

2013.11.04

QIIS3013



Subscribe



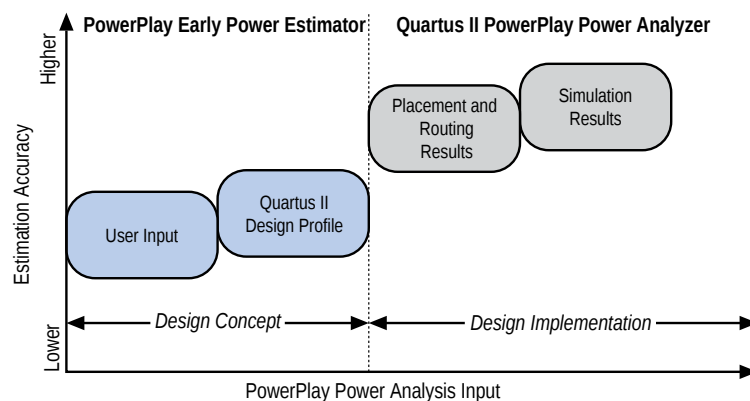
Send Feedback

The PowerPlay Power Analysis tools allow you to estimate device power consumption accurately.

As designs grow larger and process technology continues to shrink, power becomes an increasingly important design consideration. When designing a PCB, you must estimate the power consumption of a device accurately to develop an appropriate power budget, and to design the power supplies, voltage regulators, heat sink, and cooling system.

The following figure shows the PowerPlay Power Analysis tools ability to estimate power consumption from early design concept through design implementation.

Figure 8-1: PowerPlay Power Analysis From Design Concept Through Design Implementation



For the majority of the designs, the PowerPlay Power Analyzer and the PowerPlay EPE spreadsheet have the following accuracy after the power models are final:

- PowerPlay Power Analyzer— $\pm 20\%$ from silicon, assuming that the PowerPlay Power Analyzer uses the Value Change Dump File (.vcd) generated toggle rates.
- PowerPlay EPE spreadsheet— $\pm 20\%$ from the PowerPlay Power Analyzer results using .vcd generated toggle rates. 90% of EPE designs (using .vcd generated toggle rates exported from PPPA) are within $\pm 30\%$ silicon.

The toggle rates are derived using the PowerPlay Power Analyzer with a .vcd file generated from a gate level simulation representative of the system operation.

© 2013 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at www.altera.com/common/legal.html. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO
9001:2008
Registered



Related Information

- [About Power Estimation and Analysis](#)
- [PowerPlay Early Power Estimators \(EPE\) and Power Analyzer](#)

Types of Power Analyses

Understanding the uses of power analysis and the factors affecting power consumption helps you to use the PowerPlay Power Analyzer effectively. Power analysis meets the following significant planning requirements:

- **Thermal planning**—Thermal power is the power that dissipates as heat from the FPGA. You must use a heatsink or fan to act as a cooling solution for your device. The cooling solution must be sufficient to dissipate the heat that the device generates. The computed junction temperature must fall within normal device specifications.
- **Power supply planning**—Power supply is the power needed to run your device. Power supplies must provide adequate current to support device operation.

Note: For power supply planning, use the PowerPlay EPE at the early stages of your design cycle. Use the PowerPlay Power Analyzer reports when your design is complete to get an estimate of your design power requirement.

The two types of analyses are closely related because much of the power supplied to the device dissipates as heat from the device; however, in some situations, the two types of analyses are not identical. For example, if you use terminated I/O standards, some of the power drawn from the power supply of the device dissipates in termination resistors rather than in the device.

Power analysis also addresses the activity of your design over time as a factor that impacts the power consumption of the device. The static power (P_{STATIC}) is the thermal power dissipated on chip, independent of user clocks. P_{STATIC} includes the leakage power from all FPGA functional blocks, except for I/O DC bias power and transceiver DC bias power, which are accounted for in the I/O and transceiver sections. Dynamic power is the additional power consumption of the device due to signal activity or toggling.

Related Information

- [PowerPlay Early Power Estimator \(EPE\) User Guide](#)

Differences between the PowerPlay EPE and the Quartus II PowerPlay Power Analyzer

The following table lists the differences between the PowerPlay EPE and the Quartus II PowerPlay Power Analyzer.

Table 8-1: Comparison of the PowerPlay EPE and Quartus II PowerPlay Power Analyzer

Characteristic	PowerPlay EPE	Quartus II PowerPlay Power Analyzer
Phase in the design cycle	Any time, but it is recommended to use Quartus II PowerPlay Power Analyzer for post-fit power analysis.	Post-fit
Tool requirements	Spreadsheet program	The Quartus II software

Characteristic	PowerPlay EPE	Quartus II PowerPlay Power Analyzer
Accuracy	Medium	Medium to very high
Data inputs	<ul style="list-style-type: none"> Resource usage estimates Clock requirements Environmental conditions Toggle rate 	<ul style="list-style-type: none"> Post-fit design Clock requirements Signal activity defaults Environmental conditions Register transfer level (RTL) simulation results (optional) Post-fit simulation results (optional) Signal activities per node or entity (optional)
Data outputs ⁽¹⁾	<ul style="list-style-type: none"> Total thermal power dissipation Thermal static power Thermal dynamic power Off-chip power dissipation Current drawn from voltage supplies 	<ul style="list-style-type: none"> Total thermal power Thermal static power Thermal dynamic power Thermal I/O power Thermal power by design hierarchy Thermal power by block type Thermal power dissipation by clock domain Off-chip (non-thermal) power dissipation Device supply currents

The result of the PowerPlay Power Analyzer is only an estimation of power. Altera does not recommend using the result as a specification. The purpose of the estimation is to help you establish guidelines for the power budget of your design. It is important that you verify the actual power during device operation as the information is sensitive to the actual device design and the environmental operating conditions.

Note: The PowerPlay Power Analyzer does not include the transceiver power for features that can only be enabled through dynamic reconfiguration (DFE, ADCE/AEQ, EyeQ). Use the EPE to estimate the incremental power consumption by these features.

Related Information

- [PowerPlay Early Power Estimators \(EPE\) and Power Analyzer Page](#)
For more information, refer to the device-specific PowerPlay Early Power Estimators (EPE) page on the Altera website.
- [PowerPlay Power Analyzer Reports](#)
For more information, refer to this page for device-specific information about the PowerPlay Early Power Estimator.

⁽¹⁾ PowerPlay EPE and PowerPlay Power Analyzer outputs vary by device family. For more information, refer to the device-specific PowerPlay Early Power Estimators (EPE) and Power Analyzer Page and PowerPlay Power Analyzer Reports in the Quartus II Help.

Factors Affecting Power Consumption

Understanding the following factors that affect power consumption allows you to use the PowerPlay Power Analyzer and interpret its results effectively:

- [Device Selection](#)
- [Environmental Conditions](#)
- [Device Resource Usage](#)
- [Signal Activities](#)

Device Selection

Device families have different power characteristics. Many parameters affect the device family power consumption, including choice of process technology, supply voltage, electrical design, and device architecture.

