

インテル® Fortran コンパイラー 16.0 for Linux* リリースノート (インテル® Parallel Studio XE 2016)

2015 年 12 月 7 日時点

このドキュメントは、インテル® デベロッパー・ゾーンに公開されている「[Intel Fortran Compiler 16.0 Update 1 for Linux* Release Notes for Intel Parallel Studio XE 2016](#)」の日本語参考訳です。

このドキュメントでは、新機能、変更された機能、注意事項、および製品ドキュメントに記述されていない既知の問題について説明します。

詳細は、パッケージに含まれるライセンスと本リリースノートの「著作権と商標について」を参照してください。本リリースのインテル® Fortran コンパイラー 16.0 についての詳細は、次のリンクを参照してください。

- [動作環境](#)
- [使用方法](#)
- [ドキュメント](#)
- [サンプル](#)
- [テクニカルサポート](#)
- [互換性](#)
- [新機能と変更された機能](#)
- [新規および変更されたコンパイラー・オプション](#)
- [終了予定のサポート](#)
- [終了したサポート](#)
- [既知の問題](#)
- [Fortran 2008 および Fortran 2015 機能の概要](#)
- [著作権と商標について](#)

変更履歴

このセクションでは製品アップデートにおける重要な変更内容を説明します。

Update 1 (インテル® Fortran コンパイラー 16.0.1)

- [omp declare simd linear 節の新しい修飾子](#)
- [OpenMP* 4.1 の機能を追加サポート](#)
- [報告された問題を修正](#)
- [ドキュメントを更新](#)

インテル® Fortran コンパイラー 15.0 以降 (インテル® Fortran コンパイラー 16.0.0 での変更)

- OpenMP* 4.1 のディレクティブ
- 異なる型/種別型値を持つ型仕様のない配列コンストラクターの診断
- ATTRIBUTES FASTMEM ディレクティブ
- OpenMP* TR3 Proposal で定義されている非同期オフロードおよびデバイス仕様のサポート
- Linux* 分割 DWARF デバッグ情報 (DWARF Fission) をサポート
- BLOCK_LOOP および NOBLOCK_LOOP ディレクティブ、unroll_and_jam ディレクティブの private 節を追加
- Debian* 6.0 のサポートを終了
- OpenMP* 4.1 の機能を追加サポート
- 新規および変更されたコンパイラー・オプション
- Fortran 2008 の機能をサポート
- Fortran 2015 の機能をサポート
- 報告された問題を修正

動作環境

アーキテクチャー名についての説明は、「[インテル® アーキテクチャー・プラットフォームの用語](#)」(英語)を参照してください。

- インテル® ストリーミング SIMD 拡張命令 2 (インテル® SSE2) 対応の IA-32 またはインテル® 64 アーキテクチャー・プロセッサをベースとするコンピューター (インテル® Pentium® 4 プロセッサ以降、または互換性のあるインテル以外のプロセッサ)
 - 64 ビット・アプリケーションおよびインテル® Xeon Phi™ コプロセッサに作業をオフロードするアプリケーションの開発は、64 ビット・バージョンの OS でのみサポートしています。32 ビット・アプリケーションの開発は、32 ビット・バージョンまたは 64 ビット・バージョンの OS のいずれかでサポートしています。
 - 64 ビット・バージョンの OS で 32 ビット・アプリケーションを開発する場合は、Linux* ディストリビューションからオプションのライブラリー・コンポーネント (ia32-libs、lib32gcc1、lib32stdc++6、libc6-dev-i386、gcc-multilib、g++-multilib) をインストールする必要があります。
- 機能を最大限に活用できるよう、マルチコアまたはマルチプロセッサ・システムの使用を推奨します。
- RAM 2GB (4GB 推奨)
- 4GB のディスク空き容量 (すべての機能をインストールする場合)
- インテル® Xeon Phi™ コプロセッサの開発/テストの場合:
 - インテル® Xeon Phi™ コプロセッサ
 - インテル® メニーコア・プラットフォーム・ソフトウェア・スタック (インテル® MPSS)
- IA-32 対応アプリケーションまたはインテル® 64 対応アプリケーションを開発する場合は、次の Linux* ディストリビューションのいずれか (本リストは、

