEGAL OPERAND IN VARIABLE LIST ∘ DATA 文で変数の並びが正しくありません。 〔例〕 ∘ DATA 5/5/
1 3 7	UNLABELED COMMON EXCEEDS 262K ∘ 無名共通ブロックの大きさが262K を越えています。 〔例〕 ∘ DIMENSION A(8200), B(8200), E(8200), 1F(8200) COMMON A, B, E, F ↑↑ カラム67

エラー番号	エラーメッセージ・意味・例
1 3 8	COMMON BLOCK "$" EXCEEDS 262K ○ 共通ブロックの大きさが262Kを越えています。 〔例〕 ○ DIMENSION A(8200), B(8200), E(8200), 1F(8200) COMMON /BLK/A, B, E, F ↑↑ カラム67
1 3 9	FIXED POINT CONSTANT OUT OF RANGE ○ 整数型定数が範囲外にあります。 (±8388607までです) 〔例〕 ○ I＝8388608
1 4 0	"$" IS A DUPLICATE COMMON ENTRY OR IS DECLARED IN MORE THAN ONE COMMON BLOCK ○ 共通ブロックの宣言で，同一名が2 重にあるいは二つ以上の共通ブロックで宣言されています。 〔例〕 ○ COMMON/BLK1/A, B/BLK2/B, C
1 4 1	ARRAY "$" EXCEEDS 262K ○ 配列の大きさが 262K を越えています。 〔例〕 ○ COMMON/BL/A (512, 512)
1 4 2	NO PUNCTUATION FOLLOWING REFERENCE TO ELEMENT OF ARRAY "$" ○ 配列要素の参照に文法違反があります。 (たとえば，"," がありません) 〔例〕 ○ WRITE (3,4) (DIM(I) I＝1, 10)
1 4 3	EXPRESSION NESTED TOO DEEPLY ○ 式の重ねが多すぎます。
1 4 4	SYNTAX ERROR IN DO STATEMENT ○ DO 文に文法違反があります。 〔例〕 ○ DO 123 I＝J, K/ ○ DO 123 I＝J, K, M,
1 4 5	BACKWARD STATEMENT REFERENCE IN DO STATEMENT ○ DO 文の端末文が，その DO 文よりも前にあります。 〔例〕 ○ 145 CONTINUE DO 145 I＝1, 10, 2

エラー番号	エラーメッセージ・意味・例
1 4 6	ILLEGL NESTING ŌF DŌ LŌŌPS ○ DŌ 文の重ねが正しくありません。 〔例〕 ○ DŌ 1 I1 =J, K DŌ 2 I2 =M, N 1 CŌNTINUE 2 CŌNTINUE
1 4 7	NŌN-INTEGER USED AS DŌ LŌŌP CŌNTROL PARAMETER ○ DŌ 文の制御変数が整数型でありません。 〔例〕 ○ REAL X, Y, Z DŌ 100 X=Y, Z 100 CŌNTINUE
1 4 8	EQUIVALENCE ALTERS CŌMMŌN ALLŌCATIŌN ŌF “$” ○ EQUIVALENCE 文で，共通ブロックに対して矛盾した宣言があります。 〔例〕 ○ CŌMMŌN A, B, C, D EQUIVALENCE (A, C)
1 4 9	UNDEFINED ARRAY REFERENCE ŌR LABELED STATEMEN T FUNCTIŌN ○ 未定義な配列の参照あるいは，文番号のついた文関数があります。 〔例〕 ○ 11 ICALX (I, J, K)=(I+J) ＊K
1 5 0	UNDEFINED ARRAY REFERENCE FŌLLŌWS LŌGICAL IF ○ 論理IF文に続いて未定義な配列の参照があります。 （あるいは文関数の宣言があります） 〔例〕 ○ IF (AA. EQ. BB) ICALCX (I, J) =I +J （ICALCX が配列として宣言されていないとします）
1 5 1	SYNTAX ERRŌR IN READ / WRITE LIST ○ 入出力文の並びに文法違反があります。 〔例〕 ○ READ (2, 1) A ＊B
1 5 2	ILLEGAL SYNTAX IN FŌRMAT STATEMENT ○ FŌRMAT 文に文法違反があります。 〔例〕 ○ 1521 FŌRMAT (1H1, K 10 . 3) ○ 1522 FŌRMAT (1H2, 21 ＊F 10 . 3)
1 5 3	SCALE FACTŌR IS TŌŌ LARGE ○ FŌRMAT 文の桁移動数が範囲外にあります。 （±2047までです） 〔例〕 ○ 153 FŌRMAT (1H3, 2048 PF 10. 3)

