



Title	科学計算用サブプログラム・ライブラリィ (SSL)の追加
Author(s)	
Citation	大阪大学大型計算機センターニュース. 1974, 12, p. 103-119
Version Type	VoR
URL	<a href="https://hdl.handle.net/11094/65229">https://hdl.handle.net/11094/65229</a>
rights	
Note	

*The University of Osaka Institutional Knowledge Archive : OUKA*

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

The University of Osaka

## 科学計算用サブプログラム・ライブラリィ (SSL) の追加

ここに示されたプログラムは、大阪大学大型計算機センターにおいて研究開発計画の一環として開発され、新たに科学計算用サブプログラム・ライブラリィとして追加になったものです。これらサブプログラムの呼び出し方は、従来のサブプログラム・ライブラリィと同じくフォーTRAN・プログラム中に CALL 文で引用するのみでおこなえます。そしてリンクロード時にディスク・バック上のサブプログラムと結合処理されます。

追加になったプログラムは、全部で4グループから成っています。以下でそれらの内訳及び内容の紹介を致します。

追加ライブラリィ・プログラム内訳  
作成者 大阪大学工学部 安井 裕・大中幸三郎

誤差評価の可能な多種精度演算ルーチン	
M 型*	内部操作の自由度を上げるためパラメターをなるべく多く仮引数として出した。
S 型*	つかいやすさのために引数をオペランドのみとした
E 型*	誤差項の計算を独立しておこなっている
I 型*	Interval arithmetic をおこなっている

\* すべて多種精度演算、可変長である。

### 概 説

ライブラリィ・プログラムの開発や新しい数値計算プログラムをつくる時にはしばしば精度、確度の確認、またプログラムでもちいられている諸係数の算出等を厳密におこなうことが必要になる。数値計算によって得られた解の確度の評価は自動計算機による数値計算上、重要な事柄である。高速に処理するための工夫も勿論、興味深い事柄ではあるが、数値解析上の主要な関心は打ち切り誤差、丸め誤差、さらに入力データのもつ誤差の伝播の解析的な決定にもある。

しかしながらどのような計算についても、意味ある計算の過程としてのプログラムにおける、

誤差の振舞いを知ろうとするときには、計算過程の個々について何らかの形で直接的に誤差の影響をつかんだ演算を行なわなければならない。

したがって、ここに我々は誤差評価を可能とする機能として、演算中に丸めの制御、誤差の計算、そして Interval Arithmetic 等を行なう機能をもった一連の基本演算プログラムを作った。そして、これらのプログラム・パッケージはすべて多種精度演算プログラムとしても高精度演算が容易にでき、他のルーチンにはみられない実用上有用な特徴を有している。

したがってはじめに述べた種々の目的に対して非常に強力な tool として設立つものと考え

る。なお、このプログラムは FORTRAN で作られており、全国共同利用大型計算機センター間での互換性は保証されている。

**M型: Machine language like arithmetic**

## 1. 数の表現

M型の多重精度実数は一次元整数配列をもちいて表わされる。その例として $\pi$ を10進で31桁について示すと下図のようになる。

配列名

MX

(1)	1
(2)	0
(3)	3
(4)	1 4 1 5 9
(5)	2 6 5 3 5
(6)	8 9 7 9 3
(7)	2 3 8 4 6
(8)	2 6 4 3 3
(9)	8 3 2 8 0

符号部  $\rightarrow \{MX(1)\} = \begin{cases} 1 & \text{正数} \\ 0 & \text{第2要素以下の内容にかかわらずゼロ} \\ -1 & \text{負数} \end{cases}$

指数部

整数部

小数部

仮数部

$\{MX\}^{(注)} \cong 3.14159 \cdots 83280 \times (10^5)^0$

(注)  $\{name\}$  と書くことにより name の内容を表わすものとする。

上図は仮数部の配列一要素に10進で5桁（一般にはN桁）はいるときの図である。そして、

指数部は $10^5$  毎（一般には $10^N$  毎）の指数をあらわしている。したがって  $\{MX(2)\} = 1$  ならば  $\{MX\} = 314159.26 \dots\dots$ となる。また、配列の長さは上図では“9”であるが、一般にはプリセットパラメータ KM で与えられる。