インテル社により動作確認が行われたディストリビューションのリストです。その他のディストリビューションでも動作する可能性はありますが、推奨しません。ご質問は、[テクニカルサポート](#)までお問い合わせください。)

- Debian* 7.0、8.0
- Fedora* 21、22
- Red Hat* Enterprise Linux* 5、6、7
- SUSE* Linux* Enterprise Server 11、12
- Ubuntu* 12.04 LTS (64 ビットのみ)、13.10、14.04 LTS、15.04
- インテル® Cluster Ready
- Linux* 開発ツール・コンポーネント (gcc、g++ および関連ツールを含む)。(本リストは、インテル社により動作確認が行われたコンポーネント・バージョンのリストです。その他のバージョンでも動作する可能性はありますが、推奨しません。ご質問は、[テクニカルサポート](#)までお問い合わせください。)
 - gcc 4.3-5.1
 - binutils 2.20-2.25
- -traceback オプションを使用するには、libunwind.so が必要です。一部の Linux* ディストリビューションでは、別途入手して、インストールする必要があります。

注

- インテル® コンパイラーは、さまざまな Linux* ディストリビューションと gcc バージョンで動作確認されています。一部の Linux* ディストリビューションには、動作確認されたヘッダーファイルとは異なるバージョンのものが含まれており、問題を引き起こすことがあります。使用する glibc のバージョンは、gcc のバージョンと同じでなければなりません。最良の結果を得るため、上記のディストリビューションで提供されている gcc バージョンのみを使用してください。
- インテル® コンパイラーは、デフォルトで、インテル® SSE2 命令対応のプロセッサ (例: インテル® Pentium® 4 プロセッサ) が必要な IA-32 アーキテクチャー・アプリケーションをビルドします。コンパイラー・オプションを使用して任意の IA-32 アーキテクチャー・プロセッサ上で動作するコードを生成できます。
- 非常に大きなソースファイル (数千行以上) を -O3、-ipo および -qopenmp などの高度な最適化オプションを使用してコンパイルする場合は、多量の RAM が必要になります。
- 一部の最適化オプションには、アプリケーションを実行するプロセッサの種類に関する制限があります。詳細は、オプションの説明を参照してください。

インテル® メニーコア・プラットフォーム・ソフトウェア・スタック (インテル® MPSS)

インテル® メニーコア・プラットフォーム・ソフトウェア・スタック (インテル® MPSS) は、インテル® Xeon Phi™ コプロセッサを使用するアプリケーションを開発している場合、インテル® Fortran コンパイラーのインストール前またはインストール後にインストールできます。

最新バージョンのインテル® MPSS を使用することを推奨します。インテル® Parallel Studio XE for Linux* を登録すると、インテル® ソフトウェア開発製品レジストレーション・センター (<http://registrationcenter.intel.com>) から入手できます。

ユーザー空間およびカーネルドライバのインストールに必要な手順については、インテル® MPSS のドキュメントを参照してください。

インテル® Fortran コンパイラーの使用方法

インテル® Fortran コンパイラーの使用方法についての情報は、『入門ガイド』(<install-dir>/documentation_2016/ps2016/getstart_comp_lf.htm) に含まれています。

ドキュメント

製品ドキュメントは、<install-dir>/documentation_2016/ja/ps2016/getstart_comp_lf.htm からリンクされています。

サンプル

製品サンプルは、/opt/intel/samples_2016/en/compiler_f/psxe ディレクトリーにあります。

テクニカルサポート

インテル® ソフトウェア開発製品レジストレーション・センターでライセンスを登録してください。登録を行うことで、サポートサービス期間中 (通常は 1 年間)、製品アップデートと新しいバージョンの入手を含む無償テクニカルサポートが提供されます。

テクニカルサポート、製品のアップデート、ユーザーフォーラム、FAQ、ヒント、およびその他のサポート情報は、<http://www.intel.com/software/products/support/> (英語) を参照してください。

注: 販売代理店がこの製品のテクニカルサポートを提供している場合、インテルではなく販売代理店にお問い合わせください。

互換性

一般に、インテル® Fortran コンパイラー for Linux* の以前のバージョン (8.0 以降) でコンパイルされたオブジェクト・コードおよびモジュールは、バージョン 16 でもそのまま使用できます。ただし、次の例外があります。

