

この資料は英語版を翻訳したもので、内容に相違が生じる場合には原文を優先します。こちらの日本語版は参考用としてご利用ください。設計の際には、最新の英語版で内容をご確認ください。

NII52001-8.0.0

はじめに

この章では、ソフトウェア開発者を対象に Nios® II プロセッサの概要を説明します。Nios II ソフトウェア開発環境、Nios II エンベデッド・デザイン・スイート (EDS) ツール、およびソフトウェア開発のプロセスを紹介します。この章は、以下の項で構成されています。

- 1-1 ページの「最新情報」
- 1-2 ページの「使用開始にあたって」
- 1-3 ページの「Nios II ソフトウェア開発環境」
- 1-3 ページの「Nios II プログラム」
- 1-6 ページの「Nios II プログラム作成のためのデザイン・フロー」
- 1-10 ページの「追加 EDS サポート」
- 1-10 ページの「サードパーティ・サポート」
- 1-10 ページの「第 1 世代の Nios プロセッサからの移行」
- 1-11 ページの「Nios II に関するその他の情報」

最新情報

以下の機能は Nios II EDS v8.0 の新機能です。

- **.sopcinfo** ファイル — v8.0 から導入。SOPC Builder のシステム・ファイルは **.sopc** ファイルではなく、**.sopcinfo** ファイルとしてエンコードされます。**.sopcinfo** ファイルは、SOPC Builder でシステム生成を実行するときに生成されます。**.sopcinfo** ファイルは、スタティック・ファイルであり、生成時にインストールされる IP コンポーネントに基づいてシステムの生成方法を記述しています。以前は **.sopc** ファイルを入力として受け入れていたユーティリティ (**nios2-bsp-create-settings** など) は、v8.0 からは **.sopcinfo** ファイルを受け入れます。

--sopc コマンドライン引数は、**.sopc** ファイルではなく **.sopcinfo** ファイルをサポートします。




SOPC Builder リリース 7.2 またはそれ以前のリリースでハードウェアが生成されている場合、BSP を作成する前に SOPC Builder リリース 8.0 で再生成してください。

BSP エディタ — コマンドライン Nios II ソフトウェア・ビルド・ツール以外に、BSP プロジェクトを作成する別の方法は、スタンドアロンのグラフィカル・ユーザー・インタフェース (GUI) である Nios II BSP エディ


タを使用することです。BSP エディタは、ソフトウェア・ビルド・ツールをドライブするグラフィカル・フロント・エンドを提供します。Nios II BSP エディタを起動するには、次のコマンドを使用します。

```
nios2-bsp-editor
```

 BSP エディタの使用法について詳しくは、ツールを起動して画面上の説明文を参照してください。

- ウェブ上で配布されるソフトウェア例—Nios II IDEのv7.2まで配布されていた特定のソフトウェア例は、v8.0からはウェブ上で配布されます。ウェブで入手可能なソフトウェア例は、以下のとおりです。

例の名称	場所
Zip ファイル・システム	Nios Community Wiki
MicroC/OS-II メッセージ・ボックス	Nios Community Wiki
Hello LED	Nios Community Wiki
ホスト・ファイル・システム	Nios Community Wiki
Dhrystone	www.altera.co.jp
MicroC/OS-II チュートリアル	www.altera.co.jp
密結合メモリ	www.altera.co.jp
カスタム命令チュートリアル	www.altera.co.jp

 Nios II プロセッサ・コアは、オプションのメモリ・マネジメント・ユニット (MMU) を備えています。Nios II HAL (Hardware Abstraction Layer) は、MMU をサポートしていません。

使用開始 にあたって

Nios II プロセッサ用ソフトウェアの作成は、他社のマイクロコントローラ・ファミリのソフトウェア開発プロセスと似ています。アルテラから開発キットを購入することで、簡単かつ効率的に設計を開始することができます。この開発キットには、資料、既成の開発ボード、および Nios II プログラムを作成するのに必要なすべての開発ツールが含まれています。

