



### 目 次

1. 利用負担金の改定のお知らせ	1
2. ジョブ制限値の変更について（再掲）	2
3. NQSのデフォルトの変更について（UXP）	5
4. VPP700/56へのライブラリの移行状況について	5
5. 新スーパーコンピュータシステム講習会開催のお知らせ	6
6. 計算化学講習会開催のお知らせ	7
7. 新スーパーコンピュータVPP700/56の利用方法について	8

大型計算機センターWWWホームページ  
<http://www.cc.kyushu-u.ac.jp>

#### 1. 利用負担金の改定のお知らせ

利用負担金の一部改定を予定していますが、以下の改定内容は現在学内等で検討・調整中であり、承認されれば平成9年1月7日より実施する予定です。

##### 1) 改定事項

平成9年1月から新スーパーコンピュータシステム（VPP700）を導入します。この導入に伴い、次の事項について負担金を改定する予定です。

- (1) 演算負担金（バッチ処理・スーパーコンピュータ）の改定
- (2) 演算負担金（会話型処理・スーパーコンピュータ）の改定
- (3) 入出力負担金（プリンタ）の追加
- (4) 現有の「並列計算機」の改称
- (5) 可視化用サーバ及びユーザーインターフェースワークステーションは現行のワークステーション負担金を適用
- (6) 「備考」欄の整備

##### 2) 改定内容

- (1) 演算負担金（バッチ処理・スーパーコンピュータ）  
以下のように改定します。

1 件の演算時間のうち

5 分まで	1 秒につき	3 円
5 分を超え 15 分まで	1 秒につき	1 円
15 分超	1 秒につき	0.4 円

ただし、17 並列以上のプログラムで使用したときの演算負担金については、上記の演算時間の区分に従って、各々その単価の 2 倍にします

また、併せて、「備考」欄中に「1 演算負担金バッチ処理の項中スーパーコンピュータを利用して単一プログラムにより複数の演算処理を並列に実行した場合の負担金については、各々演算時間中その最長の演算時間をその算出基準とする。」の項目を追加します。

(2) 演算負担金(会話型処理・スーパーコンピュータ)

1 秒につき 3 円 に改定します。

(3) 入出力負担金

新スーパーコンピュータシステムの一部として導入される大型カラープリンタ(A0判)は既存の高画質カラープリンタと同様に消耗品実費相当額を受益者負担で賄う方式を採用し、「大型カラープリンタ 4,000円/頁」の項目を追加します。

(4) 現有の「並列計算機」の改称

新スーパーコンピュータシステムも並列計算機となるため、「会話型処理」中の「並列計算機」を「分散メモリ型小規模並列コンピュータ」に改称します。

併せて、「備考」欄中の「並列計算機」についても同様に改称します。

(5) 新規導入の可視化用サーバ及びユーザーインターフェースワークステーションは現行のワークステーション負担金を適用します。

(6) 「備考」欄の整備

現行「備考」欄の 3 項目中「利用負担金額」を「一月の利用負担金額」にし、前回改正(平成 8 年 2 月)時の不明確さを是正し、別表の整備を図ります。

(システム運用掛 ダイヤルイン 092-642-2307)

## 2. ジョブ制限値の変更について(再掲)

平成 9 年 1 月 7 日(火)から新スーパーコンピュータシステム FUJITSU VPP700/56 のサービス開始に伴いジョブクラスと制限値を変更いたします。

新スーパーコンピュータシステムでは、最大 32 並列処理、主記憶 54.4 GB (32PE × 1.7GB) のジョブ投入が可能になります。また、非並列でも主記憶が 1.7 GB まで利用できます。

平成 8 年 10 月 1 日からテスト運用しております M-VPP 連携機能(詳細はセンターニュース NO.547 をご覧ください)は M-1800 の MSP と VP2600 の UX P 間の連携機能でしたが、平成 9 年 1 月 7 日からは M-1800 の MSP と VPP700 の UX P 間の連携機能に変わります。

変更箇所は次のとおりです。

### ・MSP 関連

#### 変更内容

1. VP バッチジョブ・ジョブクラス W, X のジョブ投入先を VPP700 に変更
2. M-VPP 連携機能のジョブクラス Y, Z を増設  
非並列 標準, 大型  
並列 8 PE 並列, 16 PE 並列

## 汎用コンピュータ M-1800/20U

適用ジョブ		ジョブクラス	CPU時間	ファイルアクセス	最大 ジョインズ	端末接続	ラインプリンタ		図形文字パターン数			
							出力枚数	出力行数				
バッチジョブ	標準ジョブ	A	分 10	万回 20	MB	分	枚	行	パターン			
		B	180	50								
	MTジョブ	N	10	20	10					600	36,000	100,000
	大型ジョブ	F	180	50	200							
VPPバッチジョブ	標準ジョブ	W	60	50	1.7GB	1435				非並列		
	長時間ジョブ	X	1200	50	1PE 当たり					非並列		
	並列ジョブ1	Y			1.7GB					8PE並列		
	並列ジョブ2	Z			16PE並列							
TSSジョブ		標準			60	制限	10	1435				
		大型		なし	50							

## スーパーコンピュータ VP2600/10

適用ジョブ		ジョブクラス	CPU時間	ファイルアクセス	最大 ジョインズ	端末接続	ラインプリンタ		図形文字パターン数
							出力枚数	出力行数	
バッチジョブ	大型ジョブ	A	分 10	万回 20	MB	分	枚	行	パターン
		B	180	50	50				
		V	10	50	300 (500)				
			180						

\* VP2600のジョブクラスおよび制限値は平成9年2月末まで有効です

・ UXP 関連

変更内容

1. 新スーパーコンピュータ VPP700/56 のジョブキューを新設  
 非並列 標準, 大型  
 並列 8PE並列, 16PE並列, 32PE並列

汎用コンピュータ M-1800/20U

適用ジョブ	キュー	CPU時間	リージョン	備考
バッチジョブ	ss	180分	100MB	
TSSジョブ		60分	100MB	ベクトル演算可

スーパーコンピュータ VPP700/56

適用ジョブ	キュー	CPU時間	リージョン	備考
バッチジョブ	c	60分	100MB	コンパイル専用
	s	60分	1.7GB	非並列
	p1	1200分	1PE あたり 1.7GB	非並列
	p8			8PE並列
	p16			16PE並列
	p32			32PE並列

スーパーコンピュータ VP2600/10

適用ジョブ	キュー	CPU時間	リージョン	備考
バッチジョブ	vs	60分	100MB	
	v1	180分	100MB	
	vx	180分	400MB	

\* VP2600のジョブキューおよび制限値は平成9年2月末まで有効です

(システム管理掛 ダイヤルイン 092-642-2308)