Power consumption also varies in a single device family. A larger device consumes more static power than a smaller device in the same family because of its larger transistor count. Dynamic power can also increase with device size in devices that employ global routing architectures.

The choice of device package also affects the ability of the device to dissipate heat. This choice can impact your required cooling solution choice to comply to junction temperature constraints.

Process variation can affect power consumption. Process variation primarily impacts static power because sub-threshold leakage current varies exponentially with changes in transistor threshold voltage. Therefore, you must consult device specifications for static power and not rely on empirical observation. Process variation has a weak effect on dynamic power.

Environmental Conditions

Operating temperature primarily affects device static power consumption. Higher junction temperatures result in higher static power consumption. The device thermal power and cooling solution that you use must result in the device junction temperature remaining within the maximum operating range for the device. The main environmental parameters affecting junction temperature are the cooling solution and ambient temperature.

The following table lists the environmental conditions that could affect power consumption.

Table 8-2: Environmental Conditions that Could Affect Power Consumption

Environmental Conditions	Description
Airflow	<p>A measure of how quickly the device removes heated air from the vicinity of the device and replaces it with air at ambient temperature.</p> <p>You can either specify airflow as “still air” when you are not using a fan, or as the linear feet per minute rating of the fan in the system. Higher airflow decreases thermal resistance.</p>

Environmental Conditions	Description
Heat Sink and Thermal Compound	A heat sink allows more efficient heat transfer from the device to the surrounding area because of its large surface area exposed to the air. The thermal compound that interfaces the heat sink to the device also influences the rate of heat dissipation. The case-to-ambient thermal resistance (θ_{CA}) parameter describes the cooling capacity of the heat sink and thermal compound employed at a given airflow. Larger heat sinks and more effective thermal compounds reduce θ_{CA} .
Junction Temperature	<p>The junction temperature of a device is equal to:</p> $T_{\text{Junction}} = T_{\text{Ambient}} + P_{\text{Thermal}} \cdot \theta_{JA}$ <p>in which θ_{JA} is the total thermal resistance from the device transistors to the environment, having units of degrees Celsius per watt. The value θ_{JA} is equal to the sum of the junction-to-case (package) thermal resistance (θ_{JC}), and the case-to-ambient thermal resistance (θ_{CA}) of your cooling solution.</p>
Board Thermal Model	The junction-to-board thermal resistance (θ_{JB}) is the thermal resistance of the path through the board, having units of degrees Celsius per watt. To compute junction temperature, you can use this board thermal model along with the board temperature, the top-of-chip θ_{JA} and ambient temperatures.

Device Resource Usage

The number and types of device resources used greatly affects power consumption.

- **Number, Type, and Loading of I/O Pins**—Output pins drive off-chip components, resulting in high-load capacitance that leads to a high-dynamic power per transition. Terminated I/O standards require external resistors that draw constant (static) power from the output pin.
- **Number and Type of Hard Logic Blocks**—A design with more logic elements (LEs), multiplier elements, memory blocks, transceiver blocks or HPS system tends to consume more power than a design with fewer circuit elements. The operating mode of each circuit element also affects its power consumption. For example, a DSP block performing 18×18 multiplications and a DSP block performing multiply-accumulate operations consume different amounts of dynamic power because of different amounts of charging internal capacitance on each transition. The operating mode of a circuit element also affects static power.
- **Number and Type of Global Signals**—Global signal networks span large portions of the device and have high capacitance, resulting in significant dynamic power consumption. The type of global signal is important as well. For example, Stratix V devices support global clocks and quadrant (regional) clocks. Global clocks cover the entire device, whereas quadrant clocks only span one-fourth of the device. Clock networks that span smaller regions have lower capacitance and tend to consume less power. The location of the logic array blocks (LABs) driven by the clock network can also have an impact because the Quartus II software automatically disables unused branches of a clock.

Signal Activities

The behavior of each signal in your design is an important factor in estimating power consumption. The following table lists the two vital behaviors of a signal, which are toggle rate and static probability:

Table 8-3: Signal Behavior

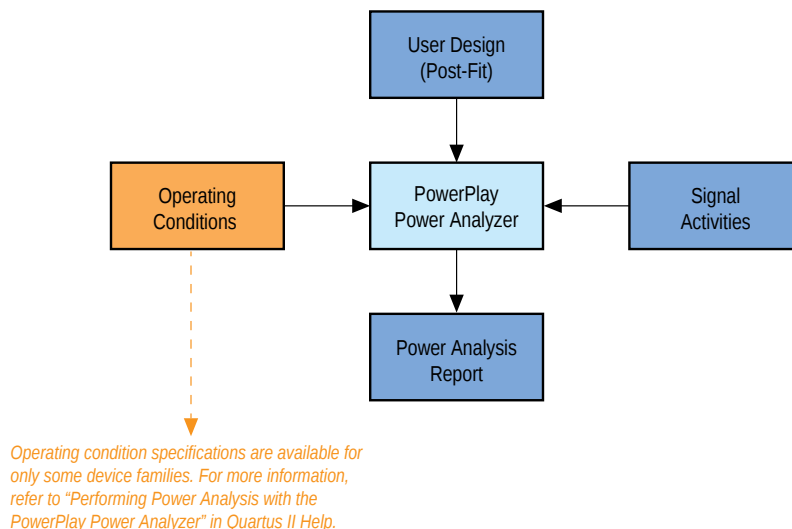
Signal Behavior	Description
Toggle rate	<ul style="list-style-type: none">The toggle rate of a signal is the average number of times that the signal changes value per unit of time. The units for toggle rate are transitions per second and a transition is a change from 1 to 0, or 0 to 1.Dynamic power increases linearly with the toggle rate as you charge the board trace model more frequently for logic and routing. The Quartus II software models full rail-to-rail switching. For high toggle rates, especially on circuit output I/O pins, the circuit can transition before fully charging the downstream capacitance. The result is a slightly conservative prediction of power by the PowerPlay Power Analyzer.
Static probability	<ul style="list-style-type: none">The static probability of a signal is the fraction of time that the signal is logic 1 during the period of device operation that is being analyzed. Static probability ranges from 0 (always at ground) to 1 (always at logic-high).Static probabilities of their input signals can sometimes affect the static power that routing and logic consume. This effect is due to state-dependent leakage and has a larger effect on smaller process geometries. The Quartus II software models this effect on devices at 90 nm or smaller if it is important to the power estimate. The static power also varies with the static probability of a logic 1 or 0 on the I/O pin when output I/O standards drive termination resistors.

Note: To get accurate results from the power analysis, the signal activities for analysis must represent the actual operating behavior of your design. Inaccurate signal toggle rate data is the largest source of power estimation error.

PowerPlay Power Analyzer Flow

The PowerPlay Power Analyzer supports accurate power estimations by allowing you to specify the important design factors affecting power consumption. The following figure shows the high-level PowerPlay Power Analyzer flow.

Figure 8-2: PowerPlay Power Analyzer High-Level Flow



To obtain accurate I/O power estimates, the PowerPlay Power Analyzer requires you to synthesize your design and then fit your design to the target device. You must specify the electrical standard on each I/O cell and the board trace model on each I/O standard in your design.