Nios II ソフトウェア開発ハンドブックでは、読者にエンベデッド・プロセッサのコンセプトに関する基本的な知識があることを想定して作成されていますが、アルテラの特定のテクノロジーやアルテラの開発ツールに関する知識は必要ありません。アルテラ・ハードウェア開発ツールに関する知識があると、Nios II ソフトウェア開発環境の背景にある論理をよ

り深く理解することができます。ただし、ソフトウェア開発者は、アルテラ・テクノロジの知識がなくても、アプリケーションの開発やデバッグを行うことができます。

ソフトウェア設計者が新しい環境でプログラムの作成方法を取得するには、おそらく既存のコードを書き換えるのが最も一般的で安心できる方法と考えられます。Nios II EDS は、ソフトウェア設計例を提供しており、設計者はこれらを自身のプログラムで検証、修正、および使用できます。提供されている例は、単純な Hello world プログラムから実用的なリアル・タイム・オペレーティング・システム (RTOS) の例、そしてウェブ・サーバーを稼働させる完全な TCP/IP (Transmission Control Protocol/Internet Protocol) スタックなど、広範囲に及びます。それぞれの例は、ドキュメント化されており、すぐにコンパイルできます。

Nios II ソフトウェア 開発環境

Nios II EDS は、すべての Nios II プロセッサ・システムに役立つ一貫性のあるソフトウェア開発環境を提供します。PC、アルテラ FPGA、および JTAG (Joint Test Action Group) ダウンロード・ケーブル (アルテラの USB-Blaster™ ダウンロード・ケーブルなど) を使用して、任意の Nios II プロセッサ・システムのためにプログラムを記述したり、それと通信することができます。Nios II プロセッサの JTAG デバッグ・モジュールは、JTAG ダウンロード・ケーブルを使用してプロセッサと通信するための一貫性のある 1 つの方法を提供します。デバイスが Nios II プロセッサ・システムのみを実装しているかどうか、または Nios II プロセッサが複雑なマルチプロセッサ・システムに深く埋め込まれているかどうかに関係なく、プロセッサへのアクセスは同じです。したがって、エンベデッド・プロセッサのためにインタフェース・メカニズムを手動で作成する時間は必要ありません。

Nios II EDS は異なる 2 つのデザイン・フローを提供し、Nios II プログラムを作成するための独自のオープン・ソース・ツール (GNU C/C++ ツール一式など) を多数備えています。Nios II EDS は、Nios II-をベースとしたシステムのためのボード・サポート・パッケージ (BSP) 作成を自動化しており、BSP を手動で作成するための時間は不要です。アルテラの BSP には、アルテラの HAL (Hardware Abstraction Layer)、オプションの RTOS、およびデバイス・ドライバが含まれています。BSP は C/C++ ランタイム環境を提供し、エンベデッド・システムのハードウェアからユーザーを隔離します。

Nios II プログラム

いずれかの Nios II EDS デザイン・フローで開発する各 Nios II プログラムは、アプリケーション・プロジェクト、オプションのライブラリ・プロジェクト、および BSP プロジェクトで構成されます。Nios II プログラ

ムを、Nios II プロセッサ上で動作する Executable And Linked Format File (.elf) に作成します。2つのデザイン・フローで専門用語が異なることがあります。開発する Nios II プログラムは概念的には同じです。

以下の項では、Nios II プログラムを構成するプロジェクトのタイプについて説明します。

アプリケーション・プロジェクト


Nios II C/C++ アプリケーション・プロジェクトは、1つの実行 (.elf) ファイルに結合されたソース・コードの集合から成ります。アプリケーションの標準的な特色は、ソース・ファイルの1つに関数 main() が含まれていることです。アプリケーションには、ライブラリおよび BSP で関数を呼び出すコードが含まれています。

ライブラリ・プロジェクト

ライブラリ・プロジェクトは、1つのライブラリ・アーカイブ (.a) ファイル内に含まれるソース・コードの集合です。ライブラリには多くの場合、再使用可能で、複数のアプリケーションが共有できる汎用関数が含まれています。一般的な演算関数の集合が1つの例です。ライブラリには関数 main() は含まれていません。

BSP プロジェクト

Nios II BSP プロジェクトは、システム固有のサポート・コードを含む特殊なライブラリです。BSP は SOPC Builder システムで1個のプロセッサに対してカスタマイズされた、ソフトウェア・ランタイム環境を提供します。Nios II EDS は、BSP の動作をコントロールする設定を変更するためのツールを提供します。