### 3. NQS のデフォルトの変更について (UXP)

平成9年1月7日(火)から、新スーパーコンピュータ VPP700/56 のサービスを開始します。これを契機にNQSのデフォルトを以下のように変更します。ご注意ください。

#### 1) qsub コマンドのデフォルトバッチキュー名

(新) 新スーパーコンピュータの p1 キュー

(旧) 汎用コンピュータの ss キュー

この変更によりバッチキュー名を省略して qsub コマンドを利用しているユーザは、キュー名 ss を必ず指定してください。

例) kyu-cc% qsub -q ss test.sh

#### 2) qstat, qdel コマンドのデフォルトマシン名

(新) kyu-vpp (新スーパーコンピュータ)

(旧) kyu-cc (汎用コンピュータ)

この変更により汎用コンピュータを利用しているユーザは必ずマシン名 kyu-cc を指定してください。

例) kyu-cc% qstat @kyu-cc

kyu-cc qdel -r kyu-cc 5478.kyu-cc

(システム管理掛 ダイヤルイン 092-642-2308)

### 4. VPP700/56 へのライブラリの移行状況について

現在 VP2600/10 で公開しているライブラリの新スーパーコンピュータ VPP700/56 への移行状況をお知らせします。なお、詳しい利用方法は移行作業が完了次第、センターニュース、LOGON, login 時のメッセージなどで随時お知らせします。また、VP2600/10 は 1997 年 2 月まで並行運用されます。

#### MARC

OS が MSP から UXP に変更される予定です。利用方法は汎用計算機の UXP(ホスト名 kyu-cc) から MARC ジョブを VPP700/56 に投入します。従来の 4 倍のメモリーが確保できるようになります。ただし、SSU はサポートされません。

#### Gaussian94

VP2600/10 と同様、汎用機からバックグラウンドで利用する形態です。ジョブキューとマシン名が変更になる他は、従来と同じ利用方法です。

#### FSPICE/VP

FSPICE/VP は過去のライブラリ統計の結果、ここ数年ほとんど利用がないため、VP2600/10 の撤去に伴い、2 月末日をもって運用を停止する予定です。運用停止により支障が生じる方は早めに連絡をお願いします。

## プログラムライブラリ開発課題

UXP のライブラリの利用方法は従来と同じです。MSP からの利用については、現在動作確認および移行方法の検討を行っております。運用方法は決まり次第お知らせします。

(ライブラリ室 ダイヤルイン 092-642-2295)

e-mail : f70029a@kyu-cc.cc.kyushu-u.ac.jp

## 5. 新スーパーコンピュータシステム講習会開催のお知らせ

標記講習会を下記の要領で開催いたします。希望者は、共同利用掛 (ダイヤルイン 092-642-2305) に申し込みください。なお、第一回と第二回の講習の内容は同じです。

### 記

- |       |  |           |                       |
|-------|--|-----------|-----------------------|
| ・日 時  | 第一回  | 12月20日(金) | 10時 ~ 16時             |
|       | 第二回  | 12月24日(月) | 10時 ~ 15時             |
| ・受付時間 | 9時30分 ~ 9時50分  |           |                       |
| ・対象   | センター利用経験者で、VPP700/56 に興味をお持ちの方<br>センター利用の経験がなくても、分散メモリ型並列計算機および並列化プログラミングに興味のある方 (なお、並列化プログラミングの解説は、ある程度の Fortran 文法の知識が前提)  |           |                       |
| ・募集人員 | 70名  |           |                       |
| ・内容   | 1) VPP700/56 の概要<br>新スーパーコンピュータシステム VPP700/56 の概要をハードウェア、ソフトウェア両面から紹介<br>2) 並列化プログラミング入門<br>VPP700/56 の並列化 Fortran である Fortran 90/VPP による並列化プログラミング手法についての解説<br>3) VPP700/56 利用法<br>センターの UXP, MSP システムから VPP700/56 を利用するための手続きの解説<br>4) 講演<br>題目 「高エネ研でのVPP500」<br>講師 橋本省二 (高エネルギー物理学研究所・データ処理センター)<br>概要 高エネルギー物理学研究所における VPP500/80 の利用について紹介する。高エネ研の VPP500 は主に素粒子物理の理論的および実験的研究のために導入された。最も多く CPU を使っているのは格子 QCD (量子色力学) の数値シミュレーションのグループである。ここでは、それらのシミュレーションの成果を紹介すると共に VPP で最高の性能を引き出すための一般的注意点などについてもまとめる。 |           |                       |
| ・会場   | 第一回  | 12月20日    | 附属図書館視聴覚ホール           |
|       | 第二回  | 12月24日    | 大型計算機センター・多目的講習室 (3階) |



## 7. 新スーパーコンピュータ VPP700/56 の利用方法について

新スーパーコンピュータ VPP700/56 の利用方法を、VP2600/10 の Fortran 資産を移行するための注意点を中心に解説します。なお、並列化プログラミングの解説、チューニングツールの使用方法などは、今後のセンターニュース、講習会資料、広報で紹介する予定です。

### 1 VPP700/56 の概要

九州大学大型計算機センターの新スーパーコンピュータシステム FUJITSU VPP700/56 は、大規模数値計算を高速に処理するためのベクトル並列型計算機です。VPP700/56 は、個々のプロセッサ (PE : Processing Element) がそれぞれメモリを持ち、他の PE のデータが必要になった場合は、PE 間を相互に結合したネットワークを介して転送を行う分散メモリ型計算機です。VPP700/56 で複数の PE を用いた並列処理を実行したい場合は、並列処理のための命令、データ分割方法、PE 間のデータのやりとりなどを利用者自身でプログラムする必要があります ([3])。なお、VPP700/56 の PE の総数は 56 台ですが、同時に利用できる PE 数は 32 台です。

各 PE は 2GB の主記憶と最大処理性能 2.2GFLOPS のベクトルユニットを搭載しています。1PE あたりの処理性能は VP2600/10 の半分程度ですが、VP2600/10 に比べ 4 倍のメモリを搭載しています。さらに、複数の PE によるジョブ処理によりスループットは格段に向上するはずですが、また、各 PE は最大 570MB/秒の性能を持つ PE 間通信ユニットを 2 台装備しています。

### 2 VPP700/56 の利用方法

当面は、従来の VP2600/10 と同様、汎用計算機 M-1800/20U を経由したバックグラウンドでの利用となります。即ち、汎用計算機 M-1800/20U で作成した Fortran, C プログラムに対し、処理内容を記述したバッチリクエストあるいは JCL を qsub(UXP) あるいは submit(MSP) コマンドで投入します。投入されたプログラムおよびデータは VPP700/56 に送られ、処理結果は汎用計算機に返却されます。

