SmartPower User Guide

Libero SoC v2025.1

Introduction (Ask a Question)

SmartPower is a power-analysis tool that allows you to visualize power consumption and potential power consumption problems globally and in-depth within your design, so you can make adjustments—when possible—to reduce power.

SmartPower provides a detailed and accurate way to analyze designs for Microchip SoC FPGAs, from top-level summaries to deep-down specific functions within the design, such as gates, nets, I/Os, memories, clock domains, blocks, and power supply rails.

You can analyze the hierarchy of block instances and specific instances within a hierarchy, and each can be organized in various ways to show the respective power consumption of the component pieces.

SmartPower also analyzes power by functional modes, such as Active, Flash*Freeze, Shutdown, Sleep, or Static, depending on the specific FPGA family used. You can also create custom modes that may have been created in the design. Custom modes can be used for testing "what if" potential operating modes.

SmartPower has a unique feature that allows you to create test scenario profiles. A profile enables you to create sets of operational modes, so you can understand the average power consumed by this combination of functional modes. An example may be a combination of Active, Sleep, and Static modes—as would be used over time in an actual application.

SmartPower generates detailed hierarchical reports of the power consumption of a design for easy evaluation. This enables you to locate the power consumption source and take appropriate action to reduce the power, if possible.

SmartPower supports use of files in the Value-Change Dump (VCD) format, as specified in the IEEE® 1364 standard, generated by the simulation runs. The support for this format lets you generate switching activity information from ModelSim® or other simulators, and then use the switching activity-over-time results to evaluate average and peak power consumption for your design.

SmartPower supports SmartFusion® 2, IGLOO® 2, RTG4™, PolarFire®, and PolarFire SoC devices.

Table of Contents

1. Starting SmartPower (Ask a Question)

Set the target clock and data frequencies before you evaluate your power consumption.

1. Open the Design Flow window.
2. In the Design Flow window, right-click **Verify Power** and choose **Open Interactively**.

The SmartPower analysis tool window appears.

Figure 1-1. SmartPower User Interface (SmartFusion® 2, IGLOO® 2, and RTG4™)

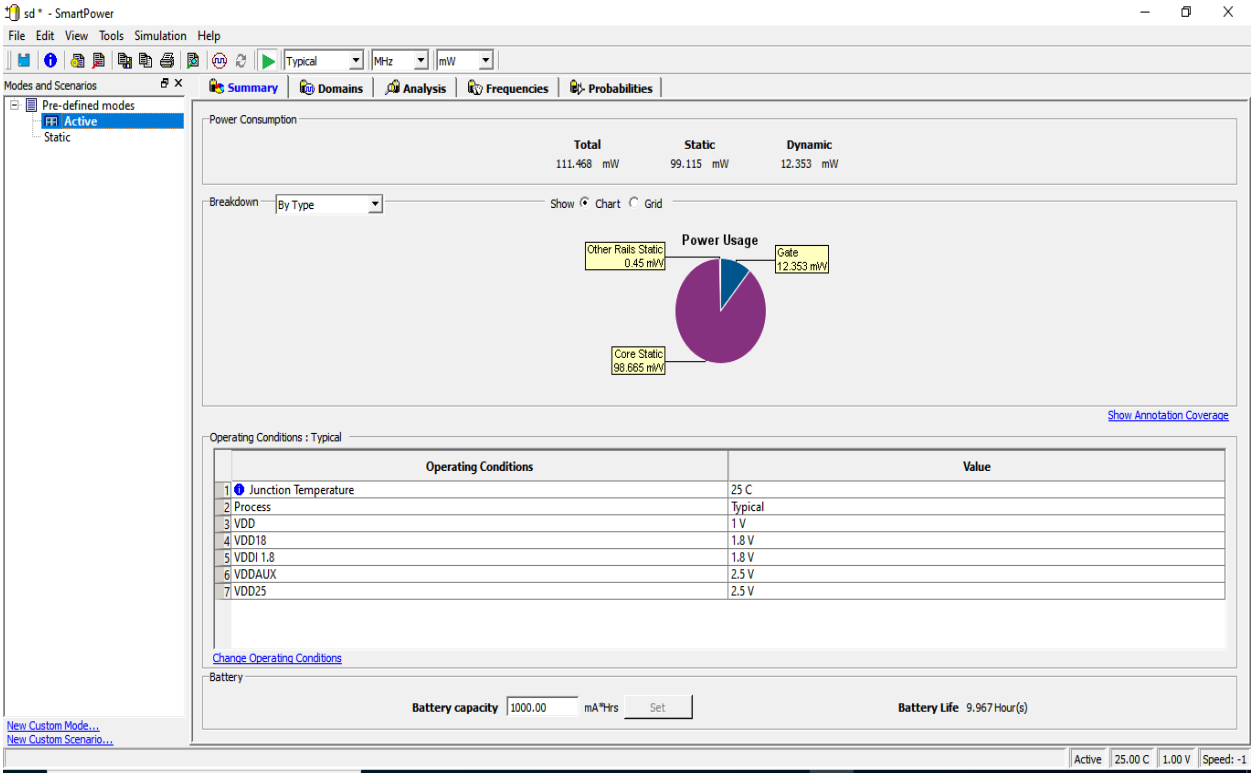
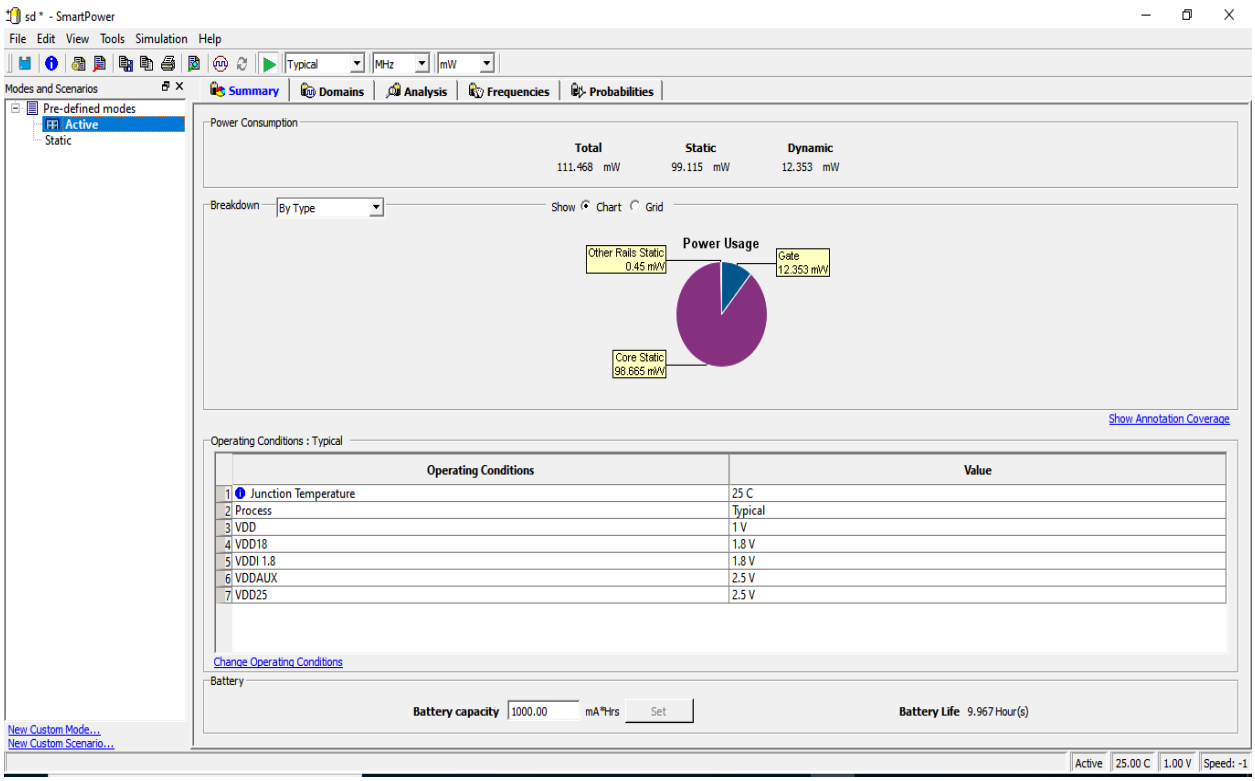


Figure 1-2. SmartPower User Interface (PolarFire®)



2. SmartPower Interface (Ask a Question)

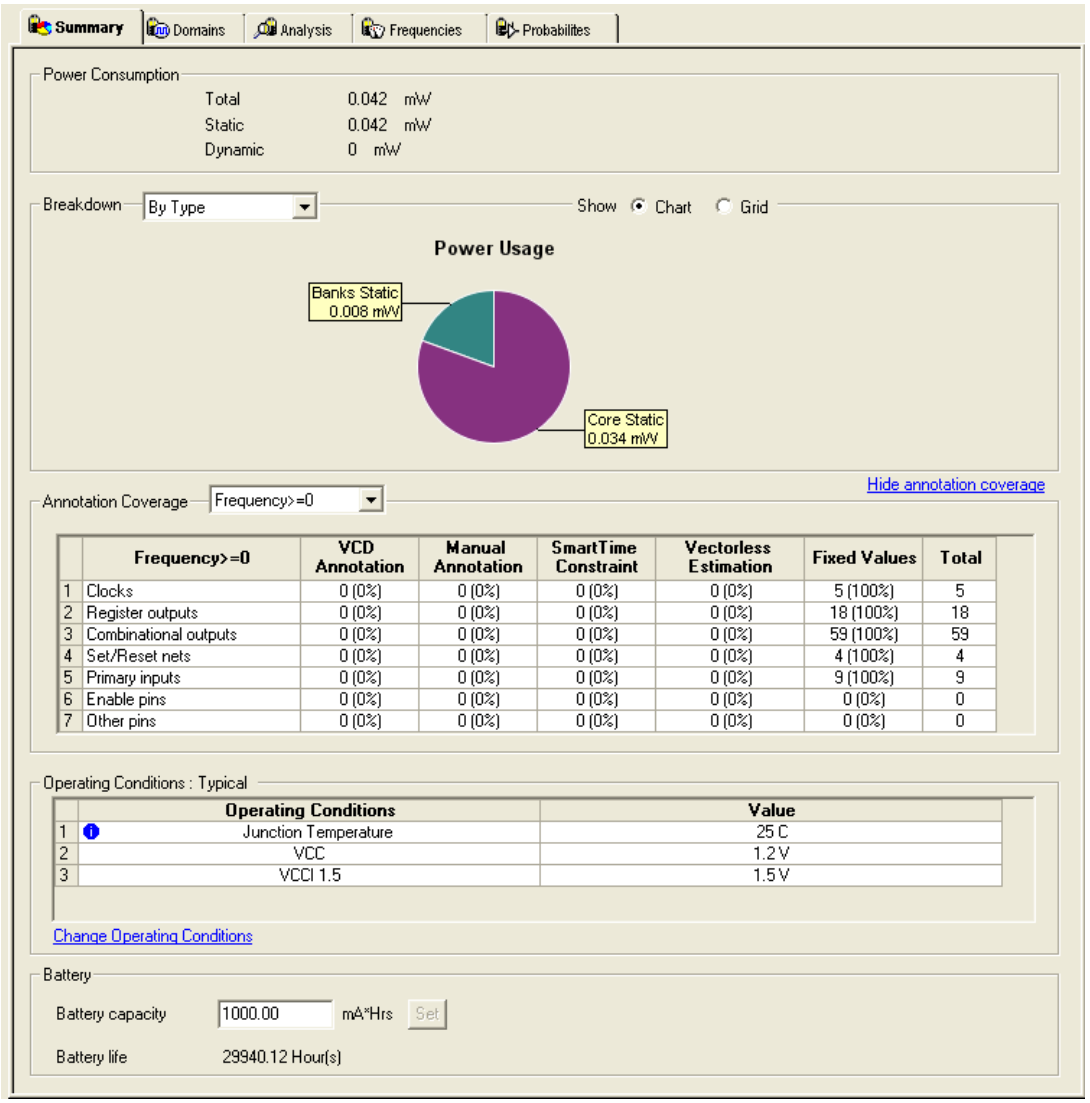
The following sections describe the SmartPower tabs, toolbars, and menu commands.

2.1. Summary Tab (Ask a Question)

The **Summary** tab is organized into the following sections:

- **Power Consumption**
- **Breakdown**
- **Annotation Coverage**
- **Operating Conditions**
- **Battery**

Figure 2-1. SmartPower Summary Tab



2.1.1. Power Consumption (Ask a Question)

Displays the total power consumption and the static and dynamic power of the design. The information displayed is accurate after you enter your target clock and data frequencies.

2.1.2. Breakdown (Ask a Question)

Displays a breakdown of power consumption in the design. A drop-down list allows you to view the power breakdown by type of component, voltage rail, or clock domain.

2.1.2.1. Show (Ask a Question)

Displays power usage as a chart or as a grid.

When you select **By Type** from the drop-down list, the **Summary** tab displays a breakdown of the design's power usage by the following types:

- Net
- Gate
- I/O
- Memory
- Core Static
- Banks Static
- Analog (for Fusion Designs)

Any static power used by instances of a component in a design will be included in the appropriate category. As a result, the sum of the power in breakdown by type will be equal to the total static and dynamic power consumption of the design. Percentages displayed are fractions of the total power. Global nets are included in the net power types. To obtain the power of a clock, select breakdown by clock domain.

When you select **By Rail** from the drop-down list, the **Summary** tab displays a breakdown of the power usage by the rail that the power is drawn from and a breakdown for the current usage. The power, voltage, and current drawn for each rail are displayed.

When you select **By Clock Domain** from the drop-down list, the **Summary** tab displays a breakdown of the power usage by the clock domain that the power is drawn from. For each clock domain, the power and percentage of power drawn are displayed. Dynamic power is accounted for in breakdown by clock, but static power is not associated with any clock.

By Mode is available only when a scenario is selected. When you select **By Mode** in the Modes and Scenarios toolbar, the **Summary** tab shows power consumption, duration, and power duration weight by mode.

After you select **By Type**, **By Rail**, **By Mode**, or **By Clock Domain**, you can customize the columns in the grid by right-clicking the column headers. To exclude a column, remove its check mark from the right-click menu. Right-click a column header to set the column default size for the selected column or for all columns, and then sort the values in ascending order, descending order, or customize the columns.

To sort the contents of a column, double-click a column header.

To export the contents of the grid to a file, from the **File** menu, choose **Export Grid** or click its associated toolbar button.

To print the grid, from the **File** menu, choose **Print Grid** or click its associated toolbar button. To copy the grid, from the **Edit** menu, choose **Copy Grid** or click its toolbar button.

To copy the chart, click the contents of the chart, and from the right-click menu, choose **Copy Chart to Clipboard**.

To export the chart, click the contents of the chart, and from the right-click menu, choose **Export Chart to File**.

To rotate the chart by 90 degrees, click the contents of the chart, and from the right-click menu, choose **Rotate Chart Clockwise** or **Rotate Chart Counter-Clockwise**.

2.1.3. Annotation Coverage (Ask a Question)

Displays the number and percentage of pins annotated by each source (VCD, manual annotation, SmartTime constraint, vectorless estimation, and fixed values) for all clocks, register outputs, combinational outputs, set/reset nets, primary inputs, enable pins, and other pins.

- **Hide annotation coverage** link: hides the annotation coverage.
- **Show annotation coverage** link: shows the annotation coverage.
- **Annotation Coverage Summary** drop-down menu: shows probability or frequency annotation statistics (frequency=0 shows statistics for pins that have a frequency of 0).
- **VCD**: shows the number/percentage of pins with frequencies/probabilities imported from a VCD file.
- **Manual annotation**: shows the percentage of pins with manually annotated frequencies/ probabilities.
- **SmartTime constraint**: shows the percentage of pins initialized with SmartTime.
- **Fixed values**: shows the percentage of pins with default frequency/probability.
- **Vectorless estimation**: shows the percentage of pins that have been annotated with vectorless estimation.

2.1.4. Operating Conditions (Ask a Question)

The **Operating Conditions** group displays the temperature and voltage operating conditions for the selected mode or scenario specified in Operating Conditions Dialog. When you select a scenario, the temperature and voltage operating conditions for each mode in the scenario are displayed.

To change the operating conditions, click the **Change Operating Conditions** link or click **Tools** from the menu,and choose **Operating Conditions**.

The VPP voltage operating range is available only for SmartFusion 2, IGLOO 2, and RTG4 device families. You can choose to change the voltage operating range for the VPP at any point from 2.5V to 3.3V or 3.3V to 2.5V.

After you change the voltage operating range, you must restart the SmartPower tool to reflect the changed voltage operating range.

To change the voltage operating range for VPP, perform the following steps:

1. In the **Design Flow** window, under the **Verify Post Layout Implementation** group, right-click **Verify Power** and choose **Configure Options**. The **Verify Power Configuration** dialog appears.
2. Choose the required voltage operating range and click **OK**. The selected voltage operating range is set as the current range.

Important: You must restart the SmartPower tool to apply the new voltage operating range.

2.1.4.1. Battery (Ask a Question)

Allows you to set the battery capacity in mA/hr and reports the battery life.

2.2. Domains Tab (Ask a Question)

The **Domains** tab lists existing domains along with their corresponding clock and data frequencies and probability data for clock domains and sets of pins. In this tab, you can add, edit, or remove domains or sets of pins, and change the clock and/or data frequency for a selected domain.

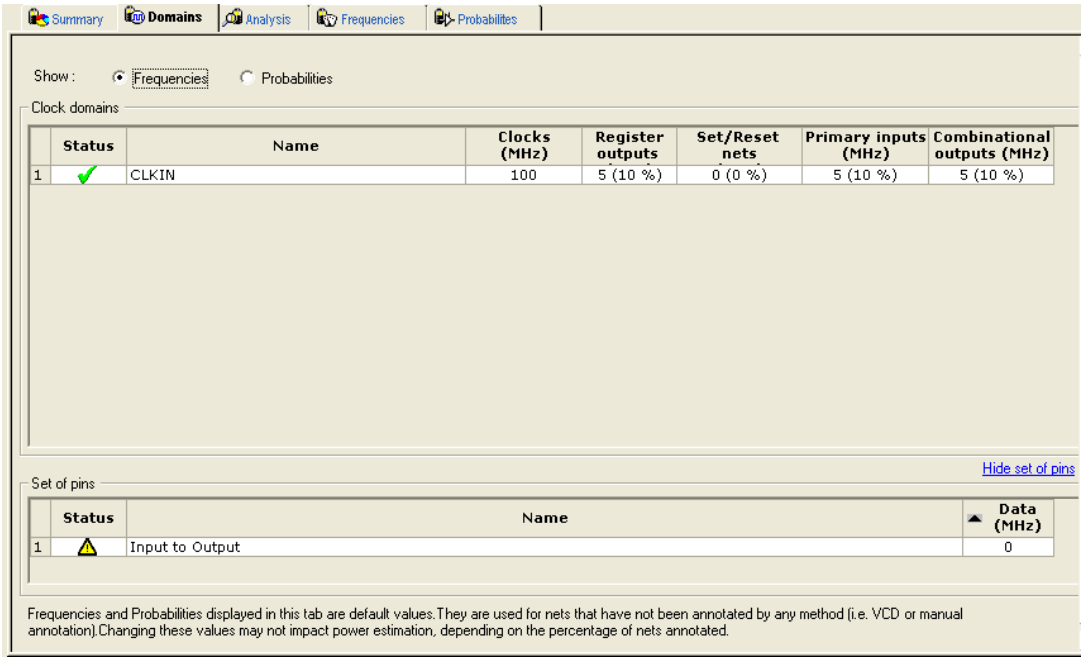
When you start SmartPower for the first time, it automatically initializes frequencies from SmartTime with a default data toggle rate of 10%. However, if you update your timing constraints in SmartTime, you can use the Initialize Frequencies and Probabilities dialog box to update

SmartPower with new frequencies from SmartTime. If you do not have timing constraints in SmartTime, you can initialize clock and data frequencies manually by directly editing the **Domains** tab grid or using the Initialize Frequencies and Probabilities dialog box.

If you import a VCD file, the **Domains** tab shows the average clock and data frequencies for each clock domain and input set. This average is calculated using VCD information.

Note: If you change a clock or data frequency in the **Domains** tab, it impacts only nets that use the default estimation source. Nets use the default estimation source when they have not been manually set to a specific frequency in the **Frequencies** tab, and when they have not been set by importing a VCD file. To see explicitly the list of nets using the default estimation source, refer to the Frequencies tab.

Figure 2-2. SmartPower Domains Tab



Note: Data and clock frequencies are set to zero automatically when in non-active modes.

To hide or display the grid for set of pins, click the **Hide/Show set of pins link**. You can edit frequency or probability data directly from the grid.

To create a new clock domain or set of pins:

1. Right-click the clock domain or set of pins and choose **create new clock domain** or **create new set of pins**.
2. Enter a name and click **OK**. The new clock domain or set of pins appears in the **Domains** tab.
3. Modify the frequency and probability data as needed.

To initialize, clock domain frequencies, right-click the clock domain name and select **Initialize Frequencies and Probabilities**. If there is no frequency constraint set for the clock domain in SmartTime, the clock frequency of the domain is not changed.

The status of a clock frequency or probability is displayed in the first column by an icon. Mouse over the icon to identify the status (see the following table).

Table 2-1. Clock Frequency/Probability Status

Icon	Description
⚠	This clock has not been initialized.
⚠	No constraints available for this clock from SmartTime.
✓	Average frequencies/probabilities imported from VCD.
✓	Clock constraints imported from SmartTime.
✓	This clock has been initialized manually.

Note: The frequency and probability distribution feature is available only in active operating mode or when in a mode based on the Active mode.

An icon in the first column shows the status of a set of pins. Mouse over the icon to identify the status (see the following table).

Table 2-2. Set of Pins Status

Icon	Description
⚠	This set has not been initialized.
✓	Average frequencies/probabilities imported from VCD.
✓	This set has been manually initialized

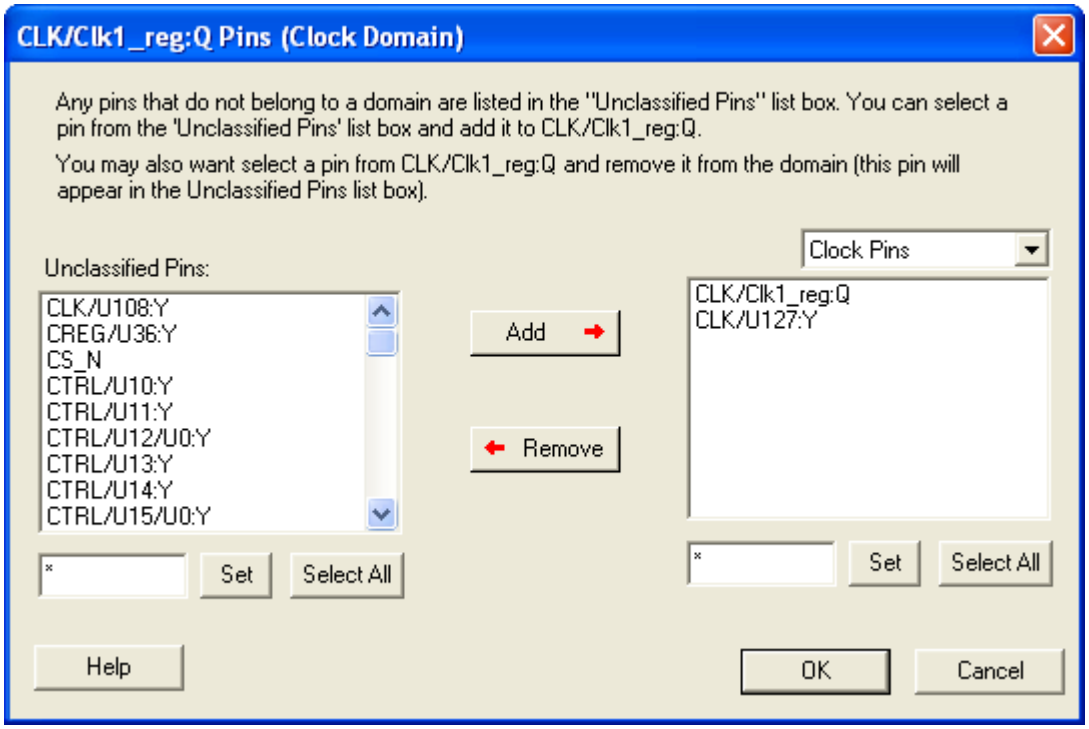
Note: The frequency and probability distribution feature is available only in active operating mode or when in a mode based on the Active mode.

To remove one or multiple clock domains, select the clock domain(s) and click **Remove Selected Domains(s)**.

To edit a clock domain or a set of pins:

1. Select the clock domain and from the right-click menu, and choose **Edit Domain**. The Clock Domain or Set-of- Pins window appears.

Figure 2-3. Clock Domain Window



Any pins that do not belong to a domain appear under **Unclassified Pins**. You can select a pin from this list and add it to the current domain. You can also select a pin from the current domain and remove it from the domain (this pin will appear under **Unclassified Pins**).

Clock Pins and Data Pins: Select **Clock Pins** from the pull-down list to display all clock pins for the selected domain, or select **Data Pins** to display registers, asynchronous signals, primary input, and combinational signals for the selected domain.

Use the **filter** boxes to narrow your search for a specific pin. The boxes are text filters; * is a wildcard.

2. Click **OK**.

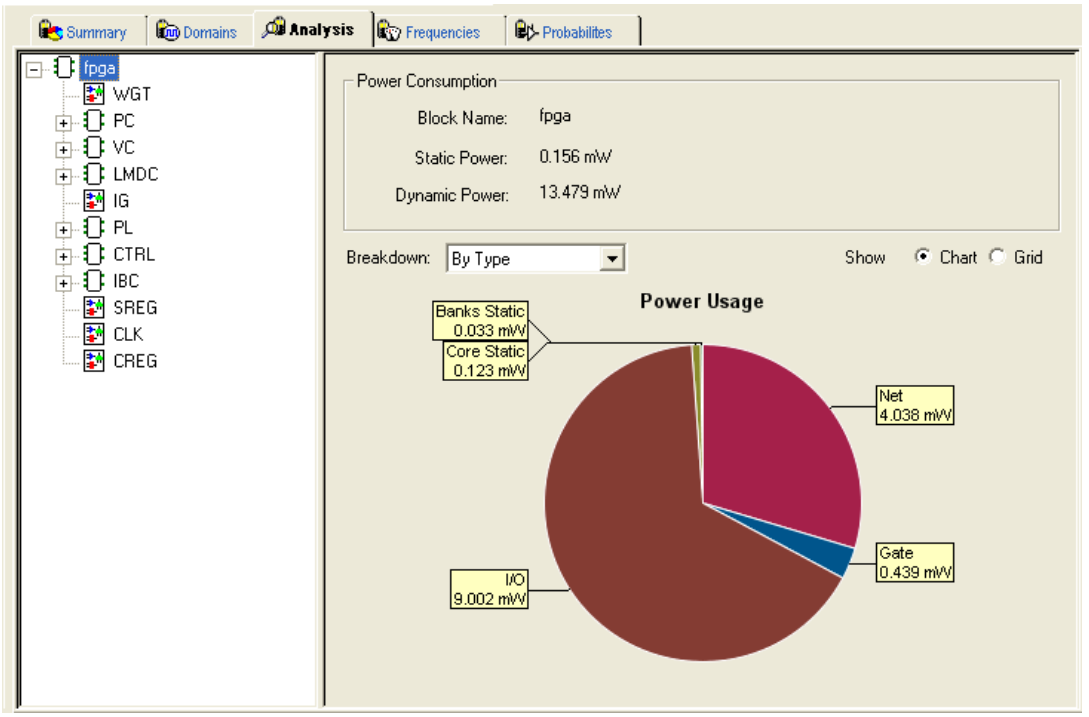
To delete a clock domain or a set-of-pins, select the clock domain or set-of-pins and choose **Remove Domain** from the right-click menu.

2.3. Analysis Tab (Ask a Question)

The **Analysis** tab allows you to inspect detailed hierarchical reports of the power consumption. The **Analysis** tab consists of two windows:

- Hierarchy of Instances window
- Analysis window

Figure 2-4. SmartPower Analysis Tab



2.3.1. Hierarchy of Instances Window (Ask a Question)

SmartPower lists the hierarchy of instances in the Hierarchy of Instances window. Clicking the plus sign (+) next to a block expands the block to show its sub-blocks. Only hierarchical blocks are displayed in this list (no gates or nets).

When you select a block of the hierarchical tree, SmartPower displays its name and power consumption in the Analysis window.

2.3.2. Analysis Window (Ask a Question)

The Analysis window is organized into two sections:

- **Power Consumption**
- **Breakdown**

2.3.2.1. Power Consumption (Ask a Question)

Displays the name of the currently selected block from the hierarchy of instances window, as well as the static and dynamic power consumption of the selected block.

2.3.2.2. Breakdown (Ask a Question)

Displays a breakdown of power usage of the currently selected block. Use the drop-down list to view the power breakdown by type of component (instance, or clock domain).

When you select **By Type** from the drop-down list, the **Analysis** tab shows a breakdown of the selected block's power usage by the following types:

- Net
- Gate
- I/O
- Memory
- Core Static

- Banks Static
- Analog

Any static power used by instances of a component in a design are included in the appropriate category.

The sum of the power in breakdown by type equals the total static and dynamic power consumption of the design. Percentages displayed are fractions of the total power. Global nets are included in the net power types. To obtain the power of a clock, select breakdown by clock domain.

When you select **By Clock Domain** from the drop-down list, the **Analysis** tab shows a breakdown of the power usage by the clock domain that the power is drawn from. For each clock domain, the power and percentage of power drawn are displayed. Dynamic power is accounted for in breakdown by clock, but static power is not associated by any clock.

When you select **By Type**, **By Instance** or **By Clock Domain**, you can customize which columns are displayed in the grid by right-clicking the column headers. To exclude a column, remove its check mark from the right-click menu. Right-click a column header to set the column default size for the selected column or for all columns, and then sort the values in ascending order, descending order, or customize the columns.

To sort the contents of a column, double-click a column header.

To export the contents of the grid to a file, choose **Export Grid** from the **File** menu, or click its associated toolbar button.

To print the grid, choose **Print Grid** from the **File** menu, or click its associated toolbar button. To copy the grid, choose **Copy Grid** from the **Edit** menu or click its toolbar button.

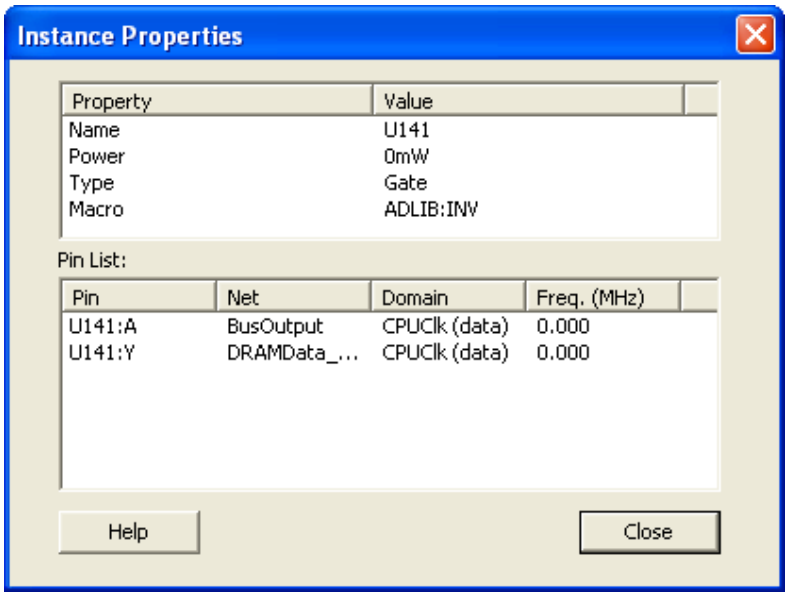
To copy the chart, right-click the contents of the chart and choose **Copy Chart to Clipboard**.

To export the chart, right-click the contents of the chart and choose **Export Chart to File**.

To rotate the chart by 90 degrees, click the contents of the chart and choose **Rotate Chart Clockwise** or right-click **Rotate Chart Counter-Clockwise**.

To view an instance's properties, right-click any instance and choose **Instance Properties**. The Instance Properties window appears, with the instance name, type and power and other properties specific to each type of instance (see the following figure).

Figure 2-5. Instance Properties: Gate



The Instance Properties window allows you to access a list of pins of a gate, and check their domain and frequency.

For more information, see:

- Analyzing Results
- Advanced Analysis of I/Os

2.3.2.2.1. Show (Ask a Question)

Displays power usage as a chart or grid.

Note: The power usage chart is not available when **breakdown by instance** is selected.