エラー番号	エラーメッセージ・意味・例
1 5 4	REPEAT FACTOR IS TOO LARGE ○ FORMAT 文の反復数が大きすぎます。 （最大 99までです） 〔例〕 ○ 154 FORMAT (1H1, 122F 10. 3)
1 5 5	WARNING: CHARACTER STRING LONGER THAN 133 CHARACTERS ○ FORMAT 文の文字ストリングの長さが133文字を越えています。 〔例〕 ○ 155 FORMAT (1H△, 134H0 ……Z)
1 5 6	FIELD WIDTH IS GREATER THAN 63 ○ FORMAT文の欄の幅が 63 を越えています。 〔例〕 ○ 156 FORMAT (1H△, F64. 40)
1 5 7	FRACTION WIDTH IS GREATER THAN 63 ○ FORMAT 文の小数部の幅が63 を越えています。 〔例〕 ○ 157 FORMAT (1H△, F10. 64)
1 5 8	WARNING: T SPECIFICATION POSITIONS OUT OF A 133 CHARACTER BUFFER ○ FORMAT 文のT変換の欄の数が 133 を越えています。 〔例〕 ○ 158 FORMAT (1H△, T140, F10. 3)
1 5 9	EXCESS RIGHT PARENTHESES IN FORMAT ○ FORMAT 文の右括弧が多すぎます。 （左括弧が足りません） 〔例〕 ○ 159 FORMAT (1H△, F10.3))
1 6 0	NUMERIC ITEM USED IN ILLEGAL FORMAT CONTEXT ○ FORMAT 文で数字だけの欄記述子があります。 〔例〕 ○ 160 FORMAT (1H△, 124P2, 10. 3)
1 6 1	WARNING: EXTRANEOUS INFORMATION FOLLOWING CONTINUE STATEMENT ○ CONTINUE 文に余分な情報があります。 〔例〕 ○ 161 CONTINUE 123
1 6 2	MISSING DECIMAL POINT IN F, E OR D FORMAT SPECIFICATION ○ FORMAT 文の F, E, D変換で "." がありません。 〔例〕 ○ 162 FORMAT (1H△, F 20)

エラー番号	エラーメッセージ・意味・例
1 6 3	COMPLEX CONSTANT NOT FOLLOWED BY DELIMITER ○ 複素数型定数に続く記述が正しくありません。 〔例〕 ○ C＝(1., 2.) X
1 6 4	EXCESS NESTED PARENTHESES IN FORMAT ○ FORMAT 文で括弧の重ねが多すぎます。 （最大3 重までです） 〔例〕 ○ 164 FORMAT (1H△, ((((F10. 3)))))
1 6 5	EXCESS LEFT PARENTHESIS IN FORMAT ○ FORMAT 文で左括弧が多すぎます。 （右括弧が足りません） 〔例〕 ○ 165 FORMAT (1H△, 2 (F10. 3)
1 6 6	NON-INTEGER USED AS UNIT NUMBER IN R/W STATEMENT ○ 入出力文で，ユニット番号として整数型以外が使用されています。 〔例〕 ○ READ(2.0,1) A ○ REAL A WRITE (A) X
1 6 7	WARNING: NON-COMMON DATA IN BLOCK DATA SUBPROGRAM ○ BLOCK DATA 副プログラムで共通ブロックを参照しない DATA 文があります。 〔例〕 ○ BLOCK DATA COMMON /BLK/I, R DATA A, B/2.2, 4.4/
1 6 8	SYNTAX ERROR IN IMPLIED DO LOOP IN R/W STATEMENT ○ 入出力文で，DO 型並びに文法違反があります。 〔例〕 ○ READ (1) (A(I), I＝10＊10)
1 6 9	SYNTAX ERROR IN UNCONDITIONAL GO TO STATEMENT ○ 無条件 GO TO 文に文法違反があります。
1 7 0	SYNTAX ERROR IN LIST OF COMPUTED GO TO STATEMENT ○ 計算形 GO TO 文の中の並びに文法違反があります。 〔例〕 ○ GO TO (10, 100 /), I ○ GO TO (100(
1 7 1	MISSING COMMA FOLLOWING LIST OE COMPUTED GO TO ○ 計算形 GO TO 文の中の並びに続く“,” がありません。 〔例〕 ○ GO TO (10, 100) I

