

AOBA-A

プログラム開発・実行留意事項

日本電気株式会社

2021年4月6日

2.0版

目次

1. はじめに	3
2. システムの主な違い.....	3
1) システム構成	3
2) コンパイラ仕様.....	3
3. プログラミングにおける留意事項.....	3
1) データ形式（エンディアン）	3
2) 変数	5
4. 開発・実行における留意事項	6
1) コンパイルコマンド.....	6
2) コンパイルオプション.....	6
3) コンパイラ指示行	7
4) コンパイラ指示行変換ツール	7
5) 実行時環境変数	7
5. 数値計算ライブラリ利用における留意事項	8

1. はじめに

本資料は、前大規模科学計算システムベクトルスーパーコンピュータ SX-ACE システムで利用していたプログラムを、AOBA-A で利用するためのプログラミングに関する留意事項を説明したものです。本説明は、Fortran プログラムを対象としています。

2. システムの主な違い

1) システム構成

AOBA-A システムは、前システム SX-ACE と同じくベクトルアーキテクチャを継承しています。アプリケーション演算処理を行うベクトルエンジン(VE) が、OS 処理を行うベクトルホスト(VH)に8枚搭載されています。

VE 内には、8つのコアが搭載され、48GBのメモリを共有しています。共有メモリ並列プログラムの実行では、最大8並列、メモリ48GBまで利用可能となります。

VE内のコアおよび、他のVEのコアを使用した、分散並列プログラムの実行は、MPIライブラリを利用することができます。

システム	共有メモリ並列		分散並列	
	最大並列数	最大使用メモリ量	最大並列数	最大使用メモリ量
AOBA-A	8	48GB	2048	12,288GB
SX-ACE	4	64GB	4096	65,536GB

2) コンパイラ仕様

AOBA-A では、Fortran、C および C++ コンパイラを利用できます。

各コンパイラともに、前システム SX-ACE と同様に、自動ベクトル化・自動並列化機能および MPI 分散並列ライブラリを利用することができます。

3. プログラミングにおける留意事項

1) データ形式 (エンディアン)

AOBA-A ではエンディアン形式が変わり、x86 系システムと同じ、リトルエンディアンとなります。x86 系システムで作成したバイナリデータがそのまま扱えます。

システム	データ形式(エンディアン)
AOBA-A	リトルエンディアン
SX-ACE	ビッグエンディアン

前システム SX-ACE で形式を指定していないバイナリデータ(ビッグエンディアン形式データ)を利用する場合には、実行時環境変数 `VE_FORT_UFMTENDIAN` を用いて形式指定をする必要があります。

尚、前システム SX-ACE でリトルエンディアン形式で出力したデータ(実行時環境変数 `F_UFMTENDIAN`)は、実行時環境変数の設定が不要となります。

実行時環境変数 `VE_FORT_UFMTENDIAN` の指定方法は次の通りです。

◆ sh 形式の場合の指定方法

引数	例・説明
ALL	<code>export VE_FORT_UFMTENDIAN=ALL</code> 全ての装置番号をビッグエンディアンとする
番号,番号	<code>export VE_FORT_UFMTENDIAN=10,11</code> 装置番号 10 番と 11 番をビッグエンディアンとする
番号-番号 (連続の範囲)	<code>export VE_FORT_UFMTENDIAN=10-12</code> 装置番号 10 番から 12 番をビッグエンディアンとする
big: 番号 little: 番号 と ; の除外	<code>export VE_FORT_UFMTENDIAN=big;little:10-12</code> 装置番号 10 番から 12 番以外をビッグエンディアンとする (;前の指定から、除外する装置番号を後ろに指定する)

◆ csh 形式の場合の指定方法

引数	例・説明
ALL	<code>setenv VE_FORT_UFMTENDIAN ALL</code> 全ての装置番号をビッグエンディアンとする
番号,番号	<code>setenv VE_FORT_UFMTENDIAN 10,11</code> 装置番号 10 番と 11 番をビッグエンディアンとする
番号-番号 (連続の範囲)	<code>setenv VE_FORT_UFMTENDIAN 10-12</code> 装置番号 10 番から 12 番をビッグエンディアンとする
big: 番号 little: 番号 と ; の除外	<code>setenv VE_FORT_UFMTENDIAN big;little:10-12</code> 装置番号 10 番から 12 番以外をビッグエンディアンとする (;前の指定から、除外する装置番号を後ろに指定する)