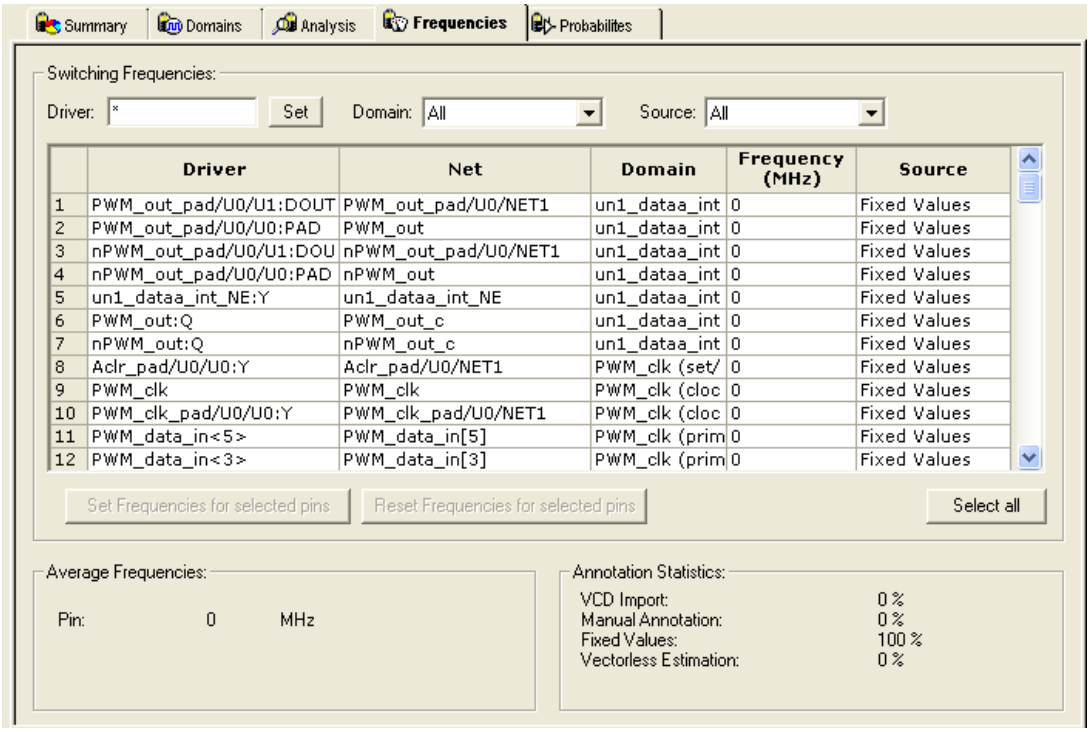
2.4. Frequencies Tab (Ask a Question)

Use the **Frequencies** tab to attach switching frequency attributes to the interconnects of the design.

The **Frequencies** tab is organized into the following areas:

- **Switching Frequencies**
- **Average Frequencies**
- **Annotation Statistics**

Figure 2-6. Frequencies Tab



	Driver	Net	Domain	Frequency (MHz)	Source
1	PWM_out_pad/U0/U1:DOUT	PWM_out_pad/U0/NET1	un1_dataa_int	0	Fixed Values
2	PWM_out_pad/U0/U0:PAD	PWM_out	un1_dataa_int	0	Fixed Values
3	nPWM_out_pad/U0/U1:DOU	nPWM_out_pad/U0/NET1	un1_dataa_int	0	Fixed Values
4	nPWM_out_pad/U0/U0:PAD	nPWM_out	un1_dataa_int	0	Fixed Values
5	un1_dataa_int_NE:Y	un1_dataa_int_NE	un1_dataa_int	0	Fixed Values
6	PWM_out:Q	PWM_out_c	un1_dataa_int	0	Fixed Values
7	nPWM_out:Q	nPWM_out_c	un1_dataa_int	0	Fixed Values
8	Aclr_pad/U0/U0:Y	Aclr_pad/U0/NET1	PWM_clk (set/	0	Fixed Values
9	PWM_clk	PWM_clk	PWM_clk (cloc	0	Fixed Values
10	PWM_clk_pad/U0/U0:Y	PWM_clk_pad/U0/NET1	PWM_clk (cloc	0	Fixed Values
11	PWM_data_in<5>	PWM_data_in[5]	PWM_clk (prim	0	Fixed Values
12	PWM_data_in<3>	PWM_data_in[3]	PWM_clk (prim	0	Fixed Values

2.4.1. Switching Frequencies (Ask a Question)

Displays the pins and lists the net, domain, frequency, and frequency source for each pin.

The **Driver** box allows you to filter the list of pins based on the criteria entered. Enter text in the filter box and click **Set** to apply this text as a filter. Use the ***** character to display all pins or as a wildcard.

Use the **Domain** drop-down list to select a different domain (All, CLKIN, Default, or Input to Output).

Use the **Source** drop-down list to select a different source (All, Default, Manual Annotation, or VCD Import).

To change the frequency, enter a new value in the **Frequency** column, or select multiple pins, click the **Set frequencies for selected pins** button, and enter a new frequency. Use **Reset frequencies for selected pins** to revert to the default frequency.

To select all the pins, click **Select All**.

2.4.2. Average Frequencies (Ask a Question)

Displays the average frequency of all pins displayed in the grid.

Average frequencies are useful when you import a VCD file. Because these files allow you to specify the frequency of each pin individually, it is useful to know the average clock pin or data pin frequency for a particular clock domain.

If you did not specify a frequency annotation for any clock pin in this clock domain, the average value is equal to the default clock frequency of the clock domain. If you set one or several clock pins, SmartPower takes those annotations to compute an average value.

If you did not specify a frequency annotation for any data pin in this clock domain, the average value is equal to the default data frequency of the clock domain. If you set one or several data pins, SmartPower considers these annotations into account to compute an average value.

2.4.3. Annotation Statistics (Ask a Question)

VCD import: displays the percentage of pins with frequencies imported from the *.vcd file.

Manual annotation: displays the percentage of pins with manually-annotated frequencies.

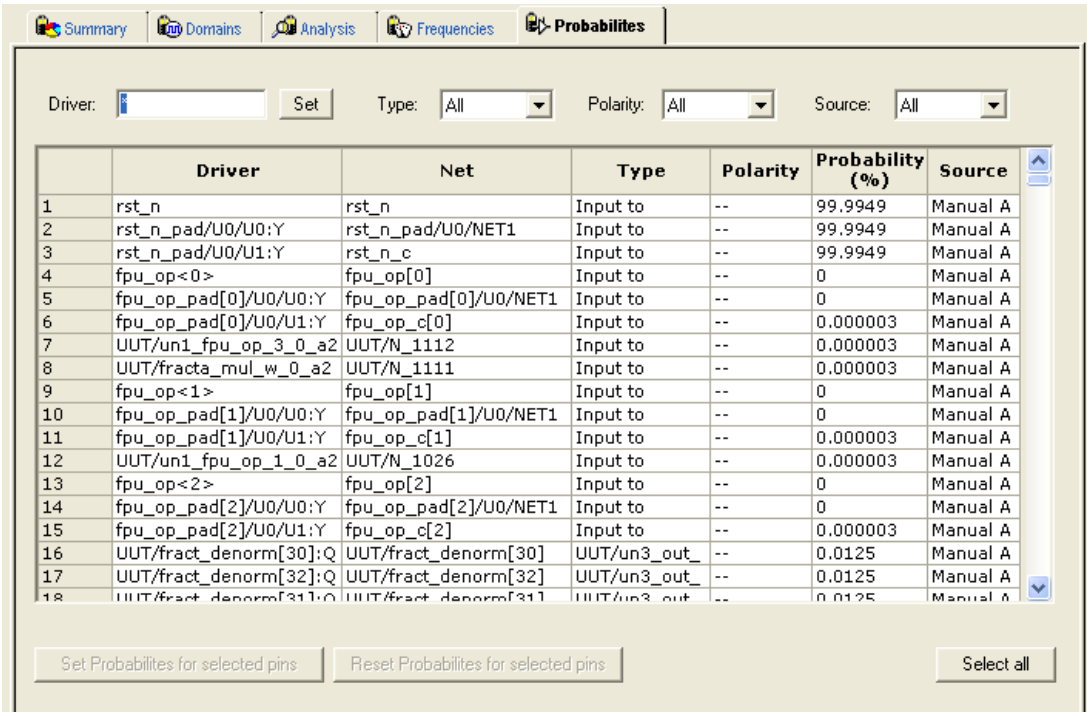
Fixed values: displays the percentage of pins with default frequency.

2.5. Probabilities Tab (Ask a Question)

The **Probabilities** tab displays the drivers, nets, types, polarity, rates, and source for each pin. Use the **Probabilities** tab to control the probabilities for all pins.

- **Bidirectional and Tristate I/Os**: controls the output probability of each tristate and bidirectional I/O. The probability is the percentage of time that the I/O is used as an output. You can change the default value and set a specific value for each bidirectional or tristate I/O.
- **Memories**: controls the read cycle probability or the write cycle probability of a memory block. The probability is the percentage of time that a memory block will be used in a read cycle or a write cycle. You can change the default value and set a specific value for each enable pin of each memory block available in the design. If an enable pin is active low, assign it a probability of 0% to completely enable the I/O or the memory block, or 100% to disable it completely.

Figure 2-7. SmartPower Probabilities Tab



	Driver	Net	Type	Polarity	Probability (%)	Source
1	rst_n	rst_n	Input to	--	99.9949	Manual A
2	rst_n_pad/U0/U0:Y	rst_n_pad/U0/NET1	Input to	--	99.9949	Manual A
3	rst_n_pad/U0/U1:Y	rst_n_c	Input to	--	99.9949	Manual A
4	fpu_op<0>	fpu_op[0]	Input to	--	0	Manual A
5	fpu_op_pad[0]/U0/U0:Y	fpu_op_pad[0]/U0/NET1	Input to	--	0	Manual A
6	fpu_op_pad[0]/U0/U1:Y	fpu_op_c[0]	Input to	--	0.000003	Manual A
7	UUT/un1_fpu_op_3_0_a2	UUT/N_1112	Input to	--	0.000003	Manual A
8	UUT/fracta_mul_w_0_a2	UUT/N_1111	Input to	--	0.000003	Manual A
9	fpu_op<1>	fpu_op[1]	Input to	--	0	Manual A
10	fpu_op_pad[1]/U0/U0:Y	fpu_op_pad[1]/U0/NET1	Input to	--	0	Manual A
11	fpu_op_pad[1]/U0/U1:Y	fpu_op_c[1]	Input to	--	0.000003	Manual A
12	UUT/un1_fpu_op_1_0_a2	UUT/N_1026	Input to	--	0.000003	Manual A
13	fpu_op<2>	fpu_op[2]	Input to	--	0	Manual A
14	fpu_op_pad[2]/U0/U0:Y	fpu_op_pad[2]/U0/NET1	Input to	--	0	Manual A
15	fpu_op_pad[2]/U0/U1:Y	fpu_op_c[2]	Input to	--	0.000003	Manual A
16	UUT/fract_denorm[30]:Q	UUT/fract_denorm[30]	UUT/un3_out_	--	0.0125	Manual A
17	UUT/fract_denorm[32]:Q	UUT/fract_denorm[32]	UUT/un3_out_	--	0.0125	Manual A
18	UUT/fract_denorm[31]:Q	UUT/fract_denorm[31]	UUT/un3_out_	--	0.0125	Manual A

In designs with a cascaded memory, all memory enable pins are shown in the **Probabilities** tab. To simulate real-world power consumption, selectively enable only some memory enable pins, as specified in the memory cascading structure.

- For purely vertical cascading, enable the pins that belong to one RAM block.
- For purely horizontal cascading, enable all pins belonging to all RAM blocks of the cascaded memory.

To find which RAM block corresponds to which pin, cross-probe with the MVN NetlistViewer.

The **Driver** box allows you to filter the list of drivers based on the criteria entered. Enter text in the filter box, and click **Set** to apply this text as a filter. Use the ***** character as a wildcard.

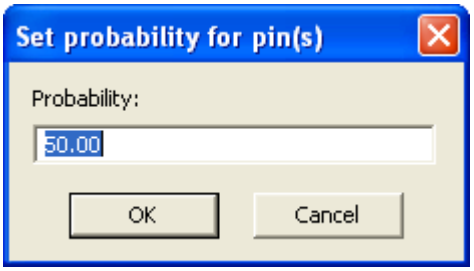
Use the **Type** drop-down list to filter the pin list by different type (All, All enable, CLKIN, Default, or Input to Output).

Use the **Polarity** drop-down list to filter the pin list by polarity (Low, High, or All).

Use the **Source** drop-down list to filter the pin list by source (All, Default Estimation, Manual Annotation, SDC, VCD Import).

To change the rate, enter a new value in the **Rate** column, or select multiple pins, click the **Set Probabilities for selected pins** button, and enter a new frequency in the Set Default Probabilities dialog box.

Figure 2-8. Set Probability for Pins Dialog Box



Use **Reset Probabilities for selected pins** to revert to the default polarity. You can change the default probability from the **Domains** tab.

To select all the drivers, click **Select All**.

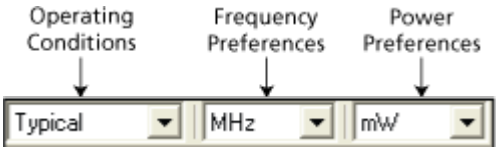
You can set or reset the probabilities, set a default probability, copy, export, or print the grid by selecting one or multiple pins and choosing an option from the right-click menu.

For more information, see Specifying Individual Pin Probabilities.

2.6. Preferences Toolbar (Ask a Question)

The SmartPower Preferences toolbar appears on the right side of the row below the menus. It provides quick access to commonly used settings. Mouse over the toolbar button to view a description of the command. Use the drop-down list to select your settings.

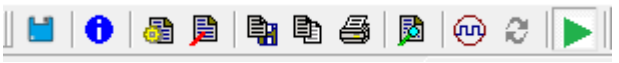
Figure 2-9. SmartPower Preferences Toolbar



- **Operating Conditions**: allows you to set the operating conditions: Best, Typical, or Worst. The actual voltage and temperature of a given operating condition is defined by the operating condition range selected for the current design (COM, IND, MIL, Auto, TGrade1, TGrade2, and Custom). For SmartPower, unlike SmartTime, the worst voltage is the upper bound of the range. That is, higher (worst-case) power comes from higher voltage, while higher voltage results in best-case speed.
  Note: Worst-case and best-case calculations are available only for certain families and devices.
- **Frequency Preferences**: sets unit preferences for frequency: Hz, KHz, MHz.
- **Power Preferences**: sets unit preferences for power: W, mW, or uW.

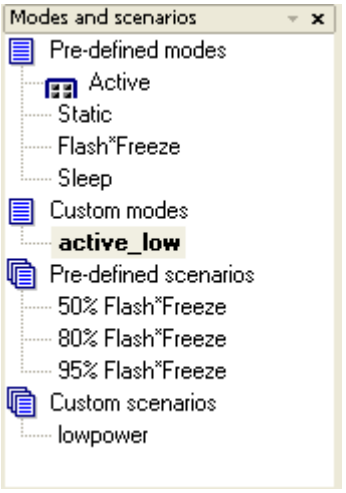
2.7. Modes and Scenarios Toolbar (Ask a Question)

The Modes and Scenarios toolbar appears as the left pane in the SmartPower interface. It displays all predefined modes, custom modes, predefined scenarios, and custom scenarios.



The Modes and Scenarios toolbar is shown by default, but can be closed or undocked. To view the toolbar, choose **Modes and Scenarios toolbar** from the **View** menu.

Figure 2-10. Modes and Scenarios Toolbar



From this toolbar, you can:

- Select a predefined mode by clicking the mode name.
- Select a custom mode by clicking the mode name.
- Select a custom mode for power-driven layout by right-clicking the active-based mode and selecting **Use for PDPR**.
- Re-import a VCD file from an Active mode or a custom mode by right-clicking the VCD file and selecting **Re- import VCD File**.
- Remove a VCD file from an Active mode or a custom mode by right-clicking the VCD file and selecting **Remove VCD File from <mode name> Mode** or by choosing **Remove VCD File <file name>** from the **Simulation** menu.
- Generate an Activity and Hazards Power report by right-clicking the VCD file and selecting **Activity and Hazards Power Report**
- Create a new custom mode by clicking **Custom modes** and selecting **New Custom Mode**, or by clicking the **New Custom Mode** hyperlink.
- Edit and delete a custom mode by right-clicking the mode name and selecting **Edit custom mode** or **Delete custom mode**.
- Copy a custom or predefined mode by right-clicking the mode name and selecting **Create a copy of this mode**.
- Select a scenario by clicking the scenario name.
- Create a new custom scenario by clicking **Scenarios** and selecting **New custom scenario**.
- Edit and delete a custom scenario by right-clicking the scenario name and selecting **Edit custom scenario**.
- Copy a scenario by right-clicking the scenario name and selecting **Create a copy of this scenario**.
  Note: When a scenario is selected, only the **Summary** tab is available.

2.8. SmartPower Standard Toolbar and Menu Commands (Ask a Question)

The SmartPower standard toolbar contains commands for performing common SmartPower operations on your designs. Roll the mouse pointer over the toolbar button to view a description of the button. Click the button to access the command.

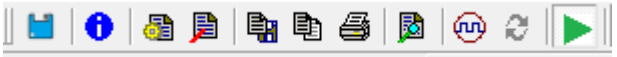


Table 2-3. SmartPower Toolbar

Toolbar Button	Description
	Commit
	Settings Summary
	Operating Conditions
	Options
	Export Path Grid
	Copy Path Grid
	Print Grid
	Generate Power Report
	Initialize Frequencies and Probabilities
	Refresh vectorless estimation
	Freeze/Unfreeze Calculations

The PC and UNIX workstation versions of SmartPower have the same menus. However, some dialog boxes, field locations and names, and buttons may look slightly different on the two platforms due to the different window environments.

2.9. File Menu (Ask a Question)

Menu	Description
Commit	Commits power information. You must commit your changes if you wish to save your settings in SmartPower. If you commit your changes, the information is stored in the .adb file, and your settings are restored the next time you open your design in SmartPower.
Export Grid	Exports the selected area of the Report window to a text (.txt) file.
Print Grid	Prints the selected area of the Report window.
Settings Summary	Displays a summary of the power settings for the current design.
Close	Closes SmartPower.

2.10. Edit Menu (Ask a Question)

Menu	Sub-Menu	Description
Domains	New Clock Domain	Adds a clock domain.
	New Set of Pins	Adds a set of pins.
	Delete Domain	Deletes a clock domain.
	Edit Domain	Edits a clock domain.

Edit Menu (continued)		
Menu	**Sub-Menu**	**Description**
Modes and Scenarios	Use for Analysis > *mode or scenario name*	Sets the selected mode or scenario to be used for analysis.
	Use for PDPR > *mode or scenario name*	Sets the selected mode or scenario to be used for a power-driven layout.
	Custom Modes > New Custom Mode	Creates a new custom mode.
	Custom Modes > Edit Custom Mode	Edits a custom mode.
	Custom Modes > Delete Custom Mode	Deletes a custom mode.
	Custom Scenarios > New Scenario	Creates a new custom scenario.
	Custom Scenarios > Edit Scenario	Edits a custom scenario.
	Custom Scenarios > Delete Scenario	Deletes a custom scenario.
Copy Grid	—	Copies the selected cells of the dynamic grid onto the Clipboard.

2.11. View Menu (Ask a Question)

Menu	Description
Standard Toolbar	Displays or hides the Standard toolbar.
Preferences Toolbar	Displays or hides the Preferences toolbar.
Modes and Scenarios Toolbar	Displays or hides the Modes and Scenarios toolbar.
Unprocessed Simulation Files Toolbar	Displays or hides the Unprocessed Simulation Files toolbar.
Customize Columns	Selects and orders the columns displayed in the user interface.
Reset Window Layout	Resets the window layout to the default layout.

2.12. Tools Menu (Ask a Question)

Menu	Sub-Menu	Description
Initialize Frequencies and Probabilities	—	Imports clock constraints from SmartTime, and allows you to enter frequencies and probabilities for the design, as shown in the Initializing Frequencies and Probabilities section.
Operating Conditions	—	Allows you to customize the operating conditions used in SmartPower.
Options	—	Displays the Options dialog box, where you can set analysis and display preferences.
Reports	Power Report	Generates a power report.
	Scenario Power Report	Generates a scenario power report.
	Activity and Hazards Power Report	Generates an activity and hazards power report.
Export Report for MPE	—	Exports a report for Microchip Power Estimator (MPE).
Freeze/Unfreeze Calculations	—	Freezes or unfreezes the computations that happen to avoid the SmartPower tool from slowing down

2.13. Simulation Menu (Ask a Question)

Menu	Description
Import VCD File	Displays the Import VCD Options dialog box.
Remove VCD file	Removes the selected VCD file from the design.

Simulation Menu (continued)	
Menu	**Description**
Audit files	Audits simulation files within the project.

2.14. Help Menu (Ask a Question)

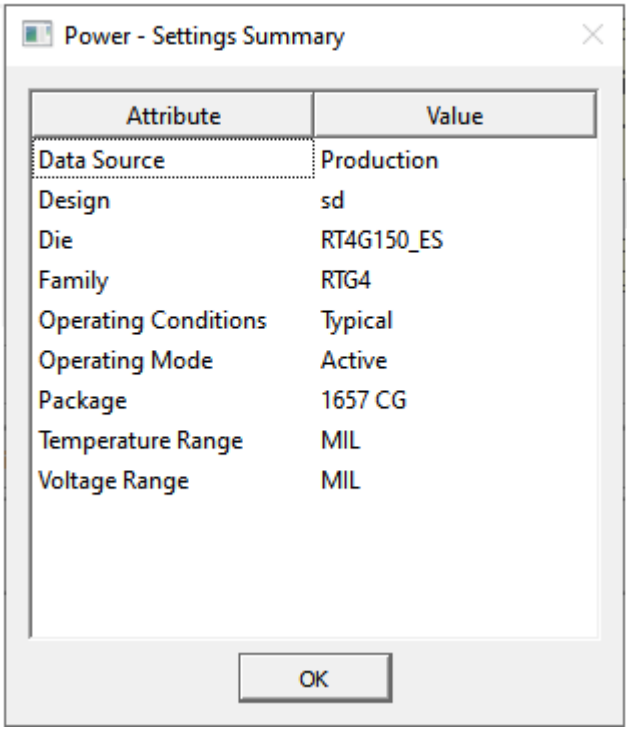
Menu	Description
Help on *current tab*	Displays the help content for the current tab.
Help Topics	Opens the help files for SmartPower.
SmartPower User's Guide	Displays the SmartPower User's Guide.
Data Change History	Displays the data change history report.

For more information, see the SmartPower Preferences Toolbar.

2.15. Power-Settings Summary (Ask a Question)

The Power-Settings Summary dialog box summarizes the power settings for the current design. To open the dialog box, choose **Settings Summary** from the **File** menu or click the icon in the SmartPower toolbar.

Figure 2-11. Power-Settings Summary Dialog Box



The Power-Settings Summary dialog box shows the following information.

Table 2-4. Power-Settings Summary Dialog Box Attributes

Attribute	Description
Data Source	Displays the data source.
Design	Displays the name of the design.

Table 2-4. Power-Settings Summary Dialog Box Attributes (continued)

Attribute	Description
Family	Displays the device family used in the design.
Die	Displays the die.
Package	Displays the package.
Temperature Range	Displays the temperature range.
Voltage Range	Displays the core supply voltage.
Operating Condition	Displays the operating conditions that SmartPower uses to calculate static and dynamic power contributions.
Operating Mode	Displays the operating mode that SmartPower uses to calculate static and dynamic power contributions.

Note: The status bar also shows the ambient temperature, VCCA, and speed grade.

2.16. Initializing Frequencies and Probabilities (Ask a Question)

The Initialize Frequencies and Probabilities dialog box allows you to initialize the frequencies and probabilities for your design.

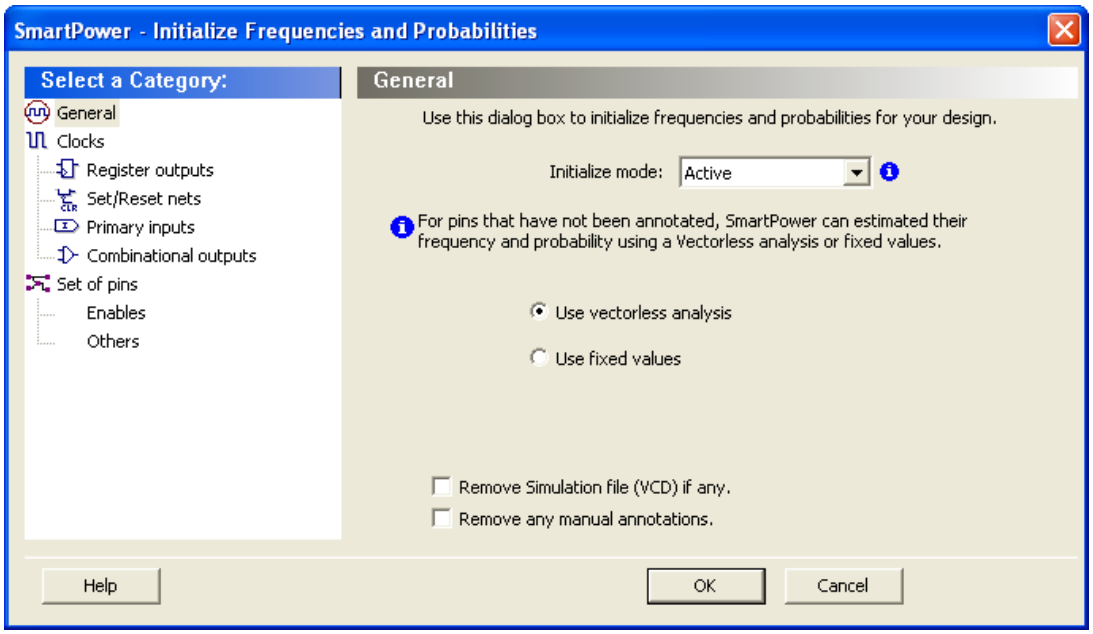
To open the Initialize Frequencies and Probabilities dialog box, choose **Initialize Frequencies and Probabilities** from the **Tools** menu, or click the  icon in the SmartPower toolbar.
Note: This feature is available only in the Active operating mode or in modes derived from the Active mode.

The dialog box is organized into the following panels:

- General
- Clocks
- Register Outputs
- Set/Reset Nets
- Primary
- Inputs
- Combinational Outputs
- Enables Sets of Pins
- Other Sets of Pins

2.16.1. General (Ask a Question)

Figure 2-12. Initialize Frequencies and Probabilities Dialog Box – General



Initialize mode: select an active-based mode for which you want to initialize the frequencies and probabilities. For pins that have not been annotated, SmartPower can estimate their frequency and probability using vectorless analysis or fixed values:

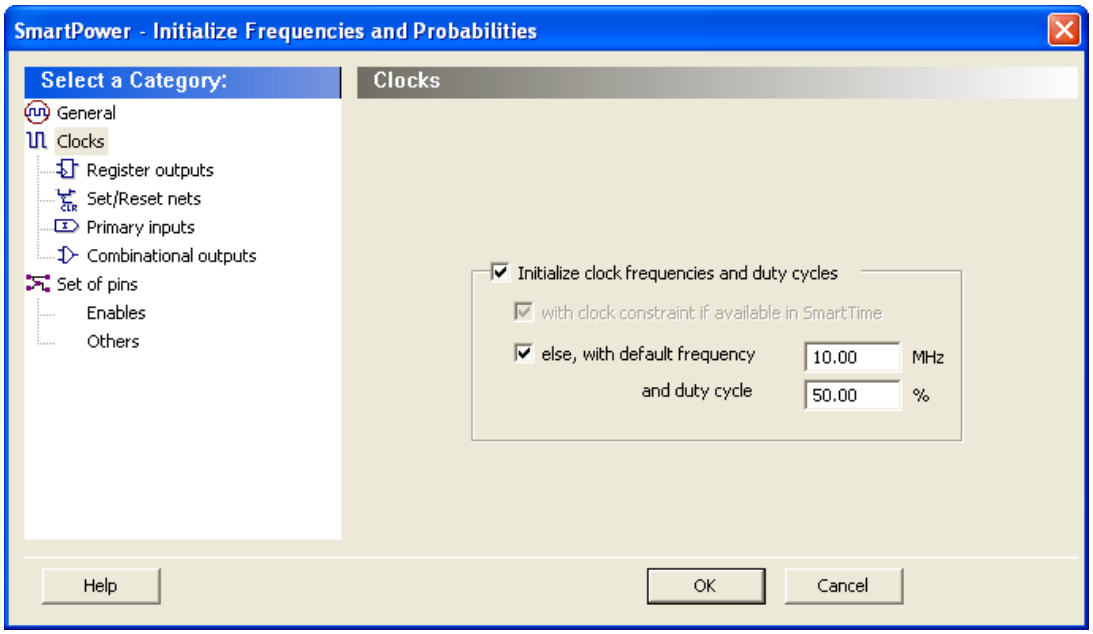
- **Use vectorless analysis**: SmartPower initializes frequencies and probabilities with vectorless analysis.
- **Use fixed values**: SmartPower initializes frequencies and probabilities (%) of all registers with a default toggle rate (when using toggle rates), frequency, and probability.

Based on the selected mode, a warning appears with information about the annotated pins. If the selected mode contains annotations, you can remove them by selecting one or both of the following options:

- **Remove Simulation file (VCD) if any**: removes the simulation file for the selected mode.
- **Remove any manual annotations**: removes all manual pin annotations for the selected mode.

2.16.2. Clocks (Ask a Question)

Figure 2-13. Frequencies and Probabilities Dialog Box – Clocks

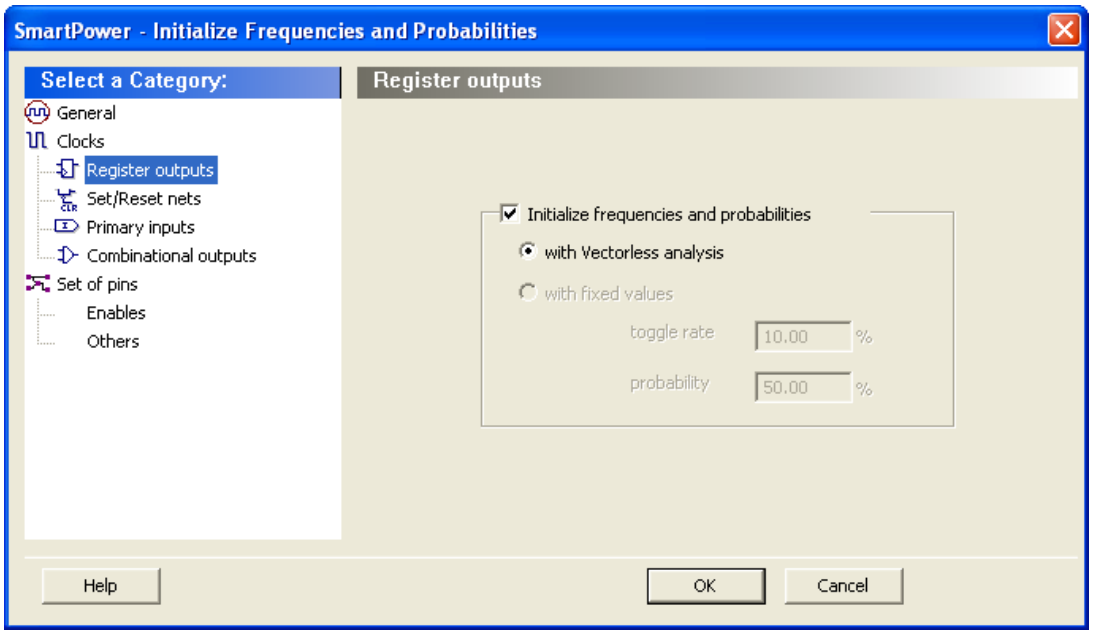


- **Initialize clock frequencies and duty cycles**: initializes clock frequencies and duty cycles with one of the following:
  - **with clock constraint if available in SmartTime**: when only this box is checked, SmartPower initializes all clock domains in SmartPower with clock constraints from SmartTime. If there is no frequency constraint set for this clock domain in SmartTime, the clock frequency of the domain will not be updated.
  - **else, with default frequency (MHz) and duty cycle (%)**: when only this box is checked, SmartPower initializes all clock domains in SmartPower with the specified default frequency and duty cycle.

Note: When both boxes are checked, and **with clock constraint if available in SmartTime**, **else default frequency**, and **and duty cycle** selected, SmartPower initializes all clock domains with clock constraints from SmartTime. If there is no frequency constraint set for a clock domain in SmartTime, it is initialized with the specified default frequency and duty cycle.

2.16.3. Register Outputs (Ask a Question)

Figure 2-14. Frequencies and Probabilities Dialog Box –Registers



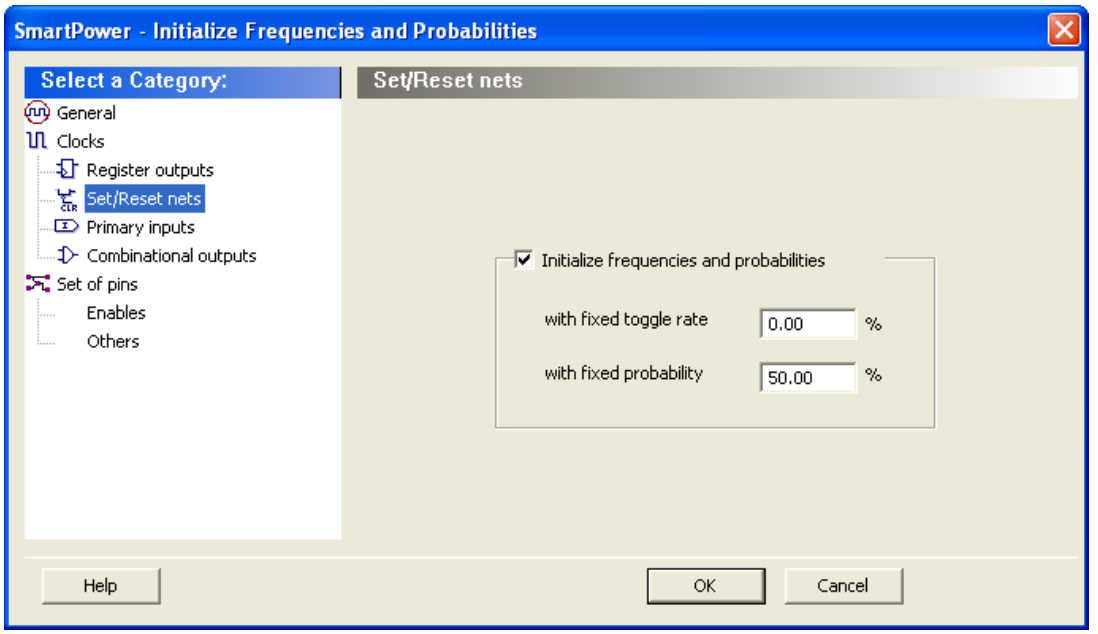
Initialize frequencies and probabilities: initializes frequencies and probabilities for register outputs in your design.

with Vectorless analysis: SmartPower initializes frequencies and probabilities using vectorless analysis.

with fixed values: SmartPower initializes frequencies and probabilities (%) of all register outputs with a default toggle rate (available only when using toggle rates) or frequency (when not using toggle rates), and a default probability.