 Nios II 統合開発環境 (IDE) および Nios II IDE デザイン・フローの資料では、BSP を表すときに、“システム・ライブラリ” という用語を使用しています。

BSP は以下の要素で構成されています。

- HAL (Hardware Abstraction Layer)
- Newlib C 標準ライブラリ
- デバイス・ドライバ
- オプションのソフトウェア・パッケージ
- オプションのリアル・タイム・オペレーティング・システム (RTOS)

HAL (Hardware Abstraction Layer)

HAL は非スレッドの UNIX ライクな C/C++ ランタイム環境を提供します。HAL によって一般的な I/O デバイスが提供されるため、`printf()` などの `newlib C` 標準ライブラリ・ルーチンを使用して、ハードウェアにアクセスするプログラムを作成できます。HAL により、ペリフェラルを制御したり、ペリフェラルと通信するために直接ハードウェア・レジスタにアクセスする必要性が最小限（あるいはまったく不要）になります。



HAL について詳しくは、「[Nios II ソフトウェア開発ハンドブック](#)」の「[The Hardware Abstraction Layer](#)」セクションおよび「[HAL API リファレンス](#)」の章を参照してください。

Newlib C 標準ライブラリ

Newlib は、エンベデッド・システムでの使用を意図した C 標準ライブラリのオープン・ソース実装です。`printf()`、`malloc()`、および `open()` などの一般的なルーチンの集合です。

デバイス・ドライバ

各デバイス・ドライバは、ハードウェア・コンポーネントを管理します。デフォルトにより、HAL はデバイス・ドライバを必要とする SOPC Builder システムの各コンポーネント用デバイス・ドライバをインスタンス化します。Nios II ソフトウェア開発環境では、デバイス・ドライバは以下の特性を備えています。

- デバイス・ドライバは特定の SOPC Builder コンポーネントに関連付けられています。
- デバイス・ドライバには、コンパイルに影響を与える設定がある場合があります。これらの設定は BSP 設定の一部になります。

オプションのソフトウェア・パッケージ

ソフトウェア・パッケージは、付加機能を提供するためにオプションで BSP プロジェクトに追加できるソース・コードです。NicheStack® TCP/IP - Nios II Edition は、ソフトウェア・パッケージの一例です。



Nios II IDE および Nios II IDE デザイン・フローの資料では、ソフトウェア・パッケージを表すときに“ソフトウェア・コンポーネント”という用語を使用しています。

Nios II ソフトウェア開発環境では、ソフトウェア・パッケージは一般に以下の特性を備えています。

- ソフトウェア・パッケージは特定のハードウェアに関連していません。
- ソフトウェア・パッケージには、コンパイルに影響を与える設定がある場合があります。これらの設定は BSP 設定の一部になります。



Nios II ソフトウェア開発環境では、ソフトウェア・パッケージはライブラリ・プロジェクトとは区別されます。ソフトウェア・パッケージは BSP プロジェクトの一部で、独立したライブラリ・プロジェクトではありません。

オプションのリアル・タイム・オペレーティング・システム (RTOS)

Nios II EDS には、オプションで BSP に組み込むことができるサードパーティの MicroC/OS-II RTOS の実装が含まれています。MicroC/OS-II は HAL 上に構築され、簡単で十分に実証された RTOS スケジューラを実装しています。BSP 設定の一部となる設定を変更できます。サードパーティ・ベンダからオペレーティング・システムを入手できます。

Nios II プログラム 作成のため のデザイン・ フロー

Nios II EDS は、Nios II プログラム作成用の 2 種類のデザイン・フローを提供しています。Nios II 統合開発環境 (IDE) 内ですべて作業を行うか、コマンドラインやスクリプト化された環境で Nios II ソフトウェア・ビルド・ツールを使用した後、デザインを IDE にインポートしてデバッグすることができます。