VPP700/56 で動作する言語処理プログラムは以下の通りです。

ソフトウェア名	機能	コマンド
Fortran 90/VP	ベクトル Fortran コンパイラ	frt
Fortran 90/VPP	ベクトル並列 Fortran コンパイラ	frt -Wx
C	C コンパイラ	cc
C/VP	ベクトル C コンパイラ	vcc
C++	C++ コンパイラ	CC

なお、VPP700/56 の対話型処理の利用開始は 1997 年 4 月を予定しています。また、VP2600/10 は 1997 年 2 月まで暫定運用されます。VP2600/10 の利用方法は従来と全く同じです。

### 3 マニュアル

#### 3.1 オンラインマニュアル

対話型の利用ができるまで、VPP700/56 のオンラインマニュアルは汎用計算機 M-1800/20U の UXP システム (ホスト名 kyu-cc) から rsh コマンドを介して参照ください。ホスト名は kyu-vpp です。なお、kyu-cc 以外からの利用はできません。

```
kyu-cc% rsh kyu-vpp man frt | less <---VPP700/56 の frt コマンドの参照
kyu-cc% rsh kyu-vpp man vcc | less <---VPP700/56 の vcc コマンドの参照
```

#### 3.2 マニュアルの購入方法

参考文献にあげた計算機マニュアルはすべて九大生協の書籍部で注文することができます。注文の際はマニュアル番号 (例えば『Fortran90/VP 使用手引書』なら “J2U5-0050”) を必ず指定してください。



## 4 UXP からのジョブの投入

VPP700/56 での計算は、VP2600/10 の利用と同様、「バッチリクエスト」と呼ばれるシェルスクリプトを汎用計算機 M-1800/20U で記述・投入することにより行います。

### 4.1 NQS

#### 4.1.1 バッチキュー

UXP のバッチ処理システムを総称して NQS(Network Queuing System) と呼びます。VPP700/56 には使用できる計算資源に応じてバッチキューが設定されています。1997 年 1 月現在の制限値は以下の通りです。

キュー名	CPU 時間	記憶域	処理形態
c	60 分	100MB	翻訳専用
s	60 分	1.7GB	1PE
p1	1200 分	1.7GB	1PE
p8	1200 分	1.7GB/PE	最大 8PE
p16	1200 分	1.7GB/PE	最大 16PE
p32	1200 分	1.7GB/PE	最大 32PE

翻訳専用の c キューは優先度が高めに設定されています。翻訳レベルのデバッグやベクトル化、並列化の翻訳情報を得るために利用します。並列プログラムの翻訳も可能です。

s, p1 キューは、非並列の 1PE ジョブの翻訳・実行が可能です。従来の Fortran プログラムや C, C++ プログラムなどを投入します。利用者が使用できる記憶領域は 1.7GB です。

p8, p16, p32 キューは並列処理を行うためのキューです。Fortran 90/VPP プログラム、メッセージパッシングライブラリを埋め込んだ Fortran, C プログラムの翻訳・実行を行います。使用できる記憶領域は、PE 数 × 1.7GB です。

#### 4.1.2 ジョブ投入にあたっての注意事項

- 1 月より、汎用機の UXP(ホスト名 kyu-cc) の利用者のホームディレクトリ (例えば /home/usr9/a79999a) に “VPP” というディレクトリが新規に作成されます。VPP は VPP700/56 の利用者ホームディレクトリとシンボリックリンクが張られています。従って、kyu-cc から VPP700/56 へジョブを投入する場合、使用するソースプログラム、データは必ずホームディレクトリ下の “VPP” 下 (例えば /home/usr9/a79999a/VPP) に作成してください。
- 並列ジョブを実行する場合は、必ず翻訳時オプション `-Wx` を指定してください。
- p で始まるキューの CPU 時間の上限は 20 時間です。CPU 時間の打ち切り時間を 20 時間以下に設定したい場合は `qsub` コマンドの `-lT` オプションで指定します。
- VP2600/10 の MSP システムでサポートしていたジョブのリスタート機能は VPP700/56 ではサポートされません。従って、実行中のジョブが定期保守時間などで実行を中断された場合、処理の始めから (自動的に) 再実行となります。
- 制限値を越えるジョブを投入したい場合は、個別対応となります。request@cc.kyushu-u.ac.jp まで連絡ください。

### 4.2 バッチリクエストの記述例

バッチリクエストはエディタでファイルとして作成します。サフィックスに特別な決まりはありませんが、認識し易いように統一することをお勧めします。

#### 4.2.1 標準的な形 (1PE)

```
#                <---csh で記述
cd EXAMPLE      <--- ディレクトリの移動
frt -Ps -Wv,-m3 test.f90  <--- 翻訳
a.out           <--- 実行
```

先頭の # は、バッチリクエストが csh の文章であることを意味します。# を指定しない場合は sh が走ります。次の行の cd は、ディレクトリを VPP700/56 の ~/EXAMPLE に移動するコマンドです (kyu-cc のディレクトリではありません)。

初期ディレクトリはバッチリクエストを投入した場所ではなく、VPP700/56 のホームディレクトリ となります。従って、プログラムを翻訳・実行するディレクトリを必ず指定してください。

次の行は frt コマンドで test.f90 を翻訳する処理です。翻訳オプションとして、ベクトル化メッセージとプログラムリストの出力を指示しています。これらはチューニングのための情報として有益です。また、VPP700/56 の Fortran システムではベクトル化オプションが標準値となっていますので、従来の -J オプションの指定は必要ありません。最後の行は作成した a.out を実行するコマンドです。

#### 4.2.2 Fortran 90/VPP の起動

```
#                <---csh で記述
cd EXAMPLE      <--- ディレクトリの移動
frt -Wx -Ps -Wv,-m3 test.f90  <--- 翻訳
a.out           <--- 実行
```

-Wx オプションの指定により、Fortran 90/VPP が起動され、並列プログラムの翻訳を行います。

#### 4.2.3 ベクトル化レベルを下げて翻訳・実行

```
#
cd EXAMPLE
frt -Ps -Wv,-an,-m3 test.f90  <--- 式の評価順序の変更を抑止
a.out
```

ベクトルモードでは、高速化のために式の評価順序を変更する最適化を行います。その際、丸め誤差の影響から計算誤差が発生することがあります。オプション -Wv,-an の指定により、式の評価順序の変更を抑止した実行結果と比較することで計算誤差の影響をチェックできます。最適化レベルは自動的に -Ob になります。また、実行時間は増大します。

#### 4.2.4 最大限の最適化を行う

```
#
cd EXAMPLE
frt -Of -KVPP700 -Ps -Eipue -Wv,-Of,-Ov,-m3 test.f90  <--- 翻訳. 最大限の最適化
a.out                                                    <--- 実行
```