2.16.4. Set/Reset Nets (Ask a Question)

Figure 2-15. Frequencies and Probabilities Dialog Box – Set/Reset nets



Initialize frequencies and probabilities: initializes frequencies and probabilities for set/reset nets in your design.

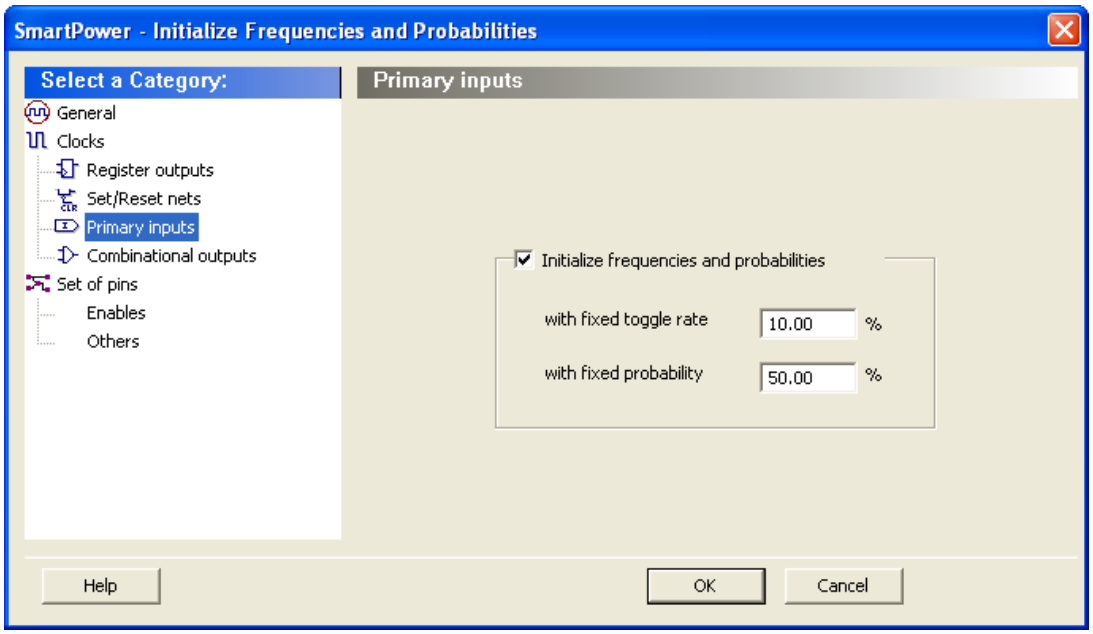
with fixed frequency (MHz): SmartPower initializes all primary inputs in SmartPower with the specified fixed frequency. This option is available only when not using toggle rates.

with fixed toggle rate (%): SmartPower initializes frequencies and probabilities of all set/reset nets with the specified fixed toggle rate. This option is available only when using toggle rates.

with fixed probability (%): SmartPower initializes probabilities of all set/reset nets with the specified fixed probability.

2.16.5. Primary Inputs (Ask a Question)

Figure 2-16. Frequencies and Probabilities Dialog Box – Primary Inputs



Initialize frequencies and probabilities: initializes frequencies and probabilities for the primary inputs in your design.

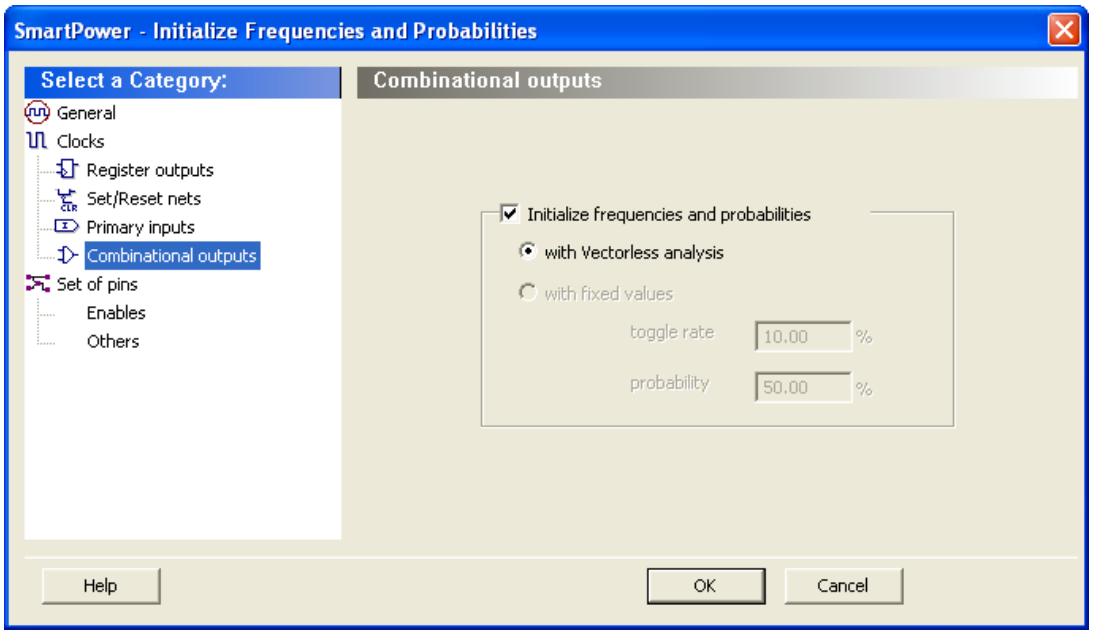
with fixed toggle rate (%): SmartPower initializes frequencies and probabilities of all primary inputs with the specified fixed toggle rate. This option is available only when using toggle rates.

with fixed frequency (MHz): SmartPower initializes all primary inputs in SmartPower with the specified default frequency. This option is available only when not using toggle rates.

with fixed probability (%): SmartPower initializes probabilities of all primary inputs with the specified fixed probability.

2.16.6. Combinational Outputs (Ask a Question)

Figure 2-17. Frequencies and Probabilities Dialog Box – Combinational Outputs



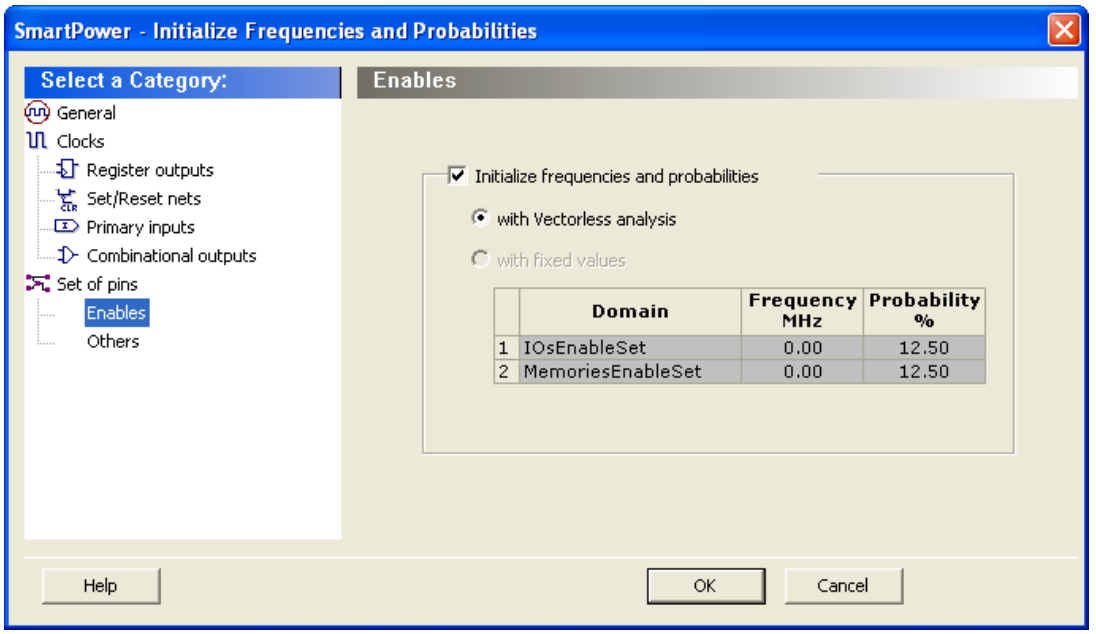
Initialize frequencies and probabilities: initializes frequencies and probabilities for the combinational outputs in your design.

with Vectorless analysis: SmartPower initializes frequencies and probabilities with vectorless analysis.

with fixed values: SmartPower initializes frequencies and probabilities (%) of all combinational outputs with a default toggle rate (available only when using toggle rates) or frequency (when not using toggle rates), and a default probability.

2.16.6.1. Enables (Ask a Question)

Figure 2-18. Frequencies and Probabilities Dialog Box – Enables



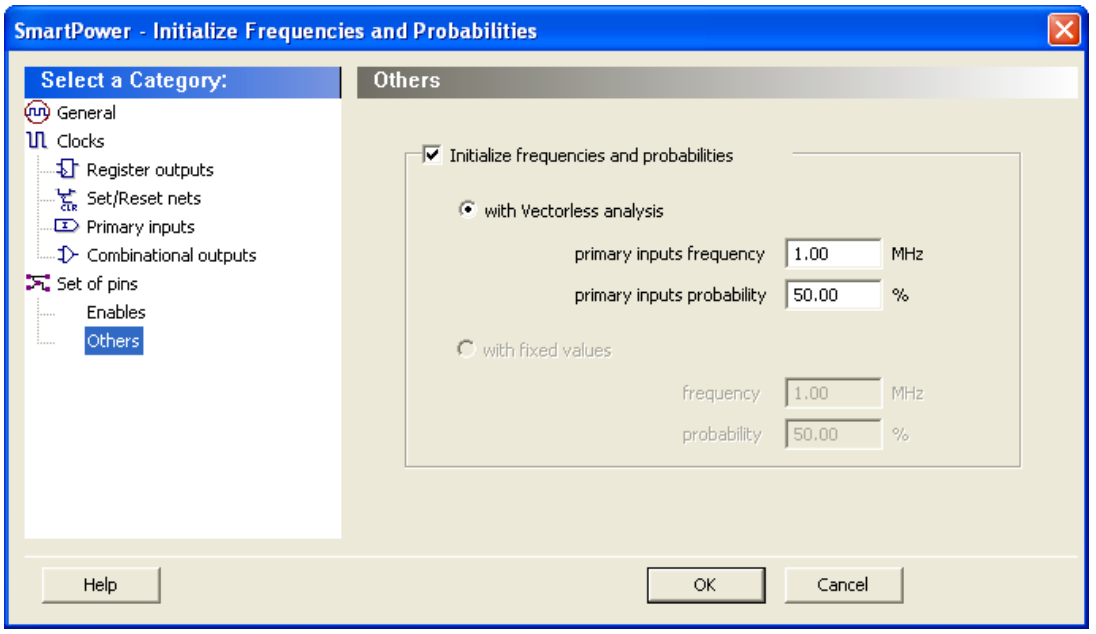
Initialize frequencies and probabilities: initializes frequencies and probabilities for the enable sets of pins in your design.

with Vectorless analysis: SmartPower initializes frequencies and probabilities with vectorless analysis.

with fixed values: SmartPower initializes frequencies and probabilities of all enables sets of pins with the specified default probability and frequency.

2.16.6.2. Others (Ask a Question)

Figure 2-19. Frequencies and Probabilities Dialog Box – Others



Initialize frequencies and probabilities: initializes frequencies and probabilities for the other sets of pins in your design with default values.

with Vectorless analysis: SmartPower initializes frequencies and probabilities with vectorless analysis as specified in the primary inputs frequency and probability.

with fixed values: SmartPower initializes frequencies and probabilities (%) of all other sets of pins with a default frequency and probability.

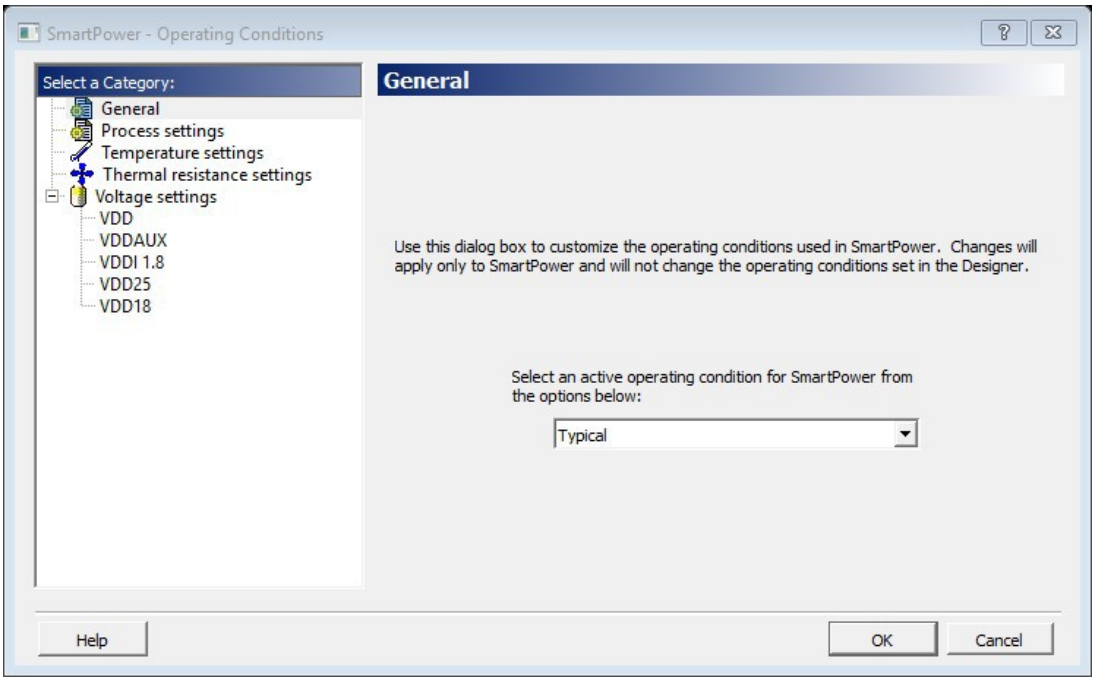
2.17. Operating Conditions Dialog (Ask a Question)

The SmartPower Operating Conditions dialog box allows you to customize the operating conditions used in SmartPower for voltage and temperature.

Note: Changes will apply only to SmartPower and will not change the operating conditions set in Designer.

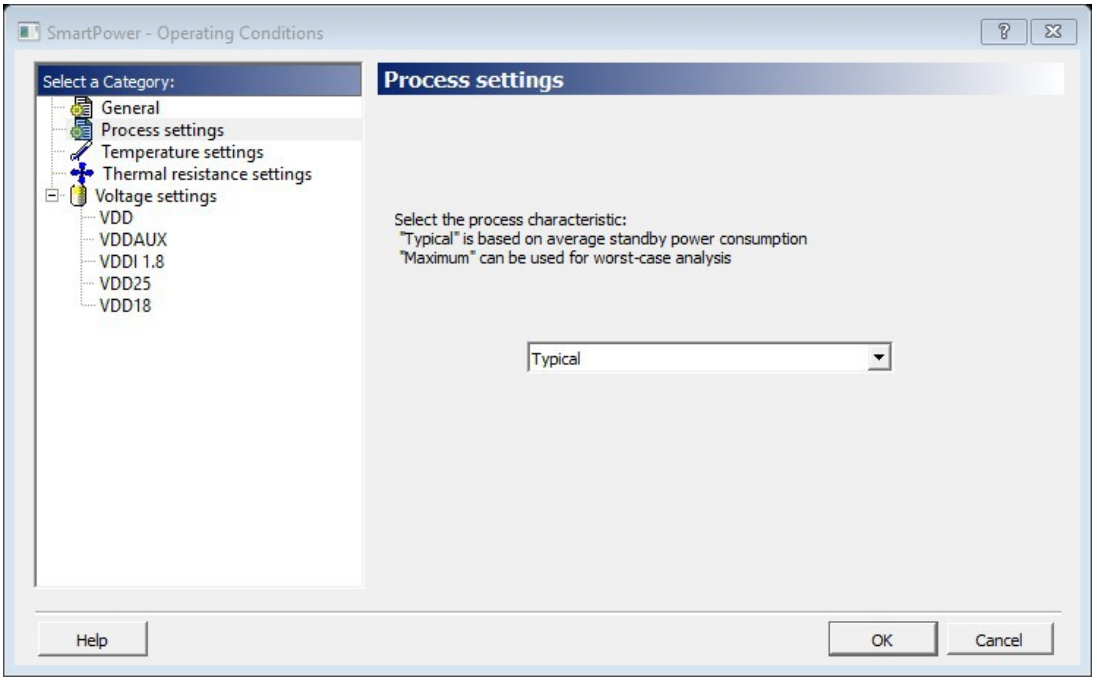
The **General** panel allows you to select the operating condition you want to apply to SmartPower: Best, Typical, or Worst.

Figure 2-20. SmartPower Operating Conditions Dialog Box - General



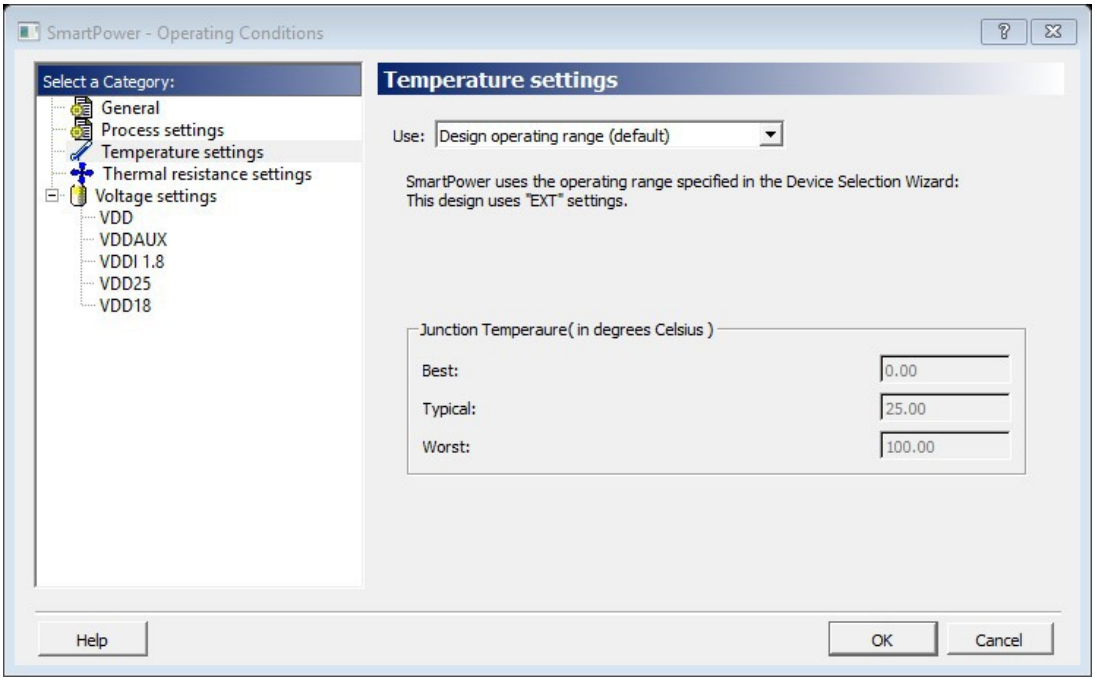
The process settings panel allows you to select the process characteristic.

Figure 2-21. SmartPower Operating Conditions Dialog Box - Process Settings



You can customize temperature settings in the Temperature Settings panel, and the voltage settings in the Voltage Settings panels—each VCC voltage used in the design appears under Voltage settings.

Figure 2-22. SmartPower Operating Conditions Dialog Box - Temperature Settings

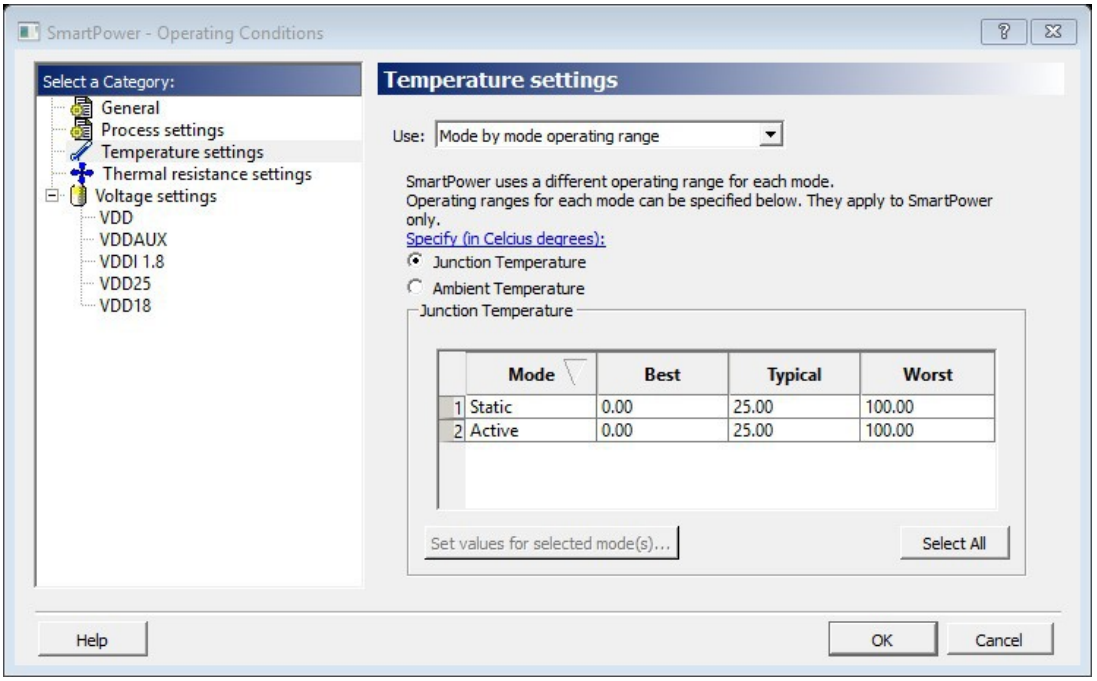


Select one of three options to specify the temperature or voltage settings:

- **Design operating range (default)**: sets the temperature or voltage operating range as specified in your Project Settings.

- **Custom operating range**: sets the temperature (junction or ambient) or voltage operating range for the current design in SmartPower only. Enter temperature values in degrees Celsius and voltage values in volts for Best, Typical, and Worst.
- **Mode by mode operating range**: sets the temperature (junction or ambient) or voltage for each mode in SmartPower only. Use the grid to enter temperature values in degrees Celsius and voltage values in volts for each mode, or select multiple modes in the grid, and click the **Set values for selected modes** button to enter the same values for all selected modes.

Figure 2-23. SmartPower Operating Conditions Dialog Box - Mode-Specific Temperature Settings



The junction temperature and static power are computed iteratively until they converge. The formulas are:

$$T_J = T_A + \theta_{JA} \cdot (P_{dynamic} + P_{static}) P_{static} = f(T_J) = P_0 + P_1 \cdot T_J$$

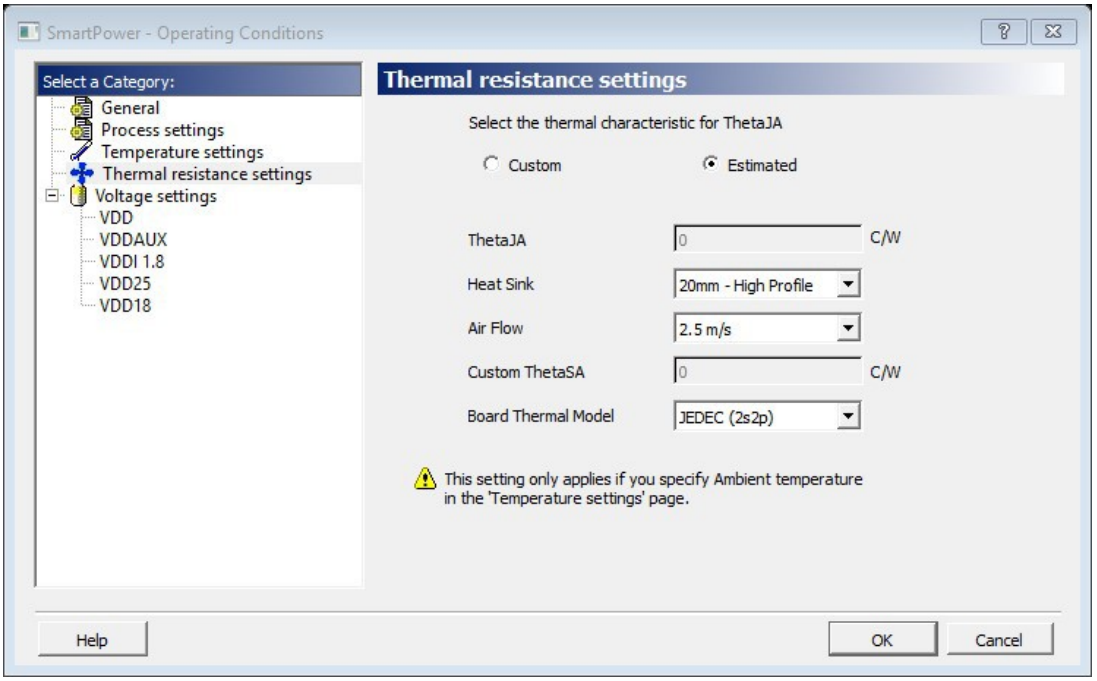
If the computed junction temperature is outside the operating condition range (smaller than the best case value or greater than the worst case value), a violation warning will be reported.

Both T_J and P_{static} is re-computed when any of the following parameters are changed:

- Ambient temperature
- Cooling style
- Custom thermal resistance or any parameters/options that affect the static or dynamic power values:
- Junction Temperature Display Mode
- Operating Condition (Affect dynamic power)
- Operating Mode (Affect dynamic power)
- Any of the clock domain frequencies (Affect dynamic power)

If you specified the Ambient temperature in the Temperature Settings panel, you can select the thermal characteristic for θ_{JA}.

Figure 2-24. SmartPower Operating Conditions Dialog Box - Thermal Resistance Settings



The thermal resistance is predefined for Still Air, 300 ft/min and Case Cooling. To enter a custom thermal resistance, select **Custom** from the thermal characteristic for θ_{JA}

SmartPower also reports the thermal resistance θ_{JA}.

2.17.1. Custom Operating Modes (Ask a Question)

SmartPower allows you to use predefined and custom operating modes. A predefined operating mode is an inherent mode of a family of devices that comes with a number of hard availability constraints on the FPGA resources. Custom operating modes are user-defined modes based on one of the predefined modes available for the family, where parameters such as clock, data, and toggle rate frequencies have been customized. A custom mode can also be linked to a VCD.

Custom modes can be used:

- To define additional modes where the chip is in the active predefined mode, but where the clock frequencies or the toggle rates are scaled up or down.
- To import multiple VCD files that capture individual steps that take place in different modes.
- To define a sequence of modes with a duration for each mode and get an estimate of the overall power consumption, from which extrapolate the battery life for the application. The Scenario Power report allows you to define a sequence of modes and estimate the weighted average of the power consumption of the chip over the entire sequence.
- To define additional modes where the chip is in a predefined mode, but the junction/ambient temperature or the rails' voltage modes are different.

2.17.1.1. Setting the Current Mode (Ask a Question)

To set the current mode:

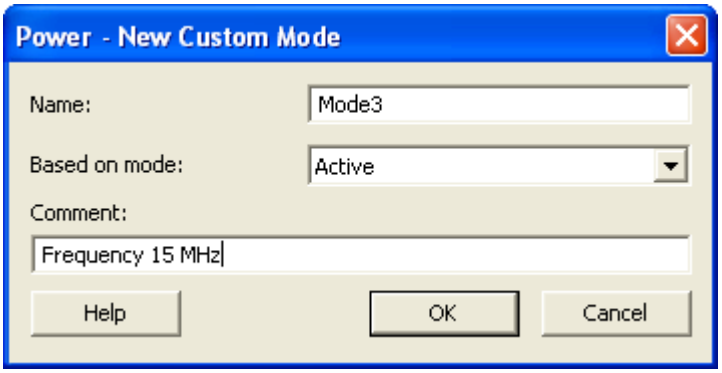
1. From the **Edit** menu in SmartPower, choose **Modes and Scenarios > Use for Analysis > *mode name*** or select the mode in the Modes and Scenarios toolbar.
   Enter the result of your step here (optional).

2.17.1.2. Creating a New Custom Mode (Ask a Question)

To create a new custom mode:

1. From the **Edit** menu in SmartPower, choose **Modes and Scenarios > Custom Modes > New Custom Mode** or click **Custom Modes** in the Modes and Scenarios toolbar, and select **New custom mode**.
   The New Custom Mode dialog box appears.

   Figure 2-25. New Custom Mode Dialog Box



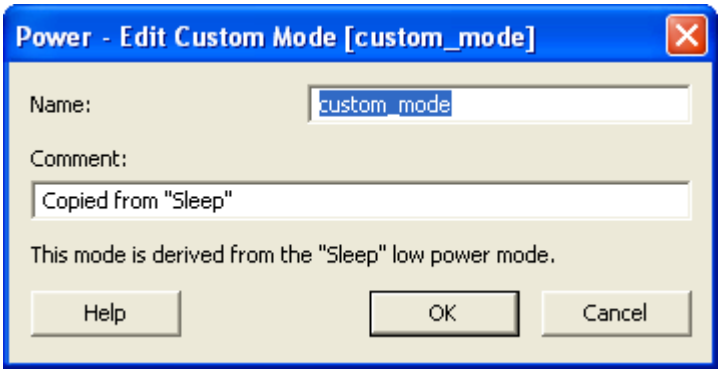
2. Enter the mode name, select the base mode, and enter comments that describe the mode. The base mode can be either a user-defined custom mode or the active mode. The new mode will inherit all clock and data frequencies and annotations of its base mode.
3. Click **OK**. The newly created operating mode is set as default in the Modes and Scenarios toolbar.

2.17.1.3. Editing a Custom Mode (Ask a Question)

To edit a custom mode:

1. From the **Edit** menu in SmartPower, choose **Modes and Scenarios > Custom Modes > Edit Custom Mode**, or right-click the mode in the Modes and Scenarios toolbar and select **Edit custom mode**.
   The Edit Custom Mode dialog box appears.

   Figure 2-26. Edit Custom Mode Dialog Box



2. Edit the mode name and comments.
3. Click **OK**.

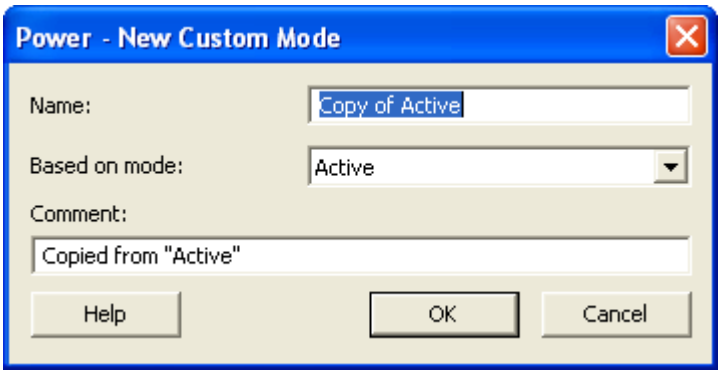
   Note: You can only use the Edit Mode dialog box to edit the name and comment of the current mode. To edit the mode properties, such as clock and data frequencies, use the SmartPower tabs.

2.17.1.4. Copying a Custom Mode (Ask a Question)

To copy a custom mode:

1. Right-click the mode in the Modes and Scenarios toolbar and choose **Create a copy of this mode**.
   The New Custom Mode dialog box appears.

   Figure 2-27. New Custom Mode Dialog Box



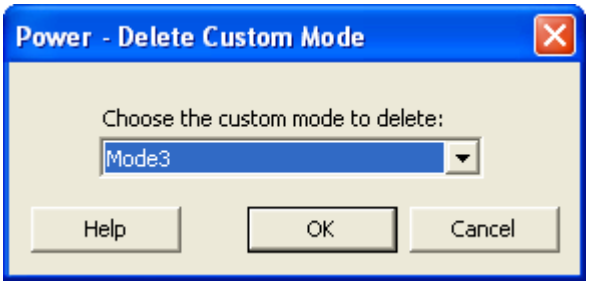
2. Edit the mode name and comments.
3. Click **OK**.

2.17.1.5. Deleting a Custom Mode (Ask a Question)

To delete a custom mode:

1. From the **Edit** menu in SmartPower, choose **Modes and Scenarios > Custom Modes > Delete Custom Mode**; or right-click the mode in the Modes and Scenarios toolbar, and select **Delete Mode**. This brings up the Delete Custom Mode dialog box.
   The Delete Custom Mode appears.

   Figure 2-28. Delete Custom Mode Dialog Box



2. Select the mode from the pull down and click **OK**.
   Note: You can delete only custom operating modes. You cannot delete predefined modes.

2.18. Scenarios (Ask a Question)

A scenario is a sequence of previously defined operating modes with a specific duration for each mode. When you create a new design, SmartPower will create predefined scenarios automatically that can be used as is, or copied to create a new scenario based on the predefined one.
Note: You cannot edit or delete predefined scenarios.