エラー番号	エラーメッセージ・意味・例
1 7 2	MISSING ŌR ILLEGAL ŌPERAND FŌLLŌWING LIST ŌF CŌMPUTED GŌ TŌ ○ 計算形 GŌ TŌ 文の中の並びに続く整数型変数がないか，あるいは正しくありません。 〔例〕 ○ GŌ TŌ (100 , 200), ○ REAL A GŌ TŌ (100, 200), A
1 7 3	WARNING: EXTRANEŌUS INFŌRMATIŌN AT END ŌF CŌMPUTED GŌ TŌ ○ 計算形 GŌ TŌ 文に続いて余分な情報があります。 〔例〕 ○ GŌ TŌ (10, 100), K ＊J ○ GŌ TŌ (20, 200), I /
1 7 4	MISSING ŌR ILLEGAL ŌPERAND BEFŌRE LIST ŌF ASSIGNED GŌ TŌ ○ 割当て形 GŌ TŌ 文で，整数型変数がないかあるいは，正しくありません。 〔例〕 ○ GŌ TŌ, (10 , 100) ○ REAL A GŌ TŌ A, (10 , 100)
1 7 5	MISSING LIST AND/OR PRECEDING CŌMMA ŌF ASSIGNED GŌ TŌ ○ 割当て形 GŌ TŌ 文の中の並びあるいは，その前のコンマのいずれかがないか，あるいはその両方がありません。
1 7 6	FIRST ŌPERAND ŌF ASSIGN STATEMENT MISSING ŌR ILLEGAL ○ ASSIGN 文で文番号がないか，あるいは正しくありません。
1 7 7	MISSING DELIMITER FŌLLŌWING FIRST ŌPERAND ŌF ASSIGN STATEMENT ○ ASSIGN 文で文番号の次にあるべき“TŌ” がありません。 〔例〕 ○ ASSIGN 100 T
1 7 8	SECŌND ŌPERAND ŌF ASSIGN STATEMENT MISSING ŌR ILLEGAL ○ ASSIGN 文で整数型変数がないか，あるいは正しくありません。 〔例〕 ○ REAL A ASSIGN 100 TŌ A
1 7 9	WARNING: EXTRANEŌUS INFŌRMATIŌN FŌLLŌWING SECŌND ŌPERAND ŌF ASSIGN STATEMENT ○ ASSIGN 文で整数型変数に続いて余分な情報があります。 〔例〕 ○ ASSIGN 10 TŌ K＋J ○ ASSIGN 10 TŌ K, J

エラー番号	エラーメッセージ・意味・例
1 8 0	MISSING ŌR ILLEGAL UNIT DESIGNATŌR IN I/Ō STATEMENT ∘ 入出力文でユニット番号の指定がないか，あるいは正しくありません。 〔例〕 ∘ { REAL A BACKSPACE A ∘ { REAL B END FILE B
1 8 1	WARNING: EXTRANEŌUS INFŌRMATIŌN FŌLLŌWING I/Ō UNIT DESIGANATŌR ∘ 入出力文でユニット番号に続いて余分な情報があります。 〔例〕 ∘ REWIND I+1 ∘ BACKSPACE I*J ∘ END FILE I+J
1 8 2	WARNING: ILLEGAL DISPLAY STRING IN STŌP ŌR PAUSE STATEMENT ∘ STŌP 文あるいは PAUSE 文で表示文字が正しくありません。 〔例〕 ∘ STŌP (100) ∘ PAUSE A*B
1 8 3	WARNING: EXTRANEŌUS INFŌRMATIŌN FŌLLŌWING DISPLAY STRING IN STŌP ŌR PAUSE STATEMENT ∘ STŌP 文あるいは PAUSE 文の表示文字に続いて余分な情報があります。 〔例〕 ∘ STŌP :123: 123 ∘ PAUSE 1−2
1 8 4	WARNING: DISPLAY STRING TŌŌ LŌNG IN STŌP ŌR PAUSE STATEMENT ∘ STŌP 文あるいは PAUSE 文で表示文字が 40字を越えています。 〔例〕 ∘ STŌP :----------: 41 字以上
1 8 6	EXCESS LEFT PARENTHESIS ŌR ILLEGAL END ŌF FŌRMAT ∘ FŌRMAT 文で左括弧が多すぎるか，あるいは途中で終っています。 〔例〕 ∘ 186 FŌRMAT (1H△, F10 . 3, 3X
1 8 7	WARNING: EXTRANEŌUS INFŌRMATIŌN FŌLLŌWING ŌPERAND ŌF UNCŌNDITIŌNAL GŌ TŌ ∘ 無条件 GŌ TŌ 文の後に余分な情報があります。 〔例〕 ∘ GŌ TŌ 100 XYZ