VPP700/56 用の最大限の最適化を指示します。最適化に伴う副作用の可能性を指摘するオプション (-E) もあわせて指定します。通常は省略値で十分ですが、プログラムによってはかなりの高速化が得られることもあります。

#### 4.2.5 実行ファイルを作成する

```
# <--- csh で記述
cd EXAMPLE <--- ディレクトリの移動
frt -o b.out -Ps -Wv,-m3 test.f90 <--- 翻訳
```

test.f90 を翻訳して、実行ファイル b.out を作成します。実行はしません。

#### 4.2.6 実行のみ

```
#
cd EXAMPLE
b.out <--- 実行のみ
```

既に翻訳が完了している b.out を実行します。このように、翻訳・実行を一つのスクリプトに続けて記述する必要はなく、別個に処理を依頼することもできます。

#### 4.2.7 標準入出力の例

```
#
cd EXAMPLE
a.out < in.data > out.data <--- 実行
```

UNIX のリダイレクション機能はそのまま使えます。装置参照番号 5 番が標準入力 (stdin), 6 番が標準出力 (stdout) に対応しています。例では in.data からデータを読み込み、out.data にデータを書き出します。

#### 4.2.8 環境変数としてのファイル処理

プログラム中に陽にファイルを指定する以外にも、環境変数によって装置参照番号とファイルを対応づけることもできます。環境変数は

fuNN<sub>i</sub> ファイル名

です。NN は 2 桁の装置参照番号 (00 ~ 99) を指定します。ファイル名はパス名込みでも構いません。例では装置参照番号 1 番に ex1.data を対応づけています。

```
# <--- csh で記述
cd EXAMPLE <--- ディレクトリの移動
frt -Ps -Wv,-m3 test.f90 <--- 翻訳
setenv fu01 ex1.data <--- 装置参照番号 1 番に ex1.data を割り当てる
a.out <--- 実行
```

sh の場合は割り当て方法が異なります。バッチリクエストファイルの先頭の “#” が無いことに注意ください。

```
cd EXAMPLE <--- ディレクトリの移動
frt -Ps -Wv,-m3 test.f90 <--- 翻訳
fu01=ex1.data
export fu01 <--- 装置参照番号 1 番に ex1.data を割り当てる
a.out <--- 実行
```

#### 4.2.9 M 形式のファイル処理

VPP700/56 の浮動小数点形式はワークステーションで広く使われている IEEE 形式と呼ばれるものです。一方、汎用計算機 M-1800/20U とベクトル計算機 VP2600/10 の浮動小数点形式は M 形式と呼ばれるものです。M-1800/20U および VP2600/10 で作成した M 形式のバイナリーデータを VPP700/56 で読み込んだり、VPP700/56

の出力する IEEE 形式のバイナリーデータを M-1800/20U で読み込んだりする場合には、2つの形式を実行時オプションによって変換する必要があります。ただし、変換可能な型は実数型、複素数型データのみです。

実行時オプションは -W1 のあとにカンマで区切って指定します。

-W1,-Cuno	装置参照番号 <i>uno</i> からのバイナリーデータの入出力を M 形式で行います。 <i>uno</i> の指定がない場合は、すべての装置参照番号を指定したものとします。
-W1,-M	IEEE ⇄ M 浮動小数点形式の入出力変換後に浮動小数点データの仮数部の一部が 損失した場合、診断メッセージを出力します。

```
#
cd EXAMPLE
b.out -W1,-C,-M          <---b.out を実行。入出力は M 形式
```

b.out を実行する際に、バイナリーデータの入出力を M 形式で行います。

#### 4.2.10 SSL II/VP の結合

```
#
cd EXAMPLE
frt -Ps -Wv,-m3 test.f90 -lssl2vp  <--- 翻訳。SSL II/VP を結合
a.out                               <--- 実行
```

SSL II/VP のサブルーチンを使用している場合は、-lssl2vp を指定します。

#### 4.2.11 NUMPAC の結合

```
#
cd EXAMPLE
frt -Ps -Wv,-m3 test.f90 -lnumpac  <--- 翻訳。NUMPAC を結合
a.out                               <--- 実行
```

NUMPAC のサブルーチンを使用している場合は、-lnumpac を指定します。

#### 4.2.12 SSL II/VPP の結合

```
#
cd EXAMPLE
frt -Wx -Ps -Wv,-m3 test.f90 -lssl2vpp <--- 翻訳。SSL II/VPP を結合
a.out                               <--- 実行
```

SSL II/VPP のサブルーチンを使用している場合は、-lssl2vpp を指定します。

#### 4.2.13 私有ライブラリの結合

```
#
cd EXAMPLE
setenv LD_LIBRARY_PATH /home/usr9/a79999a/MYLIB  <--- 検索パスの指定
frt -Ps -Wv,-m3 test.f90 -lmylib                <--- 翻訳。私有ライブラリを結合
a.out                                           <--- 実行
```

VPP700/56 の ~/MYLIB 下の私有ライブラリ libmylib.a を結合します。検索パスを通していない場合は、フルパスで例えば、

```
frt -Ps -Wv,-m3 test.f90 -L/home/usr9/a79999a/MYLIB -lmylib
```

と指定します。VPP700/56 のホームディレクトリまでのパスは `rsh kyu-vpp pwd` で参照できます。

#### 4.2.14 C/VP の利用

ベクトル版 C(コマンド `vcc`) を利用する場合も、バッチリクエストの記述方法は Fortran と同様です。

```
#
cd EXAMPLE
vcc test.c
a.out
```

オプションは `rsh kyu-vpp man vcc` で参照ください。

### 4.3 バッチリクエストの投入

#### 4.3.1 qsub コマンド

バッチリクエストファイルの処理は `qsub(/usr/bin/qsub)` コマンドで依頼します。形式は

`qsub_options_scriptfile`

です。 `options` はオプションを空白で区切って指定します。 `scriptfile` はバッチリクエストのスクリプトファイル名です。主なオプションは次の通りです。