2.18.1. Creating a New Custom Scenario (Ask a Question)

To create a new custom scenario:

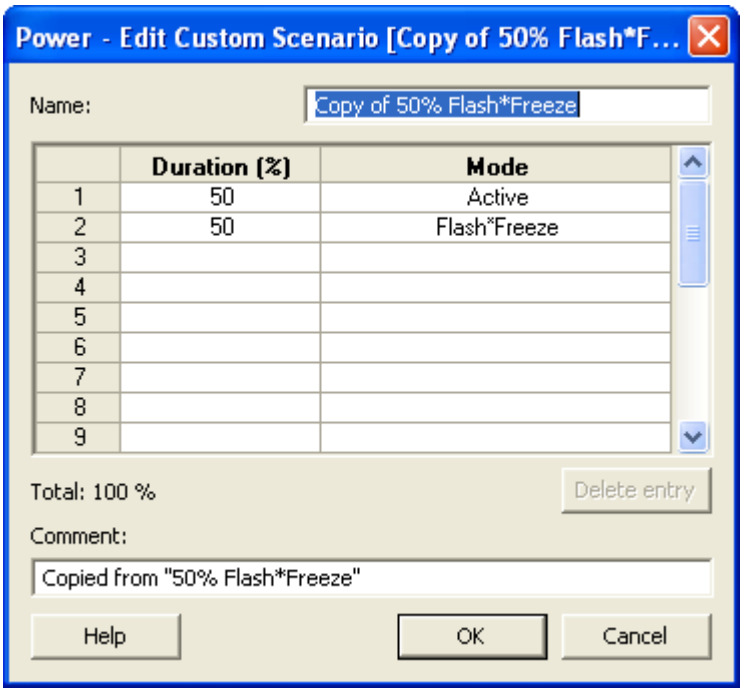
1. From the **Edit** menu in SmartPower, choose **Modes and Scenarios > Custom Scenarios > New Custom Scenario**; or click **Scenarios** in the Modes and Scenarios toolbar, and select **New custom scenario**.
2. Enter the scenario name, enter a duration (total duration for the sequence must equal to 100%) and select previously defined operating modes for this sequence.
3. Click **OK**.
   The newly created scenario is set as default in the Modes and Scenarios toolbar.

2.18.2. Editing a Custom Scenario (Ask a Question)

To edit a custom scenario:

1. From the **Edit** menu in SmartPower, choose **Modes and Scenarios > Custom Scenarios > Edit Custom Scenario**; or right-click the scenario in the Modes and Scenarios toolbar, and select **Edit custom scenario**.
   The Edit Custom Scenario dialog box appears.

Figure 2-29. Edit Custom Scenario Dialog Box



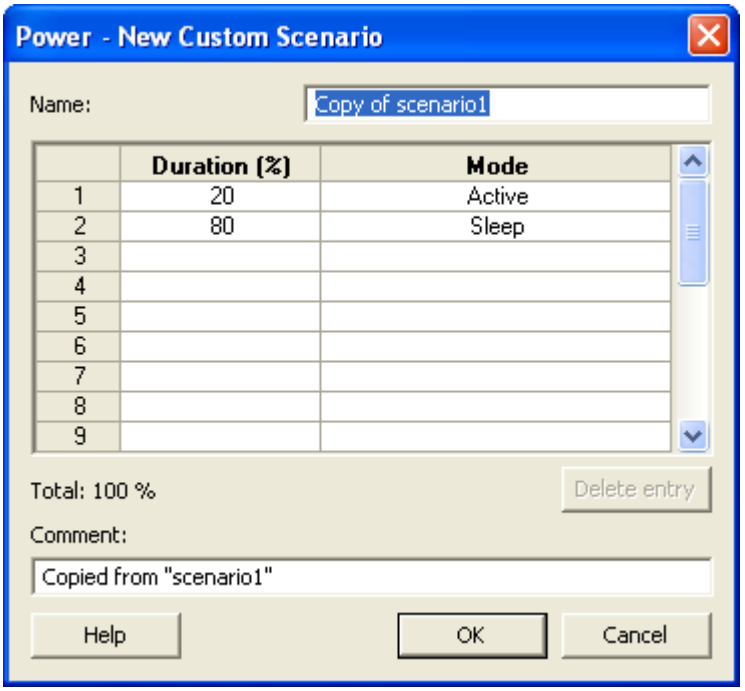
2. Edit the information.
3. Click **OK**.

2.18.3. Copying a Scenario (Ask a Question)

To copy a scenario:

1. Right-click the scenario in the Modes and Scenarios toolbar, and select **Create a copy of this scenario**.
   The New Custom Scenario dialog box appears.

Figure 2-30. New Custom Scenario Dialog Box



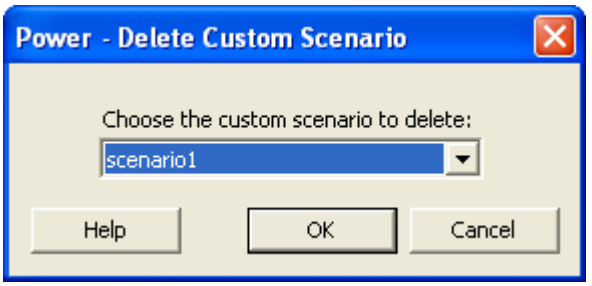
2. Edit the information.
3. Click **OK**.

2.18.4. Deleting a Custom Scenario (Ask a Question)

To delete a custom scenario:

1. From the **Edit** menu in SmartPower, choose **Modes and Scenarios > Custom Scenarios > Delete Custom Scenario** or right-click the scenario in the Modes and Scenarios toolbar, and select **Delete Scenario**.
   The Delete Custom Scenario dialog box appears.

Figure 2-31. Delete Custom Scenario Dialog Box

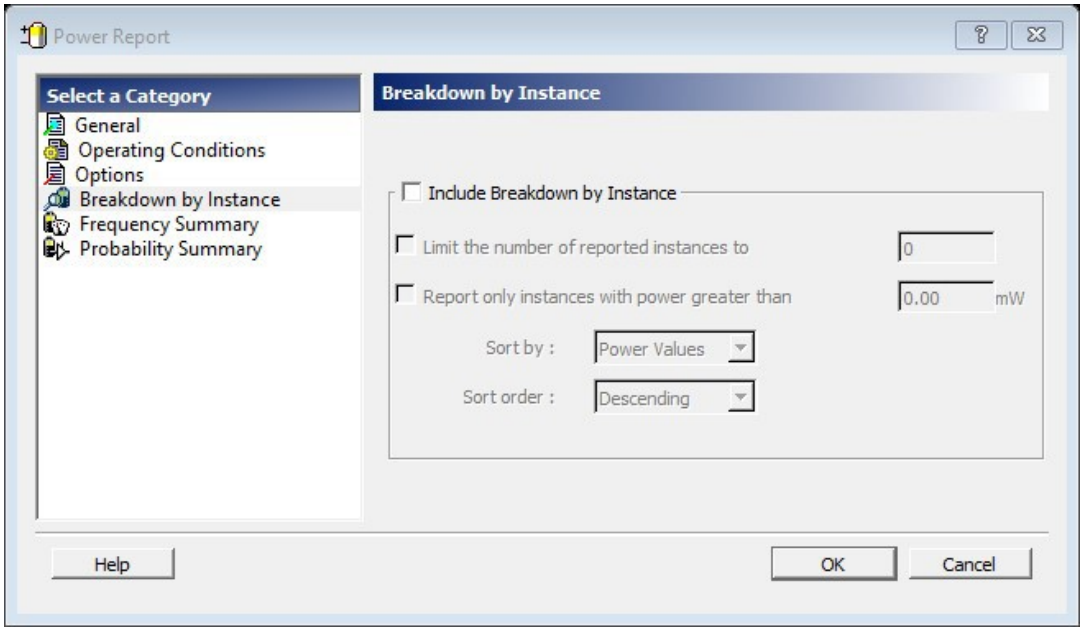


2. Select the scenario from the pull down and click **OK**.

2.18.5. Breakdown by Instance (Ask a Question)

Using the **Breakdown by Instance** panel, you can include the breakdown by instance in the report and set specific options.

Figure 2-32. SmartPower Power Report Dialog Box - Breakdown By Instance Panel



Include Breakdown by Instance: shows the power consumption of each element that has been instantiated in the design: gates, nets, memories, and I/Os. The breakdown by instance can be filtered by:

- **Limit the number of reported instances to**: limits the number of instances reported to the specified number.
- **Report only instances with power greater than**: does not report instances with power consumption below the selected value.

This section can be sorted by selecting the preferred method:

- **Sort by**: sorts by name (alphabetical) or power values.
- **Sort order**: sorts by ascending or descending order.
  Note: The filter reduces the number of lines in the report by one per instance.

2.19. Power Reports (Ask a Question)

The Power report allows you to identify power consumption problems in your design. You can generate this report from the **Tools** menu and from SmartPower.

The Power Report dialog box is organized into the following panels:

- General
- Operating Conditions
- Options
- Breakdown by Instance
- Frequency Summary
- Probability Summary

2.19.1. Generating Power Reports from the Tools Menu (Ask a Question)

To generate Power reports from the **Tools** menu:

1. From the **Tools** menu, choose **Reports > Power** .

The Power Report dialog box appears.

2. Select the options you want to include in the report.
3. Click **OK**.
   The Power Report appears in a separate window.

2.19.2. Generating Power Reports from within SmartPower (Ask a Question)

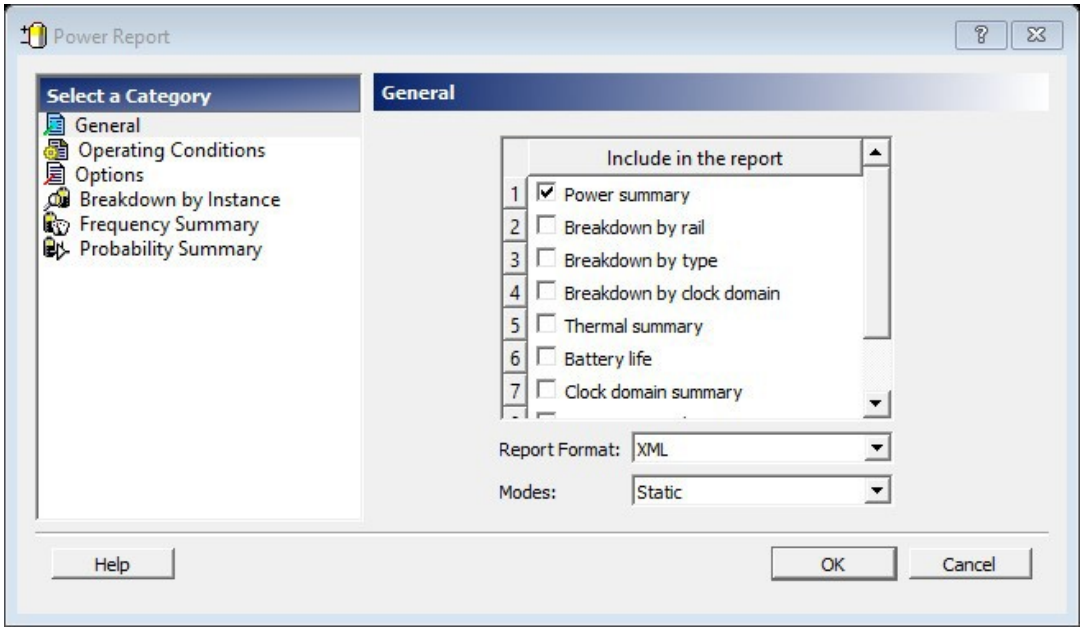
To generate Power reports from within SmartPower:

1. From the **Tools** menu, choose **Reports > Power Report** or click the **Report** button to open the Report dialog box. By default, the report includes global design information and a power summary.
2. Specify which results you want to display by selecting the categories and their options.

2.19.3. General (Ask a Question)

The **General** panel allows you to select what to include in the report, the report format, and the report mode.

Figure 2-33. SmartPower Power Report Dialog Box - General Panel



2.19.4. Include in the Report (Ask a Question)

Select the options you want to include in the Power report:

Power summary: reports the static, dynamic, and total power consumption of the design.

Breakdown by rail: shows the power consumption of each rail.

Breakdown by type: reports power consumption according to gates, nets, clocks, core static, I/Os, and memories.

Breakdown by clock domain: reports power consumption of each clock domain.

Thermal summary: includes a thermal report. The ambient temperature can be defined by operating conditions or ambient temperature.

When operating conditions is selected, the following characteristics are reported:

- Operating conditions
- Temperature range
- Junction temperature

When ambient temperature is selected, the following characteristics are reported:

- Ambient temperature
- Cooling style
- Package
- Thermal resistance Theta-JA
- Junction temperature
- Temperature range
- Junction temperature range limits specification

Battery life: reports the battery life.

Clock Domain summary: reports clock and data frequencies for each clock domain.

Operating Condition Summary: reports operating conditions.

Annotation Coverage: reports the number and percentage of pins annotated by each source (VCD, manual annotation, SDC constraint, vectorless estimation, and fixed values) for all clocks, register outputs, combinational outputs, set/reset nets, primary inputs, enable pins, and other pins.

2.19.5. Report Format (Ask a Question)

Select **Text**, **XML**, or **CSV** (comma-separated value) as the desired export format.

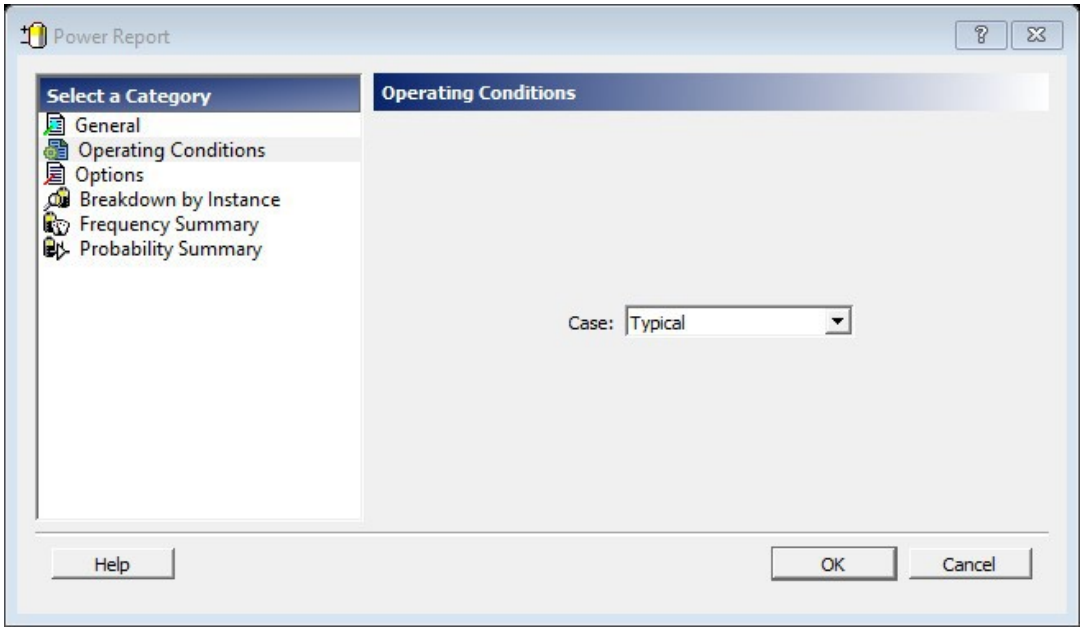
2.19.6. Mode (Ask a Question)

Select a mode for generating the report.

2.19.7. Operating Conditions (Ask a Question)

The **Operating Conditions** panel allows you to select the operating conditions for the current report.

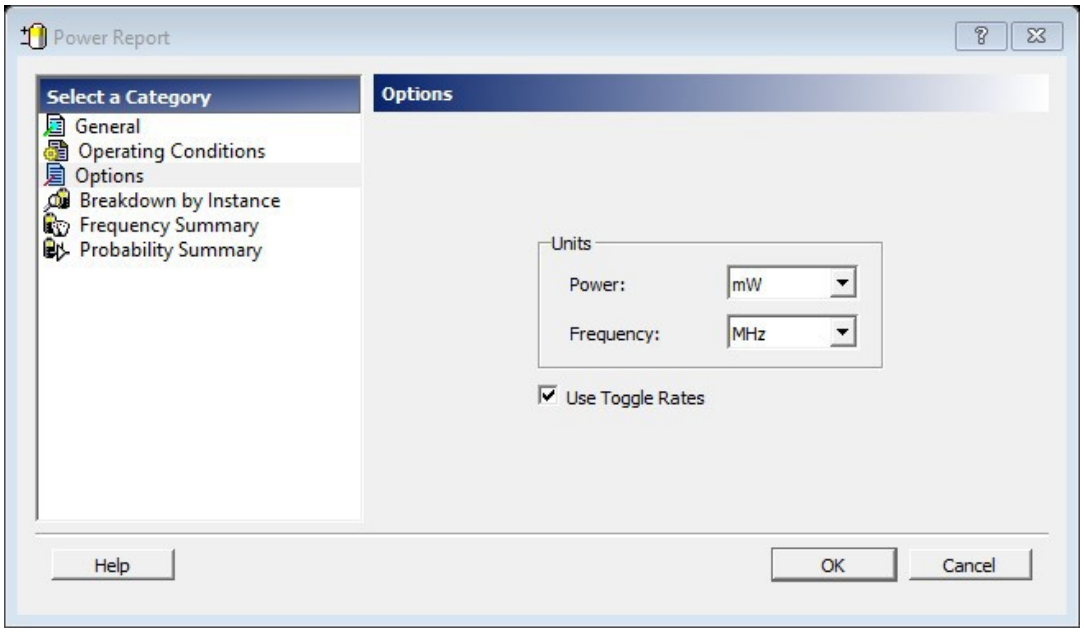
Figure 2-34. SmartPower Power Report Dialog Box - Operating Conditions Panel



2.19.8. Options (Ask a Question)

The **Options** panel allows you to select power and frequency units and to use toggle rates.

Figure 2-35. SmartPower Power Report Dialog Box - Options Panel



2.19.8.1. Units (Ask a Question)

Frequency: sets unit preferences for frequency: Hz, KHz, MHz.

Power: sets unit preferences for power: W, mW, or uW.

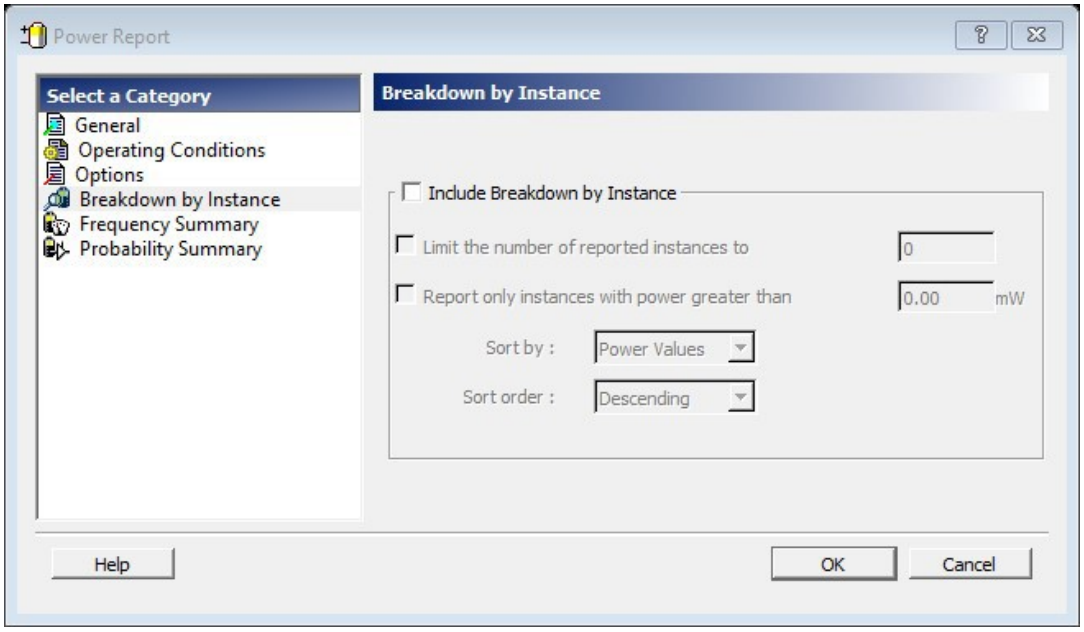
2.19.8.2. Use Toggle Rates (Ask a Question)

When toggle rates are active (**Use Toggle Rates** box is checked), the data frequency of all the clock domains is defined as a function of the clock frequency. This updates the data frequency automatically when you update the clock frequency. Toggle rates allow you to specify the data frequency as a percentage of clock frequency; however, you can specify the data frequency only as a percentage of the clock frequency, not as a number. To set the data frequency again, clear the **Use Toggle Rates** option.

2.19.9. Breakdown by Instance (Ask a Question)

Using the **Breakdown by Instance** panel, you can include the breakdown by instance in the report and set specific options.

Figure 2-36. SmartPower Power Report Dialog Box - Breakdown By Instance Panel



Include Breakdown by Instance: shows the power consumption of each element that has been instantiated in the design: gates, nets, memories, and I/Os. The breakdown by instance can be filtered by:

- **Limit the number of reported instances to**: limits the number of instances reported to the specified number.
- **Report only instances with power greater than**: does not report instances with power consumption below the selected value.

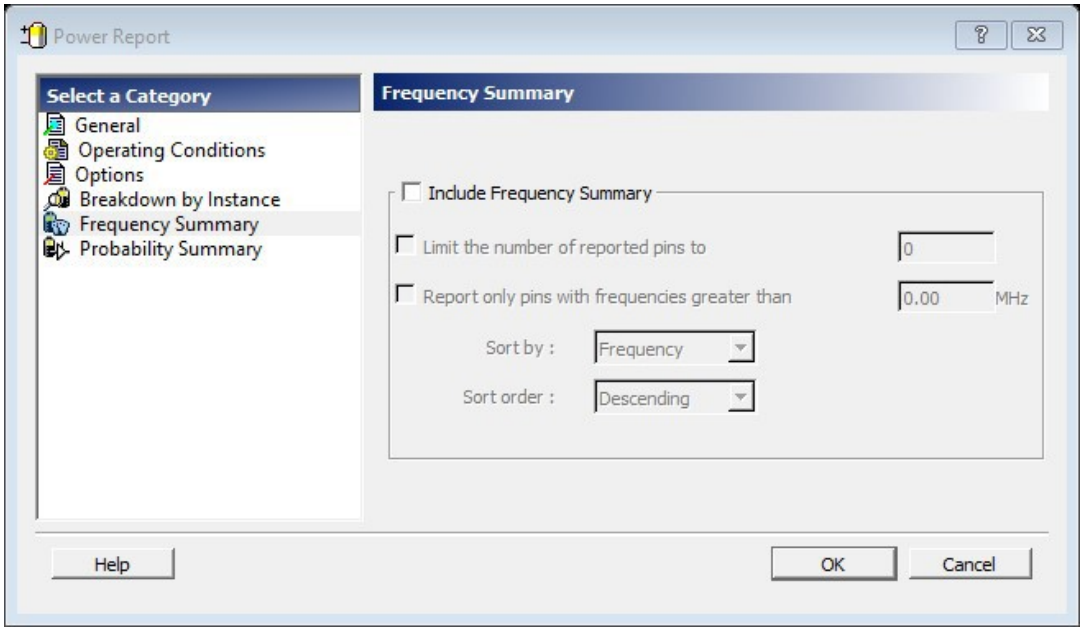
This section can be sorted by selecting the preferred method:

- **Sort by**: sorts by name (alphabetical) or power values.
- **Sort order**: sorts by ascending or descending order.
  Note: The filter reduces the number of lines in the report by one per instance.

2.19.10. Frequency Summary (Ask a Question)

Using the **Frequency Summary** panel you can include the frequency summary in the report and set specific options.

Figure 2-37. SmartPower Power Report Dialog Box - Frequency Summary Panel



Include Frequency Summary: section shows the frequency summary and reports the pin, net, domain, frequency, and frequency source for each pin.

The frequency summary can be filtered by:

- **Limit the number of reported pins to**: limits the number of pins reported to the specified number.
- **Report only pins with frequencies greater than**: any pin with a frequency below the selected value is not reported.

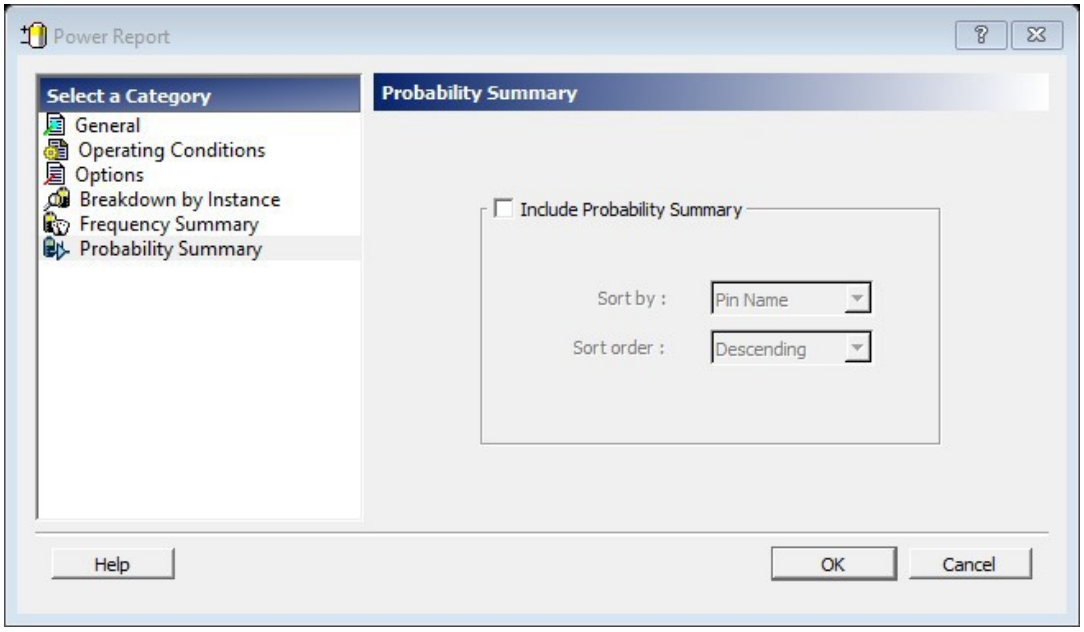
This section can be sorted by selecting the preferred method:

- **Sort by**: sorts by pin name, net name, domain, frequency, or source.
- **Sort order**: sorts by ascending or descending order.

2.19.11. Probability Summary (Ask a Question)

Using the **Probability Summary** panel you can include the probability summary in the report and set specific options.

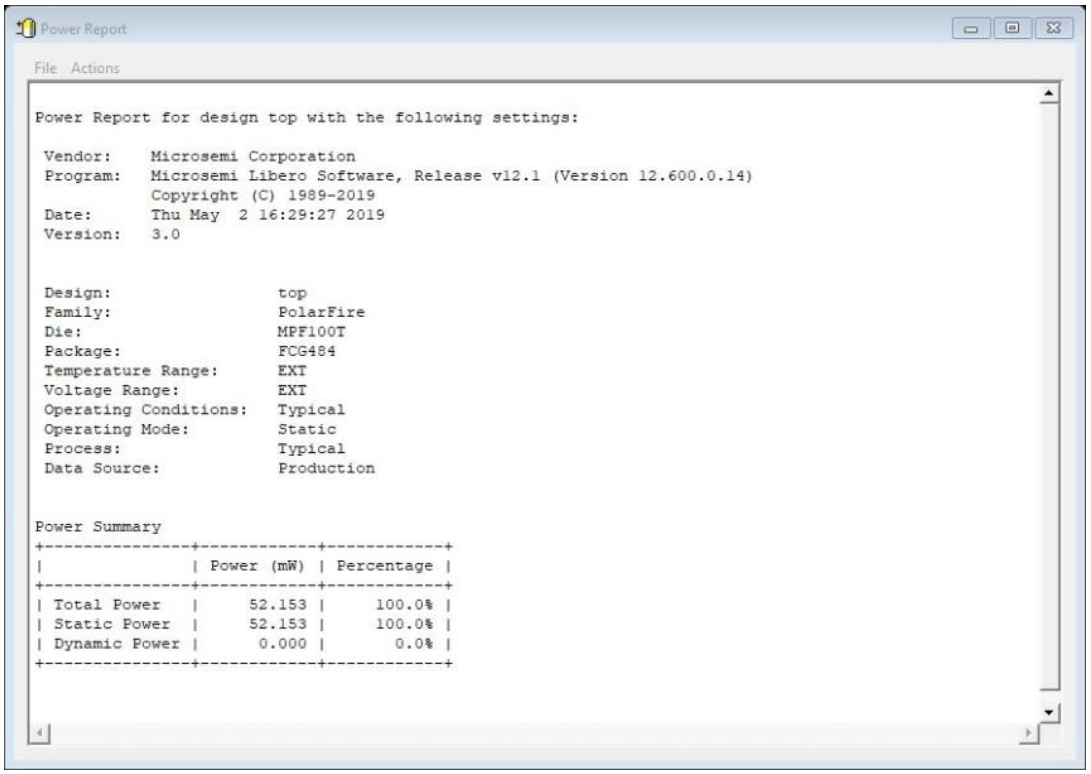
Figure 2-38. SmartPower Power Report Dialog Box - Probability Summary Panel



Include Probability Summary: shows the probability summary and reports the driver, net, rate, source, and type for each pin. This section can be sorted by selecting the preferred method:

- **Sort by**: sorts by pin name, net name, rate, source, or type.
- **Sort order**: sorts by ascending or descending order.

Figure 2-39. SmartPower Report



In addition to the information selected on the Power Reports dialog box, the report contains global design information.

Global design information: shows the target family, the package and the die. It also shows information about the operating conditions, speed grade, and power mode. This option is set by default.

2.20. Scenario Power Report (Ask a Question)

The Scenario Power report allows you to select a previously defined scenario and calculate the average power consumption and the battery life for this scenario. You can generate this report from the **Tools** menu and from within SmartPower.

The Scenario Power Report dialog box is organized into the following panels:

- General
- Operating Conditions
- Options
- Battery life

2.20.1. Generating Scenario Power Reports from the Tools Menu (Ask a Question)

Make sure your design contains one or more scenarios.

To generate a Scenario Power report from the **Tools** menu:

1. From the **Tools** menu, choose **Reports > Power > Scenario Power Report**.
   The Scenario Power Report dialog box appears.
2. Select the options you want to include in the report, and then click **OK**.
   The Scenario Power report appears in a separate window.

2.20.2. Generating Scenario Power Reports from within SmartPower (Ask a Question)

Make sure your design contains one or more scenarios.

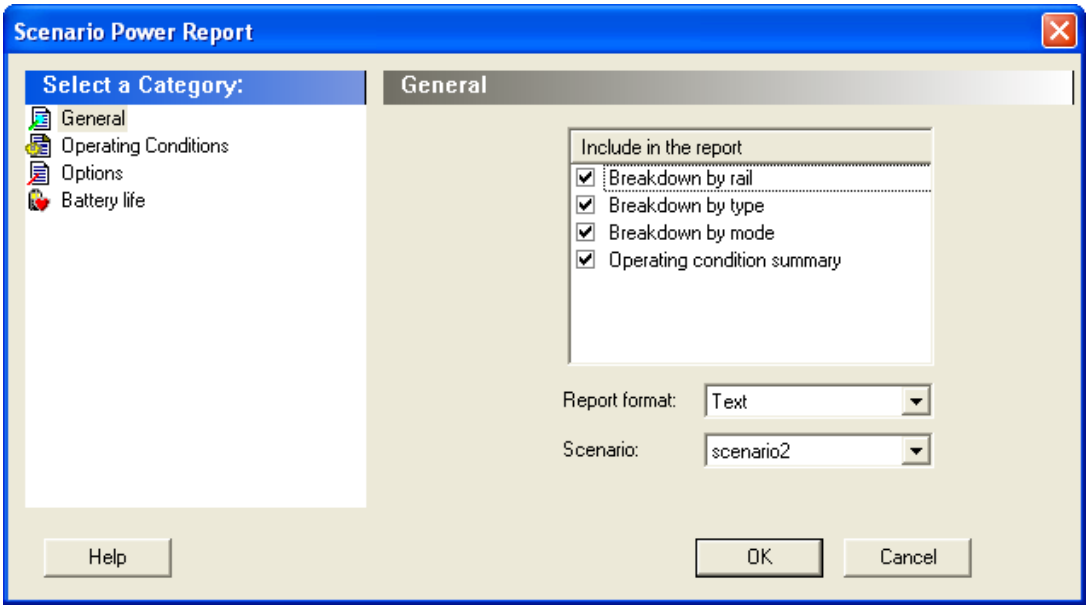
To generate Scenario Power reports from within SmartPower:

1. From the **Tools** menu, choose **Reports > Scenario Power Report** or click the **Scenario Power Report** button to open the Power Scenarios dialog box. By default, the report includes global design information and power sequencer summary.
2. Specify which results you want to display by checking the boxes to be included in the report.

2.20.3. General (Ask a Question)

The **General** panel allows you to select what to include in the report, the report format, and the scenario for which you want to generate the report.

Figure 2-40. SmartPower Scenario Power Report Dialog Box – General Panel



2.20.4. Include in the report (Ask a Question)

Breakdown by rail: shows the power consumption of each rail.

Breakdown by type: reports on the power consumption according to: gates, nets, clocks, core static, I/Os, and memories.

Breakdown by mode: reports on power consumption by mode.

Operating condition summary: reports the operating conditions.

2.20.4.1. Report format (Ask a Question)

Select **Text** or **CSV** (comma-separated value) as the desired export format.