Related Information

- [Performing Power Analysis with the PowerPlay Power Analyzer](#)

Operating Settings and Conditions

You can specify device power characteristics, operating voltage conditions, and operating temperature conditions for power analysis in the Quartus II software.

On the **Operating Settings and Conditions** page of the **Settings** dialog box, you can specify whether the device has typical power consumption characteristics or maximum power consumption characteristics.

On the **Voltage** page of the **Settings** dialog box, you can view the operating voltage conditions for each power rail in the device, and specify supply voltages for power rails with selectable supply voltages.

Note: The Quartus II Fitter may override some of the supply voltages settings specified in this chapter. For example, supply voltages for several Stratix V transceiver power supplies depend on the data rate used. If the Fitter detects that voltage required is different from the one specified in the **Voltage** page, it will automatically set the correct voltage for relevant rails. The Quartus II PowerPlay Power Analyzer uses voltages selected by the Fitter if they conflict with the settings specified in the **Voltage** page.

On the **Temperature** page of the **Settings** dialog box, you can specify the thermal operating conditions of the device.

Related Information

- [Operating Settings and Conditions Page \(Settings Dialog Box\)](#)

- [Voltage Page \(Settings Dialog Box\)](#)
- [Temperature Page \(Settings Dialog Box\)](#)

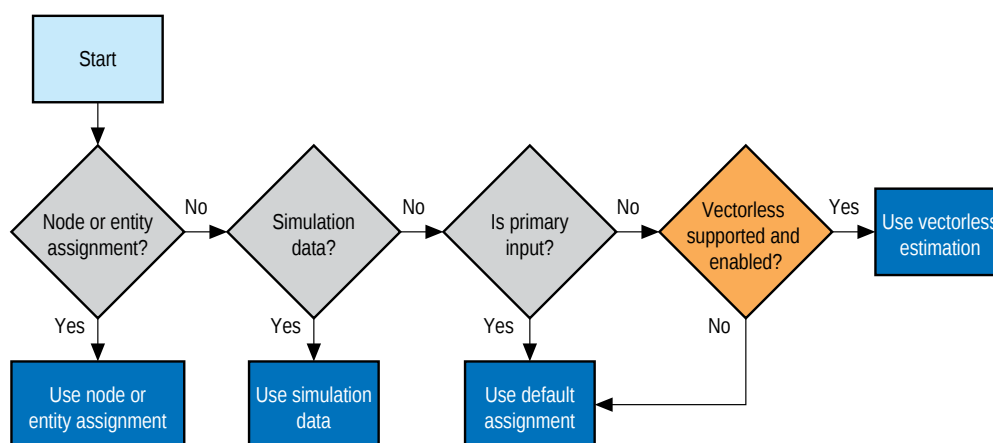
Signal Activities Data Sources

The PowerPlay Power Analyzer provides a flexible framework for specifying signal activities. The framework reflects the importance of using representative signal-activity data during power analysis. Use the following sources to provide information about signal activity:

- Simulation results
- User-entered node, entity, and clock assignments
- User-entered default toggle rate assignment
- Vectorless estimation

The PowerPlay Power Analyzer allows you to mix and match the signal-activity data sources on a signal-by-signal basis. The following figure shows the priority scheme applied to each signal.

Figure 8-3: Signal-Activity Data Source Priority Scheme



Related Information

- [Performing Power Analysis with the PowerPlay Power Analyzer](#)

Simulation Results

The PowerPlay Power Analyzer directly reads the waveforms generated by a design simulation. Static probability and toggle rate can be calculated for each signal from the simulation waveform. Power analysis is most accurate when you use representative input stimuli to generate simulations.

The PowerPlay Power Analyzer reads results generated by the following simulators:

- ModelSim®
- ModelSim-Altera
- QuestaSim
- Active-HDL
- NCSim

- VCS
- VCS MX
- Riviera-PRO

Signal activity and static probability information are derived from a Verilog Value Change Dump File (**.vcd**). For more information, refer to [Signal Activities](#) on page 8-6.

For third-party simulators, use the **EDA Tool Settings** to specify the Generate Value Change Dump (VCD) file script option in the Simulation page of the Settings dialog box. These scripts instruct the third-party simulators to generate a **.vcd** that encodes the simulated waveforms. The Quartus II PowerPlay Power Analyzer reads this file directly to derive the toggle rate and static probability data for each signal.

Third-party EDA simulators, other than those listed, can generate a **.vcd** that you can use with the PowerPlay Power Analyzer. For those simulators, you must manually create a simulation script to generate the appropriate **.vcd**.

Note: You can use a **.vcd** created for power analysis to optimize your design for power during fitting by utilizing the appropriate settings in the PowerPlay power optimization list, available in the **Fitter Settings** page of the **Settings** dialog box.

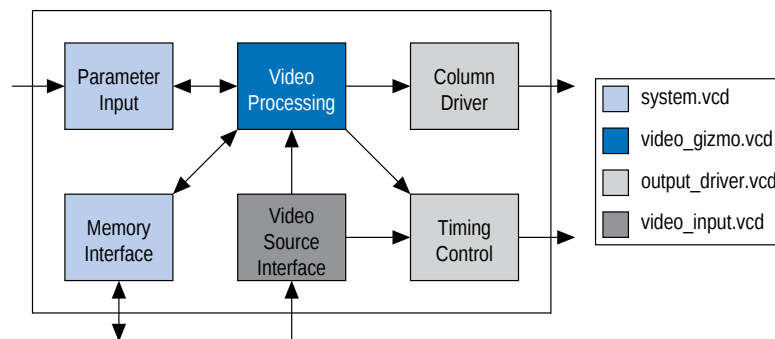
Related Information

- [Power Optimization](#)
- [Section I. Simulation](#)

Using Simulation Files in Modular Design Flows

A common design practice is to create modular or hierarchical designs in which you develop each design entity separately, and then instantiate these modules in a higher-level entity to form a complete design. You can perform simulation on a complete design or on each module for verification. The PowerPlay Power Analyzer supports modular design flows when reading the signal activities from simulation files. The following figure shows an example of a modular design flow.

Figure 8-4: Modular Simulation Flow



When specifying a simulation file (a **.vcd**), the software provides support to specify an associated design entity name, such that the PowerPlay Power Analyzer imports the signal activities derived from that file for the specified design entity. The PowerPlay Power Analyzer also supports the specification of multiple **.vcd**

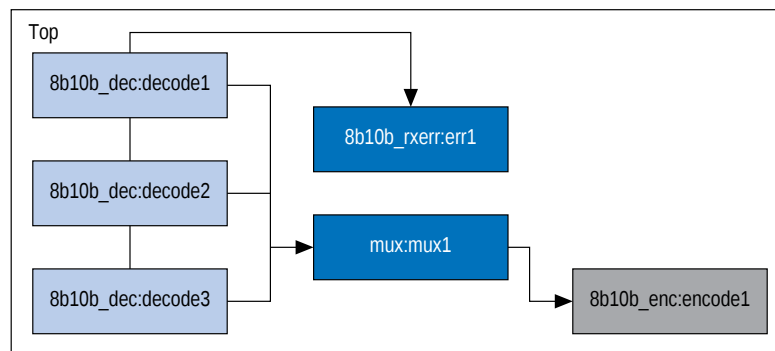
files for power analysis, with each having an associated design entity name to enable the integration of partial design simulations into a complete design power analysis. When specifying multiple **.vcd** files for your design, more than one simulation file can contain signal-activity information for the same signal.