エラー番号	エラーメッセージ・意味・例
1 8 8	ILLEGAL STATEMENT TERMINATING A DŌ LŌŌP ∘ DŌ 文の端末文に許されない文が使用されています。 （許されない文とは，GŌ TŌ 文，算術 IF 文，STŌP 文，RETURN 文，PAUSE 文あるいは DŌ文，もしくはこれらを含んだ論理 IF 文です） 〔例〕 ∘ { DŌ 1 I＝J,K 1 STŌP
1 8 9	NŌN－CŌMMŌN DATA AND PRŌGRAM STRING ARE TŌŌ LARGE ∘ オブジェクトプログラムが大きすぎます。
1 9 0	ILLEGAL SIZE RELATIŌNSHIP BETWEEN D AND W IN FŌRMAT SPECIFICATIŌN ∘ FŌRMAT 文で“D”と“W”の関係が正しくありません。 〔例〕 ∘ 190 FŌRMAT (1H△, F10 . 13)
1 9 1	STRING VALUE ASSIGNED TŌ DP ŌR CŌMPLEX VARIABLE ∘ 文字型の値が倍精度実数型あるいは複素数型変数に代入されています。 〔例〕 ∘ { DŌUBLE D STRING S D＝S
1 9 2	LŌGICAL VALUE ASSIGNED TŌ NŌN－LŌGICAL VARIABLE ∘ 論理型の値が非論理型変数に代入されています。 〔例〕 ∘ { LŌGICAL L REAL R R＝L ∘ { INTEGER I I＝.TRUE.
1 9 3	NUMERIC VALUE ASSIGNED TŌ NŌN－NUMERIC VARIABLE ∘ 数値型の値が非数値型変数に代入されています。 〔例〕 ∘ { LOGICAL L REAL R L＝R
1 9 4	NUMERIC ŌPERATIŌN ŌN NŌN－NUMERIC DATA ∘ 非数値型データに対する算術演算があります。 〔例〕 ∘ { LŌGICAL L1, L2 REAL A A＝L1＋L2

エラー番号	エラーメッセージ・意味・例
1 9 5	ILLEGAL EXPŌNENTIATIŌN ○ 許されない巾乗演算があります。 （複素数型を指数とする巾乗演算，及び複素数型を基数とした場合の整数型の指数以外の巾乗演算は許されません） 〔例〕 ○ { CŌMPLEX C1, C2 REAL R C1＝R＊＊C2
1 9 6	RELATIŌN ENTAILS CŌMPARISŌN ŌF LŌGICAL ŌR CŌMPLEX DATA ○ 関係演算で，論理型あるいは複素数型データの比較があります。 （複素数型データに対する関係演算は，.EQ. 及び.NE. のみ許されます） 〔例〕 ○ { CŌMPLEX C1, C2 IF(C1.GT.C2) GŌ TŌ 100
1 9 7	INCŌMPATIBLE MIXTURE ŌF DATA TYPES IN EXPRESSIŌN ○ 式の中でデータ型の不調和な混合演算があります。 〔例〕 ○ { REAL A, B LŌGICAL L A＝L＋B
1 9 8	NŌN-LŌGICAL CŌNDITIŌN FŌR LŌGICAL IF ○論理 IF 文に対して非論理型を使用しています。 〔例〕 ○ { REAL A, B IF(A＋B) GŌ TŌ 100
1 9 9	LŌGICAL ŌPERATIŌN ŌN NŌN-LŌGICAL DATA ○ 非論理型に対して論理演算があります。 〔例〕 ○ { REAL R1, R2 IF(R1.AND.R2) GŌ TŌ 100
2 0 0	CŌMPLEX ŌR NŌN-NUMERIC ARITHMETIC IF EXPRESSIŌN ○ 算術IF文で，複素数型データあるいは非数値型データが使用されています。 〔例〕 ○ { CŌMPLEX C IF(C) 1, 2, 3
2 0 1	ILLEGAL UNIT SPECIFICATIŌN IN I/Ō STATEMENT-SHŌULD BE INTEGER LESS THAN 17 ○ 入出力文でユニット番号が正しくありません。 （ユニット番号は，1～16でなければなりません） 〔例〕 ○ WRITE(17, LIST) A, B, C

