Show

Quartus II Help v13.0 > Using the Quartus II Executables in Shell Scripts

Using the Quartus II Executables in Shell Scripts

When you install the Quartus II software, the installer modifies your PATH environment variable so that you can run the executables from any directory, or from a script. One of the most basic uses of the Quartus II executables is to compile your project. You can run various processing commands with the Quartus II command-line executable **quartus_sh** with the `--flow` option. To direct the Quartus II software to run any flow predefined in the GUI, use the following syntax:

```
    quartus_sh --flow <flow_name> <project_name> [-c <revision_name>]
```

For example, to compile a project **top**, type the following:

```
quartus_sh --flow compile top
```

To compile the rev1 revision of the **top** project, type the following:

```
quartus_sh --flow compile top -c rev1
```

To compile the FPGA project named **top**, migrate to HardCopy, and compile the resulting HardCopy project, type the following:

```
quartus_sh --flow hardcopy_full_compile top
```

To compile the FPGA project named **top**, migrate to HardCopy, and compile the resulting HardCopy project using 3 separate Quartus II commands, type the following:

```
quartus_sh --flow compile top
quartus_sh --flow migrate_to_hardcopy
cd top_hardcopy_optimization
quartus_sh --flow compile top
```

For each argument passed to the --flow option, the executable calls a predefined script that opens the project and executes the given flow using the ::quartus::flow Tcl package.

Note: Tcl API functions for **quartus_cmd** are retained in the Quartus II software for backward compatibility only. The Quartus II Tcl Console supports both sets of API functions. Altera recommends that you use the executables listed in the table below for scripts that use both command-line executables and Tcl API functions.

More information about Tcl is available on the Altera website.

The following table describes each of the command-line executables.

Executable and Name	**Description**
Analysis & Synthesis **quartus_map**	Builds a single project database that integrates all the design files in a design entity or project hierarchy, performs logic synthesis to optimize the logic of the design, and performs technology mapping to implement the design logic with device resources such as logic elements.
Fitter **quartus_fit**	Performs place and route by fitting the logic of a design into a device. The Fitter selects appropriate interconnection paths, pin assignments, and logic cell assignments. You must run Analysis & Synthesis successfully before you run the Fitter.
JAM STAPL Player **quartus_jli**	Reads and executes Jam Files (**.jam**). A single JAM/STAPL File can perform several functions, such as programming, configuring, verifying, erasing, and blank-checking a programmable device.
JAM Compiler **quartus_jli**	The Quartus II JAM Compiler converts Jam/STAPL files to Jam Byte Code Files (**.jbc**).
TimeQuest Timing Analyzer **quartus_sta**	Validates the timing performance of all logic in the design using an industry-standard constraint, analysis, and reporting methodology. You can use the TimeQuest analyzer graphical user interface (GUI) or command-line interface to constrain, run, and view results for all timing paths in the design. The GUI version of **quartus_sta** is called **quartus_staw**. You can run **quartus_staw**

at the command-line or in the Quartus II GUI by clicking **TimeQuest Timing Analyzer** on the Tools menu. The **quartus_sta** executable includes Tcl support.

Assembler
quartus_asm

Generates a device programming image, in the form of one or more Altera Programming Unit (APU). Other programming hardware manufacturers that provide support for Altera devices can use Hexadecimal (Intel-Format) Output Files, Tabular Text Files, and Raw Binary Files. You must run the Fitter successfully before running the Assembler.

Design Assistant
quartus_drc

Checks the reliability of a design based on a set of design rules. The Design Assistant is especially useful for checking the reliability of a design before converting the design for HardCopy devices. The Design Assistant supports designs that target any Altera device supported by the Quartus II software. You must run Analysis & Elaboration, Analysis & Synthesis, or the Fitter successfully before running the Design Assistant.

Compiler Database Interface
quartus_cdb

Manages version-compatible database files and generates incremental netlists for use with LogicLock back-annotation. Alternately, **quartus_cdb** back-annotates device and resource assignments to preserve the fit for future compilations. The **quartus_cdb** executable includes Tcl support. You must run Analysis & Synthesis successfully before running the Compiler Database Interface.

EDA Netlist Writer
quartus_eda

Generates netlist and other output files for use with other EDA tools. The **quartus_eda** executable includes Tcl support. You must run Analysis & Synthesis, the Fitter, or the Timing Analyzer successfully before running the EDA Netlist Writer, depending on the arguments used.

SignalTap II Logic Analyzer
quartus_stp

Captures signals from internal device nodes while the device is running at speed. The captured data appears as a waveform within the SignalTap II Logic Analyzer. The SignalTap II Logic Analyzer adds or removes logic analyzer instances in a design, runs data acquisition on a device with Tcl commands, and creates a SignalTap II File from a SignalTap II Logic Analyzer megafunction compiled in a design. The **quartus_stp** executable includes Tcl support. You can add or remove an instance from a design before running Analysis & Synthesis. You must run Analysis & Synthesis successfully before you create a SignalTap II File from a design that contains a SignalTap II Logic Analyzer megafunction.

Programmer
quartus_pgm

The Programmer configures Altera devices using one of the supported file formats: Programmer Object File, SRAM Object File, Jam File, or Jam Byte Code File. You must specify a valid programming mode, programming cable, and programming operation for a given device.

Convert Programming File
quartus_cpf

Converts one programming file format to another format.

PowerPlay Power Analyzer
quartus_pow

The PowerPlay Power Analyzer produces detailed reports that you can use to analyze and optimize thermal power dissipation on a block-type or design-hierarchy basis.

Quartus II Shell
quartus_sh

Acts as a simple Quartus II Tcl interpreter. The Shell has a smaller memory footprint than the other executables that support Tcl: **quartus_cdb**, **quartus_eda**, **quartus_sim**, **quartus_sta**, and **quartus_stp**. You can start the Shell as an interactive Tcl interpreter (shell), use it to run a Tcl script, or use it to evaluate individual Tcl commands interactively.

Quartus II SSN Analyzer
quartus_si

The Quartus II SSN Analyzer estimates and reports the simultaneous switching noise for I/O in your design.

See Also ▸

Rate This Page

Contact Altera|Legal Notice