## 2. プリセットパラメータと宣言文

利用者が定めるプリセットパラメータは、NW, N, IR, ITE, IDE, KM の6つ（いずれも単精度整数型）であって、その役割と条件は次の通りである。

NW … WRITE 文のユニット番号。

エラーの印字の時に必要である。

N … 仮数部の配列一要素にはいる10進の桁数であって、次の3つの条件をみたすように N を定めること。

(i)  $N \geq 1$

(ii)  $(10^N)^2$  が単精度整数型でオーバーフローせずにあらわされること。

(iii)  $N < (\text{単精度実数型の仮数部の有効桁数} - 1)$

IR … 丸め処理を定めるもの。  $0 \leq IR \leq 10$

IR = 0 切り上げ<sup>(注)</sup>

1 0 捨 1 入

⋮ ⋮

5 4 捨 5 入

⋮ ⋮

9 8 捨 9 入

10 切り捨て

(注) 小数以下1位で切り上げて10とったとすれば  $9.0 < X \leq 10.0$  であるが、

IR = 0 のときの切り上げは  $9.0 \leq X < 10.0$  なり、真の切り上げと多少

異なる

ITE …… 指数部の上限

$2 \times \text{ITE}$  が単精度整数型でオーバーフローしないこと。

IDM … 倍長精度実数型の指数部の上限。

$10^{-\text{IDM}} < |X| < 10^{\text{IDM}}$  となる X が倍長精度実数型で表現できること。

KM … M 型の多重精度の数を表わす整数型一次元配列の長さ。条件は次の2つである。

(i)  $2 \times KM - 2$  が単精度整数型でオーバーフローしないこと。

(ii)  $KM \geq 7$

利用者はメインプログラムで次の宣言をする必要がある。

DOUBLE PRECISION DN, DIT, DIT 2

COMMON /BLKMS/ITE, ITE 1, IT, IT11, IR 1, KETA, N, IDE,  
FIT, DN, DIT, DIT 2, IERROR, NW

### 3. 各種サブルーチン

記号の説明

MACC 1 , MACC 2

M型の数を表わす長さKMの整数型一次元配列の配列名。

MACCW

ワーキングエリア・長さKKMの整数型一次元配列の配列名。

ただし,  $KKM \equiv 2 \times KM - 2$

INT

| {INT} | <  $10^N$  をみたす整数型変数名。

X, Z

倍長精度実数型変数名。

( i ) M型の多重精度演算ルーチンを使用するのに必須のサブルーチン。

SUBROUTINE START (IR, KM, KKM )

2. で定められたプリセットパラメータから計算される定数の計算と条件の判定を行う。

SUBROUTINE ERROR (NS 1 ,NS 2 )

エラーが起った時の印字ルーチンであって、このルーチンでエラーインディケータを印字し、プログラムを停止する。利用者は直接にこのルーチンを呼ぶことはない。

( ii ) M型の多重精度の2数の四則演算を行うサブルーチン。

SUBROUTINE ADD (MACC 1 ,MACC 2 ,KM )

$MACC 1 \leftarrow \{MACC 1\} + \{MACC 2\}$

桁落ちのインディケータKETA (COMMON領域) がセットされる。このインディケータは {KETA} だけの要素分が桁落ちしたことを示している。M型でKETAがセットされるのは、このルーチンと次に示すSUBROUTINE SUB だけである。このルーチンは内部でSUBROUTINE SUB を呼んでいるので、使用に際してはSUBROUTINE SUB が必要である。

SUBROUTINE SUB (MACC 1 ,MACC 2 ,KM )

$MACC1 \leftarrow \{MACC1\} - \{MACC2\}$

使用に際しては SUBROUTINE ADD が必要である。

KETAについては SUBROUTINE ADD と同じである。

SUBROUTINE MULT1 (MACC1, MACC2, MACCW, KM, KKM)

$MACC1 \leftarrow \{MACC1\} \times \{MACC2\}$

SUBROUTINE DIVI1 (MACC1, MACC2, MACCW, KM, KKM)

$MACC1 \leftarrow \{MACC1\} \div \{MACC2\}$

(iii) M型の多重精度の数と整数の乗除算

SUBROUTINE MULT2 (MACC1, INT, KM)

$MACC1 \leftarrow \{MACC1\} \times \{INT\}$

SUBROUTINE DIVI2 (MACC1, INT, KM)