Note: When you apply multiple **.vcd** files to the same design entity, the signal activity used in the power analysis is the equal-weight arithmetic average of each **.vcd**.

Note: When you apply multiple simulation files to design entities at different levels in your design hierarchy, the signal activity in the power analysis derives from the simulation file that applies to the most specific design entity.

The following figure shows an example of a hierarchical design. The top-level module of your design, called **Top**, consists of three 8b/10b decoders, followed by a mux. The software then encodes the output of the mux to produce the final output of the top-level module. An error-handling module handles any 8b/10b decoding errors. The Top module contains the top-level entity of your design and any logic not defined as part of another module. The design file for the top-level module might be a wrapper for the hierarchical entities below it, or it might contain its own logic. The following usage scenarios show common ways that you can simulate your design and import the **.vcd** into the PowerPlay Power Analyzer.

Figure 8-5: Example Hierarchical Design



Complete Design Simulation

You can simulate the entire design and generate a **.vcd** from a third-party simulator. The PowerPlay Power Analyzer can then import the **.vcd** (specifying the top-level design). The resulting power analysis uses the signal activities information from the generated **.vcd**, including those that apply to submodules, such as `decode [1-3]`, `err1`, `mux1`, and `encode1`.

Modular Design Simulation

You can independently simulate of the top-level design, and then import all the resulting **.vcd** files into the PowerPlay Power Analyzer. For example, you can simulate the `8b10b_dec` independent of the entire design and `mux`, `8b10b_rxerr`, and `8b10b_enc`. You can then import the **.vcd** files generated from each simulation by specifying the appropriate instance name. For example, if the files produced by the

simulations are **8b10b_dec.vcd**, **8b10b_enc.vcd**, **8b10b_rxerr.vcd**, and **mux.vcd**, you can use the import specifications in the following table:

Table 8-4: Import Specifications

File Name	Entity
8b10b_dec.vcd	Top 8b10b_dec:decode1
8b10b_dec.vcd	Top 8b10b_dec:decode2
8b10b_dec.vcd	Top 8b10b_dec:decode3
8b10b_rxerr.vcd	Top 8b10b_rxerr:err1
8b10b_enc.vcd	Top 8b10b_enc:encode1
mux.vcd	Top mux:mux1

The resulting power analysis applies the simulation vectors in each file to the assigned entity. Simulation provides signal activities for the pins and for the outputs of functional blocks. If the inputs to an entity instance are input pins for the entire design, the simulation file associated with that instance does not provide signal activities for the inputs of that instance. For example, an input to an entity such as **mux1** has its signal activity specified at the output of one of the decode entities.

Multiple Simulations on the Same Entity

You can perform multiple simulations of an entire design or specific modules of a design. For example, in the process of verifying the top-level design, you can have three different simulation testbenches: one for normal operation, and two for corner cases. Each of these simulations produces a separate **.vcd**. In this case, apply the different **.vcd** file names to the same top-level entity, as shown in the following table.

Table 8-5: Multiple Simulation File Names and Entities

File Name	Entity
normal.vcd	Top
corner1.vcd	Top
corner2.vcd	Top

The resulting power analysis uses an arithmetic average of the signal activities calculated from each simulation file to obtain the final signal activities used. If a signal **err_out** has a toggle rate of zero transition per second in **normal.vcd**, 50 transitions per second in **corner1.vcd**, and 70 transitions per second in **corner2.vcd**, the final toggle rate in the power analysis is 40 transitions per second.

If you do not want the PowerPlay Power Analyzer to read information from multiple instances and take an arithmetic average of the signal activities, use a **.vcd** that includes only signals from the instance that you care about.

Overlapping Simulations

You can perform a simulation on the entire design, and more exhaustive simulations on a submodule, such as **8b10b_rxerr**. The following table lists the import specification for overlapping simulations.

Table 8-6: Overlapping Simulation Import Specifications

File Name	Entity
full_design.vcd	Top
error_cases.vcd	Top 8b10b_rxerr:err1

In this case, the software uses signal activities from **error_cases.vcd** for all the nodes in the generated **.vcd** and uses signal activities from **full_design.vcd** for only those nodes that do not overlap with nodes in **error_cases.vcd**. In general, the more specific hierarchy (the most bottom-level module) derives signal activities for overlapping nodes.

Partial Simulations

You can perform a simulation in which the entire simulation time is not applicable to signal-activity calculation. For example, if you run a simulation for 10,000 clock cycles and reset the chip for the first 2,000 clock cycles. If the PowerPlay Power Analyzer performs the signal-activity calculation over all 10,000 cycles, the toggle rates are only 80% of their steady state value (because the chip is in reset for the first 20% of the simulation). In this case, you must specify the useful parts of the **.vcd** for power analysis. The **Limit VCD Period** option enables you to specify a start and end time when performing signal-activity calculations.

Specifying Start and End Time when Performing Signal-Activity Calculations using the Limit VCD Period Option

To specify a start and end time when performing signal-activity calculations using the **Limit VCD period** option, follow these steps:

1. In the Quartus II software, on the Assignments menu, click **Settings**.
2. Under the Category list, click **PowerPlay Power Analyzer Settings**.
3. Turn on the **Use input file(s) to initialize toggle rates and static probabilities during power analysis** option.
4. Click **Add**.
5. In the **File name** and **Entity** fields, browse to the necessary files.
6. Under Simulation period, turn on **VCD file** and **Limit VCD period** options.
7. In the **Start time** and **End time** fields, specify the desired start and end time.
8. Click **OK**.

You can also use the following tcl or qsf assignment to specify **.vcd** files:

```
set_global_assignment -name POWER_INPUT_FILE_NAME "test.vcd" -section_id test.vcd

set_global_assignment -name POWER_INPUT_FILE_TYPE VCD -section_id test.vcd

set_global_assignment -name POWER_VCD_FILE_START_TIME "10 ns" -section_id test.vcd

set_global_assignment -name POWER_VCD_FILE_END_TIME "1000 ns" -section_id test.vcd
```

```
set_instance_assignment -name POWER_READ_INPUT_FILE test.vcd -to  
test_design
```

Related Information

- [set_power_file_assignment](#)
- [Add/Edit Power Input File Dialog Box](#)

Node Name Matching Considerations

Node name mismatches happen when you have **.vcd** applied to entities other than the top-level entity. In a modular design flow, the gate-level simulation files created in different Quartus II projects might not match their node names with the current Quartus II project.