---

<code>-q_queue</code>	<code>queue</code> はキュー名です。省略すると、VPP700/56 の p1 キューに投入されます。
<code>-mb</code>	バッチリクエストの実行開始をメールで通知します。
<code>-me</code>	バッチリクエストの実行終了をメールで通知します。
<code>-mi</code>	バッチリクエストの統計情報をメールで通知します。
<code>-mu_user</code>	バッチリクエストに関するメールを <code>user</code> に送ります。 <code>user</code> は、九州大学大型計算機センターの利用者であれば login name で、他のマシンに送る場合は @ 付きのアドレスを指定します。
<code>-e_errfile</code>	標準エラー出力 (翻訳結果、timex の情報など) を指定したファイル <code>errfile</code> に出力します。指定がない場合は、リクエストを投入したディレクトリ下に「スクリプトファイル名.e リクエスト番号」という名前のファイルが自動的に作成されます。
<code>-o_outfile</code>	標準出力を指定したファイル <code>outfile</code> に出力します。指定がない場合は、リクエストを投入したディレクトリ下に「スクリプトファイル名.o リクエスト番号」という名前のファイルが自動的に作成されます。
<code>-eo</code>	標準エラー出力を標準出力と同じファイルに出力します。
<code>-lT_time</code>	使用するバッチリクエストの CPU 時間の上限を設定します。設定は [[時間:] 分:] 秒です。 【制限値の指定例】 <code>-lT 10:35:20</code> 10 時間 35 分 20 秒 <code>-lT 12345</code> 12345 秒 <code>-lT 59:45</code> 59 分 45 秒
<code>-lPv</code>	並列処理で使用する PE 数を指定します。

---

メールでの通知オプションとは別に、利用者の課金システムよりバッチリクエストの実行に要した課金情報が実行終了後メールで送付されます。

なお、1月7日から `qsub` コマンドのキューの省略値が `ss`(汎用機) から `p1` に変更されますので、`kyu-cc` の `ss` キューに投入する場合は `-q ss` オプションは省略できなくなります。

#### 4.3.2 qsub コマンドオプションの記述

`qsub` コマンドのオプションは、バッチリクエストファイルに記述することができます。オプションは # @\$ に続けて空白を置かずに指定します。

```

#                               <--- csh で記述
# Q$-eo -me                     <--- qsub オプションの指定例 (1)
# Q$-lT 1:30:00                 <--- qsub オプションの指定例 (2)
cd EXAMPLE                       <--- ディレクトリの移動
firt -Wx -Ps -Wv, -m3 test.f90   <--- 翻訳
a.out                             <--- 実行

```

#### 4.3.3 バッチリクエストの投入例 1

以下、バッチリクエストの記述されたファイル名を a.sh とします。

```

kyu-cc% qsub -q s a.sh           <--- バッチリクエストの投入例 1
Request 5466.kyu-cc submitted to queue: s.

```

汎用計算機 M-1800/20U から VPP700/56 の s キューに投入しました。下線部の “5466” がリクエスト番号，“kyu-cc” がホスト名です。“5466.kyu-cc” でリクエスト名を構成します。実行が終了すると、スクリプトファイル名とリクエスト番号に対応した標準エラー出力ファイル a.sh.e5466 と標準出力ファイル a.sh.o5466 が返ってきます。

```

kyu-cc% ls
test.f90  a.out      a.sh      a.sh.e5466  a.sh.o5466

```

#### 4.3.4 バッチリクエストの投入例 2

```

kyu-cc% qsub -q p1 -o out.data -eo a.sh <--- バッチリクエストの投入例 2
Request 5467.kyu-cc submitted to queue: p1.

```

p1 キューへ投入しました。-q p1 は省略可能です。標準出力ファイル名を out.data に、また標準エラー出力を標準出力と同じファイル out.data に出力します。実行が終了すると、標準エラー出力と標準出力が同一のファイル out.data に格納されて返ってきます。

```

kyu-cc% ls
test.f90  a.out      a.sh      out.data

```

#### 4.3.5 バッチリクエストの投入例 3

```

kyu-cc% qsub -q p16 -me -mi a.sh <--- バッチリクエストの投入例 3
Request 5467.kyu-cc submitted to queue: p16.

```

VPP700/56 の p16 キューへ投入しました。オプションとして、実行終了と統計情報をメールで知らせることを指定しています。

#### 4.3.6 メールを送り先

VPP700/56 で対話型処理のサービスを開始するまでは、汎用計算機 M-1800/20U の UXP/M システム (ホスト名 kyu-cc) にメールが転送されます。

### 4.4 ジョブの状態表示

qsub コマンドにより依頼したバッチリクエストの処理状況は、qstat コマンドによって調べることができます。qstat コマンドはバッチリクエストの処理状況をキューごとに出力します。形式は

qstat\_@machine

です。 *machine* は計算機のマシン名を指定します。なお、1月7日からマシン名の省略値が VPP700/56 のマシン名 *kyu-vpp* に変更されますので、VPP700/56 の処理状況は *qstat* のみで調べることができます。また、*kyu-cc* に投入したスカラージョブの処理状況が知りたい場合は *qstat @kyu-cc* を指定するようになります。

```
kyu-cc% qstat | less  <--- VPP700/56 の処理状況を表示
```

## 4.5 リクエストのキャンセル

### 4.5.1 *qdel* コマンド

誤って投入したバッチリクエストや、これ以上実行する必要がないと判断したジョブのキャンセルは *qdel* コマンドで行います。形式は

```
qdel options request-id
```

です。 *options* はオプションの並び、 *request-id* はリクエスト名です。


リクエスト名は *qstat* コマンドの REQUEST ID で確認することができます。 *qdel* コマンドのオプションは次の通りです。

---


-k 実行中のバッチリクエストに対してキャンセルを行います。 -k の指定がない場合は、実行待ちのリクエストだけをキャンセルします。

---

### 4.5.2 リクエストのキャンセル例

```
kyu-cc% qdel 5478.kyu-cc  <---VPP700/56 の実行待ちリクエストを汎用機からキャンセル  
Request 5478.kyu-cc has been deleted.
```

実行中の場合は -k オプションの指定が必要です。

```
kyu-cc% qdel -k 5479.kyu-cc  <---VPP700/56 の実行中のリクエストを汎用機からキャンセル  
Request 5479.kyu-cc is running, and has been signalled.
```

## 4.6 NQS の注意点

バッチリクエストを投入した段階で、スクリプトに記述したプログラムおよびデータファイルの修正は絶対に行わないでください。 NQS はバッチリクエストが投入された段階でのファイル群のコピーや排他的な処理をしません。従って、リクエストを投入した後でこれらのファイルを修正すると、修正後のファイルを翻訳・実行することがあり、全く異なる結果や異常終了を起こす可能性があります。

その場合、別なディレクトリにコピーを持つか、同じディレクトリの場合でもバッチリクエストを投入する前の段階で別な名前のコピーを持つなどのファイル管理をお願いします。