$MACC1 \leftarrow \{MACC1\} \div \{INT\}$

(iv) M型の多重精度の数と倍長精度実数の変換を行うサブルーチン。

SUBROUTINE CVSHO (MACC1.Z, KM)

$Z \leftarrow \{MACC1\}$

SUBROUTINE CVLNG (X, MACC1, KM)

$MACC1 \leftarrow \{X\}$

#### 4. 諸 注 意

- (i) 2. で述べたプリセットパラメータを定めた後、演算ルーチンに飛ぶ前に必ずSTARTを呼んで定数の計算を行うこと。
- (ii) ERRORは各演算ルーチンから呼ばれている。
- (iii) ADDとSUBは1組になって働く。
- (iv) MULT1, DIVI1におけるKKMは $KKM \equiv 2 \times KM - 2$  ( $KM \geq 7$ ) である。
- (v) CVSHO, CVLNG は必ず変換誤差をとまう。
- (vi) 配列の長さKM ( $KM \geq 7$  , KKM ( $KKM \equiv 2 \times KM - 2$ ) はSTART で定めたものでなくてもよいので、演算毎に長さを変化させてもよい。
- (vii) サブルーチン内の配列は全て整合配列になっている。
- (viii) KM, KKMはDO 文の制御変数になっているので、HAPPで使用するときは注意が必要

である。

(ix) DFLOAT をもちいた以外はJIS FORTRAN (水準7000) にもとずいている。

S 型 : Symbolic language like arithmetic

## 1. 数の表現

S 型の多重精度実数はM型と同じく整数型一次元配列で表わされる。M型とのちがいは小数部一要素に  $2N$  桁 ( $N$  についてはM型参照) を入れたことである。したがって、M型と同じ例をS型で示すと下図のようになる。

配列名

LX

(1)	1	符号部	
(2)	0	指数部	
(3)	3	整数部	
(4)	1415926535	小数部	} 仮数部 $\{LX\} = 3.14159\cdots 83280 \times (10^5)^0$
(5)	8979323846		
(6)	2643383280		

配列の第1要素 (符号部) , 第2要素 (指数部) , 第3要素 (整数部) の定義はM型と同一である。したがって、指数部が  $10^{2N}$  毎でなく、 $10^N$  毎の指数を表わし、整数部には  $N$  桁 (小数部には  $2N$  桁) はいることに注意する必要がある。

## 2. プリセットパラメータと宣言文

利用者が定めるプリセットパラメータは、NW, N, IR, ITE, IDE, Kの6個であるが、Kをのぞく5個はM型と同一である。

K……………S型の多重精度実数を表わす整数型一次元配列の長さを示す単精度整数。条件は次の通りである。

(i)  $4 \times K - 8$  が単精度整数型でオーバーフローしないこと。

(ii)  $5 \leq K \leq 8$  (Kの上限は暫定的)

利用者はメインプログラムで次の宣言をする必要がある。

```
DOUBLE PRECISION DN, DIT, DIT 2
COMMON /BLKMS /ITE, ITE 1, IT, IT 1 1, IR 1, KETA, N, IDE, FIT,
      DN, DIT, DIT 2, IERROR, NW
      /BLKS / K, KK, KKK, IT 2, IT 2 1
```

### 3. 各種サブルーチン

記号の説明

LX, LY, LZ

S型の数を表わす長さKの整数型一次元配列の配列名。

サブルーチンと呼ぶときの実引数の配列名は各々異なる名でなくてよい。

INT

$|\{INT\}| \leq 10^N$ をみたす整数型変数名。

X, Z

倍長精度実数変数名。

(i) S型の多重精度演算ルーチンを使用するのに必要のサブルーチン。

```
SUBROUTINE STARTS (IR)
```

2. で定められたプリセットパラメータから計算される定数の計算と条件の判定を行なう。

```
SUBROUTINE ERROR (NS1, NS 2)
```

M型のERRORと同一の内容と役割をもつ。

(ii) S型の多重精度の2数の四則演算を行うサブルーチン。

```
SUBROUTINE ADDS (LX, LY, LZ)
```

$LZ \leftarrow \{LX\} + \{LY\}$

桁落ちのインディケータKETAについてはM型のADD, SUBと同じである。また、使用に際してはSUBROUTINE SUBSが必要である。

```
SUBROUTINE SUBS (LX, LY, LZ)
```