For example, you may have a file named **8b10b_enc.vcd**, which the Quartus II software generates in a separate project called **8b10b_enc** while simulating the 8b10b encoder. If you import the **.vcd** into another project called **Top**, you might encounter name mismatches when applying the **.vcd** to the 8b10b_enc module in the **Top** project. This mismatch happens because the Quartus II software might name all the combinational nodes in the **8b10b_enc.vcd** differently than in the **Top** project.

You can avoid name mismatching with only RTL simulation data, in which register names do not change, or with an incremental compilation flow that preserves node names along with a gate-level simulation.

Note: To ensure accuracy, Altera recommends that you use an incremental compilation flow to preserve the node names of your design.

Related Information

- [Quartus II Incremental Compilation for Hierarchical and Team-Based Design](#)

Glitch Filtering

The PowerPlay Power Analyzer defines a glitch as two signal transitions so closely spaced in time that the pulse, or glitch, occurs faster than the logic and routing circuitry can respond. The output of a transport delay model simulator contains glitches for some signals. The logic and routing structures of the device form a low-pass filter that filters out glitches that are tens to hundreds of picoseconds long, depending on the device family.

Some third-party simulators use different models than the transport delay model as the default model. Different models cause differences in signal activity and power estimation. The inertial delay model, which is the ModelSim default model, filters out more glitches than the transport delay model and usually yields a lower power estimate.

Note: Altera recommends that you use the transport simulation model when using the Quartus II software glitch filtering support with third-party simulators. Simulation glitch filtering has little effect if you use the inertial simulation model.

Glitch filtering in a simulator can also filter a glitch on one logic element (LE) (or other circuit element) output from propagating to downstream circuit elements to ensure that the glitch does not affect simulated results. Glitch filtering prevents a glitch on one signal from producing non-physical glitches on all downstream logic, which can result in a signal toggle rate and a power estimate that are too high. Circuit elements in

which every input transition produces an output transition, including multipliers and logic cells configured to implement XOR functions, are especially prone to glitches. Therefore, circuits with such functions can have power estimates that are too high when glitch filtering is not used.

Note: Altera recommends that you use the glitch filtering feature to obtain the most accurate power estimates. For .vcd files, the PowerPlay Power Analyzer flows support two levels of glitch filtering.

Enabling First Level of Glitch Filtering

To enable the first level of glitch filtering in the Quartus II software for supported third-party simulators, follow these steps:

1. On the Assignments menu, click **Settings**.
2. In the **Category** list, select **Simulation under EDA Tool Settings**.
3. Select the **Tool name** to use for the simulation.
4. Turn on **Enable glitch filtering**.

Enabling Second Level of Glitch Filtering

The second level of glitch filtering occurs while the PowerPlay Power Analyzer is reading the .vcd generated by a third-party simulator. To enable the second level of glitch filtering, follow these steps:

1. On the Assignments menu, click **Settings**.
2. In the **Category** list, select **PowerPlay Power Analyzer Settings**.
3. Under **Input File(s)**, turn on **Perform glitch filtering on VCD files**.

The .vcd file reader performs filtering complementary to the filtering performed during simulation and is often not as effective. While the .vcd file reader can remove glitches on logic blocks, the file reader cannot determine how a given glitch affects downstream logic and routing, and may eliminate the impact of the glitch completely. Filtering the glitches during simulation avoids switching downstream routing and logic automatically.

Note: When running simulation for design verification (rather than to produce input to the PowerPlay Power Analyzer), Altera recommends that you turn off the glitch filtering option to produce the most rigorous and conservative simulation from a functionality viewpoint. When performing simulation to produce input for the PowerPlay Power Analyzer, Altera recommends that you turn on the glitch filtering to produce the most accurate power estimates.

Node and Entity Assignments

You can assign toggle rates and static probabilities to individual nodes and entities in the design. These assignments have the highest priority, overriding data from all other signal-activity sources.

You must use the Assignment Editor or Tcl commands to create the **Power Toggle Rate** and **Power Static Probability** assignments. You can specify the power toggle rate as an absolute toggle rate in transitions per second using the **Power Toggle Rate** assignment, or you can use the **Power Toggle Rate Percentage** assignment to specify a toggle rate relative to the clock domain of the assigned node for a more specific assignment made in terms of hierarchy level.

Note: If you use the **Power Toggle Rate Percentage** assignment, and the node does not have a clock domain, the Quartus II software issues a warning and ignores the assignment.

Assigning toggle rates and static probabilities to individual nodes and entities is appropriate for signals in which you have knowledge of the signal or entity being analyzed. For example, if you know that a 100 MHz data bus or memory output produces data that is essentially random (uncorrelated in time), you can directly enter a 0.5 static probability and a toggle rate of 50 million transitions per second.

The PowerPlay Power Analyzer treats bidirectional I/O pins differently. The combinational input port and the output pad for a pin share the same name. However, those ports might not share the same signal activities. For reading signal-activity assignments, the PowerPlay Power Analyzer creates a distinct name `<node_name~output>` when configuring the bidirectional signal as an output and `<node_name~result>` when configuring the signal as an input. For example, if a design has a bidirectional pin named `MYPIN`, assignments for the combinational input use the name `MYPIN~result`, and the assignments for the output pad use the name `MYPIN~output`.

Note: When you create the logic assignment in the Assignment Editor, you cannot find the `MYPIN~result` and `MYPIN~output` node names in the Node Finder. Therefore, to create the logic assignment, you must manually enter the two differentiating node names to create the assignment for the input and output port of the bidirectional pin.

Related Information

- [Constraining Designs](#)

For more information about how to use the Assignment Editor in the Quartus II software, refer to this document.

Timing Assignments to Clock Nodes

For clock nodes, the PowerPlay Power Analyzer uses timing requirements to derive the toggle rate when neither simulation data nor user-entered signal-activity data is available. f_{MAX} requirements specify full cycles per second, but each cycle represents a rising transition and a falling transition. For example, a clock f_{MAX} requirement of 100 MHz corresponds to 200 million transitions per second for the clock node.

Default Toggle Rate Assignment

You can specify a default toggle rate for primary inputs and other nodes in your design. The PowerPlay Power Analyzer uses the default toggle rate when no other method specifies the signal-activity data.

The PowerPlay Power Analyzer specifies the toggle rate in absolute terms (transitions per second), or as a fraction of the clock rate in effect for each node. The toggle rate for a clock derives from the timing settings for the clock. For example, if the PowerPlay Power Analyzer specifies a clock with an f_{MAX} constraint of 100 MHz and a default relative toggle rate of 20%, nodes in this clock domain transition in 20% of the clock periods, or 20 million transitions occur per second. In some cases, the PowerPlay Power Analyzer cannot determine the clock domain for a node because either the PowerPlay Power Analyzer cannot determine a clock domain for the node, or the clock domain is ambiguous. For example, the PowerPlay Power Analyzer may not be able to determine a clock domain for a node if the user did not specify sufficient timing assignments. In these cases, the PowerPlay Power Analyzer substitutes and reports a toggle rate of zero.