この 2 つのデザイン・フローを置き換えることはできません。アプリケーション、ライブラリ、およびドライバのソース・コードは、どちらのフローでも動作しますが、2 つのフローの `makefile` は異なり、互換性はありません。1 つのデザイン・フローを使用することに決めた場合、そのプロジェクトで別のデザイン・フローを使用するには、最初からやり直さなければなりません。

Nios II IDE デザイン・フロー

Nios II IDE デザイン・フローでは、Nios II IDE グラフィカル・ユーザー・インタフェース (GUI) を使用して、Nios II プログラムを作成し、変更、構築、実行し、デバッグします。IDE はユーザー用にプロジェクトの `makefile` を作成し、管理します。このデザイン・フローは、構築されたプロセスおよびプロジェクト設定を限定的に制御するだけの場合に最適で、カスタマイズされたスクリプトは必要ありません。

Nios II IDE は、一般的な Eclipse IDE フレームワークと Eclipse C/C++ 開発ツール・キット (CDT) プラグインをベースにしています。Nios II IDE は、バックグラウンドで他のツールを動作させ、低レベル・ツールの詳細からユーザーを遮蔽し、統一された開発環境を提供します。

Nios II IDE は、プロジェクトの作成およびコンフィギュレーションに役立つウィザードを備えており、使いやすく、特に Nios II の初心者にとって便利です。Nios II IDE は、Windows および Linux オペレーティング・システムの両方で使用できます。



Nios II IDE について詳しくは、「[Nios II ソフトウェア開発ハンドブック](#)」の「[Nios II Integrated Development Environment](#)」の章を参照してください。

Nios II ソフトウェア・ビルド・ツールのデザイン・フロー

Nios II ソフトウェア・ビルド・ツールのデザイン・フローでは、コマンドラインに入力するか、またはスクリプトに記述されたコマンドを使用して、Nios II プログラムを作成し、変更、構築し、実行します。このデザイン・フローは、構築されたプロセスおよびプロジェクト設定に対して緻密なコントロールが必要な場合、またはカスタマイズされたスクリプトが必要な場合に最適です。

Nios II ソフトウェア・ビルド・ツールは、Nios II IDE の **New Project** ウィザードおよび **System Library** プロパティ・ページに、同様の機能を提供するユーティリティおよびスクリプトです。Nios II ソフトウェア・ビルド・ツールのデザイン・フローにより、Nios II ソフトウェア開発を開発フローの他の部分と統合できます。スクリプトを使用すると、ソフトウェア開発フローは完全に繰り返し可能かつアーカイブ可能になります。

デバッグ時には、Nios II ソフトウェア・ビルド・ツールのプロジェクトを Nios II IDE プロジェクトとして IDE にインポートして、デバッグを行います。IDE でインポートされたアプリケーション・プロジェクトをさらに編集し、再構築、実行し、デバッグすることができます。また、ライブラリおよび BSP プロジェクトをインポートして、デバッグでソース・コードを表示することも可能ですが、インポートされたライブラリおよび BSP プロジェクトは IDE で直接構築することはできません。

Nios II IDE と同様に、Nios II ソフトウェア・ビルド・ツールは Windows および Linux オペレーティング・システムの両方で使用できます。Nios II ソフトウェア・ビルド・ツールは、アルテラの将来の Nios II 開発の基礎となるものです。



Nios II ソフトウェア・ビルド・ツールについて詳しくは、「[Nios II ソフトウェア開発ハンドブック](#)」の「[Introduction to the Nios II Software Build Tools](#)」の章を参照してください。

デザイン・フロー・ツール

この項では、各デザイン・フローのための Nios II プログラムの作成に使用するツールを紹介します。ここで示す表は必要なレベルのコントロールを決定するのに役立つように構成されています。これらの表をクイック・リファレンス・ガイドとして使用して、デザイン・フロー間の違いを把握することができます。

表 1-1 に、各 Nios II デザイン・フローでのタスクに対して、高度に自動化されたレベルのコントロールを提供するツールを示します。このレベルのコントロールでは、Nios II プログラム例全体（アプリケーション・プロジェクトと BSP プロジェクトから成る）、またはデフォルト設定を使用したアルテラ提供ソフトウェア例からの BSP プロジェクト例だけを作成します。この高度に自動化されたツールは、開発の出発点として、あるいはカスタマイズが不要なときに使用します。