$LZ \leftarrow \{LX\} - \{LY\}$

桁落ちのインディケータKETAについてはM型の ADD, SUB, S 型の ADDS と同じである。また、使用に際しては SUBROUTINE ADDS が必要である。

SUBROUTINE MULTIS (LX, LY, LZ)

$LZ \leftarrow \{LX\} \times \{LY\}$

SUBROUTINE DIVI1S (LX, LY, LZ)

$LZ \leftarrow \{LX\} \div \{LY\}$

(iii) S 型の多重精度の数と整数の乗除算。

SUBROUTINE MULT2S (LX, INT, LZ)

$LZ \leftarrow \{LX\} \times \{INT\}$

SUBROUTINE DIVI2S (LX, INT, LZ)

$LZ \leftarrow \{LX\} \div \{INT\}$

(iv) S 型の多重精度の数と倍長精度実数の変換を行うサブルーチン。

SUBROUTINE CVSHOS (LX, Z)

$Z \leftarrow \{LX\}$

SUBROUTINE CVLNGS (X, LZ)

$LZ \leftarrow \{X\}$

#### 4. 諸 注 意

- (i) 2. で述べたプリセットパラメータを定めた後、演算ルーチンに飛ぶ前に必ず STARTS を呼んで定数の計算を行なうこと。
- (ii) ERROR は各演算ルーチンから呼ばれている。
- (iii) ADDS, SUBS は 1 組になって働く。
- (iv) ADDS, SUBS, MULT1S, DIVI1S は引数以外にワーキングエリアが必要である。
- (v) CVSHOS, CVLNGS は必ず変換誤差をとまう。
- (vi) (iv) で述べたワーキングエリア、引数の両方の配列は整合配列とはなっていないので、利用者は各サブルーチン内の DIMENSION 宣言の大きさを定めなければならない。配列の寸法は K ( $K \geq 5$ ), KK ( $KK \equiv 2 \times K - 3$ ), KKK ( $KKK \equiv 4 \times K - 8$ ) の 3 種類がある。
- (vii) 配列の長さ K, KK, KKK は STARTS で定めたものでなくてもよいので、演算毎に長



## 2. プリセットパラメータと宣言文

利用者が定める プリセットパラメータはNW, N, IR, ITE, KMAX, INDの6個であるが, NW, IR, ITE はM型と同一である。

N……………仮数部の配列一要素にはいる10進の桁数であり, 次の3つの条件をみたす単精度整数である。

(i)  $1 \leq N \leq 10$

(ii)  $(10^N)^2$  が単精度整数型でオーバーフローしないこと。

(iii)  $N < (\text{単精度実数型の仮数部の有効桁数} - 1)$

KMAX ……仮数部の長さの最大値を示す単精度整数。条件は次の2つである。

(i)  $5 \leq KMAX \leq 11$  (KMAX の上限は暫定的)

(ii)  $2 \times KMAX$  が単精度整数型でオーバーフローしないこと。

IND……………誤差の伝播や桁落ちによって, (仮数部の長さ) $< 4$  となったときの処理を与える単精度整数。

IND = 0 仮数部の長さを4とし, 誤差項を $10^{2N} - 1$ として演算を続行する。

IND = 1 警告を出す。その後の処理はIND = 0と同一。

IND = 2 エラーを印字し、演算を停止する。

利用者はメインプログラムで次の宣言をする必要がある。

```
DOUBLE PRECISION DIT, DIT 2, DIT 3, DIR
COMMON KMAX, KMAX 1, KMAX 2, KMAX 3, KMAXW, IT, IT11, IT21,
        DIT, DIT 2, DIT 3, FIT, FIT 2, ITE, ITE 1, IR 1, DIR, IND,
        IERROR, NW
```

## 3. 各種サブルーチン

記号説明

MACC 1, MACC 2, MACC 3

E 型の数を表わす長さ  $KMAX + 3$  の整数型一次元配列の配列名。

サブルーチンを呼ぶときの実引数の配列名は各々異なる名でなくてもよい。

INT

$\{|INT\} < 10^N$  をみたす整数型変数名。

(i) E 型の多重精度演算ルーチンを使用するのに必須のサブルーチン。

```
SUBROUTINE STARTE (N, IR)
```

2. で定められたプリセットパラメータから計算される定数の計算と条件の判定を行う。

SUBROUTINE ERRORE (NS 1, NS 2)