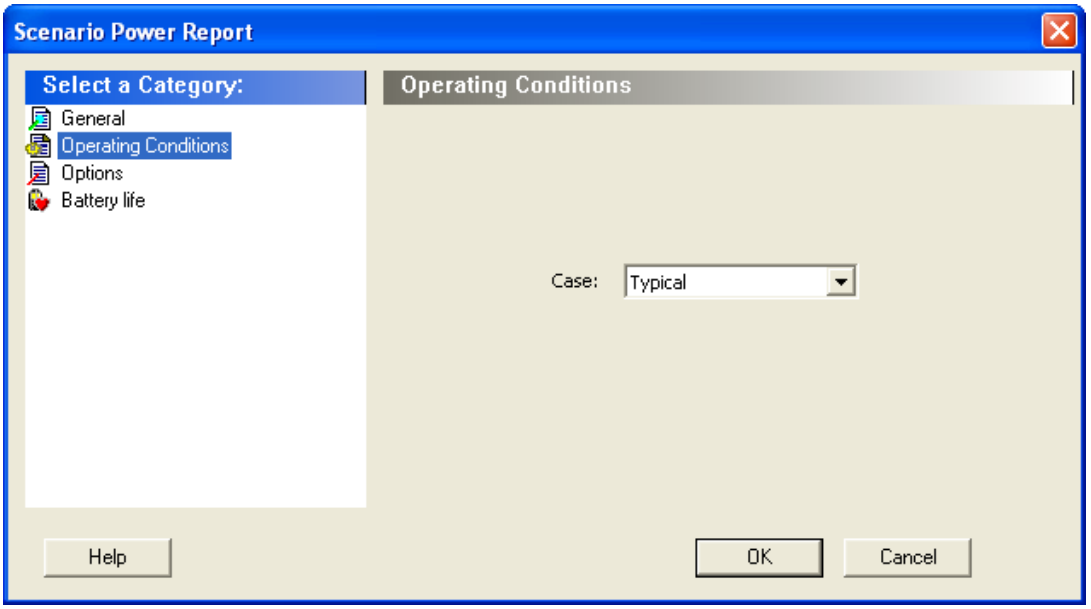
2.20.4.2. Scenario (Ask a Question)

Select a previously defined scenario to generate the report from.

2.20.5. Operating Conditions (Ask a Question)

The **Operating Conditions** panel allows you to select the operating conditions case for the current design.

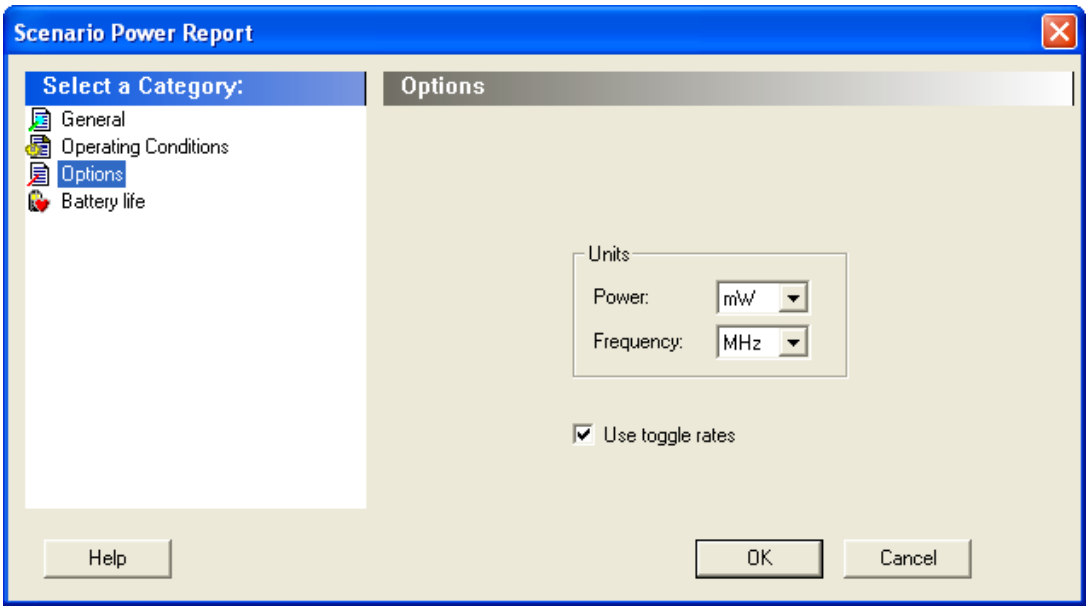
Figure 2-41. SmartPower Scenario Power Report Dialog Box – Operating Conditions Panel



2.20.6. Options (Ask a Question)

The **Options** panel allows you to select power and frequency units and use toggle rates.

Figure 2-42. SmartPower Scenario Power Report Dialog Box – Options Panel



2.20.7. Units (Ask a Question)

Frequency: sets unit preferences for frequency (Hz, KHz, MHz).

Power: sets unit preferences for power (W, mW, or ʊW).

2.20.8. Use Toggle Rates (Ask a Question)

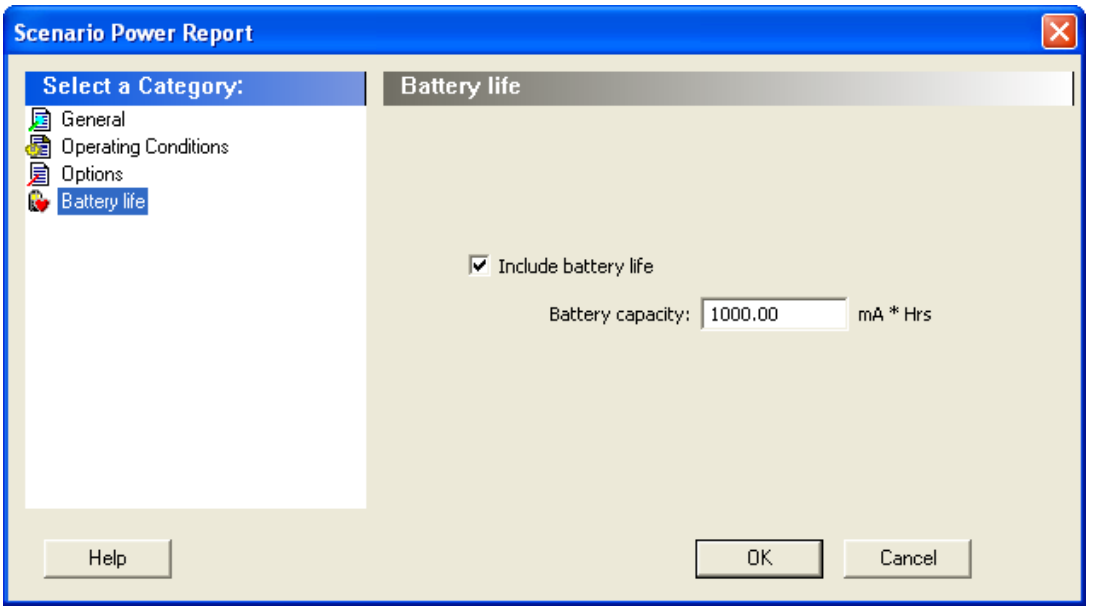
When toggle rates are active (**Use Toggle Rates** box is checked), the data frequency of all the clock domains is defined as a function of the clock frequency. This updates the data frequency automatically when you update the clock frequency. Toggle rates allow you to specify the data

frequency as a percentage of clock frequency; however, you can specify the data frequency only as a percentage of the clock frequency, not as a number. To set the data frequency again, clear the **Use Toggle Rates** option.

2.20.9. Battery life (Ask a Question)

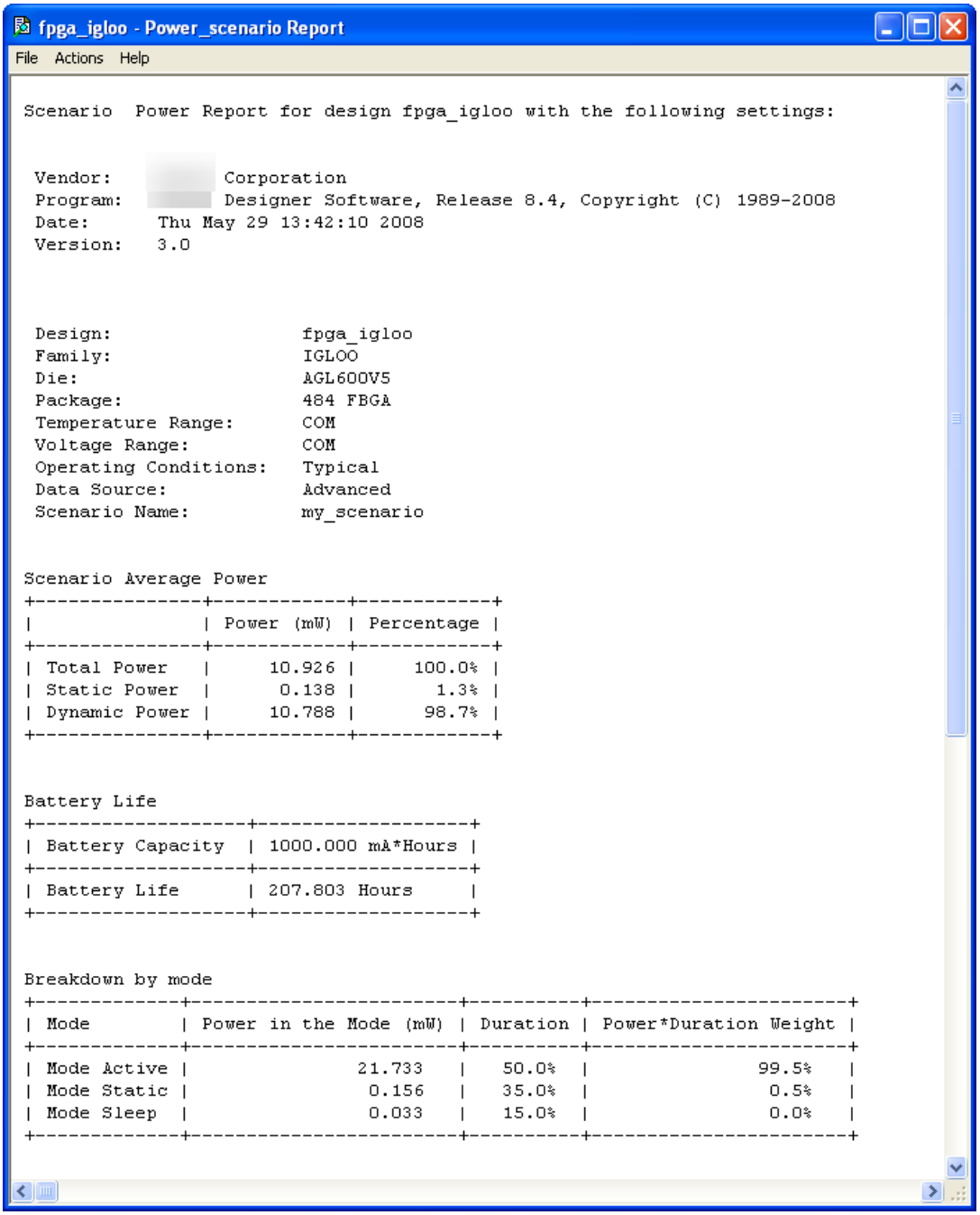
The **Battery life** panel reports the battery capacity and the battery life. Enter a battery capacity in MA/Hrs.

Figure 2-43. SmartPower Scenario Power Report Dialog Box – Battery Life Panel



The SmartPower Scenario Power report returns the average power consumption and battery life for this sequence.

Figure 2-44. SmartPower Scenario Power Report



fpga_igloo - Power_scenario Report

File Actions Help

```
Scenario  Power Report for design fpga_igloo with the following settings:


 Vendor:        Corporation
 Program:       Designer Software, Release 8.4, Copyright (C) 1989-2008
 Date:      Thu May 29 13:42:10 2008
 Version:   3.0



 Design:                  fpga_igloo
 Family:                  IGLOO
 Die:                     AGL600V5
 Package:                 484 FBGA
 Temperature Range:       COM
 Voltage Range:           COM
 Operating Conditions:    Typical
 Data Source:             Advanced
 Scenario Name:           my_scenario


Scenario Average Power
+---------------+------------+------------+
|               | Power (mW) | Percentage |
+---------------+------------+------------+
| Total Power   |     10.926 |     100.0% |
| Static Power  |      0.138 |       1.3% |
| Dynamic Power |     10.788 |      98.7% |
+---------------+------------+------------+


Battery Life
+-------------------+-------------------+
| Battery Capacity  | 1000.000 mA*Hours |
+-------------------+-------------------+
| Battery Life      | 207.803 Hours     |
+-------------------+-------------------+


Breakdown by mode
+-------------+------------------------+----------+-----------------------+
| Mode        | Power in the Mode (mW) | Duration | Power*Duration Weight |
+-------------+------------------------+----------+-----------------------+
| Mode Active |                 21.733 |    50.0% |                 99.5% |
| Mode Static |                  0.156 |    35.0% |                  0.5% |
| Mode Sleep  |                  0.033 |    15.0% |                  0.0% |
+-------------+------------------------+----------+-----------------------+
```

In addition to the information selected in the Scenario Power Reports dialog box, the report contains global design information, a mode summary, and the sequence average power.

Global design information: shows the target family, the package and the die. It also shows information about the operating conditions, speed grade, and power mode. This option is set by default.

Power Summary: reports the power consumption of the sequence by mode. This option is set by default.

Sequence Average Power: reports the average power consumption of the sequence. This option is set by default.

2.21. Activity and Hazards Reports (Ask a Question)

Traditional Power Analysis based on a VCD simulation file reports an average power value that will account for all nets switching in the design. This switching includes functional transitions and spurious transitions. Due to the delay of each gate, paths arriving at one internal gate may have different propagation delays. Therefore, a gate may exhibit multiple spurious transitions before settling to the correct logic level.

The Activity and Hazards Power report allows to identify gates and nets of the design that consume power because of spurious transitions. This is helpful to understand and further minimize power consumption. The activity and hazards report reads a VCD file and reports transitions and hazards for each clock cycle of the VCD file.

The Activity and Hazards report dialog box is organized into the following panels:

- General
- Partial Parsing
- Top-Level Name
- Glitch Filtering
- Clock Domains

2.21.1. Generating Activity and Hazard Reports from the Tools Menu (Ask a Question)

To generate Activity and Hazard reports from the **Tools** menu:

1. From the **Tools** menu, choose **Reports > Power > Activity and Hazards Report**.
   The Activity and Hazards Power Report dialog box appears.
2. Select the options you want to include in the report, and then click **OK**.
   The Activity and Hazards report appears in a separate window.

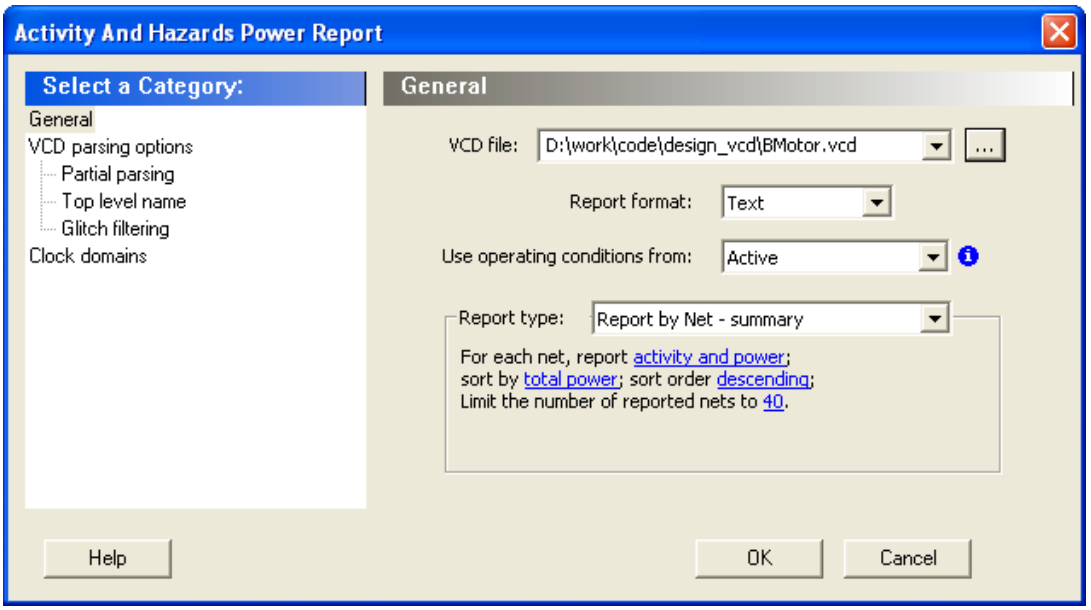
2.21.2. Generating Activity and Hazard Reports from within SmartPower (Ask a Question)

To generate Activity and Hazard reports from within SmartPower:

1. From the **Tools** menu, choose **Reports > Activity and Hazards Report**, or select a VCD file from the Modes and Scenarios toolbar and from the right-click menu, select **Tools > Power Activity and Hazards**.

2.21.3. General (Ask a Question)

Figure 2-45. SmartPower Activity and Hazards Report Dialog Box – General



VCD file: select the VCD file you want to import.

Report format: select **Text** or **CSV** (comma-separated value) as the desired export format.

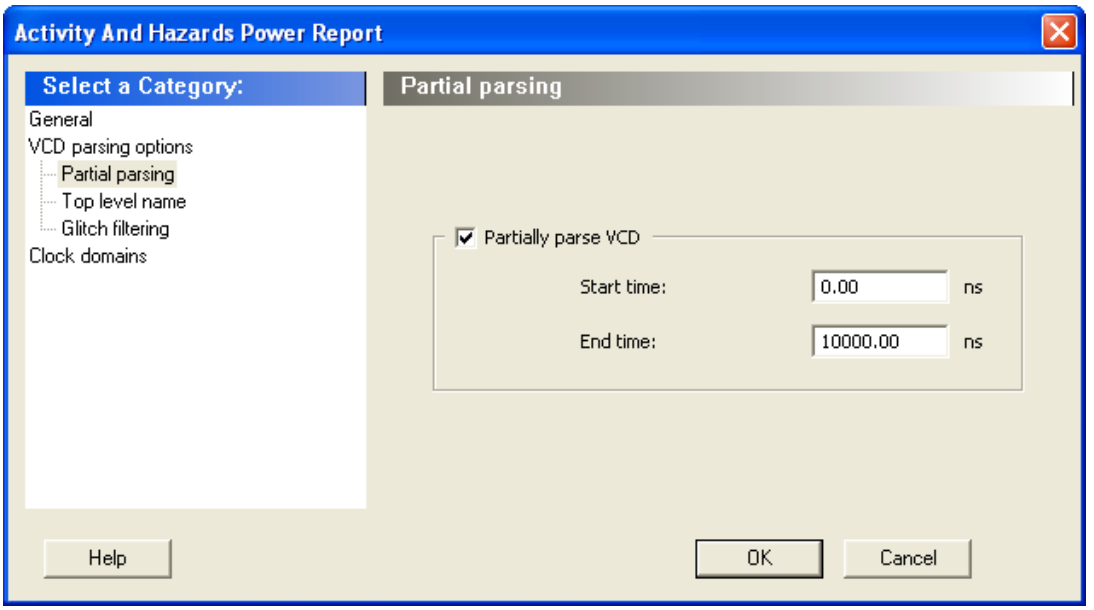
Use operating conditions from: select the mode from which the operating conditions will be used.

Report type: select the report type:

- Report by Net – summary: summary report by net.
- Report by Net – detailed: detailed report by cycle.
- Report by Cycle – summary: summary report by net.
- Report by Cycle – detailed: detailed report by cycle. The selected report type reports activity and power for each net sorted by power in descending order and limits the number of reported nets to 40 by default. To change these options, click each option and from the pop-up menu, select the desired option:
  - Report: select the query report type: activity, power, or activity and power.
  - Sort by: select the query sort by functional power, functional transitions, spurious power, spurious transitions, or total power.
  - Sort order: select the query sort order: ascending or descending.
  - Limit the number of reported nets: enter the query filter limit.

2.21.4. Partial parsing (Ask a Question)

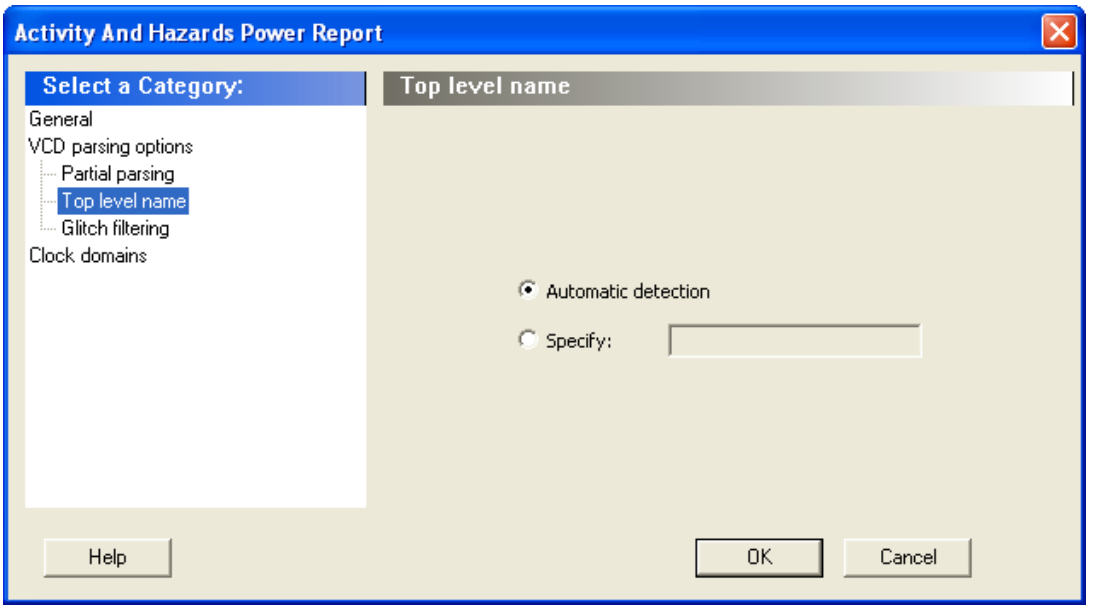
Figure 2-46. SmartPower Activity and Hazards Report Dialog Box – Partial Parsing



Partially parse VCD: specify the **Start time** and **End time** to partially parse the VCD file. This option can be used for large VCD files.

2.21.5. Top level name (Ask a Question)

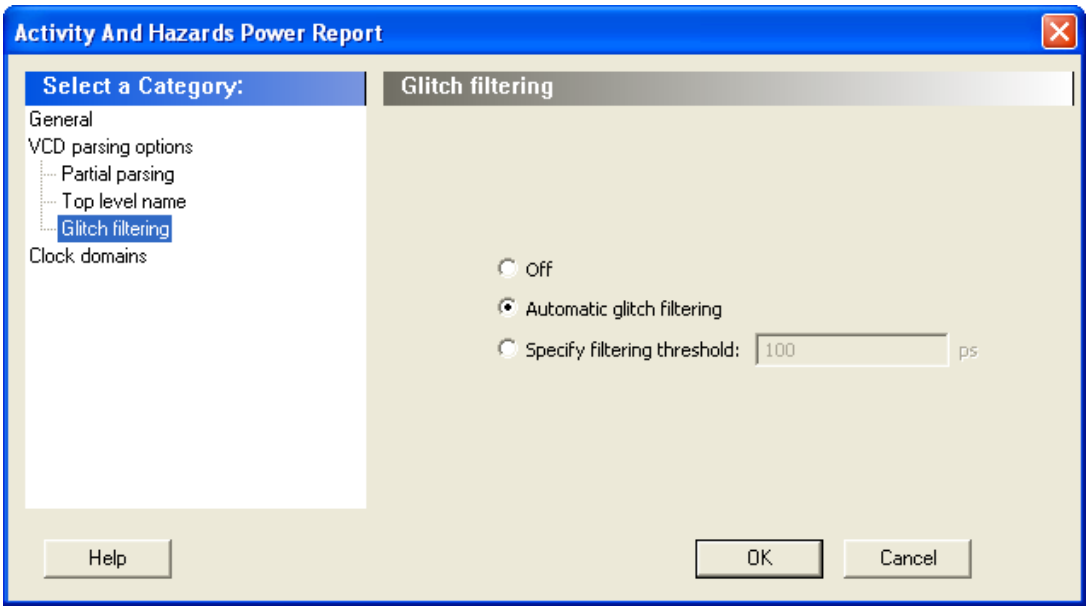
Figure 2-47. SmartPower Activity and Hazards Report Dialog Box – Top-Level Name



This option allows you to select how the top-level name is specified. Select **Automatic detection** to let the VCD reader detect the top-level name of the design automatically or select **Specify** to specify the top-level name manually.

2.21.6. Glitch Filtering (Ask a Question)

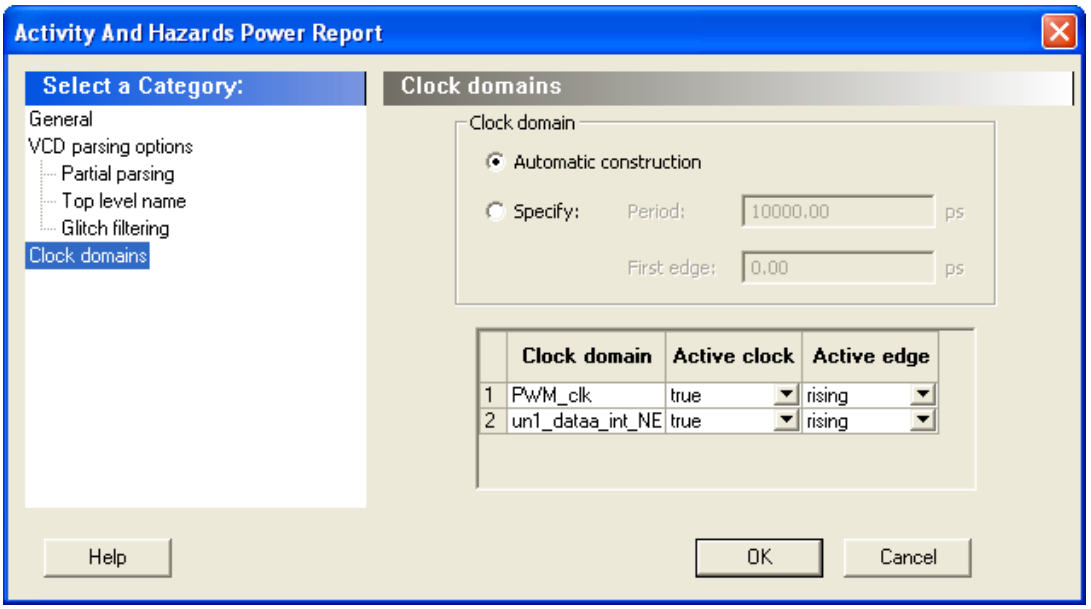
Figure 2-48. SmartPower Activity and Hazards Report Advanced Options Dialog Box – Glitch Filtering



This option allows you to filter out pulses of short durations by selecting **Automatic glitch filtering** or by entering a value in the **Specify filtering threshold** field. The default glitch filtering option is **Automatic glitch filtering**.

2.21.7. Clock Domains (Ask a Question)

Figure 2-49. SmartPower Activity and Hazards Report Advanced Options Dialog Box – Clock Domains



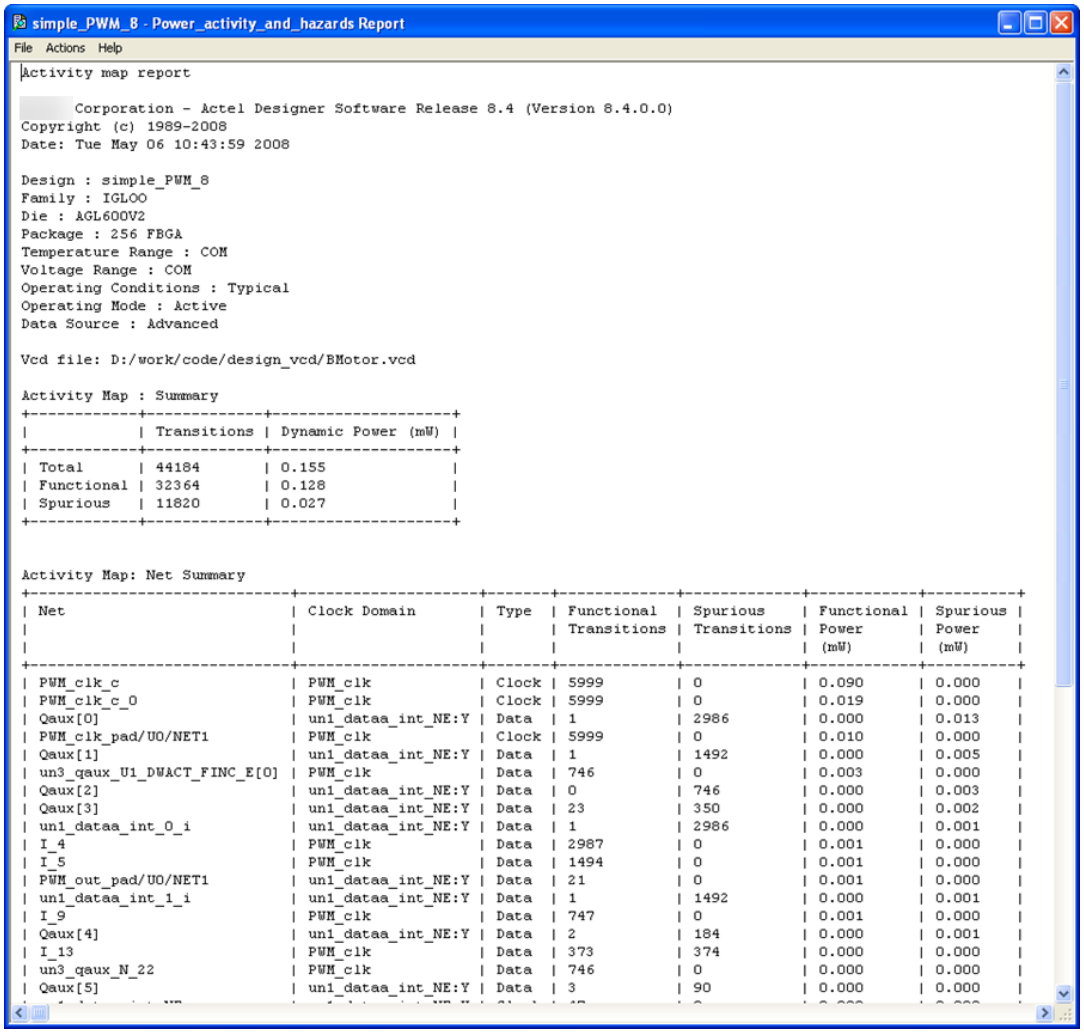
The clock domain can be automatically constructed or specified by you.

Automatic construction: constructs the clock domain automatically. SmartPower analyzes your design automatically to assess if a clock is active and what is the active edge.

Specify: select this option to specify the period and first edge.

Use the clock domain table to set the active edge (rising, falling or both) and to set a clock as transparent. The results are displayed in the following Activity and Hazards Power report.