## 5 MSP からのジョブの投入

### 5.1 M-VPP 連携機能

MSP から VPP700/56 の利用は「M-VPP 連携機能」と呼ばれるソフトウェアを介して行います。

従来の VP2600/10 は UXP と MSP の 2 つの OS が動作していましたが、VPP700/56 では UNIX OS である UXP に一本化されました。 M-VPP 連携機能は、汎用計算機 M-1800/20 の MSP システム (ホスト名 *kyu-msp*) から VPP700/56 の UXP/V システム (ホスト名 *kyu-vpp*) へバッチジョブを投入し、処理結果を MSP へ返すという MSP 利用者のためのシステムです。浮動小数点形式の変換および並列ジョブの翻訳は自動的にオプションが設定されますので、利用者が陽に指定する必要はありません。

## 5.2 制限事項

M-VPP 連携機能には以下の制限があります。なお、制限事項は適宜改善していく予定ですので、以降のセンターニュースに注意願います。

- システム記憶装置 (SSU) はサポートされません。
- ジョブのリスタート機能 (CPR) はサポートされません。
- 翻訳/結合編集/実行を行う STEP=CLG, および翻訳を行う STEP=C の形態でのみ利用できます。ロードモジュールを作成して保存したり, ロードモジュールの実行のみを行うことはできません。
- 使用可能な数値計算ライブラリは SSL II/VP, SSL II/VPP と NUMPAC です。その他のライブラリは移行・代替方法が決まり次第お知らせします。
- PKS や GRAPHMAN などの図形ライブラリは利用できません。汎用機での処理をお願いします。
- 私用ライブラリは利用できません。
- 実行時オプションは指定できません。

## 5.3 ジョブクラス

VPP700/56 専用のジョブクラス W, X, Y, Z を新設します。制限値は以下の通りです。

ジョブクラス名	CPU 時間	記憶域	処理形態
W	60 分	1.7GB	1PE
X	1200 分	1.7GB	1PE
Y	1200 分	1.7GB/PE	最大 8PE
Z	1200 分	1.7GB/PE	最大 16PE

16 台を超える PE を利用した並列ジョブを実行する場合は、UXP を利用ください。

## 5.4 JCL の記述方法

### 5.4.1 カタログドプロシジャ

```
FORTRAN, VPP=YES  
[ , OPT=B|E|F ]  
[ , OPTION='option-list' ]  
[ , UCS=UCS 名 ]  
[ , FCB=FCB 名 ]
```

#### 【パラメータの説明】

VPP=YES	Fortran ジョブの実行 (翻訳/結合編集/実行) を VPP700/56 で行うことを指定
OPT={B E F}	最適化のレベルを指定。省略値は E
OPTION='option-list'	翻訳時オプションを UXP のコマンドライン形式で指定する。
UCS=UCS 名	プリンタ出力時の UCS 名を指定
FCB=FCB 名	プリンタ出力時の FCB 名を指定

### 5.4.2 翻訳時オプションの指定方法

翻訳時オプションを指定する方法は、ソースプログラムの @FORTRAN 行を利用する方法と、OPTION パラメータに UXP の翻訳時オプションリストを記述する方法があります。



```
//A79999AW JOB CLASS=Z
// EXEC FORT,VPP=YES,OPTION='-Wv,-m3'
//FORT.SYSIN DD DSN=A79999A.PROG.FORT,DISP=SHR
//
```

UXP のコマンドラインのオプションは、大文字/小文字を区別します。小文字を入力できない端末の場合は、直前に EBCDIC-ASCII の “¥” を記述します。

```
//A79999AV JOB CLASS=X
// EXEC FORT,VPP=YES
//FORT.SYSIN DD *
@FORTRAN NOADVANCED(NOEVL),DEBUG(SUBCHK)
// DD DSN=A79999A.PROG.FORT,DISP=SHR
//
```

@FORTRAN 行は翻訳されるすべての翻訳単位に対して有効となる翻訳指示行です。なお、UXP のコマンドラインでのオプションと比べ、翻訳指示行で指定できるオプションはかなり制限されています(翻訳指示行オプション形式とコマンドオプション形式との対比表は [2] を参照)。

### 5.4.3 データセットの指定方法

ジョブの実行に必要なデータセット(プログラム、データ)は、DD 文で指定します。インクルードデータセット FORT.SYSINC および OPTION='ELM(\*)' を指定した場合の FORT.SYSIN にのみ区分編成データセットの指定が許されます。その他のデータセットには順編成データセット、または、メンバ名指定の区分編成データセットを指定してください。

DD 文で指定したデータセットは、連携機能の標準的な解釈によりシステム間で自動的に転送されます。この解釈が適当でない場合は、DD 文の DATATYPE パラメータでデータ形式および転送方法を指定してください。

#### 【DATATYPE パラメータ】

$$\text{DATATYPE} = \left\{ \begin{array}{l} \text{SRC} \\ \text{TXT} \\ \text{FBIN} \\ \text{BIN} \end{array} \right\}, \left\{ \begin{array}{l} \text{CNV} \\ \text{NOCNV} \end{array} \right\}, \left\{ \begin{array}{l} \text{EXP} \\ \text{NOEXP} \end{array} \right\}, \left\{ \begin{array}{l} \text{IMP} \\ \text{NOIMP} \end{array} \right\}$$

#### 【パラメータの説明】

- 第1パラメータは指定したデータセットのデータ形式を指定します。

SRC	ソースプログラム
TXT	書式付き順次入出力データ
FBIN	書式なし順次入出力データ
BIN	書式なし直接入出力データ

FORT.SYSIN または FORT.SYSINC で指定したデータセットは SRC、その他のデータセットは TXT が省略値となります。

- 第2パラメータは文字コードの変換 (EBCDIC-ASCII, JEF ⇔ ASCII, EUC) の有無を指定します。

CNV	文字コードを変換する
NOCNV	文字コードを変換しない

データ形式が SRC または TXT の場合は CNV、FBIN または BIN の場合は NOCNV が省略値となります。

- 第3パラメータは MSP から UXP へのデータセットの転送の有無を指定します。

EXP	データセットを転送する
NOEXP	データセットを転送しない

DISP パラメータに OLD または SHR を指定した場合は EXP, NEW または MOD の場合は NOEXP が省略値となります。

- 第4パラメータは UXP から MSP へのデータセットの転送の有無を指定します。

IMP	データセットを転送する
NOIMP	データセットを転送しない

データセットが UXP で更新された場合は IMP が省略値となります。

#### 【パラメータの指定例】

- データセットのデータ形式が書式なし順次入出力データの場合。

```
//GD.FT01F001 DD DSN=A79999A.IN.DATA,DISP=SHR,DATATYPE=(FBIN)
```

- データセットのデータ形式が書式なし直接入出力データの場合。

```
//GD.FT02F001 DD DSN=A79999A.IO.DATA,DISP=SHR,DATATYPE=(BIN)
```