エラー、警告の印字ルーチンである。このルーチンで印字した後、各サブルーチンにもどって引数を印字する。利用者は直接にこのルーチンを呼ぶことはない。

(ii) E型の多重精度の2数の四則演算を行うサブルーチン。

SUBROUTINE ADDE (MACC 1, MACC 2, MACC 3)

$MACC\ 3 \leftarrow \{MACC\ 1\} + \{MACC\ 2\}$

使用に際してはSUBROUTINE SUBEが必要である。

SUBROUTINE SUBE (MACC 1, MACC 2, MACC 3)

$MACC\ 3 \leftarrow \{MACC\ 1\} - \{MACC\ 2\}$

使用に際してはSUBROUTINE ADDEが必要である。

SUBROUTINE MULTIE (MACC 1, MACC 2, MACC 3)

$MACC\ 3 \leftarrow \{MACC\ 1\} \times \{MACC\ 2\}$

SUBROUTINE DIVIE (MACC 1, MACC 2, MACC 3)

$MACC\ 3 \leftarrow \{MACC\ 1\} \div \{MACC\ 2\}$

(iii) E型の多重精度の数と整数の乗除算

SUBROUTINE MULT2E (MACC 1, INT, MACC 3)

$MACC\ 3 \leftarrow \{MACC\ 1\} \times \{INT\}$

SUBROUTINE DIVI2E (MACC 1, INT, MACC 3)

$MACC\ 3 \leftarrow \{MACC\ 1\} \div \{INT\}$

#### 4. 諸 注 意

(i) 2.で述べたプリセットパラメータを定めた後、演算ルーチンに飛ぶ前に必ずSTARTEを呼んで定数を計算すること。

- (ii) `ERRORE` は各演算ルーチンから呼ばれている。
- (iii) `ADDE`と`SUBE`は1組になって働く。
- (iv) 各演算ルーチンは引数以外にワーキングエリアが必要である。
- (v) (iv)で述べたワーキングエリア、引数の両方の配列は整合配列とはなっていないので、利用者は各サブルーチン内の `DIMENSION` 宣言の大きさを定めなければならない。配列の寸法は $KMAX\ 3$  ( $KMAX\ 3 \equiv KMAX + 3$ ),  $KMAXW$  ( $KMAXW \equiv 2 \times KMAX$ ) の2種類がある。
- (vi) 誤差の伝播によっておこる有効桁数の変化に伴う調整はプログラム内で自動的にやっている。
- (vii)  $KMAX$ ,  $KMAX\ 1$  ( $KMAX\ 1 \equiv KMAX + 1$ ),  $KMAX2$  ( $KMAX\ 2 \equiv KMAX + 2$ ),  $KMAX\ 3$ ,  $KMAXW$  は `DO` 文の制御変数になっているので、`HARP`で使用する時には注意が必要である。
- (viii) `DFLOAT`をもちいた以外は`JIS FORTRAN` (水準7000) にもとずいている。
- (ix) 記憶場所を節約するために、ワーキングエリアを`COMMON`領域に出すことができる。  
(例外…`ADDE`, `SUBE`における`MACC 4`は`COMMON`領域に出せない。)
- (x) 誤差項の処理が困難なので多重精度 $\leftrightarrow$ 倍長精度の変換ルーチンは製作しなかった。誤差項の処理ができなくても変換を行いたいときにはM型の`CVSHO`, `CVLNG`を改造すればよい。
- (xi) E型の多重精度の数の配列の大きさは  $KMAX$  の値にかかわらず14 word とすること。  
(暫定的)

## I 型 : Interval arithmetic

### 1. 数の表現

多重精度の `interval arithmetic` を行うのがI型である。したがって、I型の多重精度実数は `interval number` となるので、S型の2数をもちいて表わし、整数型二元配列となる。

配列各

IX	( , 1 )	( , 2 )		
( 1 , )	1	1	符号部	
( 2 , )	0	0	指数部	
( 3 , )	3	3	整数部	
( 4 , )	1 4 1 5 9 2 6 5 3 5	1 4 1 5 9 2 6 5 3 5	小数部	} 仮数部
( 5 , )	8 9 8 9 3 2 3 8 4 6	8 9 7 9 3 2 3 8 4 6	⋮	
( 6 , )	2 6 4 3 3 8 3 2 8 1	2 6 4 3 3 8 3 2 7 9	⋮	