- バージョン 12.0 よりも前のコンパイラーを使用してビルドされた CLASS キーワードを使用して多相変数を宣言しているソースは再コンパイルする必要があります。

- マルチファイルのプロシージャラー間の最適化 (-ipo) オプションを使用してビルドされたオブジェクトは、最新のバージョンで再コンパイルする必要があります。
- バージョン 12.0 よりも前のコンパイラーを使用してビルドされた REAL(16)、REAL*16、COMPLEX(16)、COMPLEX*32 データ型を使用しているオブジェクトは再コンパイルする必要があります。
- バージョン 10.0 よりも前のコンパイラーを使用してインテル® 64 アーキテクチャー用にビルドされたモジュール変数を含むオブジェクトは再コンパイルする必要があります。Fortran 以外のソースからこれらの変数を参照する場合、不正な先頭の下線を削除するように外部名を変更する必要があります。
- バージョン 11.0 よりも前のコンパイラーを使用してコンパイルされた、派生型宣言の外部で ATTRIBUTES ALIGN ディレクティブを指定したモジュールは再コンパイルする必要があります。この問題が発生した場合、問題を通知するメッセージが表示されます。
- 派生型宣言の内部で ATTRIBUTES ALIGN ディレクティブを指定したモジュールは 13.0.1 以前のコンパイラーでは使用できません。
- Fortran 2008 サブモジュール機能を実装するため、バイナリー .mod ファイルの内部フォーマットが大幅に変更されました。このため、バージョン 16.0 の Fortran コンパイラーで作成されたモジュールファイルは、バージョン 15.0 以前の Fortran コンパイラーで使用することはできません。

REAL(16) および COMPLEX(16) データ型のスタック・アライメントの変更

バージョン 12.0 よりも古いコンパイラーでは、REAL(16) または COMPLEX(16) (REAL*16 または COMPLEX*32) 項目が値で渡されたとき、スタックアドレスは 4 バイトでアラインされていました。パフォーマンスを向上させるため、バージョン 12 以降のコンパイラーは、これらの項目を 16 バイトでアラインし、引数が 16 バイト境界でアラインされていると仮定します。この変更は、gcc と互換です。

この変更は、主にライブラリーが生成した REAL(16) 値の計算を行うライブラリー (組込み関数を含む) の呼び出しに影響します。以前のバージョンでコンパイルしたコードをバージョン 12 のライブラリーとリンクする場合、またはアプリケーションをインテルのランタイム・ライブラリーの共有バージョンにリンクする場合、正しくない結果が返される可能性があります。

コンパイラーのバージョン 12.0 以前でコンパイルされている場合、この問題を回避するには、REAL(16) および COMPLEX(16) データ型を使用しているすべての Fortran ソースを再コンパイルしてください。

新機能と変更された機能

一部の言語機能に関する説明はコンパイラーのドキュメントにはまだ含まれていません。必要に応じて、[Fortran 2008 Standard](#) (PDF、英語) および [Proposed draft Fortran 2015 Standard](#) を参照してください。

Fortran 2008 の機能

- サブモジュール
- IMPURE

Proposed draft Fortran 2015 の機能

- 「Technical Specification 29113 Further Interoperability with C」のすべての機能。
 - 型引き継ぎ (TYPE(*))
 - ランク引き継ぎ (DIMENSION(..))
 - 互換性のある仮引数の制約の緩和
 - Fortran で使用される C コード操作「C 記述子」を定義する C インクルード・ファイル ISO_Fortran_binding.H

OpenMP* 機能

OpenMP* 4.0 および OpenMP* 4.1 の次のディレクティブ、節、プロシージャがコンパイラーでサポートされました。これらの機能の一部は、暫定仕様に基づきインテル® Fortran Composer XE 2013 Update 2 でサポートされました。また、以前サポートされていたいくつかの構文 (DECLARE TARGET MIRROR, DECLARE TARGET LINKABLE, MAPTO, MAPFROM, SCRATCH) はサポートされなくなりました。さらに、一部の構文は以前の仕様から変更されています。