2) 変数

AOBA-A では、前システム SX-ACE と Fortran プログラムのローカル変数のメモリ上の割り付け領域が変更となります。

これにより、ローカル変数の内容状態が変わるため、実行結果が変わる、または実行が異常終了となる場合があります。

システム	ローカル変数割り付け領域	特性
AOBA-A	スタック領域	初期値は不定となる。 save 属性を指定しないローカル(局所)変数は手続きが終了したら不定になる。 (save 属性漏れ、初期化漏れのローカル(局所)変数がある場合、移行後に結果相違、実行エラー等が発生する可能性がある。)
SX-ACE	bbs 領域	スタティックな領域。手続き実行でセットされた値が以降も保持される。 初期値なしスタティック領域であるが、実行に先立ち 値 0(論理型は FALSE)に初期化される。 (ローカル(局所)変数であっても手続き終了後も値が残っている。)

◆ 確認方法

前システム SX-ACE で利用していたプログラムで、SX-ACE とプログラム結果が異なる、実行が異常終了する場合には、上記の影響も考えられます。

次の確認を行うことで、結果が同じとなる、実行が正常に終了する場合には、ローカル変数の扱いの影響(プログラムにおけるローカル変数の初期化漏れ)が考えられます。

① コンパイルオプションを設定

コンパイルオプション	説明
-bss	局所変数を .bss セクションに割り付ける。
-save	各プログラム単位において(RECURSIVE であるものは例外とする)、SAVE 文がすべての局所変数に指定されているものとする

-minit-stack=zero	実行時にスタックに割り付ける領域を 0(ゼロ)で初期化する
-------------------	-------------------------------

② 実行時環境変数を指定し実行

実行時環境変数	説明
VE_INIT_HEAP=ZERO	実行時にヒープに割り付ける領域を 0(ゼロ)で初期化する

4. 開発・実行における留意事項

1) コンパイルコマンド

AOBA-A では、前システム SX-ACE とコンパイルコマンドが異なります。

システム	Fortran	C	C++
AOBA-A	nfort	ncc	nc++
(MPI プログラム)	mpinfort	mpincc	mpinc++
SX-ACE	sxf90	sxcc	sxc++
(MPI プログラム)	sxmpif90	sxmpicc	sxmpic++

コンパイル方法の詳細については、別途、プログラム開発と実行に関する資料を参照下さい。

2) コンパイルオプション

AOBA-A のコンパイラと、前システム SX-ACE のコンパイラでは、コンパイルオプションが異なります。主なオプションの違いは次の通りです。

機能	AOBA-A コンパイラ	SX-ACE コンパイラ
最適化レベル	-O3	-Chopt
	-O2	-Cvopt
自動並列化	-mparallel	-Pauto
OpenMP 並列化	-fopenmp	-Popenmp
自動インライン展開	-finline-functions	-pi auto
自動インライン展開をしない	-fno-inline-functions	-Npi
バウンズチェック	-fbounds-check	-eR
ループアンロール	-floop-unroll	-O unroll
編集リスト出力	-report-format	-L fmtlist

コンパイラオプションの違いの詳細については、以下のドキュメント(付録 B)を参照下さい。

<https://www.hpc.nec/documentation>
(Fortran Compiler ユーザーズガイド)

3) コンパイラ指示行

AOBA-A のコンパイラと、前システム SX-ACE のコンパイラでは、コンパイラの最適化を指示するコンパイラ指示行が異なります。

コンパイラ指示行の違いについては、以下のドキュメント(付録 B)を参照下さい。

<https://www.hpc.nec/documentation>
(Fortran Compiler ユーザーズガイド)