タスク	Nios II IDE デザイン・フロー	Nios II ソフトウェア・ビルド・ツールのデザイン・フロー
Nios II プログラム例の作成	File > New > Nios II C/C++ Application	create-this-app スクリプト
BSP 例の作成	File > New > Nios II System Library	create-this-bsp スクリプト
デバッグ	Run > Debug As > Nios II Hardware	<ul style="list-style-type: none"> ● File > Import > Altera Nios II >ワークスペースに読み込む既存の Nios II ソフトウェア・ビルド・ツールのプロジェクトまたはフォルダ ● Run > Debug As > Nios II Hardware

表 1-2 に、各 Nios II デザイン・フローでのタスクに対する中間レベルのコントロールを提供するツールを示します。このレベルのコントロールでは、カスタム・コードまたはアルテラが提供するソフトウェア例か

ら Nios II プログラムを作成します。デフォルト設定で得られる以上のコントロールが必要な場合は、開発の出発点として中間ツールを使用します。

タスク	Nios II IDE デザイン・フロー	Nios II ソフトウェア・ビルド・ツールのデザイン・フロー
アプリケーションの構築	File > New > Nios II C/C++ Application	nios2-app-generate-makefile ユーティリティ
ライブラリの構築	File > New > Nios II C/C++ Library	nios2-lib-generate-makefile ユーティリティ
BSP の構築	<ul style="list-style-type: none"> File > New > Nios II System Library Project > Properties > System Library 	nios2-bsp スクリプト
デバッグ	Run > Debug As > Nios II Hardware	<ul style="list-style-type: none"> File > Import > Altera Nios II > ワークスペースに読み込む既存の Nios II ソフトウェア・ビルド・ツールのプロジェクトまたはフォルダ Run > Debug As > Nios II Hardware

表 1-3 に、各 Nios II デザイン・フローで BSP に関連するタスクに対して高度なレベルのコントロールを提供するツールを示します。このレベルのコントロールでは、高機能のスクリプタブル・コントロールを使用して Nios II BSP を作成します。BSP 構築プロセスと BSP プロジェクト設定に対する全面的なコントロールが必要なときは、高度なツールを使用します。

タスク	Nios II IDE デザイン・フロー	Nios II ソフトウェア・ビルド・ツールのデザイン・フロー
BSP の構築	システム・ライブラリ・プロジェクトのスクリプタブル・コントロールはサポートされていません。	<ul style="list-style-type: none"> nios2-bsp-create-settings ユーティリティ nios2-bsp-generate-files ユーティリティ Tcl スクリプタブル
BSP のアップデート	システム・ライブラリ・プロジェクトのスクリプタブル・コントロールはサポートされていません。	<ul style="list-style-type: none"> nios2-bsp-update-settings ユーティリティ nios2-bsp-generate-files ユーティリティ Tcl スクリプタブル
BSP の照会	サポートされていません。	<ul style="list-style-type: none"> nios2-bsp-query-settings ユーティリティ Tcl スクリプタブル
newlib のカスタマイズ	サポートされていません。	nios2-bsp スクリプト： CUSTOM_NEWLIB_FLAGS 設定を使用

追加 EDS サポート

Nios II IDE および Nios II ソフトウェア・ビルド・ツールに加えて、Nios II EDS には以下のアイテムが含まれています。

- GNU ツール一式
- 命令セット・シミュレータ
- デザイン例

GNU ツール一式

Nios II のコンパイラ・ツール一式は、標準的な GNU GCC コンパイラ、アセンブラ、リンカ、および make 機能をベースにしています。



GNU について詳しくは、www.gnu.org を参照してください。

命令セット・シミュレータ

Nios II 命令セット・シミュレータ (ISS) を使うことで、ターゲットのハードウェア・プラットフォームが完成する前に、プログラムの開発を開始することができます。実際のハードウェア・ターゲット上で実行するのと同様に、Nios II IDE から ISS を使用して簡単にプログラムを実行できます。