Figure 2-50. SmartPower Activity and Hazards Report



simple_PWM_8 - Power_activity_and_hazards Report

File Actions Help

```
Activity map report

      Corporation - Actel Designer Software Release 8.4 (Version 8.4.0.0)
Copyright (c) 1989-2008
Date: Tue May 06 10:43:59 2008

Design : simple_PWM_8
Family : IGLOO
Die : AGL600V2
Package : 256 FBGA
Temperature Range : COM
Voltage Range : COM
Operating Conditions : Typical
Operating Mode : Active
Data Source : Advanced

Vcd file: D:/work/code/design_vcd/BMotor.vcd

Activity Map : Summary
+------------+-------------+--------------------+
|            | Transitions | Dynamic Power (mW) |
+------------+-------------+--------------------+
| Total      | 44184       | 0.155              |
| Functional | 32364       | 0.128              |
| Spurious   | 11820       | 0.027              |
+------------+-------------+--------------------+


Activity Map: Net Summary
+-----------------------------+---------------------+-------+-------------+-------------+-------------+----------+
| Net                         | Clock Domain        | Type  | Functional  | Spurious    | Functional  | Spurious |
|                             |                     |       | Transitions | Transitions | Power       | Power    |
|                             |                     |       |             |             | (mW)        | (mW)     |
+-----------------------------+---------------------+-------+-------------+-------------+-------------+----------+
| PWM_clk_c                   | PWM_clk             | Clock | 5999        | 0           | 0.090       | 0.000    |
| PWM_clk_c_0                 | PWM_clk             | Clock | 5999        | 0           | 0.019       | 0.000    |
| Qaux[0]                     | un1_dataa_int_NE:Y  | Data  | 1           | 2986        | 0.000       | 0.013    |
| PWM_clk_pad/U0/NET1         | PWM_clk             | Clock | 5999        | 0           | 0.010       | 0.000    |
| Qaux[1]                     | un1_dataa_int_NE:Y  | Data  | 1           | 1492        | 0.000       | 0.005    |
| un3_qaux_U1_DWACT_FINC_E[0] | PWM_clk             | Data  | 746         | 0           | 0.003       | 0.000    |
| Qaux[2]                     | un1_dataa_int_NE:Y  | Data  | 0           | 746         | 0.000       | 0.003    |
| Qaux[3]                     | un1_dataa_int_NE:Y  | Data  | 23          | 350         | 0.000       | 0.002    |
| un1_dataa_int_0_i           | un1_dataa_int_NE:Y  | Data  | 1           | 2986        | 0.000       | 0.001    |
| I_4                         | PWM_clk             | Data  | 2987        | 0           | 0.001       | 0.000    |
| I_5                         | PWM_clk             | Data  | 1494        | 0           | 0.001       | 0.000    |
| PWM_out_pad/U0/NET1         | un1_dataa_int_NE:Y  | Data  | 21          | 0           | 0.001       | 0.000    |
| un1_dataa_int_1_i           | un1_dataa_int_NE:Y  | Data  | 1           | 1492        | 0.000       | 0.001    |
| I_9                         | PWM_clk             | Data  | 747         | 0           | 0.001       | 0.000    |
| Qaux[4]                     | un1_dataa_int_NE:Y  | Data  | 2           | 184         | 0.000       | 0.001    |
| I_13                        | PWM_clk             | Data  | 373         | 374         | 0.000       | 0.000    |
| un3_qaux_N_22               | PWM_clk             | Data  | 746         | 0           | 0.000       | 0.000    |
| Qaux[5]                     | un1_dataa_int_NE:Y  | Data  | 3           | 90          | 0.000       | 0.000    |
```

2.22. Export Report for Microchip Power Estimator (MPE) (Ask a Question)

This option exports an .xml report from SmartPower, which can be imported into the MPE tool. To export the MPE report, go to **Tools > Export Report for MPE.**

The generated .xml report contains the following information:

- Device Settings
- Thermal Settings
- Voltage Source
- Clocks
- Logic Breakdown
- LSRAM Breakdown with Advanced Settings
- uSRAM Breakdown with Advanced Settings
- Math Breakdown with Advanced Settings

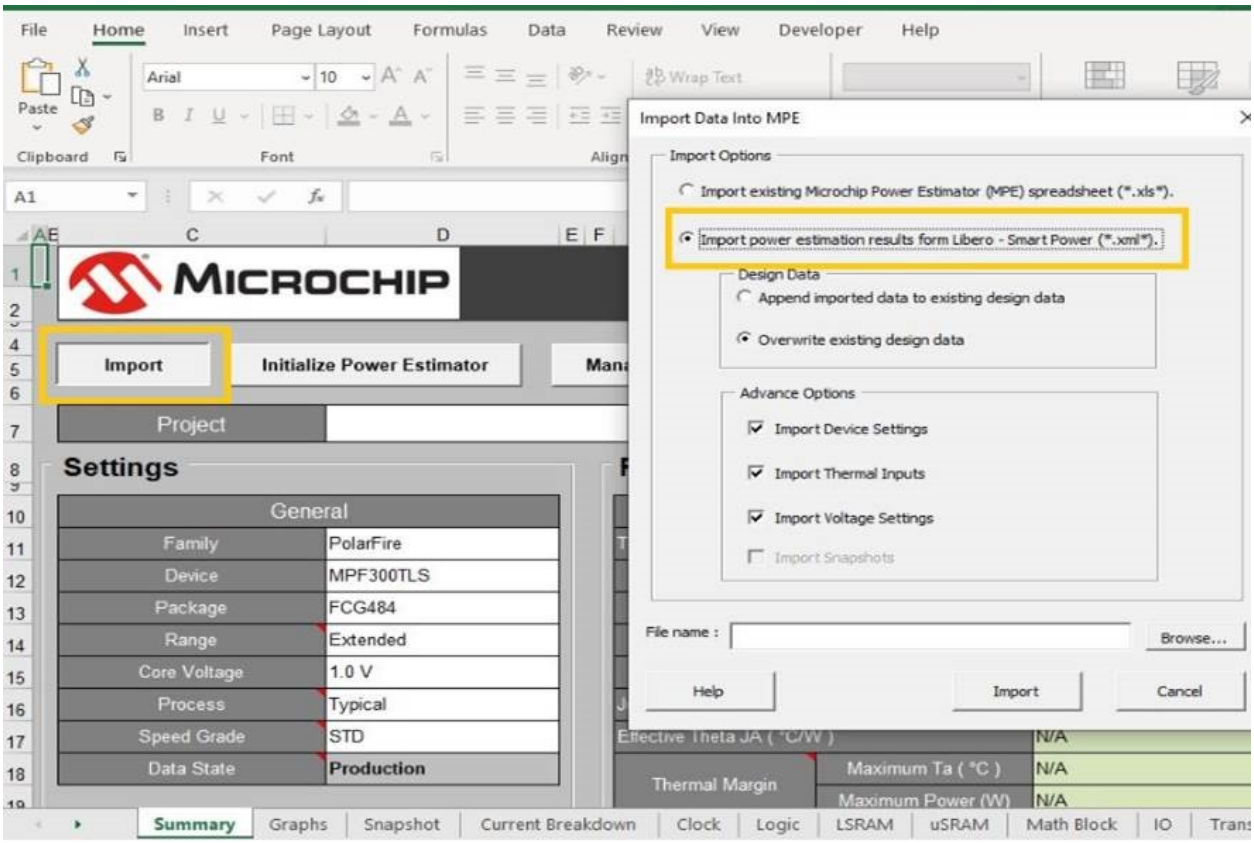
- PLL and DLL
- I/Os
- Crypto
- Transceivers

In addition, the following information is available for PolarFire SoC devices:

- MSS RISC-V (Quad U54)
- AXI MSS/Fabric Interfaces
- MDDR
- MSS I/O Interfaces
- User Crypto

MPE has an import feature, which allows the user to import either the existing MPE spreadsheet (.xls format) or Power estimation results from Libero® SoC SmartPower (.xml format), as shown in the following figure.

Figure 2-51. MPE with Import Data

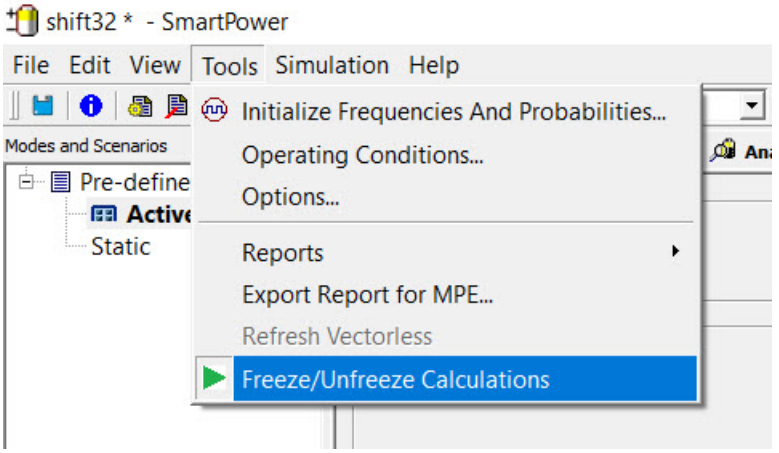


2.23. Freeze/Unfreeze Calculations (Ask a Question)

Freeze/Unfreeze calculation option allows a more responsive User Interface and avoids runtime slowness due to a re-computation that occurs on every value changes. A toggle play pause button is provided to freeze the computation until the user is ready to re-compute. Only when the user clicks the play button, all the changes done will be computed and the User Interface gets updated.

To Freeze/Unfreeze Calculations, go to **Tools > Freeze/Unfreeze Calculations**.

Figure 2-52. Freeze/Unfreeze Calculations



2.24. Importing a VCD File (Ask a Question)

The VCD file is a simulation file. The format of this file is specified in the IEEE® 1364 standard.

You can configure the Project Manager to generate VCD files automatically using QuestaSim. You can also generate a VCD file with a QuestaSim simulator using the following commands:

```
vcd file example.vcd
vcd add -r /testbench/<top>_0/*
run 1 us
vcd flush
```

This example creates a VCD file called `example.vcd`, adds all signals recursively, runs the simulation for one microsecond, and quits. You must quit QuestaSim to obtain an accurate result from SmartPower.

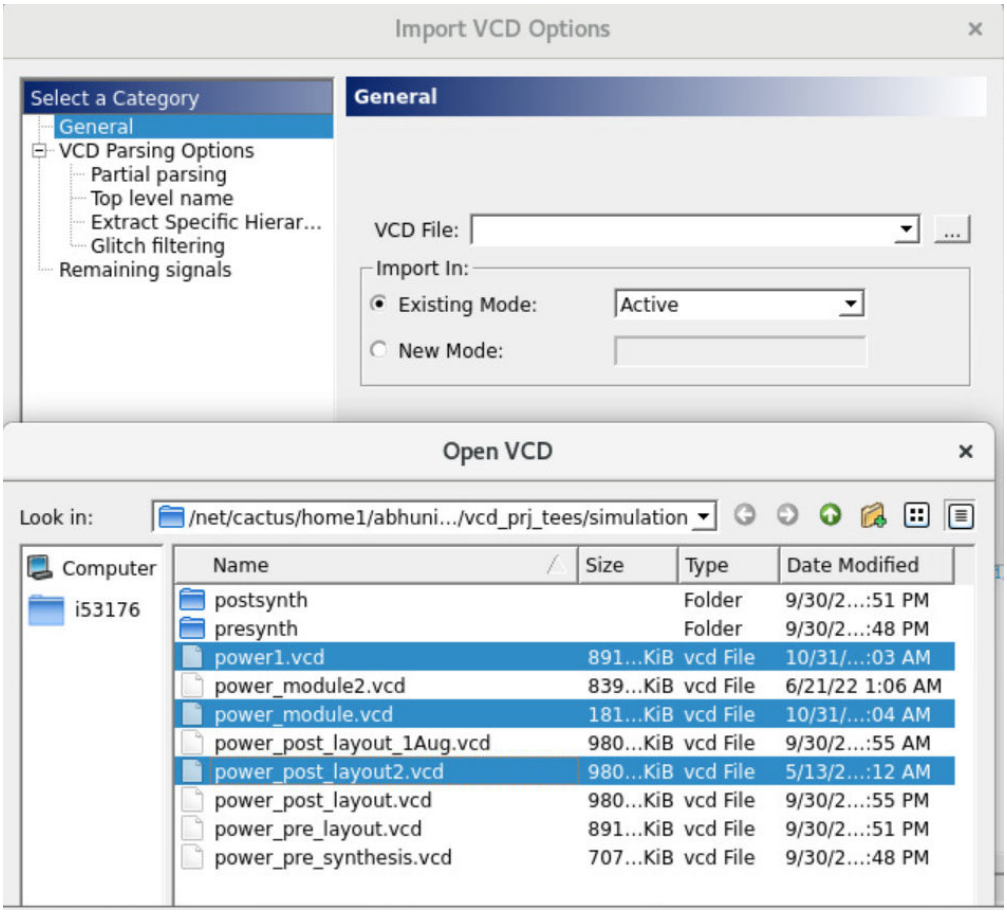
If you have not completed the layout of the design, the design software guides you through Place-and-Route. Hence, you can import the VCD file. To annotate your VCD values to the design, complete Place-and-Route even if you had generated VCD files using timing simulation (pre-layout).
Note: SmartPower is validated with the QuestaSim generated VCD files. However, you might use any Verilog/VHDL simulator that offers a VCD dump feature.

For more information about how to generate a VCD file, see the user manual for your simulation tool.

To Import VCD files in SmartPower and access the available importing options, perform the following steps:

1. From the **Simulation** menu, choose **Import VCD File**.
   The Import VCD Options dialog box appears.

Figure 2-53. Import VCD File Dialog Box

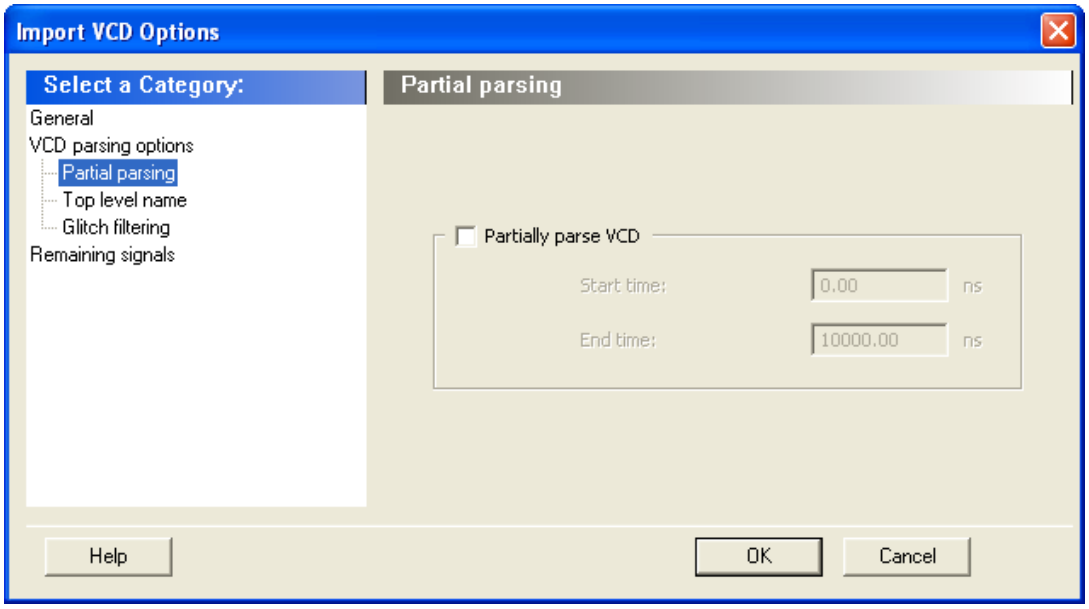


2. Select the VCD file that you want to import and select a mode to import it in, or click **New mode** and enter a mode name.

Important: The user can select multiple VCD files from the same directory. This will also help the user to import multiple VCDs for different blocks from different designs, and multiple VCDs for different modules for the same design.

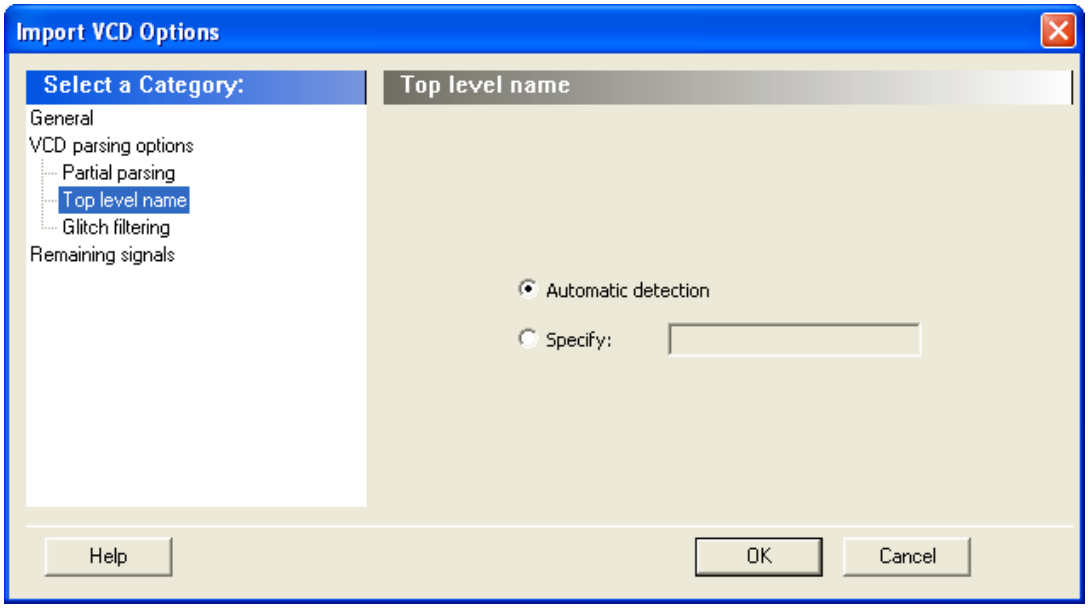
3. Select the options you want to specify:
   - **Partial parsing**: Specify the **Start time** and **End time** to partially parse the VCD file. This option can be used for large VCD files.

Figure 2-54. Import VCD File Dialog Box—Partial Parsing



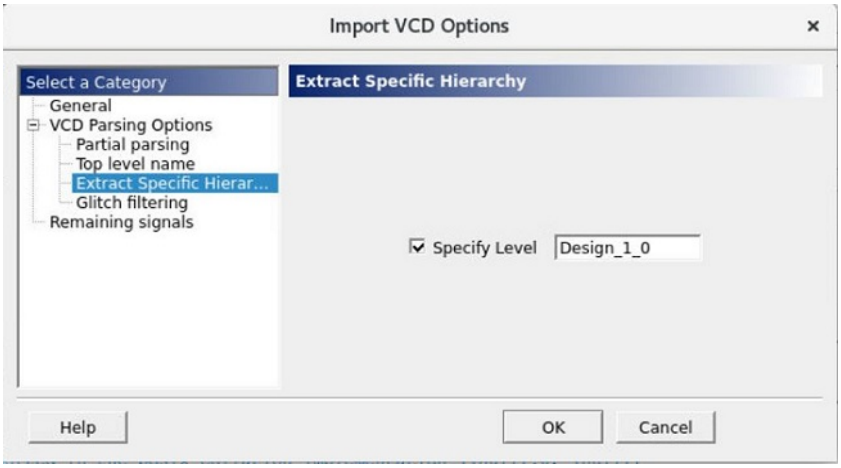
- **Top level name**: Allows you to select how the top-level name is specified. The **Top level name** is the instance name of your design instantiated in the simulation testbench. Select **Automatic detection** to let the VCD reader detect the top-level name of the design automatically or select **Specify** to specify the top-level name manually.

Figure 2-55. Import VCD File Dialog Box—Top-Level Name



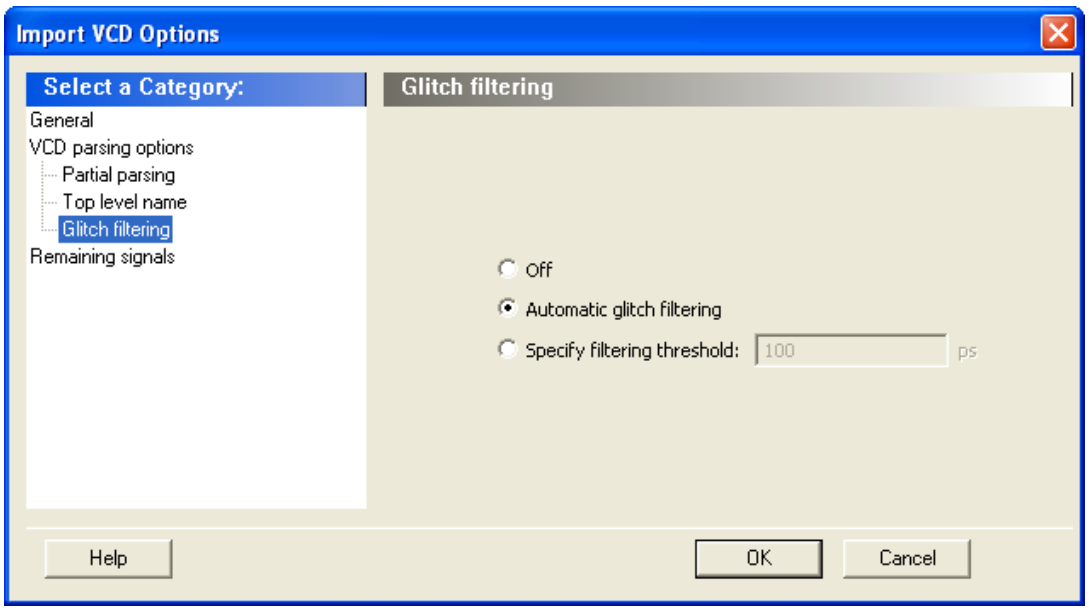
- **Extract Specific Hierarchy**: Allows you to specify the hierarchy level of the design to be annotated.
  Note: Currently, this option is supported for VCD generated for full design and not for a block/module VCD.

Figure 2-56. Import VCD File Dialog Box—Extract Specific Hierarchy



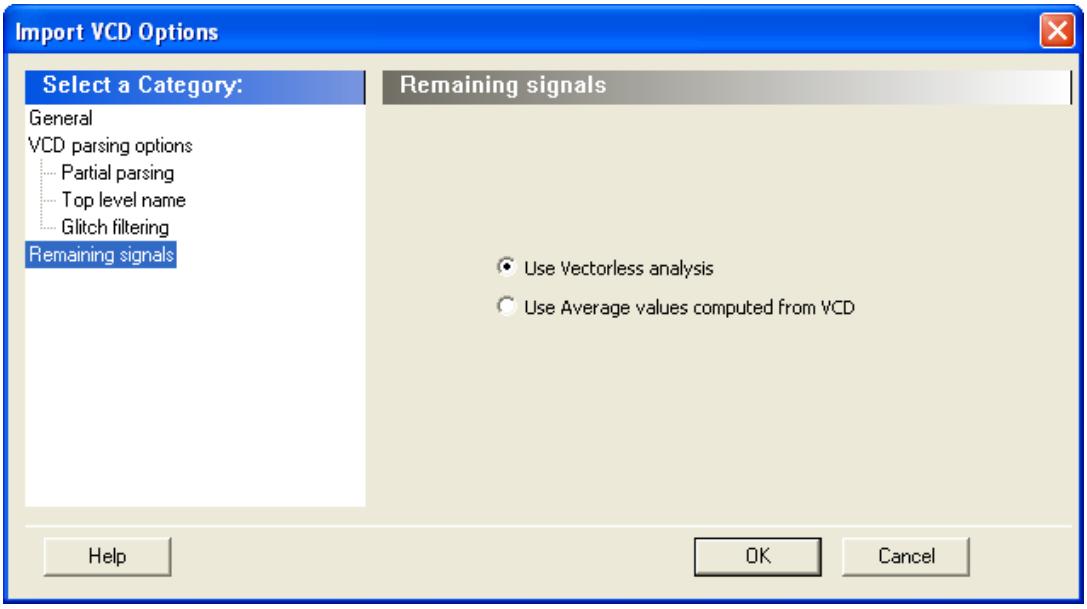
- **Glitch filtering**: Allows you to filter pulses of short durations by selecting **Automatic glitch filtering** or by entering a value in the **Specify filtering threshold** field. The default option is **Automatic glitch filtering**.

Figure 2-57. Import VCD File Dialog Box — Glitch Filtering



- **Remaining signals**: A few signals might not be annotated by the VCD reader. It generally happens with VCD files created from a behavioral or a post-synthesis simulation. For those signals, you can either complete the annotation with the vectorless analysis or with the average values computed from the VCD file.

Figure 2-58. Import VCD File Dialog Box—Remaining Signals



4. Click **OK**.
   When the VCD file is imported successfully, the file appears under the Imported mode in the Modes and Scenarios toolbar, and the following messages appear in the Designer Log window:

```
Info:
VCD:34 glitch(es) filtered with 1000 ps threshold
```

   The preceding message reports the number of glitches that are filtered by the VCD reader.

```
Info:
VCD: Annotation Statistics Percentage of Annotated Pins:100.00 % Percentage of Unannotated
Pins:0.00 %
Percentage of Annotated Pins with Zero Frequency: 25.99 %
```

   If the percentage of annotated pins is less than 50.00%, a warning message is generated.

   If the percentage of annotated pins is low, you might want to verify that signals at all levels of hierarchy are added recursively (for example, using `vcd add -r` in QuestaSim).

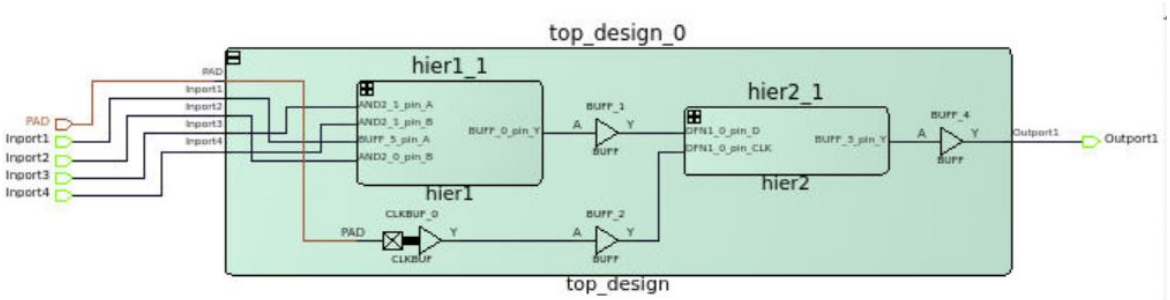
   If you simulate a pre-synthesis netlist or a post-synthesis RTL netlist, it is possible to get a low percentage of annotated pins. This happens because not all logic elements are in the pre-synthesis netlist and the post-synthesis RTL netlist. For accurate power estimation, it is best to run post-layout simulation with a back-annotated netlist.

   To see which pins are not annotated, open the SmartPower **Frequencies** tab. If your file is imported successfully, a list of pins with annotated individual frequencies, displayed with **VCD Import,** appear as a source. The unannotated pins are displayed with **Default Estimation** as source. If your design has enabled pins, open the SmartPower **Probabilities** tab.

   Another feature enables to annotate different parts of the hierarchy depending on the instructions.

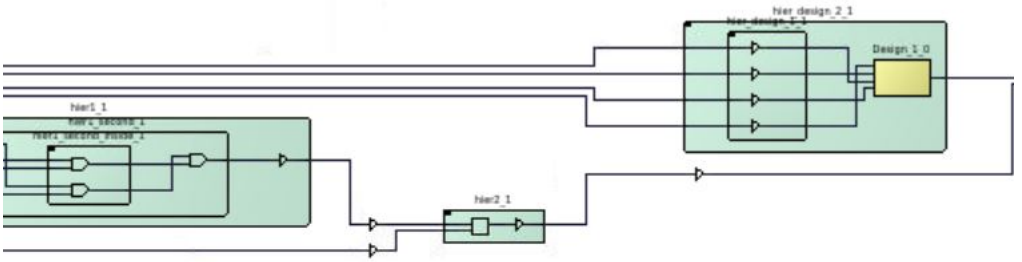
   - When a VCD is generated at a lower level of hierarchy inside the design, import the VCD and annotate the level of hierarchy present in the VCD.

Figure 2-59. VCD Generated for hier2_1 for a Testbench



- The design contains a block and the VCD is generated for that block, only for the annotated block.

Figure 2-60. VCD Generated for Block Design_1_0 (Colored in Yellow)



After completing the import VCD process, you can create a custom mode based on the active mode. This custom mode inherits all the clock and toggle frequencies of the active mode you set through the VCD import. This final step is optional. It gives you the flexibility to modify the active mode frequencies while saving the VCD scenario in SmartPower. For more information, see Custom Operating Modes.

2.25. Removing a VCD File (Ask a Question)

The VCD file is a simulation file whose format conforms to the IEEE 1364 standard.

To remove a VCD file, perform one of the following steps:

- Select the VCD file from Active mode or a custom mode, right-click the VCD file and select **Remove VCD File from <mode name> Mode**.
- From the **Simulation** menu, choose **Remove VCD File > <file name>**
  Note: The annotation is removed when you remove the VCD file.

2.26. Auditing Files (Ask a Question)

When you start SmartPower, it performs an audit automatically. During the audit, SmartPower checks the time stamps of VCD files and reports any missing or out-of-date files. You can also initiate a file audit manually from the **Simulation** menu by choosing **Audit Files**. After an audit is run, an icon displaying the file status appears in the Modes and Scenarios toolbar next to the VCD file. Mousing over the icon displays more information.

Figure 2-61. Audit Files Results

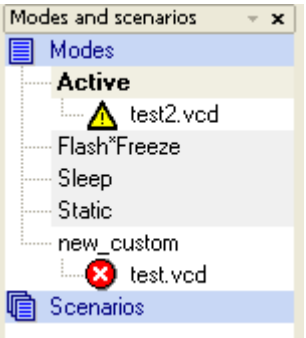


Table 2-5. Audit Files Icons and Descriptions

Icon	Description
	Indicates that the VCD file has been modified
	Indicates that the VCD file is missing

To re-import the VCD file, right-click the VCD file and select **Re-import VCD File**. Then use the Import VCD Options dialog box to re-import the file.

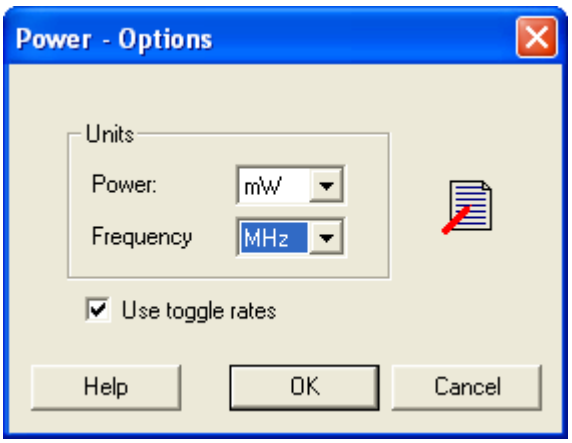
3. Power Options (Ask a Question)

Allows you to set options that affect graphical and textual reports.

To open the SmartPower Power Options dialog box:

1. From the **File** menu, choose **Options**.
2. Select the **Options** panel in the Power Report dialog box.
3. Click the  icon in the SmartPower toolbar.

Figure 3-1. SmartPower Preferences Dialog Box

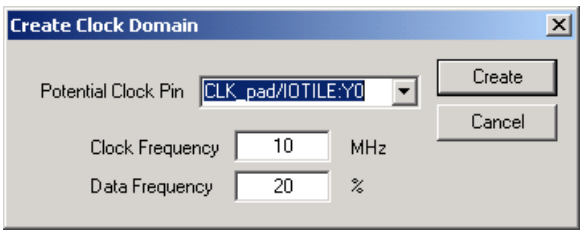


The dialog box has two sections: **Units** and **Use Toggle Rates**.

- **Units**: sets unit preferences for power (W, mW, and uW) and frequency (Hz, KHz, and MHz).
- **Use toggle rates**: when checked, the data frequency of all the clock domains is defined as a function of the clock frequency. This updates the data frequency automatically when you update the clock frequency. Toggle rates allow you to specify the data frequency as a percentage of clock frequency, but you can no longer specify the data frequency as a number, only as a percentage of the clock frequency. To specify data frequencies, clear the **Use toggle rates** option in the Preferences window.

You can set the data frequency percentage when you create a new clock domain with toggle rates active. In addition, when toggle rates are active you can set the data frequency percentage in the **Domain and Frequencies** tabs.

Figure 3-2. Create Clock Domain – Toggle Rates Enabled



4. Calculating Power (Ask a Question)

4.1. Calculating Power (Ask a Question)

Use the following steps to calculate the power consumption of your design. The list of screens and steps appear in the order in which you should view them to analyze your power accurately.

1. **Summary** tab: view global power at the design level and view its impact on junction temperature.
2. **Domains** tab: define clock domains and specify a clock frequency and a data frequency for each clock domain.
3. **Analysis** tab (optional): view detailed hierarchical analysis of your power consumption. If your power consumption exceeds your budget, this step helps you understand where there is room for improvement.
4. **Frequencies** tab (optional): specify individual pin frequencies. This step provides pin-by-pin control of the frequency.
5. **Probabilities** tab (optional): specify probabilities. You can use the default probabilities or set your own.

4.2. Extracting Power Consumption of a Specific Clock Domain (Ask a Question)

To calculate the power consumption of a single clock domain (clock tree and data path) in a design:

1. Set SmartPower in toggle-rate mode. This is the default setting. To confirm that SmartPower is in toggle-rate mode, choose **Preferences** from the **File** menu. Select the **Use Toggle Rates** check box.
2. In the **Domains** tab, set all the **Clock Frequencies** to zero except for the clock domain for which you are calculating the power consumption.

4.3. Advanced Analysis of I/Os (Ask a Question)

SmartPower allows you to estimate the power consumption related to I/Os. You can then analyze power consumption in detail.

To display I/O power consumption:

1. In the **Summary** tab, select **By Type** from the **Breakdown** drop-down list. The grid displays the total power consumption of all I/Os in the design, as well as all other components. If you use different I/O standards, and different Vcci power rails, you can also display the power consumption for each rail by selecting **By Rail** from the **Breakdown** drop-down list. **Breakdown By Rail** shows the power consumption and current for each Vcci rail in the design.
2. In the **Analysis** tab, select the highest hierarchy level in the Hierarchy tree in the left pane. Select **By Instance** from the Breakdown drop-down list.
3. In the **Analysis** tab, unselect all categories under **Instances Contributions** except I/Os, and then click **Apply**. The table then displays a list of all I/Os in the design, along with the power contribution of each individual I/O. The columns display important I/O attributes: external port name, load, standard, drive- strength, slew, and macro.
4. In the **Probabilities** tab, you can control the output probability of each tri-state and bidirectional I/O. The probability is the percentage of time that the I/O is used as an output. The default value can be changed and a specific value can also be set for each bidirectional or tri-state I/O.

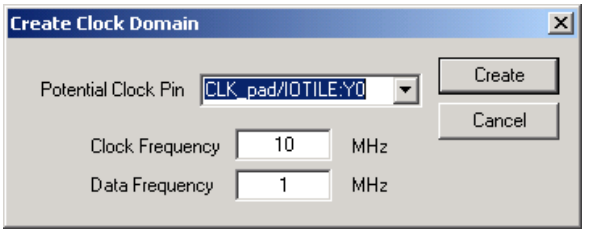
4.4. Adding a New Clock Domain (Ask a Question)

When you run SmartPower, it researches your existing clock domains and partitions your design automatically.

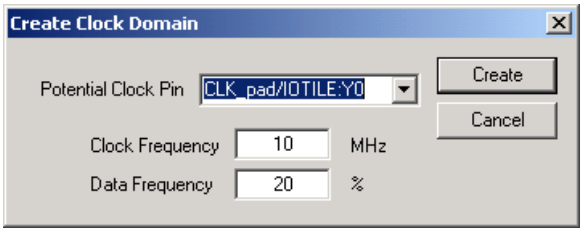
You may want to review the list of clock domains in the **Domains** tab to ensure that all the clocks of your design are included in the list. Add or remove clocks as necessary.
To add a new clock domain:

1. Click the **Domains** tab, and then click the **Add Domain** button.
2. Select **Clock Domain** from the drop-down list.
   The Create Clock Domain dialog box appears.

Figure 4-1. Create Clock Domain Dialog Box – Toggle Rates Disabled



3. To create a new clock, select a **Potential Clock Pin**, specify a clock and data frequency and click **Create**.
   The new clock domain appears in the Domains window. If you select an existing clock pin from the drop- down list, the lists of clock pins and data pins of this new clock domain are computed automatically based on the netlist topology.
   Note: Select **Use Toggle Rates** in the SmartPower Preferences to define your data frequency as a percentage of your clock frequency. If your data frequency is 20% of your clock frequency, type **20** in the **Data Frequency** text box.