インテル® コンパイラー 16.0 における OpenMP* 4.1 ドラフト仕様 TR3 の新しい機能のサポートは、OpenMP* 4.5 仕様 (2015 年 11 月にリリース予定) の規格に合わせて変更される可能性があります。

詳細は、コンパイラー・ドキュメントまたは上記の OpenMP* 仕様へのリンクを参照してください。

OpenMP* 4.1 のディレクティブ:

- TARGET ENTER DATA
- TARGET EXIT DATA

節:

- OMP TARGET および OMP TARGET UPDATE ディレクティブの DEPEND
- OMP TARGET および OMP TARGET UPDATE ディレクティブの NOWAIT
- OMP SIMD ディレクティブの SIMDLLEN
- OMP ORDERED ディレクティブの SIMD

OpenMP* TR3 Proposal で定義されている非同期オフロードおよびデバイス仕様のサポート

インテル® Fortran コンパイラー 16.0 では、OpenMP* TR3 Technical Report (<http://openmp.org> (英語)) で定義されている非同期オフロードおよびデバイス仕様

をサポートする、新しいオフロード節とディレクティブが追加されました。詳細は、『インテル® Fortran コンパイラー・ユーザー・リファレンス・ガイド』を参照してください。

omp declare simd linear 節の新しい修飾子

omp declare simd ディレクティブの linear 節を新しい修飾子で拡張

linear (*linear-list* [: *linear-step*])

linear-list は次のいずれかです。

list

modifier (*list*)

modifier は次のいずれかです。

ref

val

uval

- すべての *list* 項目は各 SIMD レーンで同時に呼び出される関数の仮引数でなければなりません。
- *modifier* が指定されない場合や **val** または **uval modifier** が指定された場合、各レーンの各 *list* 項目の値は、関数に入るときの *list* 項目の値とレーンの論理番号の倍数 *linear-step* に相当します。
- **uval modifier** が指定された場合、各呼び出しは各 SIMD レーンと同じメモリー位置を使用します。このメモリー位置は論理的な最終レーンの最後の値で更新されます。
- **ref modifier** が指定された場合、各レーンの各 *list* 項目のメモリー位置は、レーンの論理番号の倍数 *linear-step* でインデックスされた関数に入るときのメモリー位置の配列に相当します。

新しいディレクティブと追加されたディレクティブ

インテル® Parallel Studio XE 2015 Composer Edition for Fortran Linux* では、次のコンパイラー・ディレクティブが追加、変更されています。詳細は、ドキュメントを参照してください。

- BLOCK_LOOP [clause [[,] clause...]]
- NOBLOCK_LOOP
- unroll_and_jam (n) [private (var1,[var2]...)]

BIND(C) と ATTRIBUTES STDCALL をともに使用可能

コンパイラー 15.0 では、互換性のあるプロシージャー (宣言に BIND(C) 言語バインド属性を含むプロシージャー) で ATTRIBUTES STDCALL ディレクティブを指定することができます。

STDCALL によるその他の影響 (値渡しなど) はありません。必要に応じて、(ATTRIBUTES VALUE ではなく) Fortran 標準の VALUE 属性を利用できます。その他

のプラットフォームでは、STDCALL と BIND(C) をともに指定しても効果はありません。

ATTRIBUTES FASTMEM ディレクティブ

割り当てられたオブジェクトで高帯域幅 (HBW) メモリ割り当てを有効にします。このディレクティブ・オプションは、インテル® MIC アーキテクチャーにのみ適用され、Linux* でのみ利用可能です。

-init=snan を使用した初期化されていない変数の確認をローカルおよびヒープ変数に拡張

-init=snan はこれまで組込み数値型のスタティック変数を初期化していました。この対象が拡張され、組込み数値型のローカル、自動、割り当て済み変数を含むようになりました。