4) コンパイラ指示行変換ツール

前システム SX-ACE のコンパイラ指示行を、AOBA-A のコンパイラ指示行に変換するツール `nfdirconv` を提供しています。

```
$ nfdirconv sample.f
```

ツールの詳細については、以下のドキュメント(付録 C)を参照下さい。

<https://www.hpc.nec/documentation>
(Fortran Compiler ユーザーズガイド)

5) 実行時環境変数

AOBA-A のコンパイラと、前システム SX-ACE のコンパイラでは、性能情報の出力などを指定する、実行時環境変数が異なります。

主な実行時環境変数の違いは次の通りです。

機能	AOBA-A コンパイラ	SX-ACE コンパイラ
実行解析情報の出力	VE_PROGINF	F_PROGINF
入出力バッファサイズ指定	VF_FORT_SETBUF	F_SETBUF
ファイル装置接続	VE_FORT <i>n</i>	F_FF <i>n</i>
ビッグエンディンモード指定	VE_FORT_UFMTENDIAN	F_UFMTENDIAN

実行時環境変数の違いの詳細については、以下のドキュメント(付録 B)を参照下さい。

<https://www.hpc.nec/documentation>
(Fortran Compiler ユーザーズガイド)

5. 数値計算ライブラリ利用における留意事項

AOBA-A では、数値計算ライブラリとして、NLC(NEC Numeric Library Collection)が利用できます。NLCは、前システム SX-ACE で提供されていた 数値計算ライブラリ ASL、LAPACK などの数学ライブラリの機能を包含したライブラリです。

ライブラリの有するサブルーチンを組み込んだプログラムをコンパイルする場合には、NLC を呼び出すコンパイルオプション(リンクオプション)を指定します。前システム SX-ACE と指定方法が異なります。

①コンパイル環境設定の実施

プログラムのコンパイルを行う前に、コンパイル環境設定スクリプトを実行します。

例(sh 形式の場合) :

```
$ source /opt/nec/ve/nlc/2.3.0/bin/nlcvars.sh
```

例(csh 形式の場合) :

```
% source /opt/nec/ve/nlc/2.3.0/bin/nlcvars.csh
```

尚、コンパイルするプログラムの分散並列の有無、および整数型のサイズによって、本スクリプトに引数を付けて実行する必要があります。

分散(MPI)並列	プログラムでの既定値整数型サイズ	スクリプト指定引数
無	32bit(kind=4)	(なし)
無	64bit(kind=8)	i64
有	32bit(kind=4)	mpi
有	64bit(kind=8)	mpi i64

例) 64bit 整数型を使用した、MPI プログラムをコンパイルする場合(sh 形式の場合)

```
$ source /opt/nec/ve/nlc/2.3.0/bin/nlcvars.sh mpi i64
```

例) 64bit 整数型を使用した、MPI プログラムをコンパイルする場合(csh 形式の場合)

```
% source /opt/nec/ve/nlc/2.3.0/bin/nlcvars.csh mpi i64
```

②コンパイル実行

・ASL ライブラリを利用する例

```
----- 逐次実行プログラムをコンパイルする場合(フロントエンドサーバで実行)
$ nfort sample.f -lasl_sequential_i64

----- OpenMP 並列プログラムをコンパイルする場合(フロントエンドサーバで実行)
$ nfort -fopenmp sample.f -lasl_openmp_i64

----- 64bit 整数型を使用した、MPI 並列プログラムをコンパイルする場合
      (フロントエンドサーバで実行)
$ mpinfort sample.f -fdefault-integer=8 -fdefault-real=8 -lasl_mpi_sequential_i64f
```

・LAPACK ライブラリを利用する例

```
----- 逐次実行プログラムをコンパイルする場合(フロントエンドサーバで実行)
$ nfort sample.f -llapack_i64 -lblas_sequential_i64

----- OpenMP 並列プログラムをコンパイルする場合(フロントエンドサーバで実行)
$ nfort -fopenmp sample.f -llapack_i64 -lblas_openmp_i64
```

NLC の詳細および利用方法は、以下を参照下さい。

<https://www.hpc.nec/documentation>

(NLC (NEC Numeric Library Collection) ユーザーズガイド)