IX の上限≡ {IX ( , 1 ) }=3.14159…83281×(10<sup>5</sup>)<sup>0</sup>

IX の下限≡ {IX ( , 2 ) }=3.14159…83279×(10<sup>5</sup>)<sup>0</sup>

IX ( , 1 ) に上限を, IX ( , 2 ) に下限を入れることに注意すること。

## 2. プリセットパラメータと宣言文

利用者が定めるプリセットパラメータはNW, N, ITE, IDE, IDM, Kの6個であるが, NW, ITE, IDE, KはS型と, NはE型と同一である。

IDM……倍長精度実数型の仮数部の有効桁数を10進で示した値。

利用者はメインプログラムで次の宣言をする必要がある。

```
SUUBLE PRECISION DIT, DIT 2, DN
COMMON  K, KK, KKK, IT, IT11, IT 2, FIT, DIT, DIT 2, ITE 1, N, DN,
        IDE, IDM, IERROR, NW
```

## 3. 各種サブルーチン

記号の説明

LXI, LYI, LZI

I 型の数を表わす大きさ K×2 の整数型二次元配列の配列名。

サブルーチンを呼ぶときの実引数の配列名は各々異なる名でなくてよい。

MACC 1, MACC 2, MACC 3

M型の数を表わす長さ KK (KK≡2×K-3) の整数型一次元配列の配列名。

サブルーチンを呼ぶときの実引数の配列名は各々異なる名でなくてよい。

INT

| {INT} | < 10<sup>N</sup>をみたす整数型変数名。

## IR 2

丸め処理を与える変数名。

ただし、この場合の切り上げは M 型の 2. で述べた

IR = 0 の場合ではなく真の切り上げである。

	負 数	正数, 0
0	切り上げ	切り捨て
1	切り捨て	切り上げ

(i) I 型の多重精度演算ルーチンを使用するのに必須のサブルーチン。

### SUBROUTINE STARTI (ITE)

2. で定められたプリセットパラメータから計算される定数の計算と条件の判定を行う。

### SUBROUTINE EXPAN (LXI, MACC 1, I)

I 型→M 型の変換ルーチン

$MACC\ 1 \leftarrow \{LXI\ ( , I)\}$

### SUBROUTINE COMP (MACC 1, LXI, I)

M 型→I 型の変換ルーチン

$LXI\ ( , I) \leftarrow \{MACC\ 1\}$

(ii) I 型の多重精度の 2 数の四則演算を行うサブルーチン。

### SUBROUTINE ADDI (LXI, LYI, LZI)

$LZI \leftarrow \{LXI\} + \{LYI\}$

必要なサブルーチン…ADDIP, SUBIP

### SUBROUTINE SUBI (LXI, LYI, LZI)

$LZI \leftarrow \{LXI\} - \{LYI\}$

必要なサブルーチン…ADDIP, SUBIP

### SUBROUTINE MULTII (LXI, LYI, LZI)

$LZI \leftarrow \{LXI\} \times \{LYI\}$

必要なサブルーチン…MULTIP

### SUBROUTINE DIVII (LXI, LYI, LZI)

$LZI \leftarrow \{LXI\} \div \{LYI\}$

必要なサブルーチン…DIVIP

(iii) I 型の多重精度の数と整数の乗除算。

SUBROUTINE MULT 2I (LXI, INT, LZI)

$LZI \leftarrow \{LXI\} \times \{INT\}$

必要なサブルーチン…MU 2IP

SUBROUTINE DIVI 2I (LXI, INT, LZI)