その他の機能

これらの機能に関する詳細は、コンパイラー・ドキュメントを参照してください。

- 新しい環境変数 INTEL_PROF_DYN_PREFIX。異なる実行で生成される PGO の ".dyn" ファイルを簡単に区別できるように、任意のプリフィックスを追加できます。インストールされたアプリケーションを開始する前に、この環境変数に任意の文字列を設定すると、.dyn ファイル名に指定した文字列がプリフィックスとして追加されます。
- SIMD ベクトル内の "レーン ID" を示す新しい `__intel_simd_lane()` 組込み関数。この組込み関数は、ショートベクトル・ハイパーオブジェクトのレデューサー実装の記述をサポートします。また、SIMD 対応関数内でリダクション操作の実行を可能にします。
- コンパイラーの動作が変更され、型仕様のない配列コンストラクターが異なる型/種別型値を持っている場合は常に診断を行うようになりました。次に例を示します。
 - `(/integer:: 0, 1., 2./)` は有効
 - `(/0, 1., 2./)` は無効
 - `(/real::1., 0, 123, 4./)` は有効
 - `(/1., 0, 123, 4./)` は無効

Fortran ライブラリー・バージョンを取得するための新しいランタイムルーチン

- `FOR_IFCORE_VERSION` は、Fortran ランタイム・ライブラリー (ifcore) のバージョンを返します。
- `FOR_IFPORT_VERSION` は、Fortran 移植ライブラリー (ifport) のバージョンを返します。

新規および変更されたコンパイラー・オプション

詳細は、コンパイラーのドキュメントを参照してください。

- `-xCOMMON-AVX512`
- `-axCOMMON-AVX512`
- `-fpp-name=<実行ファイル名およびオプションのパス>`
- `-gen-depshow=[no]intr_mod`
- `-gsplit-dwarf`
- `-qopt-prefetch-issue-excl-hint`

廃止予定のコンパイラー・オプションのリストは、ドキュメントのコンパイラー・オプションのセクションを参照してください。

新しい `-xCOMMON-AVX512`、`-axCOMMON-AVX512` コンパイラー・オプション

`-x` および `-ax` コンパイラー・オプションに `COMMON-AVX512` が追加されました。これにより、インテル® メニー・インテグレートッド・コア・アーキテクチャーとインテル® Core™ マイクロアーキテクチャーの両方でサポートされるインテル® アドバンスト・ベクトル・エクステンション 512 (インテル® AVX-512) のサブセットを生成できます。これには、インテル® AVX-512 の基本命令および競合検出命令が含まれます。

新しい `-fpp-name=<実行ファイル名およびオプションのパス>` コンパイラー・オプション

この新しいオプションにより、ユーザーは Fortran とともに使用する異なるプリプロセッサを指定できます。

新しい `-gsplit-dwarf` コンパイラー・オプション

インテル® Fortran コンパイラー 16.0 では、主要デバッグ情報を含む追加のオブジェクト・ファイル (拡張子 `.dwo`) を生成する `-gsplit-dwarf` コンパイラー・オプションが追加されました。現在は、32 ビット・ターゲットと 64 ビット・ターゲットのみサポートしています。このオプションを指定してコンパイルするには `binutils 2.23` 以降が必要です。また、デバッグするには `gdb 7.5` 以降が必要です。

新しい `-gen-depshow=[no]intr_mod` コンパイラー・オプション

この新しいオプションにより、ユーザーは依存性の解析から組込みモジュールを除外できます。このオプションは、組込みモジュールの依存関係を表示するかどうかを制御します。`-gen-depshow=nointr_mod` がデフォルトです。

新しい `-qopt-prefetch-issue-excl-hint` コンパイラー・オプション

`-qopt-prefetched` オプションと同時に使用すると、インテル® マイクロアーキテクチャ Broadwell (開発コード名) 以降のプロセッサ向けの `prefetchW` 命令を生成します。