デザイン例

Nios_II EDS には、Nios II プロセッサおよび開発環境のすべての重要な機能を実証できるように、ソフトウェア事例が提供されています。

サードパーティ・ サポート

サードパーティ・ベンダー数社は Nios II プロセッサをサポートしており、デザイン・サービス、オペレーティング・システム、スタック、その他のソフトウェアライブラリや開発ツールなどを提供しています。



Nios II プロセッサのサードパーティ・サポートに関する最新情報は、Nios II プロセッサのホーム・ページ (www.altera.co.jp/nios2) を参照してください。

第 1 世代の Nios プロセッサ からの移行

第 1 世代の Nios プロセッサ・ユーザーの方には、今後のデザインでは Nios II プロセッサへ移行することをお勧めします。分かりやすい移行プロセスについては、「AN 350: Upgrading Nios Processor Systems to the Nios II Processor」で説明しています。

Nios II に関する その他の情報

このハンドブックは、Nios II プロセッサの関連資料の一部です。Nios II プロセッサに関するその他の情報については、次の参考資料をご覧ください。

- Nios II プロセッサ・リファレンス・ハンドブックでは、命令セット・アーキテクチャを含む、プロセッサのハードウェア・アーキテクチャおよび機能が定義されています。
- Quartus® II ハンドブック Volume 5 の「エンベデッド・ペリフェラル」は、Nios II プロセッサとともに配布されるペリフェラルの参考資料を提供します。このハンドブックでは、ハードウェア構造と各ペリフェラル用 Nios II ソフトウェア・ドライバについて説明します。
- Nios II IDE は、GUI の機能を使用するためのチュートリアルと包括的な情報を提供します。Nios II IDE 内ではヘルプ・システムを使用できます。
- アルテラ・ナレッジ・データベースは、使いやすい検索エンジンによって、FAQ に対する回答を提供するインターネット・リソースです。www.altera.co.jp/support/kdb/kdb-index.jsp を参照してください。
- アルテラのアプリケーション・ノートとチュートリアルでは、特定のアプリケーションまたは目的に対する Nios II プロセッサの使用方法について段階的に説明しています。これらの資料は、オンライン資料: Nios II プロセッサ・ページ (www.altera.co.jp/literature/lit-nio2.jsp) で提供されています。

参考資料

この章では以下のドキュメントを参照しています。

- 「The Hardware Abstraction Layer」セクション (Nios II ソフトウェア開発ハンドブック)
- 「HAL API リファレンス」の章 (Nios II ソフトウェア開発ハンドブック)
- 「Nios II Integrated Development Environment」の章 (Nios II ソフトウェア開発ハンドブック)
- 「Introduction to the Nios II Software Build Tools」の章 (Nios II ソフトウェア開発ハンドブック)
- AN 350: Upgrading Nios Processor Systems to the Nios II Processor
- Nios II プロセッサ・リファレンス・ハンドブック
- Quartus II ハンドブック Volume 5: エンベデッド・ペリフェラル

改訂履歴

表 1-4 に、本資料の改訂履歴を示します。

表 1-4. 改訂履歴		
日付およびドキュメント・バージョン	変更内容	概要
2008 年 5 月 v8.0.0	1-1 ページの「最新情報」を追加。	<ul style="list-style-type: none"> ● .sopcinfo ファイル ● EDS から削除されたデザイン例 ● Nios II コアに MMU を追加
2007 年 10 月 v7.2.0	前バージョンからの内容の変更はありません。	
2007 年 5 月 v7.1.0	<ul style="list-style-type: none"> ● Nios II EDS デザイン・フロー、Nios II プログラム、Nios II ソフトウェア・ビルド・ツール、および Nios II BSP を紹介する章全体を改訂。 ● 「はじめに」の項に目次を追加。 ● 「参考資料」の項を追加。 	Nios II ソフトウェア・ビルド・ツール
2007 年 3 月 v7.0.0	前バージョンからの内容の変更はありません。	
2006 年 11 月 v6.1.0	前バージョンからの内容の変更はありません。	
2006 年 5 月 v6.0.0	前バージョンからの内容の変更はありません。	
2005 年 10 月 v5.1.0	前バージョンからの内容の変更はありません。	
2005 年 5 月 v5.0.0	前バージョンからの内容の変更はありません。	
2004 年 5 月 v1.0	初版	