Vectorless Estimation

For some device families, the PowerPlay Power Analyzer automatically derives estimates for signal activity on nodes with no simulation or user-entered signal-activity data. Vectorless estimation statistically estimates the signal activity of a node based on the signal activities of nodes feeding that node, and on the actual logic

function that the node implements. Vectorless estimation cannot derive signal activities for primary inputs. Vectorless estimation is accurate for combinational nodes, but not for registered nodes. Therefore, the PowerPlay Power Analyzer requires simulation data for at least the registered nodes and I/O nodes for accuracy.

The **PowerPlay Power Analyzer Settings** dialog box allows you to disable vectorless estimation. When turned on, vectorless estimation takes precedence over default toggle rates. Vectorless estimation does not override clock assignments.

To disable vectorless estimation, perform the following steps:

1. In the Quartus II software, on the Assignments menu, click **Settings**.
2. In the Category list, select **PowerPlay Power Analyzer Settings**.
3. Turn off the **Use vectorless estimation** option.

Related Information

- [Performing Power Analysis with the PowerPlay Power Analyzer](#)

Using the PowerPlay Power Analyzer

For flows that use the PowerPlay Power Analyzer, you must first synthesize your design, and then fit it to the target device. You must either provide timing assignments for all the clocks in your design, or use a simulation-based flow to generate activity data. You must specify the I/O standard on each device input and output and the board trace model on each output in your design.

Related Information

- [Performing Power Analysis with the PowerPlay Power Analyzer](#)

Common Analysis Flows

You can use the analysis flows in this section with the PowerPlay Power Analyzer. However, vectorless activity estimation is only available for some device families.

Signal Activities from Full Post-Fit Netlist (Timing) Simulation

Timing Simulation flow provides the most accuracy because all node activities reflect actual design behavior if supplied input vectors are representative of typical design operation. Results are better if the simulation filters glitches. The disadvantage of this method is that the simulation time is long.

Signal Activities from Full Post-Fit Netlist (Zero Delay) Simulation

You can use the zero delay simulation flow with designs for which signal activities from a full post-fit netlist (timing) simulation are not available. Zero delay simulation is as accurate as timing simulation in 95% of designs with no glitches.

Note: If your design has glitches, the power estimation may not be accurate. Altera recommends that you use the signal activities from a full post-fit netlist (timing) simulation to achieve an accurate power estimation of your design.

The following designs might exhibit glitches:

- Designs with many XOR gates (for example, an encryption core)
- Designs with arithmetic blocks without input and output registers (DSPs and carry chains)

Related Information

- [Generating a .vcd from Full Post-Fit Netlist \(Zero Delay\) Simulation](#) on page 8-19
For more information about creating zero delay simulation signal activities, refer to this page.

Signal Activities from RTL (Functional) Simulation, Supplemented by Vectorless Estimation

In the functional simulation flow, simulation provides toggle rates and static probabilities for all pins and registers in your design. Vectorless estimation fills in the values for all the combinational nodes between pins and registers, giving good results. This flow usually provides a compilation time benefit when you use the third-party RTL simulator.

RTL Simulation Limitation

RTL simulation may not provide signal activities for all registers in the post-fitting netlist because synthesis loses some register names. For example, synthesis might automatically transform state machines and counters, thus changing the names of registers in those structures.

Signal Activities from Vectorless Estimation and User-Supplied Input Pin Activities

The vectorless estimation flow provides a low level of accuracy, because vectorless estimation for registers is not entirely accurate.

Signal Activities from User Defaults Only

The user defaults only flow provides the lowest degree of accuracy.

Importance of .vcd

Altera recommends that you use a **.vcd** or a **.saf** generated by gate-level timing simulation for an accurate power estimation because gate-level timing simulation takes all the routing resources and the exact logic array resource usage into account.

Generating a .vcd

In previous versions of the Quartus II software, you could use either the Quartus II simulator or an EDA simulator to perform your simulation. The Quartus II software no longer supports a built-in simulator, and you must use an EDA simulator to perform simulation. Use the **.vcd** as the input to the PowerPlay Power Analyzer to estimate power for your design.

To create a **.vcd** for your design, follow these steps:

1. On the Assignments menu, click **Settings**.
2. In the **Category** list, under **EDA Tool Settings**, click **Simulation**.
3. In the **Tool name** list, select your preferred EDA simulator.
4. In the **Format for output netlist** list, select **Verilog HDL**, or **SystemVerilog HDL**, or **VHDL**.
5. Turn on **Generate Value Change Dump (VCD) file script**.

This option turns on the **Map illegal HDL characters** and **Enable glitch filtering** options. The **Map illegal HDL characters** option ensures that all signals have legal names and that signal toggle rates are available later in the PowerPlay Power Analyzer. The **Enable glitch filtering** option directs the EDA Netlist Writer to perform glitch filtering when generating VHDL Output Files, Verilog Output Files, and the corresponding Standard Delay Format Output Files for use with other EDA simulation tools. This option is available regardless of whether or not you want to generate .vcd scripts.

Note: When performing simulation using ModelSim, the **+nospecify** option for the `vsim` command disables the **specify path delays and timing checks** option in ModelSim. By enabling glitch filtering on the **Simulation** page, the simulation models include specified path delays. Thus, ModelSim might fail to simulate a design if you enabled glitch filtering and specified the **+nospecify** option. Altera recommends that you remove the **+nospecify** option from the ModelSim `vsim` command to ensure accurate simulation for power estimation.

6. Click **Script Settings**. Select the signals that you want to output to the .vcd.

With **All signals** selected, the generated script instructs the third-party simulator to write all connected output signals to the .vcd. With **All signals except combinational lcell outputs** selected, the generated script tells the third-party simulator to write all connected output signals to the .vcd, except logic cell combinational outputs.

Note: The file can become extremely large if you write all output signals to the file because the file size depends on the number of output signals being monitored and the number of transitions that occur.

7. Click **OK**.
8. In the **Design instance name** box, type a name for your testbench.
9. Compile your design with the Quartus II software and generate the necessary EDA netlist and script that instructs the third-party simulator to generate a .vcd.
10. Perform a simulation with the third-party EDA simulation tool. Call the generated script in the simulation tool before running the simulation. The simulation tool generates the .vcd and places it in the project directory.

Related Information

- [Simulation Results](#) on page 8-8
- [Glitch Filtering](#) on page 8-13
- [Section I. Simulation](#)

Generating a .vcd from ModelSim Software

To generate a .vcd with the ModelSim software, follow these steps:

1. In the Quartus II software, on the Assignments menu, click **Settings**.
2. In the **Category** list, under **EDA Tool Settings**, click **Simulation**.
3. In the **Tool name** list, select your preferred EDA simulator.
4. In the **Format for output netlist** list, select **Verilog HDL**, or **SystemVerilog HDL**, or **VHDL**.
5. Turn on **Generate Value Change Dump (VCD) file script**.
6. To generate the .vcd, perform a full compilation.
7. In the ModelSim software, compile the files necessary for simulation.