終了予定のサポート

Red Hat* Enterprise Linux 5* のサポートを終了予定

Red Hat* Enterprise Linux* 5 のサポートは、将来のリリースで終了する予定です。

32 ビット・ホストへのインストールのサポートを終了予定

32 ビット・ホストへのインストールは、将来のリリースで終了する予定です。ただし、32 ビット・ターゲット用コード生成のサポートは引き続き行われます。

終了したサポート

Debian* 6 のサポートを終了

これらのオペレーティング・システム・バージョンのサポートを終了しました。新しいバージョンのオペレーティング・システムに移行してください。

Fedora* 20 のサポートを終了

これらのオペレーティング・システム・バージョンのサポートを終了しました。新しいバージョンのオペレーティング・システムに移行してください。

スタティック解析のサポートを終了

スタティック解析のサポートを終了しました。ご意見やお問い合わせは、[こちら](#) (英語) までお寄せください。

既知の問題

パラメーター化された派生型で文字長引数の特定の使用法がまだ完全に実装されていない

パラメーター化された派生型 (PDT) では、文字長引数の次の使用法はまだ完全に実装されていません。

- 文字長引数を含む PDT 引数定数
- `%RE` と `%IM` は未実装

16.0 初期リリースに含まれていたユーザー定義 I/O に関連するエラー状態を検出する修正により大量のリグレッションが引き起こされる

インテル® Fortran コンパイラー 16.0 初期リリースに含まれていたユーザー定義 I/O に関連するエラー状態を検出する修正、DPD200243620 "Missing error for inaccessible components of derived type in I/O list" により大量のリグレッションが引き起こされることが判明し、16.0 Update 1 リリースで元に戻されました。

Fortran 2008 および Fortran 2015 機能の概要

インテル® Fortran コンパイラーは、Fortran 2008 規格の多くの機能と Proposed draft Fortran 2015 規格の機能をサポートします。その他の機能は将来のリリースでサポートされる予定です。現在のコンパイラーでは、以下の Fortran 2008 機能がサポートされています。

- 配列の最大次元数が 31 次元に (Fortran 2008 では 15 次元)
- Co-Array
- CODIMENSION 属性
- SYNC ALL 文
- SYNC IMAGES 文
- SYNC MEMORY 文
- CRITICAL および END CRITICAL 文
- LOCK および UNLOCK 文
- ERROR STOP 文
- ALLOCATE および DEALLOCATE で Co-Array を指定
- 組込みプロシージャ: ATOMIC_DEFINE、ATOMIC_REF、IMAGE_INDEX、LCOBOUND、NUM_IMAGES、THIS_IMAGE、UCOBOUND
- CONTIGUOUS 属性
- ALLOCATE の MOLD キーワード
- DO CONCURRENT
- OPEN の NEWUNIT キーワード
- GO および GO.d フォーマット編集記述子
- 無制限のフォーマット項目繰り返しカウント指定子
- CONTAINS セクションは空にすることも可能
- 組込みプロシージャ: BESSEL_J0、BESSEL_J1、BESSEL_JN、BESSEL_YN、BGE、BGT、BLE、BLT、DSHIFTL、DSHIFTR、ERF、ERFC、ERFC_SCALED、GAMMA、HYPOT、IALL、IANY、IPARITY、IS_CONTIGUOUS、LEADZ、LOG_GAMMA、MASKL、MASKR、MERGE_BITS、NORM2、PARITY、POPCNT、POPPAR、SHIFTA、SHIFTL、SHIFTR、STORAGE_SIZE、TRAILZ
- 組込みモジュール ISO_FORTRAN_ENV の追加: ATOMIC_INT_KIND、ATOMIC_LOGICAL_KIND、CHARACTER_KINDS、INTEGER_KINDS、INT8、INT16、INT32、INT64、LOCK_TYPE、LOGICAL_KINDS、REAL_KINDS、REAL32、REAL64、REAL128、STAT_LOCKED、STAT_LOCKED_OTHER_IMAGE、STAT_UNLOCKED
- ALLOCATABLE または POINTER 属性を持たない OPTIONAL 仮引数は、対応する実引数に ALLOCATABLE 属性があるのに割り当てられない場合、

POINTER 属性があるのに関連付けが解除されている場合、または NULL 組込み関数への参照の場合、無視されます。

- 仮引数がプロシージャ・ポインターの場合、そのポインターの有効な参照先か、または組込み関数 NULL への参照である実引数に関連付けられます。実引数がポインターではない場合、仮引数に INTENT (IN) 属性が含まれていなければなりません。
- BLOCK 構造
- EXECUTE_COMMAND_LINE 組込みサブルーチン
- サブモジュール
- IMPURE