$LZI \leftarrow \{LXI\} \div \{INT\}$

必要なサブルーチン…DI 2IP

(iv) M 型の数の演算を行うサブルーチン。

(ii), (iii) のサブルーチンの内部で呼ばれているものであり，利用者は必ずしも利用しなくてもよいが，知っているると便利なので，ここに示す。

サブルーチン	引 数	役 割
ADDIP	<sup>(注)</sup> MACC1, <sup>(注)</sup> MACC2, MACC3, IR2	$MACC3 \leftarrow \{MACC1\} + \{MACC2\}$
SUBIP	<sup>(注)</sup> MACC1, <sup>(注)</sup> MACC2, MACC3, IR2	$MACC3 \leftarrow \{MACC1\} - \{MACC2\}$
MU1IP	MACC1, MACC2, MACC3, IR2	$MACC3 \leftarrow \{MACC1\} \times \{MACC2\}$
DI1IP	MACC1, MACC2, MACC3, IR2	$MACC3 \leftarrow \{MACC1\} \div \{MACC2\}$
MU2IP	MACC1, INT, MACC3, IR2	$MACC3 \leftarrow \{MACC1\} \times \{INT\}$
DL2IP	MACC1, INT, MACC3, IR2	$MACC3 \leftarrow \{MACC1\} \div \{INT\}$

(注) ADDIP, SUBIP においてはMACC1とMACC2の符号は異符号であってはならない。

#### 4. 諸 注 意

- (i) 2. で述べたプリセットパラメータを定めた後，演算ルーチンに飛ぶ前に必ずSTARTIを呼んで定数を計算すること。
- (ii) M, S, E 型の ERROR, ERROE に相当するものはなく，各ルーチンに組み込まれている。
- (iii) 各演算ルーチンは引数以外にワーキングエリアが必要である。
- (iv) (iii) で述べたワーキングエリア，引数の両方の配列は整合配列とはなっていないので，利用者は各サブルーチン内の DIMENSION 宣言の大きさを定めなければならない。配列の寸法は  $K \times 2$  (二次元),  $KK$  ( $KK \equiv 2 \times K - 3$ , 一次元),  $KKK$  ( $KKK \equiv 4 \times K - 8$ , 一次元) の 3 種類がある。



- (v) K, KK, KKKはSTARTI で定めたものでなくてもよいので、演算毎に長さを変化させてもよい。
- (vi) K, KK, KKKはDO 文の制御変数となっているので、HARPで使用する時には注意が必要である。
- (uiii) DFLOAT をもちいた以外はJIS FORTRAN (水準7000) にもとずいている。
- (viii) I 型の数 $\leftrightarrow$ 倍長精度実数の変換ルーチンは製作していないが、必要なときにはM型, S型のものを改造すればよい。この時、使いやすいようにCOMMON領域にあらかじめDIT, DIT 2, DN, IDE, IDM をとっている。この5つは変換ルーチン以外では使用されない。なお, IDMはM, S型でも使用していないが、あれば便利なのでつけ加えておいた。
- (ix) I 型の多重精度の数の配列の大きさはKの値にかかわらず、配列のサイズ(8, 2)をとること。

## 全般的注意事項

エラーインディケータ

	M 型	S 型	E 型	I 型
10	オーバーフロー <sup>(注1)</sup>	同 左	同 左	同 左
11	アンダーフロー <sup>(注1)</sup>	同 左	同 左	ナ <sup>(注3)</sup> シ
20	不定 (DIVI1)	不定 (DIVI1S)	不定 (DIVI1E)	不定 (DIVI1I)
21	不能 (DIVI1)	不能 (DIVI1S)	不能 (DIVI1E)	不能 (DIVI1I)
30	不定 (DIVI2)	不定 (DIVI2S)	不定 (DIVI2E)	不定 (DIVI2I)
31	不能 (DIVI2)	不能 (DIVI2S)	不能 (DIVI2E)	不能 (DIVI2I)
40	$ \{INT\}  \geq 10^N$	同 左	同 左	同 左
50	オーバーフロー <sup>(注2)</sup>	同 左	ナ シ	ナ シ
51	アンダーフロー <sup>(注2)</sup>	同 左	ナ シ	ナ シ
90	$IR < 0, IR > 10$	同 左	同 左	ナ シ
91	$KM < 7$	$K < 5$	$KMAX < 5$	$K < 5$

(注1) 四則演算の結果がオーバーフロー，アンダーフローしたことを示す。

(注2) 多重精度→倍長精度実数の変換において，オーバーフロー，アンダーフローしたことを示す。

(注3) I型におけるアンダーフローは0又は最小数におきかえているので“ナシ”になった。

## 参 考 文 献

大中幸三郎，安井 裕 “誤差評価の可能な多重精度演算”，情報処理 Vol. 15, No. 2, PP. 110～117, (1974).