エラー番号	エラーメッセージ・意味・例
2 0 2	NŌ VALUE RETURNED BY THIS FUNCTIŌN ∘ この関数副プログラムでは，関数値が定義されません。 （関数名が一度は左辺に現われなければなりません）
2 0 3	ARRAY NAME “$” USED AS A VARIABLE ∘ 配列名が変数として使用されています。 〔例〕 ∘ { DIMENSIŌN ARY (10) ARY = A + B
2 0 4	UNDEFINED CŌNTRŌL VARIABLE “$” ∘ 制御変数の値が未定義です。 〔例〕 ∘ DATA (A(I), I=1, J)/10 ＊3.0/
2 0 6	LIMIT ŌF NESTED DŌ STATEMENTS EXCEEDED ∘ DŌ文の重ねが多すぎます。 （最大63重までです）
2 0 7	WARNING: EXTRANEŌUS INFŌRMATIŌN FŌLLŌWING RETURN STATEMENT ∘ RETURN 文に余分な情報があります。 〔例〕 ∘ RETURN 123
2 0 8	ILLEGAL CHARACTER “$” IN NUMERIC CŌNSTANT ∘ 数値型定数に不正な文字があります。 〔例〕 ∘ A=1. 23$5 ∘ DATA E/2F1. 0 /
2 0 9	UNLABELED CŌMMŌN INITIALIZATIŌN ∘ DATA 文で，無名共通ブロックが使用されています。 〔例〕 ∘ { CŌMMŌN A(100) DATA (A(I), I=1, 100) /100 ＊1.1/
2 1 0	SYNTAX ERRŌR IN DEFINE FILE ∘ DEFINE FILE 文に文法違反があります。 〔例〕 ∘ DEFINE FILE 10 (10, 10, E)
2 1 1	ILLEGAL UNIT IN DEFINE FILE ∘ DEFINE FILE 文のユニット番号が正しくありません。 （ユニット番号は，1〜16 でなければなりません） 〔例〕 ∘ DEFINE FILE 17(10, 10, U, I)
2 1 2	DEFINE FILE FŌRMAT NŌT U, E, L ∘ DEFINE FILE 文の書式指定がU, E, Lのいづれでもありません。 〔例〕 ∘ DEFINE FILE 1(10, 10, A, IV)

エラー番号	エラーメッセージ・意味・例
2 1 3	DIRECT ACCESS RECORD NO. NON-INTEGER ∘ DIRECT ACCESS文のレコード番号が整数型でありません。 〔例〕 ∘ { REAL R READ (I:R) .A ∘ { REAL X, Y WRITE (J:X+Y) B
2 1 4	FIND STATEMENT MUST NOT HAVE LIST ∘ FIND文に並びがあります。 〔例〕 ∘ FIND (4:I5)A, B, C
2 1 5	DUPLICATE DEFINE FILE STATEMENT ∘ DEFINE FILE文で同一ユニット番号に対して2重の宣言があります。 〔例〕 ∘ { DEFINE FILE 16 (10, 10, L, I1) DEFINE FILE 16 (10, 10, L, I2)
2 1 6	RECORD SIZE OR COUNT NOT INTEGER CONSTANT IN DEFINE FILE ∘ DEFINE FILE文でレコードサイズあるいはレコード数の指定が整数型定数でありません。 〔例〕 ∘ DEFINE FILE 1(I, J, U, I)
2 1 7	ASSOC. VARIABLE IN DEFINE FILE IS NOT INTEGER VARIABLE ∘ DEFINE FILE文で記録変数指定が整数型変数でありません。 〔例〕 ∘ DEFINE FILE 1 (10, 10, U, 1)