- 既存のデータセットにデータを出力する場合 (MSP から UXP への転送が必要ない場合)。

```
//GD.FT03F001 DD DSN=A79999A.OUT.DATA,DISP=SHR,DATATYPE=(,NOEXP)
```

## 5.5 JCL の記述例

以下、ジョブ制御文 (JCL) の例を示し、従来の制御文と連携用の制御文との変更点を説明します。

### ●従来の JCL 例

汎用計算機 M-1800/20U から VP2600/10 へ翻訳/結合編集/実行処理を依頼する従来の JCL の例です。

```
//A79999A1 JOB CLASS=V
// EXEC FORT,VP=YES,OPTION='NOSOURCE'
//FORT.SYSIN DD DSN=A79999A.PROJ.FORT,DISP=SHR
//GO.SYSIN DD *
5 5 3600
//GO.FT01F001 DD DSN=A79999A.ARRAY.DATA,DISP=SHR
//GO.FT02F001 DD DSN=A79999A.REPORT.DATA,DISP=SHR
//GO.FT03F001 DD DSN=A79999A.NEWA.DATA,DISP=(NEW,CATLG),UNIT=PUB,
// SPACE=(TRK,(100,10),RLSE)
//
```

### ●連携用 JCL

汎用計算機 M-1800/20U から VPP700/56 へ翻訳/結合編集/実行処理を依頼する連携用 JCL の例です。

```

//A79999A1 JOB CLASS=X (1)
// EXEC FORT,VPP=YES (2)
//FORT.SYSIN DD * (3)
@FORTRAN NOSOURCE (3)
// DD DSN=A79999A.PROJ.FORT,DISP=SHR
//GO.SYSIN DD *
5 5 3600
//GO.FTO1F001 DD DSN=A79999A.ARRAY.DATA,DISP=SHR,DATATYPE=(FBIN) (5)
//GO.FTO2F001 DD DSN=A79999A.REPORT.DATA,DISP=SHR (5)
//GO.FTO3F001 DD DSN=A79999A.NEWA.DATA,DISP=(NEW,CATLG),UNIT=PUB, (5)
// SPACE=(TRK,(100,10),RLSE),DATATYPE=(FBIN) (5)
//

```

【説明】

- (1) 連携ジョブはジョブクラス W, X, Y または Z を指定します。
- (2) 連携ジョブはカタログプロシジャ FORT の “VPP=YES” パラメータを指定します。
- (3) 翻訳時オプションを指定する “OPTION=” パラメータは UXP のコマンドラインでのオプションか “@FORTRAN” 行を利用して指定してください。
- (4) テキスト形式以外のデータセットの場合は “DATATYPE” パラメータでデータ形式を指定します。

5.6 実行状況および結果の確認

連携ジョブの状態は STATE コマンドで確認できます。実行結果は、通常のジョブと同様に MSO コマンドなどで確認できます。連携ジョブの場合は、ジョブログ部分に KKA で始まる識別子を持つメッセージが出力されます。また、メッセージ KKA002I で UXP 側で消費した CPU 時間を知ることができます。

5.7 連携ジョブのキャンセル

キャンセル方法は従来のジョブと同じです。CANCEL コマンドを使用してください。

```

READY
CANCEL A79999AX(J1234)

```

6 VP2600/10 からの移行

6.1 VP2600/10 と VPP700/56 の差異

VP2600/10 と VPP700/56 では、以下の点が大きく異なります。

	VPP700/56	VP2600/10
計算機	ベクトル並列計算機 (1PE での実行も可能)	ベクトル計算機 (1PE)
OS	UXP/V	MSP, UXP/M
浮動小数点形式	IEEE 形式	M 形式 (IBM 形式)
Fortran	Fortran 90 (+ 富士通並列化仕様)	FORTTRAN 77 (+ 富士通仕様)
最大記憶域	1PE あたり 1.7GB* (最大 1.7×32GB)	400MB*
理論ピーク性能	1PE あたり 2.2GFLOPS (最大 2.2×32GFLOPS)	4.9GFLOPS

\*) 利用者が確保できる最大リージョンサイズです。

## 6.2 浮動小数点形式

VPP700/56 は浮動小数点を IEEE 形式で表現します。一方、VP2600/10, M-1800/20U は M 形式 (IBM 形式) です。単精度および倍精度型データに対する表現方法の違いは以下の通りです。

型名		IEEE 形式	M 形式
単精度型	指数部	8 ビット	7 ビット
	仮数部 *	24 ビット	24 ビット
倍精度型	指数部	11 ビット	7 ビット
	仮数部 *	53 ビット	56 ビット

\* 隠れビットを含む。

指数部で IEEE 形式は M 形式に比べより広い数を表現できます。その代わり仮数部では倍精度型で 3 ビット少なくなります。しかし、10 進の精度で見ればほぼ M 形式と同じ 15 桁が表現できます。IEEE 形式は、ほとんどのワークステーションで採用されている浮動小数点形式で、ワークステーション上で書式なし出力文によって作成したデータをそのまま VPP700/56 で読み込むことが可能です。

### 6.2.1 M 形式と IEEE 形式の変換

実行時オプション `-W1,-C` を指定することで、書式なし入出力文実行時に READ 文で M 形式から IEEE 形式に、WRITE 文で IEEE 形式から M 形式にデータを変換することができます。汎用計算機 M-1800/20U (M 形式) で `-C` を指定した場合はこの逆の変換になります。

ただし、変換可能な型は、実数型、倍精度実数型、複素数型、倍精度複素数型のみです。それ以外の型がデータの入出力並びに指定されている場合は形式の変換ができませんのでご注意ください。

また、MSP からジョブを投入する場合、変換は自動的に行われます。

VPP700/56 上で作成された実行ファイルを `a.out` とすると、シェルスクリプトの例は

```
#
frt -Ps -Wv,-m3 test.f90
setenv fu01 input.data
setenv fu02 output.data
a.out -W1,-C          <--- すべての入出力を M 形式で行う
```

と記述します。

`-C` に続けて装置番号を指定すると、その番号のみの入出力が M 形式で行われます。M 形式で作成されたバイナリデータを装置番号 1 番から読み込んで、処理結果を IEEE 形式で装置番号 2 番に書き出したい時は、次のように書きます。

```
#
frt -Ps -Wv,-m3 test.f90
setenv fu01 input.data
setenv fu02 output.data
a.out -W1,-C1        <--- 装置番号 1 番からの入出力を M 形式で行う
                       2 番の入出力は IEEE 形式
```

また、Fortran 90/VP システムでサポートされている IETOM, MTOIE サービスサブルーチンによる浮動小数点形式の変換も可能です。引用形式は以下の通りです。

```
CALL IETOM(R1,R2,TYPE,RETCD)
```

- R1 : 変換前の IEEE 形式データ. スカラー変数または配列要素名.  
 R2 : 変換後の M 形式データ. スカラー変数または配列要素名.  
 TYPE : データの型を指定.  
       0= 実数型, 1= 倍精度実数型  
 RETCD : 復帰コード. 4 バイト整数型.  
       0= 正常終了, 4= 仮数部が最大 3 ビット損失,  
       8= 指数オーバーフロー, または指数アンダーフロー, または非数値データが検出された.  
       12= 引数 TYPE の指定に誤りがある.