現在のバージョンでは、次の Proposed draft Fortran 2015 の機能がサポートされています。

- 「Technical Specification 29113 Further Interoperability with C」のすべての機能。
 - 型引き継ぎ (TYPE(*))
 - ランク引き継ぎ (DIMENSION(..))
 - 互換性のある仮引数の制約の緩和
 - Fortran で使用される C コード操作「C 記述子」を定義する C インクルード・ファイル ISO_Fortran_binding.H

著作権と商標について

最適化に関する注意事項

インテル® コンパイラーでは、インテル® マイクロプロセッサに限定されない最適化に関して、他社製マイクロプロセッサ用に同等の最適化を行えないことがあります。これには、インテル® ストリーミング SIMD 拡張命令 2、インテル® ストリーミング SIMD 拡張命令 3、インテル® ストリーミング SIMD 拡張命令 3 補足命令などの最適化が該当します。インテルは、他社製マイクロプロセッサに関して、いかなる最適化の利用、機能、または効果も保証いたしません。本製品のマイクロプロセッサ依存の最適化は、インテル® マイクロプロセッサでの使用を前提としています。インテル® マイクロアーキテクチャに限定されない最適化のなかにも、インテル® マイクロプロセッサ用のものがあります。この注意事項で言及した命令セットの詳細については、該当する製品のユーザー・リファレンス・ガイドを参照してください。

注意事項の改訂 #20110804

本資料に掲載されている情報は、インテル製品の概要説明を目的としたものです。本資料は、明示されているか否かにかかわらず、また禁反言によるとよらずにかかわらず、いかなる知的財産権のライセンスも許諾するものではありません。製品に付属の売買契約書『Intel's Terms and Conditions of Sale』に規定されている場合を除き、インテルはいかなる責任を負うものではなく、またインテル製品の販売や使用に関する明示または黙示の保証 (特定目的への適合性、商品適格性、あらゆる特許権、著作権、その他知的財産権の非侵害性への保証を含む) に関してもいかなる責任

も負いません。インテルによる書面での合意がない限り、インテル製品は、その欠陥や故障によって人身事故が発生するようなアプリケーションでの使用を想定した設計は行われていません。

インテル製品は、予告なく仕様や説明が変更される場合があります。機能または命令の一覧で「留保」または「未定義」と記されているものがありますが、その「機能が存在しない」あるいは「性質が留保付である」という状態を設計の前提にしないでください。これらの項目は、インテルが将来のために留保しているものです。インテルが将来これらの項目を定義したことにより、衝突が生じたり互換性が失われたりしても、インテルは一切責任を負いません。この情報は予告なく変更されることがあります。この情報だけに基づいて設計を最終的なものとししないでください。

本資料で説明されている製品には、エラッタと呼ばれる設計上の不具合が含まれている可能性があります。公表されている仕様とは異なる動作をする場合があります。現在確認済みのエラッタについては、インテルまでお問い合わせください。

最新の仕様をご希望の場合や製品をご注文の場合は、お近くのインテルの営業所または販売代理店にお問い合わせください。

本資料で紹介されている資料番号付きのドキュメントや、インテルのその他の資料を入手するには、1-800-548-4725 (アメリカ合衆国) までご連絡いただくか、インテルの Web サイト (<http://www.intel.com/design/literature.htm> (英語)) を参照してください。

インテル・プロセッサ・ナンバーはパフォーマンスの指標ではありません。プロセッサ・ナンバーは同一プロセッサ・ファミリー内の製品の機能を区別します。異なるプロセッサ・ファミリー間の機能の区別には用いません。詳細については、http://www.intel.co.jp/jp/products/processor_number/ を参照してください。

インテル® Fortran コンパイラーは、インテルのソフトウェア使用許諾契約書 (EULA) の下で提供されます。

詳細は、製品に含まれるライセンスを確認してください。

Intel、インテル、Intel ロゴ、Intel Core、Intel Xeon Phi、Pentium は、アメリカ合衆国および / またはその他の国における Intel Corporation の商標です。

* その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。

© 2016 Intel Corporation. 無断での引用、転載を禁じます。

コンパイラーの最適化に関する詳細は、[最適化に関する注意事項](#)を参照してください。