8. Load your design by clicking **Start Simulation** on the Tools menu, or use the `vsim` command.
9. Use the .vcd script created in [step 6](#) using the following command:

```
source <design>_dump_all_vcd_nodes.tcl
```
10. Run the simulation (for example, run 2000ns or run -all).
11. Quit the simulation using the `quit -sim` command, if required.
12. Exit the ModelSim software.
If you do not exit the software, the ModelSim software might end the writing process of the .vcd improperly, resulting in a corrupt .vcd.

Generating a .vcd from Full Post-Fit Netlist (Zero Delay) Simulation

To successfully generate a .vcd from the full post-fit Netlist (zero delay) simulation, follow these steps:

1. Compile your design in the Quartus II software to generate the Netlist `<project_name>.vo`.
2. In `<project_name>.vo`, search for the include statement for `<project_name>.sdo`, comment the statement out, and save the file.
Altera recommends that you use the Standard Delay Format Output File (.sdo) for gate-level timing simulation. The .sdo contains the delay information of each architecture primitive and routing element in your design; however, you must exclude the .sdo for zero delay simulation.
3. Generate a .vcd for power estimation by performing the steps in [Generating a .vcd](#) on page 8-17.

Related Information

- [Section I. Simulation](#)

PowerPlay Power Analyzer Compilation Report

The following table lists the items in the Compilation Report of the PowerPlay Power Analyzer section.

Section	Description
Summary	The Summary section of the report shows the estimated total thermal power consumption of your design. This includes dynamic, static, and I/O thermal power consumption. The I/O thermal power includes the total I/O power drawn from the V_{CCIO} and V_{CCPD} power supplies and the power drawn from V_{CCINT} in the I/O subsystem including I/O buffers and I/O registers. The report also includes a confidence metric that reflects the overall quality of the data sources for the signal activities. For example, a Low power estimation confidence value reflects that you have provided insufficient toggle rate data, or most of the signal-activity information used for power estimation is from default or vectorless estimation settings. For more information about the input data, refer to the PowerPlay Power Analyzer Confidence Metric report.
Settings	The Settings section of the report shows the PowerPlay Power Analyzer settings information of your design, including the default input toggle rates, operating conditions, and other relevant setting information.

Section	Description
Simulation Files Read	The Simulation Files Read section of the report lists the simulation output file that the .vcd used for power estimation. This section also includes the file ID, file type, entity, VCD start time, VCD end time, the unknown percentage, and the toggle percentage. The unknown percentage indicates the portion of the design module unused by the simulation vectors.
Operating Conditions Used	The Operating Conditions Used section of the report shows device characteristics, voltages, temperature, and cooling solution, if any, during the power estimation. This section also shows the entered junction temperature or auto-computed junction temperature during the power analysis.
Thermal Power Dissipated by Block	<p>The Thermal Power Dissipated by Block section of the report shows estimated thermal dynamic power and thermal static power consumption categorized by atoms. This information provides you with estimated power consumption for each atom in your design.</p> <p>By default, this section does not contain any data, but you can turn on the report with the Write power dissipation by block to report file option on the PowerPlay Power Analyzer Settings page.</p>
Thermal Power Dissipation by Block Type (Device Resource Type)	This Thermal Power Dissipation by Block Type (Device Resource Type) section of the report shows the estimated thermal dynamic power and thermal static power consumption categorized by block types. This information is further categorized by estimated dynamic and static power and provides an average toggle rate by block type. Thermal power is the power dissipated as heat from the FPGA device.
Thermal Power Dissipation by Hierarchy	This Thermal Power Dissipation by Hierarchy section of the report shows estimated thermal dynamic power and thermal static power consumption categorized by design hierarchy. This information is further categorized by the dynamic and static power that was used by the blocks and routing in that hierarchy. This information is useful when locating modules with high power consumption in your design.
Core Dynamic Thermal Power Dissipation by Clock Domain	The Core Dynamic Thermal Power Dissipation by Clock Domain section of the report shows the estimated total core dynamic power dissipation by each clock domain, which provides designs with estimated power consumption for each clock domain in the design. If the clock frequency for a domain is unspecified by a constraint, the clock frequency is listed as “unspecified.” For all the combinational logic, the clock domain is listed as no clock with zero MHz.

Section	Description
Current Drawn from Voltage Supplies	<p>The Current Drawn from Voltage Supplies section of the report lists the current drawn from each voltage supply. The V_{CCIO} and V_{CCPD} voltage supplies are further categorized by I/O bank and by voltage. This section also lists the minimum safe power supply size (current supply ability) for each supply voltage. Minimum current requirement can be higher than user mode current requirement in cases in which the supply has a specific power up current requirement that goes beyond user mode requirement, such as the V_{CCPD} power rail in Stratix III and Stratix IV devices, and the V_{CCIO} power rail in Stratix IV devices.</p> <p>The I/O thermal power dissipation on the summary page does not correlate directly to the power drawn from the V_{CCIO} and V_{CCPD} voltage supplies listed in this report. This is because the I/O thermal power dissipation value also includes portions of the V_{CCINT} power, such as the I/O element (IOE) registers, which are modeled as I/O power, but do not draw from the V_{CCIO} and V_{CCPD} supplies.</p> <p>The reported current drawn from the I/O Voltage Supplies (ICCIO and ICCPD) as reported in the PowerPlay Power Analyzer report includes any current drawn through the I/O into off-chip termination resistors. This can result in ICCIO and ICCPD values that are higher than the reported I/O thermal power, because this off-chip current dissipates as heat elsewhere and does not factor in the calculation of device temperature. Therefore, total I/O thermal power does not equal the sum of current drawn from each V_{CCIO} and V_{CCPD} supply multiplied by V_{CCIO} and V_{CCPD} voltage.</p>
Confidence Metric Details	<p>The Confidence Metric is defined in terms of the total weight of signal activity data sources for both combinational and registered signals. Each signal has two data sources allocated to it; a toggle rate source and a static probability source.</p> <p>The Confidence Metric Details section also indicates the quality of the signal toggle rate data to compute a power estimate. The confidence metric is low if the signal toggle rate data comes from poor predictors of real signal toggle rates in the device during an operation. Toggle rate data that comes from simulation, user-entered assignments on specific signals or entities are reliable. Toggle rate data from default toggle rates (for example, 12.5% of the clock period) or vectorless estimation are relatively inaccurate. This section gives an overall confidence rating in the toggle rate data, from low to high. This section also summarizes how many pins, registers, and combinational nodes obtained their toggle rates from each of simulation, user entry, vectorless estimation, or default toggle rate estimations. This detailed information helps you understand how to increase the confidence metric, letting you determine your own confidence in the toggle rate data.</p>

Section	Description
Signal Activities	<p>The Signal Activities section lists toggle rates and static probabilities assumed by power analysis for all signals with fan-out and pins. This section also lists the signal type (pin, registered, or combinational) and the data source for the toggle rate and static probability. By default, this section does not contain any data, but you can turn on the report with the Write signal activities to report file option on the PowerPlay Power Analyzer Settings page.</p> <p>Altera recommends that you keep the Write signal activities to report file option turned off for a large design because of the large number of signals present. You can use the Assignment Editor to specify that activities for individual nodes or entities are reported by assigning an on value to those nodes for the Power Report Signal Activities assignment.</p>
Messages	The Messages section lists the messages that the Quartus II software generates during the analysis.

