11. Altera-Provided Development Tools

NII520011-5.0.0

Introduction

This chapter introduces all of the development tools that Altera provides for the Nios® II processor. These tools fall into the following categories:

- The Nios II integrated development environment (IDE) and associated tools
- Altera® command-line tools
- GNU compiler tool-chain
- Libraries and embedded software components

This chapter does not describe detailed usage of any of the tools, but it refers you to the most appropriate documentation.

The Nios II IDE Tools

Table 11–1 describes the tools provided by the Nios II IDE user interface.

Table 11–1. The Nios II IDE & Associated Tools (Part 1 of 2)

Tools	Description
The Nios II IDE	The Nios II IDE is the software development user interface for the Nios II processor. All software development tasks can be accomplished within the IDE, including editing, building, and debugging programs. For more information, refer to the Nios II IDE online help system.
Flash programmer	The Nios II IDE includes a flash programmer utility that allows you to program flash memory chips on a target board. The flash programmer supports programming flash on any board, including Altera development boards and your own custom boards. The flash programmer facilitates programming flash for the following purposes: ● Executable code and data ● Bootstrap code to copy code from flash to RAM, and then run from RAM. ● HAL file subsystems ● FPGA hardware configuration data For more information, refer to the *Nios II Flash Programmer User Guide*.

Table 11–1. The Nios II IDE & Associated Tools (Part 2 of 2)

Tools	Description
Instruction set simulator	Altera provides an instruction set simulator (ISS) for the Nios II processor. The ISS is available within the Nios II IDE, and the process for running and debugging programs on the ISS is the same as for running and debugging on target hardware. For more information, refer to the Nios II IDE online help system.
Quartus II Programmer	The Quartus II programmer is part of the Quartus II software, however the Nios II IDE can launch the Quartus II programmer directly. The Quartus II programmer allows you to download new FPGA configuration files to the board. For more information, refer to the Nios II IDE online help system, or press the F1 key while the Quartus II programmer is open.

Altera Command-Line Tools

This section describes the command-line tools provided by Altera. You can run these tools from a *Nios II Software Development Kit (SDK) Shell* command prompt, for example, to write a script to automate compilation tasks. The Altera command-line tools are in the <*Nios II kit path*>**/bin/** directory.

Each tool provides its own documentation in the form of help pages accessible from the command line. To view the help, open a *Nios II SDK Shell*, and type the following command:

```
<name of tool> --help
```

Table 11–2 shows command-line utilities that create and build Nios II IDE projects without launching the Nios II IDE graphical user interface (GUI). These utilities allow you to automate Nios II IDE operations using command-line scripts. For example, with the help of these utilities, a script can check out a Nios II IDE project from source control, import the project into the Nios II IDE workspace, and build the project.

Each of these utilities launches the Nios II IDE in the background, without displaying the GUI. You cannot use these utilities while the IDE is running, because only one instance of the Nios II IDE can be active at a time.

Table 11–2. Nios II IDE Command Line Tools (Part 1 of 2)

Tool	Description
`nios2-create-system-library`	Creates a new system library project.
`nios2-create-application-project`	Creates a new C/C++ application project.

Table 11–2. Nios II IDE Command Line Tools (Part 2 of 2)

Tool	Description
nios2-build-project	Builds a project using the Nios II IDE managed-make facilities. Creates or updates the makefiles to build the project, and optionally runs make. nios2-build-project operates only on projects that exist in the current Nios II IDE workspace.
nios2-import-project	Imports a previously-created Nios II IDE project into the current workspace.
nios2-delete-project	Removes a project from the Nios II IDE workspace, and optionally deletes files from the file system.

Table 11–3 shows other Altera-provided command-line tools for developing Nios II programs.

Table 11–3. Altera Command-Line Tools

Tool	Description
nios2-download	Downloads code to a target processor for debugging or running.
nios2-flash-programmer	Programs data to flash memory on the target board.
nios2-gdb-server	Translates GNU debugger (GDB) remote serial protocol packets over TCP to joint test action group (JTAG) transactions with a target Nios II processor.
nios2-terminal	Performs terminal I/O with a JTAG universal asynchronous receiver-transmitter (UART) in a Nios II system
validate_zip	Verifies if a specified zip file is compatible with Altera's read-only zip file system.

File format conversion is sometimes necessary when passing data from one utility to another. Table 11–4 shows the Altera-provided utilities for converting file formats.

Table 11–4. File Conversion Utilities

Utility	Description
bin2flash	Converts binary files to a **.flash** file for programming into flash memory.
elf2dat	Converts an **.elf** executable file format to a **.dat** file format appropriate for Verilog HDL hardware simulators.
elf2flash	Converts an **.elf** executable file to a **.flash** file for programming into flash memory.
elf2hex	Converts an **.elf** executable file to the Intel **.hex** file format.
elf2mem	Generates the memory contents for the memory devices in a specific Nios II system.

Table 11–4. File Conversion Utilities

Utility	Description
`elf2mif`	Converts an **.elf** executable file to the Quartus II memory initialization file **(.mif)** format
`flash2dat`	Converts a **.flash** file to the **.dat** file format appropriate for Verilog HDL hardware simulators.
`mk-nios2-signaltap-mnemonic-table`	Takes an **.elf** file and an SOPC Builder system file (**.ptf**) and creates a **.stp** file containing mnemonic tables for Nios II instructions and symbols for Altera's SignalTap® II logic analyzer.
`sof2flash`	Converts an FPGA configuration file (**.sof**) to a **.flash** file for programming into flash memory.

Table 11–5 shows the Altera-provided tools that support backward-compatibility with the first-generation Nios processor SDK and tool flow.

For more information, refer to *AN 350: Upgrading Nios Processor Systems to the Nios II Processor.*

Table 11–5. Backward Compatibility Tools

Tool	Description
`nios2-build`	Compiles and links software projects based on the legacy SDK library.
`nios2-run`	Downloads a program to a Nios II processor and then performs terminal I/O to the program.
`nios2-debug`	Downloads a program to a Nios II processor and launches the Insight debugger.
`nios2-console`	Opens the FS2 command-line interface (CLI), connects to the Nios II processor, and (optionally) downloads code.

GNU Compiler Tool-chain

Altera provides and supports the standard GNU compiler tool-chain for the Nios II processor. Complete HTML documentation for the GNU tools resides in the Nios II development kit directory. The GNU tools are in the *<Nios II kit path>***/bin/nios2-gnutools** directory.

GNU tools for the Nios II processor are generally named **nios2-elf-**<*tool name*>. The following list shows some examples:

- `nios2-elf-gcc`
- `make`
- `nios2-elf-as`
- `nios2-elf-ld`
- `nios2-elf-objdump`
- `nios2-elf-size`

For a comprehensive list, refer to the GNU HTML documentation.

Libraries & Embedded Software Components

Table 11–6 shows the Nios II development kit libraries and software components.

Table 11–6. Development Kit Libraries & Software Components

Name	Description
Hardware abstraction layer (HAL) system library	See "Overview of the HAL System Library" on page 3–1.
MicroC/OS-II RTOS	See "MicroC/OS-II Real-Time Operating System" on page 8–1.
Lightweight IP TCP/IP stack	See "Ethernet & Lightweight IP" on page 9–1.
Newlib ANSI C standard library	See "Overview of the HAL System Library" on page 3–1. The complete HTML documentation for newlib resides in the Nios II development kit directory.
Read-only zip file system	See "Read-Only Zip Filing System" on page 12–1.
Example designs	The Nios II development kit provides documented software examples to demonstrate all prominent features of the Nios II processor and the development environment.