   If you want to create an empty clock domain and fill the lists of clock-pins and data pins manually, do not select a clock pin. Instead, type a new name for your clock domain.

4.5. Verifying Lists of Clock and Data Pins of a Clock Domain (Ask a Question)

Beyond the verification of the list of clock domains, you may also wish to verify that the lists of clock pins and data pins computed for each clock domain are correct.

To verify the lists of clock pins and data pins of a clock domain:

1. Click the **Domains** tab and select a specific **Domain** in the list.
2. **Display the list of clock pins or data pins of this Domain.** A drop-down list in the **Domains** tab allows you to select clock pins or data pins. SmartPower displays the list of pins corresponding to your selection below the drop-down list. You can add or remove clock pins and data pins as necessary.

3. **Remove a pin from a clock domain.** Highlight the selected pin and click the **Remove** button. The pin is removed from the clock domain and is made available in the list of pins that you can add in another clock domain.
4. Highlight the selected pin in the list of pins that are not yet in a domain and click the **Add** button to add a pin in a clock domain.
   This pin is added to the clock domain. The pin is a clock or data pin, depending on the specification of the drop-down list when you clicked the **Add** button.
   Note: You cannot add a pin that exists in another domain until you free it from its existing domain. The pin is unavailable until you remove it from that domain.

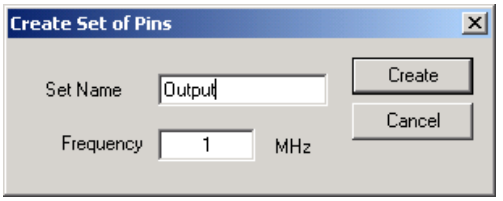
   After you verify that all the clocks of your designs are correctly identified and constructed, specify the correct clock and data frequency for each clock domain.

4.6. Adding a New Set of Pins (Ask a Question)

To add a new set of pins:

1. Open the Create Clock Domain dialog box.
2. Click the **Domains** tab and click the **Add Domain** button.
3. Select **Set of Pins** from the drop-down list.
4. **Create a Set of Pins.** Name your new set of pins, specify a data frequency, and click **Create**

Figure 4-2. Create Set of Pins Dialog Box



The new set of pins appears in the Domains window.

4.7. Specifying Clock and Data Frequencies in SmartPower (Ask a Question)

To specify a clock and data frequency:

1. Click the **Domains** tab.
2. Select the **Clock/Data** frequency cell and enter a new value.

   SmartPower defaults to 0 MHz for the clock frequency. Import your clock frequency from SmartTime by using the Initialize Frequencies dialog box in SmartPower or input your target for each clock frequency in the **Domains** tab. You must also specify a data frequency (5% of your clock frequency is a typical guideline for your data frequency—this corresponds to a toggle rate of 10%).

   Not all the pins/gates/nets of your design are associated with a specific clock. For example, the frequency of a design input port is not always correlated to a clock frequency. By extension, all pins that are upstream of the first level of sequential elements are not associated with any clock. SmartPower creates an InputSet by default that it uses to group all the pins that are controlled by design inputs (instead of sequential elements). You may want to view and verify the InputSet to further evaluate your design.

4.8. Verifying the InputSet (Ask a Question)

To verify the InputSet:

1. Click the **Domains** tab and select the domain named **InputSet** in the list.
2. Verify the list of pins of this domain. All the input ports of your design (except the clocks) belong in the **InputSet**. All pins between these input ports and the first level of sequential elements belong in the **InputSet**. You can add or remove pins as necessary.
3. Specify an average input frequency. SmartPower uses the same frequency for all pins of the InputSet. The default InputSet frequency is 1 MHz. Enter a new value to change it.

4.9. Splitting the InputSet (Ask a Question)

You might want to split the InputSet into several sets in order to specify different frequencies. For example, you might want to create a ResetSet—a reset tree with a very low frequency.

To split the InputSet:

1. Create a new set of pins: In the **Domains** tab, click the **New** button, and select **Set of Pins** from the drop- down menu.
2. In the Create Set of Pins dialog box, type a name and a frequency for the new set and click **Create**. The new set of pins appears in the Domains window. You can only create an empty set of pins, but it is possible to add pins to this domain later.
3. Remove a group of pins from the **InputSet**. Click the **Domains** tab and select the domain named **InputSet** in the list.
4. Select the pins that you want to remove and click the **Remove** button.

4.10. Specifying Individual Pin Frequencies (Ask a Question)

The **Frequencies** tab allows you to specify an average clock and data frequency for each clock domain, and also an average frequency for each set of pins. This gives you an initial estimate of the power consumption of your design. However, if this estimate is not sufficiently accurate, you can refine it with a pin-by-pin annotation of the frequency.

The following procedure describes how to set the frequency of an individual pin. Performing this procedure does not remove the pin from its clock domain. A frequency annotation merely overrides the domain level frequency.

1. Locate the pin in the **Frequencies** tab. You may need to select different clock domains from the drop-down list on the **Frequencies** tab. You can use filters to facilitate this search.
2. Select the pin(s) in the grid and click the **Set frequencies for selected pins** button.
3. Enter a new frequency value and click **OK**. This specifies a new frequency for the selected pin(s).

4.11. Changing the Frequency Annotation of a Pin (Ask a Question)

The following procedure describes how to change the frequency annotation of an individual pin. This may be useful when you import a VCD (value change-dump) file (see).

1. Locate the pin in the **Frequencies** tab**.** You may need to select different clock domains from the drop-down list on the **Frequencies** tab. You can use filters to facilitate the search.
2. Select the pin(s) in the grid and click the **Set frequency for selected pins** button.
3. Enter a new frequency value and click **OK**. This specifies a new frequency for the selected pin(s).

4.12. Removing the Frequency Annotation of a Pin (Ask a Question)

This procedure describes how to remove the frequency annotation of an individual pin. This may be useful when you import a VCD (value change-dump) file.

1. Locate the pin in the **Frequencies** tab. You may need to select different clock domains from the drop-down menu on the **Frequencies** tab. You can use filters to facilitate the search.

2. Select the pin(s) in the grid and click the **Reset frequencies for selected pins** button. This removes the specified frequency from the selected pins.

4.13. Specifying Individual Pin Probabilities (Ask a Question)

The **Probabilities** tab allows you to specify the default output probability value for memory blocks, and tristate and bidirectional I/Os. In addition, you can increase the accuracy of the power estimation by annotating the probability of specific pins.

To specify or change the probability for one or multiple pins:

1. Locate the pin(s) in the **Probabilities** tab. You can use filters to facilitate this search.
2. Select the pin(s) and click the **Set probabilities for selected pins** button.
3. Enter a new probability value and click **OK**.

4.13.1. Specifying or Changing Probability for One or More Pins (Ask a Question)

To specify or change the probability for one or multiple pins:

1. Locate the pin(s) in the **Probabilities** tab. You can use filters to facilitate this search.
2. Select the pin(s) and click the **Set probabilities for selected pins** button.
3. Enter a new probability value and click **OK**.

4.13.2. Resetting the Probability for One or More Pins (Ask a Question)

To reset the probability for one or multiple pins:

1. Locate the pin(s) in the **Probabilities** tab. You can use filters to facilitate this search.
2. Select the pin(s) and click the **Reset Probabilities for selected pins**button.

4.14. Vectorless Estimation (Ask a Question)

The Vectorless estimation method is an accurate method of annotating individual pins with frequencies and probabilities. It uses primary inputs, clock frequencies, and nets annotated with other methods as a starting point, and uses a Monte Carlo simulation to annotate all nets in the design with frequencies and probabilities.

Vectorless estimation is available for SmartFusion 2, IGLOO 2, RTG 4, PolarFire and PolarFire SoC families of devices.

Vectorless estimation can be enabled from the Initialize Frequencies and Probabilities dialog box or from the VCD Import dialog box.

Changes to probabilities or frequencies in the Probabilities, Frequencies, or Domains tabs result in an out-of-date vectorless estimation. To update vectorless estimation, press the refresh vectorless  button.

4.15. Viewing Results (Design Level) (Ask a Question)

Click the **Summary** tab to view global power consumption at the design level. The **Summary** tab shows your design's estimated power consumption and temperature information.

The power estimation reported in the **Summary** tab is the total static and dynamic power consumption of your design. For a more detailed view of this power consumption, click the **Analysis** tab.
To estimate the junction temperature:

1. Verify your package. You cannot change your package directly in SmartPower, because it may render your place-and-route information (and thus it may severely impact the total power consumption). To choose another package, use **Designer > Tools > Device Selection**.
2. Click the **Summary** tab and select a **Cooling** style in the list. Thermal resistance changes automatically when you update the cooling style.
3. Specify an ambient temperature. Enter an **ambient temperature** (default value is 25ºC) and click the **Set** button.

   Note: The junction temperature value changes according to the package, cooling style, and ambient temperature values you choose.

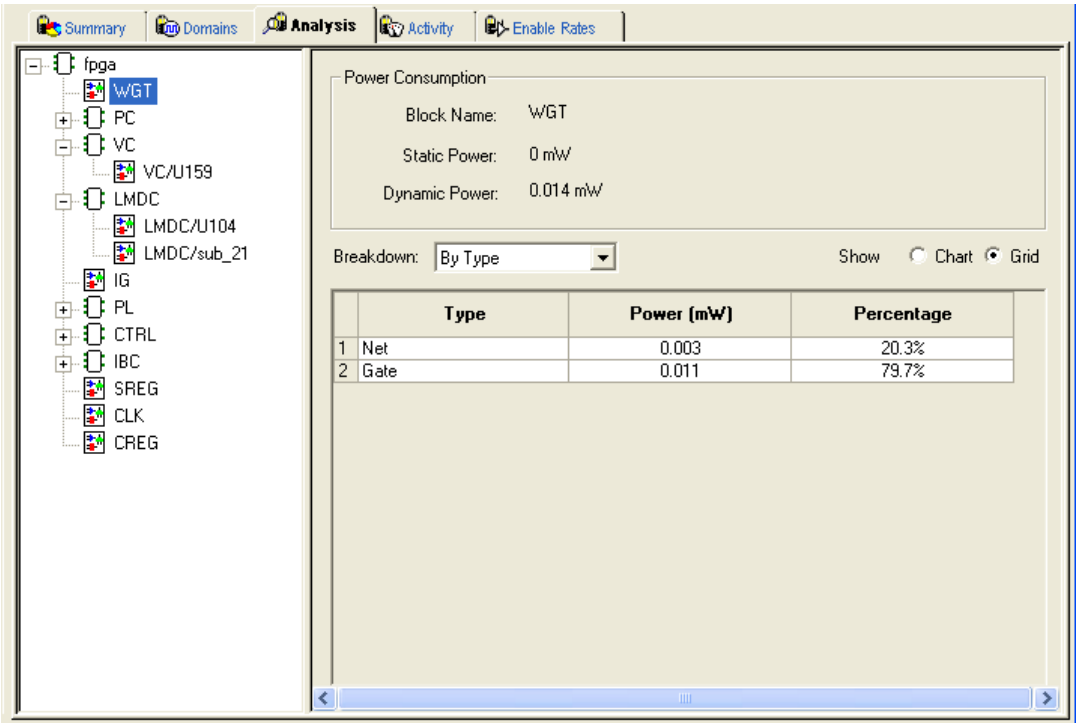
4.16. Analyzing Results (Ask a Question)

The **Analysis** tab shows the estimated power consumption of individual blocks, gates, nets, I/Os, and memory, and allows you to make a hierarchical analysis of your power consumption.

The **Analysis** tab may also help you to improve your power consumption by identifying the components that consume a significant amount of power.
To identify the components consuming the most power:

1. Use the **Analysis** tab to expand the design hierarchy. The **Analysis** tab allows you to expand your design hierarchy and view a complete list of the blocks in your design. In the hierarchy window, click the **+** next to your design to view the items in the hierarchy. Click the **+** next to a sub-block to view its sub-elements.

Figure 4-3. Analysis Tab Dialog Box



2. Click to select a block. By default, SmartPower selects the design-level block, but you can always select another block in the hierarchical tree. The Report window displays the list of sub-elements of the selected block. Initially, this list includes all sub-elements. The grid in the Analysis window displays a breakdown of the power of the selected block by type. This breakdown provides you a good overview of which areas of your design you should optimize for power.

3. To find the components that use a significant amount of power, choose **By Instance** from the **Breakdown** drop-down list.
4. Sort and filter the sub-elements to find the component that is using the most power. The Analysis window displays the list of sub-elements of the selected block. By default, this list includes all sub-elements. The grid contains columns for name, type, power, driver, fanout, macro, I/O standard, output load, output drive, slew, port, domain and frequency. You can limit the list of sub-elements to a list of sub-blocks, nets, gates, I/Os, memories, or any combination of these five classes of sub-elements. You can sort the list according to any column by double-clicking the column header.

4.17. SmartPower Calculation Equations (Ask a Question)

SmartPower calculates the static power and the dynamic power of your design, for given operating conditions and operating modes:

Static Power: summation of the static power consumed by each element of the design. SmartPower provides a static power consumption of the array. This value is die-dependent. This value is also a function of the operating mode. For some families, SmartPower also considers a static power contribution per I/O bank. For specific I/O technologies like voltage referenced I/Os or differential I/Os, SmartPower also considers a static power contribution per I/O.

Dynamic Power: summation of the dynamic power consumed by each element of the design (nets, modules, I/Os, RAMs, FIFOs, PLLs, etc.).

Operating Conditions: SmartPower calculates power consumption in Worst, Typical, or Best operating conditions. SmartPower uses the following as general guidelines:

- It applies a voltage derating to dynamic power contributions. Higher voltage typically leads to higher power consumption.
- It applies a temperature derating to static power contributions. Higher temperature typically leads to higher power consumption.
- It applies no radiation derating.

Operating Modes: SmartPower calculates power consumption in Active, Static, Flash*Freeze, Sleep, and Shutdown operating modes when applicable for your design.

4.18. Sample Equations: (Ask a Question)

The following examples are for general evaluation purposes only. They are not a precise representation of the actual calculations, since each calculation takes into account family-specific information.

- For a **net**,
  $$P = C \cdot V^2 \cdot F$$
  where C is the total capacitive loading of the net (extracted from the routing topology), V is the net's voltage swing, and F is the average switching frequency.
  Note: For the ProASICPLUS family, SmartPower extracts the capacitive loading of a net from a Wire Load model.
- For a **module**, the power is computed using a characterized library (by family and die-size) describing a specific power model for each type of module. For example, the power model of a flip-flop is given by $P = P_{CK} \cdot F_{CK} + P_{DOUT} \cdot F_{DOUT} + P_{Din} \cdot F_{Din}$
  where FCK is the average clock-input frequency for this flip-flop, FDOUT is its average data-output frequency, and PCK, PDOUT, PDin are three constants estimated by electrical simulation and silicon characterization for this flip-flop module, and FDin is its average data-input frequency.
- For an **I/O**, the formula used for computing the power consumption depends on the I/O technology and the family. For example, for a TTL output, the dynamic power is given by
  $$P = P_{INT} \cdot F + C \cdot V^2 \cdot F$$

where C is the output load (derived from what you have set in the I/O Attribute Editor GUI, typically 35 pF for TTL), V is the output's voltage swing (3.3 V for TTL), and PINT represents an internal power contribution dissipated in the pad, and F is the average switching frequency of the I/O.

- For a **complex block**, like a RAM, a FIFO, or a PLL, SmartPower uses a high-level power model that integrates design parameters.

SmartPower computes all the constant parameters of these equations automatically. However, the frequencies depend on the target frequencies of your design. Since it is impractical to enter each frequency manually, SmartPower has several flows that help you estimate the frequencies and calculate the power consumption.

4.19. Data Change History - SmartPower (Ask a Question)

The data change history report lists power data updates up to the current release that may impact the power consumption of the design.

To generate a data change history report, choose **Data Change History** from the **Help** menu to open a data change history report in text format.

5. SmartPower Tcl Commands (Ask a Question)

This section describes the SmartPower Tcl Commands for the PolarFire, PolarFire SoC, SmartFusion 2, IGLOO 2, and RTG4 device families.

5.1. smartpower_add_new_custom_mode (Ask a Question)

Description

This Tcl command creates a new custom mode.

```
smartpower_add_new_custom_mode -name {mode name} \
                               -base_mode {base mode} \
                               [-description {mode description}]
```

Arguments

Parameter	Type	Description
name	string	Specifies the name of the new custom mode. This parameter is mandatory.
base_mode	string	Specifies the name of the base mode used to create the new custom mode. It must be one of the following: Active, Standby or Flash*Freeze (RTG4, SmartFusion 2, and IGLOO 2). This parameter is mandatory.
description	string	Specifies the description of the new custom mode. This parameter is optional.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'name' is missing.
None	Parameter 'name' has illegal value.
None	base_mode: Invalid argument value: 'mode_name' (expecting Active, Static, new_mode_name1, new_mode_name2 or new_mode_name4).
None	Parameter 'base_mode' has illegal value.
None	Required parameter 'base_mode' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_add_new_custom_mode -name "name" [-description "description"] -base_mode "Active | Static | new_mode_name1 | new_mode_name2 | new_mode_name4"'.

Supported Families

Supported Families	Supported Versions
PolarFire®	v12.4+
SmartFusion® 2	v12.4+
RTG4™	v12.4+
IGLOO® 2	v12.4+
PolarFire SoC	v12.6+

Example

This example creates a new custom mode "NewCustomMode" based on the Active mode:

```
smartpower_add_new_custom_mode -name {NewCustomMode} \
                               -base_mode {Active} \
                               -description {frequency 10 MHz}
```

See Also

- smartpower_remove_custom_mode

5.2. smartpower_add_new_scenario (Ask a Question)

Description

This Tcl command creates a new custom scenario entering the scenario name, duration (total duration for the sequence must equal to 100%), and selecting previously defined operating modes for this sequence.

```
smartpower_add_new_scenario -name {New Scenario name} \
                            [-description {description of scenario}] \
                            -mode {Mode_name:duration}
```

Arguments

Parameter	Type	Description
name	string	Specifies the name of the new scenario.
description	string	Specifies the description of the new scenario. This parameter is optional.
mode	string	Specifies the previously defined mode(s) and duration(s) for the specified scenario. This parameter is mandatory. There may be multiple -mode arguments (see example below).

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'name' is missing.
None	Required parameter 'mode' is missing.
None	mode: Invalid argument value: 'mode_name' (expecting Active, Static or Flash*Freeze).
None	The sum of the duration must be 100%. Current sum: duration_value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_add_new_scenario -name "scenario name" [-description "description"] [-mode "":""]+'

Supported Families

Supported Families	Supported Versions
PolarFire®	v12.4+
SmartFusion® 2	v12.4+
RTG4™	v12.4+
IGLOO® 2	v12.4+
PolarFire SoC	v12.6+

Example

This example creates a new scenario called "MyScenario" with the specified modes and durations:

```
smartpower_add_new_scenario -name "MyScenario" -mode "Custom_1:50.00" \
                        -mode "Custom_2:25.00" -mode "Active:25.00"
```

See Also

- smartpower_edit_scenario
- smartpower_remove_scenario

5.3. smartpower_add_pin_in_domain (Ask a Question)

Description

This tcl command adds a pin into a clock or set domain.

```
smartpower_add_pin_in_domain -pin_name {pin name} \
                             -pin_type {value} \
                             -domain_name {domain name} \
                             -domain_type {value}
```

Arguments

Parameter	Type	Description
pin_name	string	Specifies the name of the pin to add to the domain.
pin_type	string	Specifies the type of the pin to add. The acceptable values for this argument are the following: • clock - The pin to add is a clock pin. • data - The pin to add is a data pin.
domain_name	string	Specifies the name of the domain in which to add the specified pin.
domain_type	string	Specifies the type of domain in which to add the specified pin. The acceptable values for this argument are the following: • clock - The domain is a clock domain. • set - The domain is a set domain.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'pin_name' is missing.
None	Required parameter 'pin_type' is missing.
None	Required parameter 'domain_type' is missing.
None	Required parameter 'domain_name' is missing.
None	domain_type: Invalid argument value: 'value' (expecting clock or set).
None	Parameter 'pin_name' has illegal value.
None	Failed to add pin "pin_name" to "domain".
None	pin_type: Invalid argument value: 'value' (expecting clock or data).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_add_pin_in_domain [-pin_name "name of pin"]+ -domain_type "clock | set" -domain_name "name of domain" -pin_type "clock | data"'.

Supported Families

Supported Families	Supported Versions
PolarFire®	v12.4+
SmartFusion® 2	v12.4+
RTG4™	v12.4+
IGLOO® 2	v12.4+
PolarFire SoC	v12.6+

Example

The following example adds a "XCMP3/U0/U1:Y" clock pin to an existing Clock "clk" domain:

```
smartpower_add_pin_in_domain -pin_name {XCMP3/U0/U1:Y} \
                             -pin_type {clock} \
                             -domain_name {clk} \
                             -domain_type {clock}
```

The following example adds a "XCMP3/U0/U1:Y" data pin to an existing Set "myset" domain:

```
smartpower_add_pin_in_domain -pin_name {XCMP3/U0/U1:Y} \
                             -pin_type {data} \
                             -domain_name {myset} \
                             -domain_type {set}
```

See Also

- smartpower_create_domain

5.4. smartpower_battery_settings (Ask a Question)

Description

This SmartPower Tcl command sets the battery capacity in SmartPower. The battery capacity is used to compute the battery life of your design.

```
smartpower_battery_settings [-capacity {decimal value}]
```

Arguments

Parameter	Type	Description
capacity	decimal	Specify the battery capacity in mA*Hours. Value must be a positive decimal. This parameter is mandatory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	capacity: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'capacity' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_battery_settings [-capacity "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2

PolarFire SoC

Example

This example sets the battery capacity to 1800 mA * Hours.

```
smartpower_battery_settings -capacity {1800}
```

5.5. smartpower_change_clock_statistics (Ask a Question)

Description

This Tcl command changes the default frequencies and probabilities for a specific domain.

Note: This command is associated with the functionality of Initialize frequencies and probabilities dialog box.

```
smartpower_change_clock_statistics -domain_name {value} \
                                   -clocks_freq {value} \
                                   -clocks_proba {value} \
                                   -registers_freq {value} \
                                   -registers_proba {value} \
                                   -set_reset_freq {value} \
                                   -set_reset_proba {value} \
                                   -primaryinputs_freq {value} \
                                   -primaryinputs_proba {value} \
                                   -combinational_freq {value} \
                                   -combinational_proba {value}
```

Arguments

Parameter	Type	Description
domain_name	string	Specifies the domain name in which to initialize frequencies and probabilities.
clocks_freq	string	Specifies the user input frequency in Hz, KHz or MHz for all clocks. Must be a positive decimal value.
clocks_proba	decimal	Specifies the user input probability in percentage(%) for all clocks. Must be a positive decimal value and less than or equal to 100.000.
registers_freq	string	Specifies the user input frequency (in Hz, KHz or MHz) or the toggle rate in percentage(%). Must be a positive decimal value and less than or equal to 100.000. If the unit is not provided and toggle rate is active, the value is handled as a toggle rate, if toggle rate is not active, the value is handled as a frequency.
registers_proba	decimal	Specifies the user input probability in percentage(%) for all registers. Must be a positive decimal value and less than or equal to 100.000.
set_reset_freq	string	Specifies the user input frequency (in Hz, KHz or MHz) or the toggle rate in percentage(%). Must be a positive decimal value. If the unit is not provided and toggle rate is active, the value is handled as a toggle rate, if toggle rate is not active, the value is handled as a frequency.
set_reset_proba	decimal	Specifies the user input probability in percentage(%) for all set/reset nets. Must be a positive decimal value and less than or equal to 100.000.
primaryinputs_freq	string	Specifies the user input frequency (in Hz, KHz or MHz) or the toggle rate in percentage(%). Must be a positive decimal value. If the unit is not provided and toggle rate is active, the value is handled as a toggle rate, if toggle rate is not active, the value is handled as a frequency.
primaryinputs_proba	decimal	Specifies the user input probability in percentage(%) for all primary inputs. Must be a positive decimal value and less than or equal to 100.000.
combinational_freq	string	Specifies the user input frequency (in Hz, KHz or MHz) or the toggle rate percentage(%). Must be a positive decimal value. If the unit is not provided and toggle rate is active, the value is handled as a toggle rate, if toggle rate is not active, the value is handled as a frequency.

smartpower_change_clock_statistics (continued)		
Parameter	**Type**	**Description**
combinational_proba	decimal	Specifies the user input probability in percentage(%) for all combinational combinational output. Must be a positive decimal value and less than or equal to 100.000.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'domain_name' is missing.
None	Parameter 'domain_name' has illegal value.
None	Parameter 'combinational_proba' has illegal value.
None	combinational_proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'combinational_proba' must be a positive decimal value.
None	Parameter 'combinational_proba' must be less than or equal to 100.000.
None	Parameter 'combinational_freq' has illegal value.
None	combinational_freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz, MHz or %)).
None	Parameter 'primaryinputs_proba' has illegal value.
None	primaryinputs_proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'primaryinputs_proba' must be a positive decimal value.
None	Parameter 'primaryinputs_proba' must be less than or equal to 100.000.
None	Parameter 'primaryinputs_freq' has illegal value.
None	primaryinputs_freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz, MHz or %)).
None	Parameter 'set_reset_proba' has illegal value.
None	set_reset_proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'set_reset_proba' must be a positive decimal value.
None	Parameter 'set_reset_proba' must be less than or equal to 100.000.
None	Parameter 'set_reset_freq' has illegal value.
None	set_reset_freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz, MHz or %)).
None	Parameter 'registers_proba' has illegal value.
None	registers_proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'registers_proba' must be a positive decimal value.
None	Parameter 'registers_proba' must be less than or equal to 100.000.
None	Parameter 'registers_freq' has illegal value.
None	registers_freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz, MHz or %)).
None	Parameter 'clocks_proba' has illegal value.
None	clocks_proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'clocks_proba' must be a positive decimal value.
None	Parameter 'clocks_proba' must be less than or equal to 100.000.
None	Parameter 'clocks_freq' has illegal value.
None	clocks_freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz or MHz)).

smartpower_change_clock_statistics (continued)	
Error Code	**Description**
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_change_clock_statistics -domain_name "domain name" \ [-clocks_freq "decimal value [unit { Hz | KHz | MHz }]"] \ [-clocks_proba "decimal value"] \ [-registers_freq "decimal value [unit { Hz | KHz | MHz | % }]"] \ [-registers_proba "decimal value"] \ [-set_reset_freq "decimal value [unit { Hz | KHz | MHz | % }]"] \ [-set_reset_proba "decimal value"] \ [-primaryinputs_freq "decimal value [unit { Hz | KHz | MHz | % }]"] \ [-primaryinputs_proba "decimal value"] \ [-combinational_freq "decimal value [unit { Hz | KHz | MHz | % }]"] \ [-combinational_proba "decimal value"] '.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all clocks with:

```
smartpower_change_clock_statistics -domain_name {my_domain} \
                                   -clocks_freq {10 MHz} \
                                   -clocks_proba {20} \
                                   -registers_freq {10 MHz} \
                                   -registers_proba {20} \
                                   -set_reset_freq {10MHz} \
                                   -set_reset_proba {20} \
                                   -primaryinputs_freq {10 MHz} \
                                   -primaryinputs_proba {20} \
                                   -combinational_freq {10 MHz} \
                                   -combinational_proba {20}
```

See Also

- smartpower_init_set_combinational_options
- smartpower_init_set_primaryinputs_options

5.6. smartpower_change_setofpin_statistics (Ask a Question)

Description

This tcl command changes the default frequencies and probabilities for a specific set.

Note: This command is associated with the functionality of Initialize frequencies and probabilities dialog box.

```
smartpower_change_setofpin_statistics \
            -domain_name "domain name" \
            [-data_freq "decimal value [unit { Hz | KHz | MHz }]"] \
            [-data_proba "decimal value"]
```

Arguments

Parameter	Type	Description
domain_name	string	Specifies the domain name in which to initialize data frequencies and probabilities.
data_freq	string	Specifies the user input data frequency in Hz, KHz, or MHz for all sets of pins.
data_proba	decimal	Specifies the user input data probability in % for all sets of pins.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'domain_name' is missing.
None	Parameter 'domain_name' has illegal value.
None	Parameter 'data_proba' has illegal value.
None	data_proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'data_freq' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_change_setofpin_statistics -domain_name "domain name" \ [-data_freq "decimal value [unit { Hz | KHz | MHz }]"] \ [-data_proba "decimal value"]'.

Supported Families

PolarFire®

SmartFusion® 2

RTG4™

IGLOO® 2

PolarFire SoC

Example

The following example initializes all clocks withs:

```
smartpower_change_setofpin_statistics -domain_name {my_domain} \
                                      -data_freq {10 MHz} \
                                      -data_proba {20}
```

5.7. smartpower_commit (Ask a Question)

Description

This Tcl command saves the changes to the design file.

```
smartpower_commit [-file "SmartPower settings file (.zip)"]
```

Arguments

Parameter	Type	Description
file	string	Path to the *.zip where power analysis details will be saved. You can specify a relative or absolute path. This parameter is optional. If the argument is not specified "smartpower.swp" file is created under <project path>/designer/<component name> directory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_commit [-file "SmartPower settings file (.zip)"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example saves changes, power analysis details in the "/prj/designer/top/sp_details.zip":

```
smartpower_commit -file "./sp_details.zip"
```

5.8. smartpower_compute_vectorless (Ask a Question)

Description

This Tcl command executes a vectorless analysis of the current operating mode.

```
smartpower_compute_vectorless
```

Arguments

Parameter	Type	Description
None	None	None

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_compute_vectorless'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

```
smartpower_compute_vectorless
```

5.9. smartpower_create_domain (Ask a Question)

Description

This Tcl command creates a new clock or set domain.

Note: The domain name cannot be the name of an existing domain. The domain type must be either clock or set.

```
smartpower_create_domain -domain_type {domain type} -domain_name {domain name}
```

Arguments

Parameter	Type	Description
domain_type	string	Specifies the type of domain to create. The acceptable values for this argument are: • clock - The domain is a clock domain. • set - The domain is a set domain.
domain_name	string	Specifies the name of the new domain.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'domain_name' has illegal value.
None	Required parameter 'domain_name' is missing.
None	domain_type: invalid argument value: 'type_name'(expecting set or clock).
None	A domain with name "domain_name" already exists.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_create_domain -domain_name "domain name" -domain_type "set | clock"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example creates a new set domain named "myset":

```
smartpower_create_domain -domain_type {set} -domain_name {myset}
```

See Also

- smartpower_init_do
- smartpower_remove_domain

5.10. smartpower_edit_custom_mode (Ask a Question)

Description

This Tcl command edits a custom mode. You should specify at least one of the following optional parameters: new name and description.

```
smartpower_edit_custom_mode -name {old mode name} \
                            [-new_name {new mode name}] \
                            [-description {mode description}]
```

Arguments

Parameter	Type	Description
name	string	Specifies the name of the custom mode you want to edit. This parameter is mandatory.

smartpower_edit_custom_mode (continued)		
Parameter	**Type**	**Description**
new_name	string	Specifies the new name of the custom mode. This parameter is optional.
description	string	Specifies the description of the new custom mode. This parameter is mandatory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'name' is missing.
None	Parameter 'name' has illegal value.
None	Parameter 'new_name' has illegal value.
None	Parameter 'description' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_edit_custom_mode -name "name" [-description "description"] [-new_name "new mode name"]'.
None	A custom mode with name "mode_name" does not exist.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example edits custom mode "CustomMode" and renames it "CustomMode2":

```
smartpower_edit_custom_mode -name {CustomMode} \
                            -new_name {CustomMode2} \
                            -description {frequency 10 MHz}
```

See Also

- smartpower_add_new_custom_mode
- smartpower_remove_custom_mode

5.11. smartpower_edit_scenario (Ask a Question)

Description

This Tcl command edits a custom scenario: scenario name, previously defined mode(s) and duration(s).

```
smartpower_edit_scenario -name {custom scenario name} \
                         [-description {description of scenario}] \
                         -mode {mode_name:duration} \
                         -new_name {New Scenario name}
```

Arguments

Parameter	Type	Description
name	string	Specifies the name of the scenario.

smartpower_edit_scenario (continued)		
Parameter	**Type**	**Description**
description	string	Specifies the description of the scenario.
mode	string	Specifies the mode(s) and duration(s) for the specified scenario. Possible values are {<operating mode>:<duration>}
new_name	string	Specifies the new name for the scenario.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'new_name' has illegal value.
None	Parameter 'name' has illegal value.
None	Required parameter 'name' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_edit_scenario -name "scenario name" [-description "description"] [-mode "":""]* [-new_name "new mode name"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example edits the name of "MyScenario" to "NewScenario":

```
smartpower_edit_scenario -name {MyScenario} -new_name {NewScenario} -mode "Active:100.00"
```

See Also

- smartpower_add_new_scenario
- smartpower_remove_scenario

5.12. smartpower_export_mpe_report (Ask a Question)

Description

This Tcl command exports the Microchip Power Estimation(MPE) report in XML format. The generated `.xml` report contains the following information:

- Device Settings
- Thermal Settings
- Voltage Source
- Clocks
- Logic Breakdown
- LSRAM Breakdown with Advanced Settings
- uSRAM Breakdown with Advanced Settings
- Math Breakdown with Advanced Settings
- PLL and DLL

- I/Os
- Crypto
- Transceivers

In addition, the following information is available for PolarFire SoC devices:

- MSS RISC-V (Quad U54)
- AXI MSS/Fabric Interfaces
- MDDR
- MSS I/O Interfaces
- User Crypto

Note: This command supported only for G5 families.

```
smartpower_export_mpe_report -filename {file_name.xml}
```

Arguments

Parameter	Type	Description
file_name	string	Name of the XML file to be exported. This argument is mandatory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'filename' has illegal value.
None	Required parameter 'filename' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_export_mpe_report -filename "filename"'.