Scripting Support

You can run procedures and create settings described in this chapter in a Tcl script. You can also run some procedures at a command prompt. For more information about scripting command options, refer to the Quartus II Command-Line and Tcl API Help browser. To run the Help browser, type the following command at the command prompt:

```
quartus_sh --qhelp
```

Related Information

- [Tcl Scripting](#)
- [API Functions for Tcl](#)
- [Quartus II Settings File Reference Manual](#)
- [Command-Line Scripting](#)

Running the PowerPlay Power Analyzer from the Command-Line

The executable to run the PowerPlay Power Analyzer is `quartus_pow`. For a complete listing of all command-line options supported by `quartus_pow`, type the following command at a system command prompt:

```
quartus_pow --help
```

or-

```
quartus_sh --qhelp
```

The following lists the examples of using the `quartus_pow` executable. Type the command listed in the following section at a system command prompt. These examples assume that operations are performed on Quartus II project called *sample*.

To instruct the PowerPlay Power Analyzer to generate a PowerPlay EPE File:

```
quartus_pow sample --output_epe=sample.csv
```

To instruct the PowerPlay Power Analyzer to generate a PowerPlay EPE File without performing the power estimate:

```
quartus_pow sample --output_epe=sample.csv --estimate_power=off
```

To instruct the PowerPlay Power Analyzer to use a .vcd as input (sample.vcd):

```
quartus_pow sample --input_vcd=sample.vcd
```

To instruct the PowerPlay Power Analyzer to use two .vcd files as input files (sample1.vcd and sample2.vcd), perform glitch filtering on the .vcd and use a default input I/O toggle rate of 10,000 transitions per second:

```
quartus_pow sample --input_vcd=sample1.vcd --input_vcd=sample2.vcd \  
--vcd_filter_glitches=on --\  
default_input_io_toggle_rate=10000transitions/s
```

To instruct the PowerPlay Power Analyzer to not use an input file, a default input I/O toggle rate of 60%, no vectorless estimation, and a default toggle rate of 20% on all remaining signals:

```
quartus_pow sample --no_input_file --default_input_io_toggle_rate=60% \  
\   
--use_vectorless_estimation=off --default_toggle_rate=20%
```

Note: No command-line options are available to specify the information found on the **PowerPlay Power Analyzer Settings Operating Conditions** page. Use the Quartus II GUI to specify these options.

The `quartus_pow` executable creates a report file, `<revision name>.pow.rpt`. You can locate the report file in the main project directory. The report file contains the same information in [PowerPlay Power Analyzer Compilation Report](#) on page 8-19.

Document Revision History

The following table lists the revision history for this chapter.

Date	Version	Changes
November 2013	13.1.0	<ul style="list-style-type: none"> Updated Multiple Simulations on the Same Entity on page 8-11 to indicate the usage of a .vcd that only includes signals from the instance that you care about if you do not want the PowerPlay Power Analyzer to read information from multiple instances and take an arithmetic average of the signal activities. Updated Differences between the PowerPlay EPE and the Quartus II PowerPlay Power Analyzer on page 8-2 to include a note. Removed “Avoiding Power Estimation and Hardware Measurement Mismatch” section. Updated Figure 8-1. Updated Types of Power Analyses on page 8-2 to remove the note about HPS Power Calculator. Removed "Creating PowerPlay EPE Spreadsheet", "PowerPlay EPE File Generator Compilation Report", "Using the HPS Power Calculator", and "Running the PowerPlay Power Analyzer Compilation Report" sections. Updated PowerPlay Power Analysis on page 8-1 to include the accuracy of the PowerPlay Power Analyzer and PowerPlay EPE spreadsheet. Updated Device Resource Usage on page 8-5. Added Differences between the PowerPlay EPE and the Quartus II PowerPlay Power Analyzer on page 8-2. Added a note in the Operating Settings and Conditions on page 8-7 to notify that the Quartus II Fitter may override some of the supply voltages settings specified in this section. Updated Using Simulation Files in Modular Design Flows on page 8-9. Updated Generating a .vcd on page 8-17. Updated PowerPlay Power Analyzer Compilation Report on page 8-19 Updated Multiple Simulations on the Same Entity on page 8-11, Glitch Filtering on page 8-13, Node and Entity Assignments on page 8-14 and Node Name Matching Considerations on page 8-13 for editorial edits. Added RTL Simulation Limitation on page 8-17. Added Specifying Start and End Time when Performing Signal-Activity Calculations using the Limit VCD Period Option on page 8-12. Updated Partial Simulations on page 8-12.
November 2012	12.1.0	<ul style="list-style-type: none"> Updated “Types of Power Analyses” on page 8–2, and “Confidence Metric Details” on page 8–23. Added “Importance of .vcd” on page 8–20, and “Avoiding Power Estimation and Hardware Measurement Mismatch” on page 8–24

Date	Version	Changes
June 2012	12.0.0	<ul style="list-style-type: none"> Updated “Current Drawn from Voltage Supplies” on page 8–22. Added “Using the HPS Power Calculator” on page 8–7.
November 2011	10.1.1	<ul style="list-style-type: none"> Template update. Minor editorial updates.
December 2010	10.1.0	<ul style="list-style-type: none"> Added links to Quartus II Help, removed redundant material. Moved “Creating PowerPlay EPE Spreadsheets” to page 8–6. Minor edits.
July 2010	10.0.0	<ul style="list-style-type: none"> Removed references to the Quartus II Simulator. Updated Table 8–1 on page 8–6, Table 8–2 on page 8–13, and Table 8–3 on page 8–14. Updated Figure 8–3 on page 8–9, Figure 8–4 on page 8–10, and Figure 8–5 on page 8–12.
November 2009	9.1.0	<ul style="list-style-type: none"> Updated “Creating PowerPlay EPE Spreadsheets” on page 8–6 and “Simulation Results” on page 8–10. Added “Signal Activities from Full Post-Fit Netlist (Zero Delay) Simulation” on page 8–19 and “Generating a .vcd from Full Post-Fit Netlist (Zero Delay) Simulation” on page 8–21. Minor changes to “Generating a .vcd from ModelSim Software” on page 8–21. Updated Figure 11–8 on page 11–24.
March 2009	9.0.0	<ul style="list-style-type: none"> This chapter was chapter 11 in version 8.1. Removed Figures 11-10, 11-11, 11-13, 11-14, and 11-17 from 8.1 version.
November 2008	8.1.0	<ul style="list-style-type: none"> Updated for the Quartus II software version 8.1. Replaced Figure 11-3. Replaced Figure 11-14.
May 2008	8.0.0	<ul style="list-style-type: none"> Updated Figure 11–5. Updated “Types of Power Analyses” on page 11–5. Updated “Operating Conditions” on page 11–9. Updated “PowerPlay Power Analyzer Compilation Report” on page 11–31. Updated “Current Drawn from Voltage Supplies” on page 11–32.