CALL MTOIE(R1,R2,TYPE,RETCD)

- R1 : 変換前の M 形式データ. スカラー変数または配列要素名.  
 R2 : 変換後の IEEE 形式データ. スカラー変数または配列要素名.  
 TYPE : データの型を指定. IETOM と同じ.  
 RETCD : 復帰コード. IETOM と同じ.

### 6.2.2 精度の損失

実行時 `-Wv`, `-C` オプションによる浮動小数点データの変換では, もとからの仮数部と指数部の持つ情報量の差から, 以下のような精度の損失が起きる可能性があります.

型名		READ(M → IEEE)	WRITE(IEEE → M)
実数型	指数部	$10^{-37} \sim 10^{38}$ の範囲で変換可能	変換可能
	仮数部		最大 3 ビットを損失する可能性がある
倍精度実数型	指数部	変換可能	$10^{-78} \sim 10^{75}$ の範囲で変換可能
	仮数部	最大 3 ビットを損失する可能性がある	

複素数型は実部と虚部それぞれに上の型に応じた情報の損失が起こる可能性があります. 実行時オプション `-W1`, `-M` を指定すると, 変換処理の過程で生じたビット損失, オーバーフローなどに関するメッセージが出力されます.

a.out -W1,-C,-M <--- ビット損失の検査

### 6.3 ハードウェアの違いによる影響

VPP700/56 と VP2600/10 とは, ハードウェアの違いから実行結果が異なる可能性があります. 例えば, かなり厳しい (マシンイプシロンに近い) 収束条件を IF 文に組み込んでいるような精度に敏感なプログラムでは, 実行結果が影響を受ける可能性があります.

また, VP2600/10 と VPP700/56 はバイナリレベルの互換性はありませんので, オブジェクトファイル, 実行ファイルは再作成となります.

### 6.4 言語仕様レベルによる差異

VPP700/56 の Fortran システムは, Fortran 90, FORTRAN 77, および FORTRAN IV の言語仕様に対応しています. ただし, 並列プログラムを実行する場合は, FORTRAN IV 言語仕様は利用できません.

UXP では, 言語レベルはサフィックスにより自動的に決まります. また, 翻訳時オプションで指定することも可能です.

言語仕様	解釈条件
Fortran 90	ファイルのサフィックスが <code>.f90</code> 翻訳時オプション <code>-X9</code> を指定
FORTRAN 77	ファイルのサフィックスが <code>.f</code> 翻訳時オプション <code>-X7</code> または <code>-Xf7</code> を指定
FORTRAN IV	翻訳時オプション <code>-X6</code> または <code>-Xf6</code> または <code>-v</code> を指定

## 6.5 FORTRAN77 EX/VP との互換性

### 6.5.1 動作を保証するには

VP2600/10 の FORTRAN システムである FORTRAN77 EX/VP システムの動作を Fortran 90/VP でも保証したい場合は、翻訳時オプションとして `-Xf7` を指定します。

```
frt -Xf7 test.f90 <---FORTRAN77 EX/VP との互換性を保証
```

ただし、この互換性は言語の解釈の互換性であって、ハードウェアの違いから生じる精度の違いは保証できません。

### 6.5.2 サポートされない機能

VP2600/10 と VPP700/56 のシステムの機能の違いによってサポートされなくなった機能で特に重要なものを以下に列挙します。

#### ●一次回帰演算のベクトル化

DO ループの 1 回前の実行で定義された値を用いた一次回帰演算は VP2600/10 ではベクトル化されていましたが、VPP700/56 ではベクトル化されません。

#### ●デバッグ機能

デバッグオプションは新システムでは `-Da`、`-Ds`、`-Du` のみがサポートされ、それ以外のデバッグオプション `-Db`、`-Di`、`-Do`、`-Dt`、`-Dv` は使えなくなります。

#### ●CLOCK, CLOCKV の引数

実行開始からの CPU 占有時間を返却する CLOCK サブルーチン、および CPU 占有時間とベクトルユニット占有時間を返却する CLOCKV サブルーチンの引数は、これまでで省略可能なものがありましたが、Fortran 90/VP では省略できなくなりました。引数を省略した場合は、実行時に異常終了します。

#### ●翻訳時間

Fortran 90/VP は、従来のベクトル化や最適化にプラスして、スカラープロセッサ向けの最適化を行っています。そのため、翻訳時間は一般的に VP2600/10 と比較して長くなります。また、実行ファイルも多少大きくなります。

## 参考文献

- [1] UXP/V Fortran 90/VP 使用手引書 V10 用, J2U5-0050, 富士通株式会社 (1995).
- [2] UXP/V Fortran 90/VP メッセージ説明書 V10 用, J2U5-0060, 富士通株式会社 (1996).
- [3] UXP/V Fortran 90/VPP 使用手引書 V10 用, J2U5-0080, 富士通株式会社 (1995).
- [4] UXP/V C/VP 使用手引書 V10 用, J2U5-0120, 富士通株式会社 (1995).
- [5] UXP/V C 言語使用手引書 V10 用, J2U5-0110, 富士通株式会社 (1995).
- [6] FUJITSU C++ 言語システム説明書, J2X0-0670, 富士通株式会社 (1994).
- [7] UXP/V アナライザ使用手引書 V11 用, J2U5-0130, 富士通株式会社 (1995).
- [8] 富士通 SSL II 使用手引書 (科学用サブルーチンライブラリ), 99SP-4020, 富士通株式会社 (1987).
- [9] FUJITSU SSL II 拡張機能使用手引書 (科学用サブルーチンライブラリ), 99SP-4070, 富士通株式会社 (1991).
- [10] FUJITSU SSL II 拡張機能使用手引書 II (科学用サブルーチンライブラリ), J2X0-1360, 富士通株式会社 (1995).

[11] FUJITSU SSL II/VPP 使用手引書 (科学用サブルーチンライブラリ)V11 用, J2X0-1370, 富士通株式会社 (1996).

[12] NUMPAC 利用手引書, 富士通株式会社 (1994).

(ライブラリ室 ダイヤルイン 092-642-2295)

e-mail : f70029a@kyu-cc.cc.kyushu-u.ac.jp