Supported Families

PolarFire®
PolarFire SoC

Example

The following command exports the Microchip Power Estimation (MPE) report in mpe_report.xml file:

```
smartpower_export_mpe_report -filename {mpe_report.xml}
```

5.13. smartpower_get_temperature (Ask a Question)

Description

Enter description here

```
smartpower_get_temperature [-what "TRUE | FALSE"] [-opcond "Best | Typical | Worst"]
```

Arguments

Parameter	Type	Description
what	boolean	

smartpower_get_temperature (continued)		
Parameter	**Type**	**Description**
opcond	string	Specifies the operating condition. The acceptable values for this argument are the following: • Worst - The operating condition is set to worst case. • Typical - The operating condition is set to typical case. • Best - The operating condition is set to best case.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'param_name' is not defined.
None	what: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0, false, ambient or opcond).
None	opcond: Invalid argument value: 'value' (expecting Best, Typical or Worst).

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2

Example

Enter example description here

```
smartpower_get_temperature  -what "TRUE" -opcond "Best"
```

5.14. smartpower_get_tetaja (Ask a Question)

Description

Enter description here

```
smartpower_get_tetaja [-style "case_cooling | still_air | 1.0_mps | 2.5_mps | custom"]
```

Arguments

Parameter	Type	Description
style	string	

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'param_name' is not defined.
None	style: Invalid argument value: 'value' (expecting case_cooling, still_air, 1.0_mps, 2.5_mps or custom).

Supported Families

PolarFire®

SmartFusion® 2
RTG4™
IGLOO® 2

Example

Enter example description here

```
smartpower_get_tetaja -style "custom"
```

5.15. smartpower_get_thermalmode (Ask a Question)

Description

Enter description here

```
smartpower_get_thermalmode
```

Arguments

Parameter	Type	Description
None	None	None

Return Type	Description
None	None

Error Codes

Error Code	Description
None	None

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

Enter example description here

```
smartpower_get_thermalmode
```

5.16. smartpower_import_vcd (Ask a Question)

Description

This Tcl command imports into SmartPower a VCD file generated by a simulation tool. SmartPower extracts the frequency and probability information from the VCD.

Note: SmartPower stops importing VCD in Static mode as now VCD basically contains signal transitions and Static mode means that the device is off. VCD file generation refer to the simulation related help in libero user guide.

```
import_vcd -file {VCD file} \
           [-opmode {mode name}] \
           [-with_vectorless {TRUE | FALSE}] \
           [-partial_parse {TRUE | FALSE}] \
           [-start_time {decimal value}] \
           [-end_time {decimalvalue}] \
```

```
            [-auto_detect_top_level_name {TRUE | FALSE}] \
            [-top_level_name {top level name}] \
            [-glitch_filtering {false | auto | true}] \
            [-glitch_threshold {integer value}] \
            [-stop_time {decimal value}]
```

Arguments

Parameter	Type	Description
file	string	Absolute path to a VCD file. Value must be a file path. This parameter is mandatory.
opmode	string	Operating mode in which the VCD will be imported. Operating mode name "Active" by default. If the mode doesn't exist, it will be created. Value must be a string. This parameter is optional.
with_vectorless	boolean	Specify the method to set the frequency and probability information for signals not annotated by the VCD TRUE(default): use the vectorless analysis, FALSE: use average value computed from the VCD. Value must be a boolean. This parameter is optional.
partial_parse	boolean	Enable partial parsing of the VCD. Specify the Smart time and End time to partially parse the VCD file. Start time and End time need to be specified when TRUE. Value must be one of TRUE, 1, true, FALSE, 0 or false. By default is FALSE. This parameter is optional.
start_time	decimal	Specify the starting timestamp of the VCD extraction in ns. It must be lower than the specified end_time. It must be lower than the last timestamp in the VCD file. Value must be a positive decimal nanoseconds(ns). This parameter is optional.
end_time	decimal	Specify the end timestamp of the VCD extraction in ns. It must be higher than the specified start_time. Value must be a positive decimal nanoseconds(ns). This parameter is optional.
auto_detect_top_level_name	boolean	Enable the auto detection of the top level name in the VCD file. Top_level_name needs to be specified when FALSE. Value must be a boolean. By default is TRUE. This parameter is optional.
top_level_name	string	Specify the full hierarchical name of the instance of the design in the VCD file. Value must be a string. This parameter is optional.
glitch_filtering	string	Enable to filter pulses of short duration by specifying automatic glitch filtering or by specifying vlaue to the filtering threshold. This parameter is optional. Value must be on of the following: • AUTO - Enable glitch filtering with predefined threshold based on the family. • TRUE - Enable glitch filtering, glitch_threshold must be specified. • FALSE - Disable glitch filtering.
glitch_threshold	integer	Specify the threshold in ps below which glitches are filtered out. Value must be a positive integer. This parameter is optional.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'file' is missing.
None	Mode "Static" is not active, please specify an active mode.
None	opmode: Invalid argument value: 'mode_name' (expecting Active or Static).
None	partial_parse: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false)
None	start_time: Invalid argument value: 'time_value' (expecting decimal value).
None	Parameter 'start_time' must be a positive decimal value.

smartpower_import_vcd (continued)	
Error Code	**Description**
None	end_time: Invalid argument value: 'time_value' (expecting decimal value).
None	Parameter 'end_time' must be a positive decimal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_import_vcd [-format "file format"] \ -file "VCD file" \ [-opmode "mode name"] \ [-with_vectorless "TRUE | FALSE"] \ [-partial_parse "TRUE | FALSE"] \ [-start_time "decimal value"] \ [-end_time "decimal value"] \ [-auto_detect_top_level_name "TRUE | FALSE"] \ [-top_level_name "top level name"] \ [-glitch_filtering "false | auto | true"] \ [-glitch_threshold "integer value"] \ [-stop_time {decimal value}]'

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The Tcl command imports the power.vcd file generated by the simulator into SmartPower:

```
smartpower_import_vcd -file {/home/example/simulation/power.vcd}
```

The Tcl command extracts information between 1ms and 2ms in the simulation, and stores the information into a custom mode:

```
smartpower_import_vcd -file {/home/example/simulation/power.vcd} -partial_parse {TRUE}
-start_time {1000000} -end_time {2000000} -opmode {power_1ms_to_2ms}
```

5.17. smartpower_init_do (Ask a Question)

Description

This Tcl command initializes the frequencies and probabilities for clocks, registers, set/reset nets, primary inputs, combinational outputs, enables and other sets of pins, and selects a mode for initialization.

```
smartpower_init_do [-opmode "Active"] \
           [-with "vectorless | default"] \
           [-clocks "TRUE | FALSE"] \
           [-registers "TRUE | FALSE"] \
           [-set_reset "TRUE | FALSE"] \
           [-primaryinputs "TRUE | FALSE"] \
           [-combinational "TRUE | FALSE"] \
           [-enables "TRUE | FALSE"] \
           [-othersets "TRUE | FALSE"]
```

Arguments

Parameter	Type	Description
opmode	string	This parameter is optional and specifies the mode in which to initialize frequencies and probabilities. The value must be Active or Flash*Freeze (RTG4, SmartFusion 2 and IGLOO 2).

smartpower_init_do (continued)

Parameter	Type	Description
with	string	This sets the option of initializing frequencies and probabilities with vectorless analysis or with fixed values. The acceptable values for this argument are the following: • vectorless - Initializes frequencies and probabilities with vectorless analysis. • fixed - Initializes frequencies and probabilities with fixed values.
clocks	boolean	This sets the option of initializing frequencies and probabilities for all clocks. The following table shows theacceptable values for this argument: • TRUE, true or 1 - Initializes frequencies and probabilities for all clocks. • FALSE, false or 0 - Does not initialize frequencies and probabilities for all clocks.
registers	boolean	This sets the option of initializing frequencies and probabilities for all registers. The following table shows theacceptable values for this argument: • TRUE, true or 1 - Initializes frequencies and probabilities for all registers. • FALSE, false or 0 - Does not initialize frequencies and probabilities for all registers.
set_reset	boolean	This sets the option of initializing frequencies and probabilities for all set/reset nets. The following table shows theacceptable values for this argument: • TRUE, true or 1 - Initializes frequencies and probabilities for all set/reset nets. • FALSE, false or 0 - Does not initialize frequencies and probabilities for all set/reset nets.
primaryinputs	boolean	This sets the option of initializing frequencies and probabilities for all primary inputs. The acceptable values for this argument are the following: • TRUE, true or 1 - Initializes frequencies and probabilities for all primary inputs. • FALSE, false or 0 - Does not initialize frequencies and probabilities for all primary inputs.
combinational	boolean	This sets the option of initializing frequencies and probabilities for all combinational outputs. The acceptable values for this argument are the following: • TRUE, true or 1 - Initializes frequencies and probabilities for all combinational outputs. • FALSE, false or 0 - Does not initialize frequencies and probabilities for all combinational outputs.
enables	boolean	This sets the option of initializing frequencies and probabilities for all enable sets of pins. The acceptable values for this argument are the following: • TRUE, true or 1 - Initializes frequencies and probabilities for all enable sets of pins. • FALSE, false or 0 - Does not initialize frequencies and probabilities for all enable sets of pins.

smartpower_init_do (continued)		
Parameter	**Type**	**Description**
othersets	boolran	This sets the option of initializing frequencies and probabilities for all other sets of pins. The acceptable values for this argument are the following: • TRUE, true or 1 - Initializes frequencies and probabilities for all other sets of pins. • FALSE, false or 0 - Does not initialize frequencies and probabilities for all other sets of pins

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'othersets' has illegal value.
None	Parameter 'enables' has illegal value.
None	enables: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	Parameter 'combinational' has illegal value.
None	combinational: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	Parameter 'primaryinputs' has illegal value.
None	primaryinputs: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	Parameter 'set_reset' has illegal value.
None	set_reset: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	Parameter 'registers' has illegal value.
None	registers: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	Parameter 'clocks' has illegal value.
None	clocks: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	Parameter 'opmode' has illegal value.
None	opmode: Invalid argument value: 'Static' (expecting Active or Flash*Freeze).
None	Parameter 'with' has illegal value.
None	with: Invalid argument value: 'value' (expecting vectorless, default or fixed).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_init_do [-opmode "Active"] \ [-with "vectorless | default"] \ [-clocks "TRUE | FALSE"] \ [-registers "TRUE | FALSE"] \ [-set_reset "TRUE | FALSE"] \ [-primaryinputs "TRUE | FALSE"] \ [-combinational "TRUE | FALSE"] \ [-enables "TRUE | FALSE"] \ [-othersets "TRUE | FALSE"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all clocks with:

```
smartpower_init_do -with {vectorless} \
                   -opmode {my_mode} \
                   -clocks {true} \
                   -registers{true} \
                   -asynchronous {true} \
                   -primaryinputs {true} \
                   -combinational {true} \
                   -enables {true} \
                   -othersets {true}
```

5.18. smartpower_initialize_clock_with_constraints (Ask a Question)

Description

This Tcl command initializes the clock frequency and the data frequency of a single clock domain with a specified clock name and the initialization options.

Notes:

- This command is associated with the functionality of Initialize frequencies and probabilities dialog box.
- This command is associated with the right click menu Synchronize Domain with SmartTime on a single clock domain in the Domains tab.

```
smartpower_initialize_clock_with_constraints -clock_name {value}
```

Arguments

Parameter	Type	Description
clock_name	string	Specifies the name of the clock that will be initialized. This parameter is mandatory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'clock_name' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_initialize_clock_with_constraints [-clock_name "name of clock"]+'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2

Example

The following example initializes "my_clock" with clock constraints from SmartTime:

```
smartpower_initialize_clock_with_constraints -clock_name {my_clock}
```

See Also

- smartpower_create_domain

5.19. smartpower_init_set_clocks_options (Ask a Question)

Description

This Tcl command initializes the clock frequency options of all clock domains.

Note: This command is associated with the functionality of Initialize frequencies and probabilities dialog box.

```
smartpower_init_set_clocks_options [-with_clock_constraints {value}] \
                                   [-with_default_values {value}] \
                                   [-freq {value}] \
                                   [-duty_cycle {value}]
```

Arguments

Parameter	Type	Description
with_clock_constraints	boolean	This sets the option of initializing the clock frequencies with frequency constraints from SmartTime. The acceptable values for this argument are the following: • true - Sets initialize clock frequencies with clock constraints ON. • false - Sets initialize clock frequencies with clock constraints OFF.
with_default_values	boolean	This sets the option of initializing the clock frequencies with a user input default value. The acceptable values for this argument are the following: • true - Sets initialize clock frequencies with default values ON. • false - Sets initialize clock frequencies with default values OFF.
freq	string	Specifies the user input frequency in Hz, KHz or MHz.
duty_cycle	decimal	Specifies the user input duty cycles in percentage(%).

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'duty_cycle' has illegal value.
None	Parameter 'duty_cycle' must be less than or equal to 100.000.
None	Parameter 'duty_cycle' must be a positive decimal value.
None	duty_cycle: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'freq' has illegal value.
None	freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz or MHz)).
None	Parameter 'with_default_values' has illegal value.
None	with_default_values: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	Parameter 'with_clock_constraints' has illegal value.
None	with_clock_constraints: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_init_set_clocks_options [-with_clock_constraints "TRUE | FALSE"] [-with_default_values "TRUE | FALSE"] [-freq "decimal value [unit { Hz | KHz | MHz }]"] [-duty_cycle "decimal value"]'.

Supported Families

PolarFire®

SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all clocks after executing "smartpower_init_do":

```
smartpower_init_do with -clocks {true}
smartpower_init_set_clocks_options -with_clock_constraints {true} \
                                   -with_default_values {true} \
                                   -freq {10 MHz} \
                                   -duty_cycle {20}
```

5.20. smartpower_init_set_combinational_options (Ask a Question)

Description

This Tcl commands initializes the frequency and probability of all combinational outputs.

```
smartpower_init_set_combinational_options [-freq {value}] \
                                          [-proba {value}]
```

Arguments

Parameter	Type	Description
freq	string	Specifies the user input frequency (in Hz, KHz or MHz) or the toggle rate in percentage(%). If the unit is not provided and toggle rate is active, the value is handled as a toggle rate, if toggle rate is not active, the value is handled as a frequency.
proba	decimal	Specifies the user input probability in percentage(%). Must be a positive decimal value and less than or equal to 100.000.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'proba' has illegal value.
None	Parameter 'proba' must be a positive decimal value.
None	Parameter 'proba' must be less than or equal to 100.000.
None	proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'freq' has illegal value.
None	freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz, MHz or %)).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_init_set_combinational_options [-freq "decimal value [unit { Hz | KHz | MHz | % }]"] [-proba "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all combinational signals after executing smartpower_init_do with -combinational {true}:

```
smartpower_init_set_combinational_options -freq {10 MHz} -proba {20}
```

See Also

- smartpower_init_set_primaryinputs_options

5.21. smartpower_init_setofpins_values (Ask a Question)

Description

This tcl command initializes the frequency and probability of all sets of pins.

```
smartpower_init_setofpins_values -domain_name "IOsEnableSet | MemoriesEnableSet" \
                         [-freq "decimal value [unit { Hz | KHz | MHz }]"] \
                         [-proba "decimal value"]
```

Arguments

Parameter	Type	Description
domain_name	string	Specifies the set of pins that will be initialized. The acceptable values for this argument are the following: • IOsEnableSet - Specifies that the IOsEnableSet set of pins will be initialized. • MemoriesEnableSet - Specifies that the MemoriesEnableSet set of pins will be initialized.
freq	string	Specifies the user input frequency in Hz, MHz, or KHz.
proba	decimal	Specifies the user input probability in percentage(%).

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'domain_name' is missing.
None	Parameter 'domain_name' has illegal value.
None	domain_name: Invalid argument value: 'value' (expecting IOsEnableSet or MemoriesEnableSet).
None	Parameter 'proba' has illegal value.
None	proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'freq' has illegal value.
None	freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz or MHz)).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_init_setofpins_values -domain_name "IOsEnableSet | MemoriesEnableSet" [-freq "decimal value [unit { Hz | KHz | MHz }]"] [-proba "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2

PolarFire SoC

Example

The following example initializes all primary inputs after executing "smartpower_init_do with":

```
smartpower_init_do with -othersets {true}
smartpower_init_setofpins_values -domain_name {IOsEnableSet} \
                                 -freq {10 MHz} \
                                 -proba{20}
```

See Also

- smartpower_init_do

5.22. smartpower_init_set_othersets_options (Ask a Question)

Description

This Tcl command initializes the frequency and probability of all other sets.

Note: This command is associated with the functionality of Initialize Frequencies and Probabilities dialog box.

```
smartpower_init_set_othersets_options \
                [-freq "decimal value [unit { Hz | KHz | MHz }]"] \
                [-proba "decimal value"] \
                [-with "fixed | default"] \
                [-input_freq "decimal value [unit { Hz | KHz | MHz }]"] \
                [-input_proba "decimal value"]
```

Arguments

Parameter	Type	Description
freq	string	Specifies the default frequency and units(possible values are Hz, KHz or MHz). This parameter is optional.
proba	decimal	Specifies the default probability. Must be positive decimal value, less than or equal to 100.000. This parameter is optional.
with	string	Specifies vectorless or default analysis. This parameter is optional.
input_freq	decimal	Specifies the input frequency(decimal possible value) and units(possible values are Hz, KHz or MHz). This parameter is optional.
input_proba	decimal	Specifies the input probability. Must be positive decimal value, less than or equal to 100.000. This parameter is optional.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	input_proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'input_proba' has illegal value.
None	Parameter 'input_proba' must be less than or equal to 100.000.
None	Parameter 'input_proba' must be a positive decimal value.
None	input_freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz or MHz)).
None	Parameter 'input_freq' has illegal value.
None	with: Invalid argument value: 'value' (expecting default or fixed).
None	Parameter 'with' has illegal value.
None	proba: Invalid argument value: 'value' (expecting decimal value).

smartpower_init_set_othersets_options (continued)	
Error Code	**Description**
None	Parameter 'proba' must be less than or equal to 100.000.
None	Parameter 'proba' must be a positive decimal value.
None	Parameter 'proba' has illegal value.
None	freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz or MHz)).
None	Parameter 'freq' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_init_set_othersets_options [-freq "decimal value [unit { Hz | KHz | MHz }]"] [-proba "decimal value"] [-input_freq "decimal value [unit { Hz | KHz | MHz }]"] [-input_proba "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all other sets after executing "smartpower_init_do" with "-othersets {true}":

Note: Warning: -with flag is obsolete; use -with argument of the command smartpower_init_do instead.

```
smartpower_init_set_othersets_options \
                         -freq "10 MHz" \
                         -proba "10" \
             [-with "fixed"] \
                         -input_freq "10 MHz" \
                         -input_proba "10"
```

See Also

- smartpower_init_do

5.23. smartpower_init_set_primaryinputs_options (Ask a Question)

Description

This Tcl command initializes the frequency and probability of all primary inputs.

```
smartpower_init_set_primaryinputs_options \
              [-freq "decimal value [unit { Hz | KHz | MHz | % }]"] \
              [-proba "decimal value"]
```

Arguments

Parameter	Type	Description
freq	string	Specifies the user input frequency (in Hz, KHz or MHz) or the toggle rate in percentage(%). If the unit is not provided and toggle rate is active, the value is handled as a toggle rate, if toggle rate is not active, the value is handled as a frequency.
proba	decimal	Specifies the user input probability in percentage(%). Must be a positive decimal value and less than or equal to 100.000.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'proba' has illegal value.
None	proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'proba' must be a positive decimal value.
None	Parameter 'proba' must be less than or equal to 100.000.
None	Parameter 'freq' has illegal value.
None	freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz, MHz or %)).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_init_set_primaryinputs_options [-freq "decimal value [unit {Hz | KHz | MHz | %}]"] [-proba "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all primary inputs after executing smartpower_init_do with -primaryinputs {true}:

```
smartpower_init_set_primaryinputs_options -freq {10 MHz} -proba {20}
```

See Also

- smartpower_init_set_combinational_options
- smartpower_change_clock_statistics

5.24. smartpower_init_set_registers_options (Ask a Question)

Description

This Tcl command initializes the frequency and probability of all register outputs.

```
smartpower_init_set_registers_options \
        [-with "default"] \
            [-freq "decimal value [unit { Hz | KHz | MHz | % }]"] \
        [-proba "decimal value"]
```

Arguments

Parameter	Type	Description
freq	string	Specifies the user input frequency (in Hz, KHz, or MHz) or the toggle rate in percentage(%). If the unit is not provided and togglerate is active, the value is handled as a toggle rate; if toggle rate is not active, the value is handled as a frequency.
proba	decimal	Specifies the user input probability in percentage(%). Must be less than or equal to 100.000.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'proba' has illegal value.
None	Parameter 'proba' must be less than or equal to 100.000.
None	proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'freq' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_init_set_registers_options [-freq "decimal value [unit { Hz | KHz | MHz | % }]"] [-proba "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all register outputs after executing "smartpower_init_do with -registers {true}":

```
smartpower_init_set_registers_options -freq {10 MHz} \
                                      -proba {20}
```

5.25. smartpower_init_set_set_reset_options (Ask a Question)

Description

This Tcl command initializes the frequency and probability of all set and reset nets.

```
smartpower_init_set_set_reset_options \
         [-freq "decimal value [unit { Hz | KHz | MHz | % }]"] \
         [-proba "decimal value"]
```

Arguments

Parameter	Type	Description
freq	decimal	Specifies the user input frequency (in Hz, KHz, or MHz) or the toggle rate (in %). If the unit is not provided and toggle rate is active, the value is handled as a toggle rate, if toggle rate is not active, the value is handled as a frequency. This parameter is optional.
proba	decimal	Specifies the user input probability in persentage(%). This parameter is optional.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'proba' must be a positive decimal value.
None	Parameter 'proba' must be less than or equal to 100.000.
None	proba: Invalid argument value: 'value' (expecting decimal value).

smartpower_init_set_set_reset_options (continued)	
Error Code	**Description**
None	proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'proba' has illegal value.
None	freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz, MHz or %)).
None	Parameter 'freq' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_init_set_set_reset_options [-freq "decimal value [unit { Hz | KHz | MHz | % }]"] [-proba "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all set/reset nets after executing "smartpower_init_do" with "-set_reset {true}":

```
smartpower_init_do -set_reset {true}
smartpower_init_set_set_reset_options -freq {10 MHz} -proba {20}
```

See Also

- smartpower_init_do

5.26. smartpower_remove_all_annotations (Ask a Question)

Description

This Tcl command removes all initialization annotations for the specified mode.

Note: This command is associated with the functionality of Initialize frequencies and probabilities dialog box.

```
smartpower_remove_all_annotations [-opmode {value}]
```

Arguments

Parameter	Type	Description
opmode	string	Removes all initialization annotations for the specified mode, where value must be Active or Flash*Freeze (RTG4, SmartFusion 2 and IGLOO 2).

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'opmode' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_remove_all_annotations [-opmode "Active"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example initializes all clocks with opmode "Acitve":

```
smartpower_remove_all_annotations -opmode {Active}
```

5.27. smartpower_remove_custom_mode (Ask a Question)

Description

This Tcl command removes a custom mode from the current design.

```
smartpower_remove_custom_mode -name {deleted mode name}
```

Arguments

Parameter	Type	Description
name	string	Specifies the name of the custom mode to be removed. This parameter is mandatory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'name' is missing.
None	Parameter 'name' has illegal value.
None	A custom mode with name "mode_name" does not exist.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_remove_custom_mode -name "mode to deleted"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example removes a "MyCustomMode" custom mode from the current design:

```
smartpower_remove_custom_mode -name {MyCustomMode}
```

See Also

- smartpower_add_new_custom_mode
- smartpower_edit_custom_mode

5.28. smartpower_remove_domain (Ask a Question)

Description

This Tcl command removes an existing clock or set domain.

```
smartpower_remove_domain -domain_name {domain name} -domain_type "set | clock"
```

Arguments

Parameter	Type	Description
domain_type	string	Specifies the type of domain to remove. The acceptable values for this argument are: • clock - The domain is a clock domain. • set - The domain is a set domain.
domain_name	string	Specifies the name of the domain to remove.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'domain_name' is missing.
None	Parameter 'domain_name' has illegal value.
None	domain_type: Invalid argument value: 'value' (expecting set or clock).
None	Could not delete domain 'domain name'.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_remove_domain -domain_name "domain name" -domain_type "set | clock"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example removes the clock domain names "myclk":

```
smartpower_remove_domain -domain_type {clock} -domain_name {myclk}
```

This example removes the set domain names "myset":

```
smartpower_remove_domain -domain_type {set} -domain_name {myset}
```

See Also

- smartpower_create_domain

5.29. smartpower_remove_file (Ask a Question)

Description

This Tcl command removes the VCD file from the specified mode or all operating mode. Frequency and probability information of signals annotated by the VCD are set back to the default value.

```
remove file -file {the name of the VCD file} \
            -format {the type of the removed file} \
            -opmode {mode name}
```

Arguments

Parameter	Type	Description
file	string	Specifies the VCD file to be removed. This is mandatory.
format	string	Specifies that the type to be removed is a VCD file. This is mandatory.
opmode	string	Specifies the operating mode. This is optional. Below shows the acceptable values for this argument: • Active(PolarFire®, SmartFusion® 2, IGLOO® 2, RTG4™, PolarFire SoC) - the operating mode is set to Active. • Flash*Freeze(SmartFusion 2, IGLOO 2) - the operating mode is set to Flash*Freeze. • Custom mode name - the operating mode is custom cerated mode.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'format' has illegal value.
None	Required parameter 'format' is missing.
None	Parameter 'file' has illegal value.
None	Required parameter 'file' is missing.
None	opmode: Invalid argument value: 'mode_name' (expecting Active, Static, Flash*Freeze, Shutdown, Sleep or CustomMode).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_remove_file -file "file name" -format "vcd" [-opmode "Active | Static"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example removes the file test.vcd from the Active mode:

```
smartpower_remove_file -file "test.vcd" -format VCD -opmode "Active"
```

This example removes the VCD file power1.vcd from all operating modes:

```
smartpower_remove_file -file “power1.vcd” -format VCD
```

See Also

- smartpower_import_vcd

5.30. smartpower_remove_pin_frequency (Ask a Question)

Description

This Tcl command removes the frequency associated with a specific pin. This pin will have a default frequency based on its domain.

Note: The pin_name must be the name of a pin that already exists in the design and already belongs to a domain. Execute the "smartpower_commit" Tcl command for preserving any changes in SmartPower.

```
smartpower_remove_pin_frequency -pin_name {pin_name}
```

Arguments

Parameter	Type	Description
pin_name	string	Specifies the name of the pin for which the frequency will be removed.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'pin_name' is missing.
None	Parameter 'pin_name' is missing or has invalid value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_remove_pin_frequency [-pin_name "name of pin"]+'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example removes the frequency from the pin named "d_in":

```
smartpower_remove_pin_frequency -pin_name {d_in}
```

See Also

- smartpower_set_pin_frequency

5.31. smartpower_remove_pin_of_domain (Ask a Question)

Description

This Tcl command removes a clock pin or a data pin from a clock or set domain, respectively.

Note: The domain name must be the name of an existing domain. The pin name must be the name of an existing pin. Execute the "smartpower_commit" Tcl command for preserving any changes in SmartPower.

```
smartpower_remove_pin_of_domain [-pin_name {name of pin}] \
                                -pin_type {clock | data} \
```

```
                                -domain_name {name of domain} \
                                -domain_type {clock | set}
```

Arguments

Parameter	Type	Description
pin_name	string	Specifies the name of the pin to remove from the domain.
pin_type	string	Specifies the type of the pin to remove. The acceptable values for this argument are the following: • clock - The pin to remove is a clock pin. • set - The pin to remove is a data pin.
domain_name	string	Specifies the name of the domain from which to remove the pin.
domain_type	string	Specifies the type of domain from which the pin is being removed. The acceptable values for this argument are the following: • clock - The domain is a clock domain. • set - The domain is a set domain.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'domain_type' is missing.
None	domain_type: Invalid argument value: 'type name' (expecting clock or set).
None	Required parameter 'domain_name' is missing.
None	Required parameter 'pin_name' is missing.
None	Required parameter 'pin_type' is missing.
None	pin_type: Invalid argument value: 'type_name' (expecting clock or data).
None	Parameter 'pin_name' is missing or has invalid value.
None	Parameter 'domain_name' is missing or has invalid value.
None	Failed to remove pin "in" from domain 'domain_value'.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_remove_pin_of_domain [-pin_name "name of pin"]+ -domain_type "clock | set" -domain_name "name of domain" -pin_type "clock | data"'.

Supported Families

PolarFire®

SmartFusion® 2

RTG4™

IGLOO® 2

PolarFire SoC

Example

The following example removes the clock pin named "XCMP3/UO/U1:Y" from the clock domain named "myclk":

```
smartpower_remove_pin_of_domain -pin_name {XCMP3/U0/U1:Y} \
                                -pin_type {clock} \
                                -domain_name {myclk} \
                                -domain_type {clock}
```

The following example removes the data pin named "count" from the set domain named "InputSet":

```
smartpower_remove_pin_of_domain -pin_name {count} \
                                -pin_type {data} \
                                -domain_name {InputSet} \
                                -domain_type {set}
```

5.32. smartpower_remove_pin_probability (Ask a Question)

Description

This tcl command removes the probability value associated with a specific pin. This pin will have a default probability based on the domain set it belongs to.

```
smartpower_remove_pin_probability -pin_name "name of pin"
```

Arguments

Parameter	Type	Description
pin_name	string	Specifies the name of the pin with the probability to remove. This pin must be the direct driver of an enable pin. There may be multiple `-pin_name` arguments. This is mandatory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'pin_name' is missing.
None	Parameter 'pin_name' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_remove_pin_probability [-pin_name "name of pin"]+'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example removes the probability of the pin driving the enable pin of a bidirectional I/O:

```
smartpower_remove_pin_probability -pin_name "mybibuf/U0/U1:EOUT"
```

See Also

- smartpower_set_pin_probability

5.33. smartpower_remove_scenario (Ask a Question)

Description

This Tcl command removes a custom scenario from the current design.

```
smartpower_remove_scenario -name {Scenario name}
```

Arguments

Parameter	Type	Description
name	string	Specifies the name of the custom scenario to be removed. This parameter is mandatory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'name' has illegal value.
None	Required parameter 'name' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_remove_scenario -name "scenario name"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example removes a "MyScenario" custom scenario from the current design:

```
smartpower_remove_scenario -name {MyScenario}
```

See Also

- smartpower_add_new_scenario
- smartpower_edit_scenario

5.34. smartpower_report_power (Ask a Question)

Description

This Tcl command creates a power report, which enables you to determine if you have any power consumption problems in your design. It includes information about the global device and SmartPower preferences selection and hierarchical details (including gates, blocks, and nets), with a block-by-block, gate-by-gate, and net-by-net power summary SmartPower results.

Notes:

- Flash*Freeze is available only for certain families and devices(RTG4, SmartFusion 2 and IGLOO 2).
- Worst and Best are available only for certain families and devices.

```
smartpower_report_power \
              [-powerunit "W | mW | uW"] \
              [-frequnit "Hz | KHz | MHz"] \
              [-opcond "Best | Typical | Worst"] \
              [-opmode "Active | Static"] \
              [-toggle "TRUE | FALSE"] \
              [-power_summary "TRUE | FALSE"] \
              [-rail_breakdown "TRUE | FALSE"] \
              [-type_breakdown "TRUE | FALSE"] \
              [-clock_breakdown "TRUE | FALSE"] \
              [-thermal_summary "TRUE | FALSE"] \
              [-battery_life "TRUE | FALSE"] \
              [-clock_summary "TRUE | FALSE"] \
```

```
[-opcond_summary "TRUE | FALSE"] \
[-annotation_coverage "TRUE | FALSE"] \
[-style "Text | CSV"] \
[-sortorder "Ascending | Descending"] \
[-sortby "Alphabetical | Power Values"] \
[-instance_breakdown "TRUE | FALSE"] \
[-power_threshold "TRUE | FALSE"] \
[-filter_instance "TRUE | FALSE"] \
[-min_power "decimal value [unit { W | mW | uW }]"] \
[-max_instance "integer value"] \
[-activity_sortorder "Ascending | Descending"] \
[-activity_sortby "Pin Name | Net Name | Domain | Frequency | Source"] \
[-activity_summary "TRUE | FALSE"] \
[-frequency_threshold "TRUE | FALSE"] \
[-filter_pin "TRUE | FALSE"] \
[-min_frequency "decimal value [unit { Hz | KHz | MHz }]"] \
[-max_pin "integer value"] \
[-enablerates_summary "TRUE | FALSE"] \
[-enablerates_sortorder "Ascending | Descending"] \
[-enablerates_sortby "Pin Name | Net Name | Type | Polarity | Rate | Source"] \
{filename}
```

Arguments

Parameter	Type	Description
powerunit	string	Specifies the unit in which power is set. The acceptable values for this argument are the following: • W - The power unit is set to watts. • mW - The power unit is set to milliwatts. • uW - The power unit is set to microwatts.
frequnit	string	Specifies the unit in which frequency is set. The acceptable values for this argument are the following: • Hz - The frequency unit is set to hertz. • kHz - The frequency unit is set to kilohertz. • MHz - The frequency unit is set to megahertz.
opcond	string	Specifies the operating condition. The following acceptable values for this argument are the following: • worst - The operating condition is set to worst case. • typical - The operating condition is set to typical case. • best - The operating condition is set to best case.
opmode	string	Specifies the operating mode. The following acceptable values for this argument are the following: • Active - The operating mode is set to Active. • Static - The operating mode is set to Static. • Flash*Freeze - The operating mode is set to Flash*Freeze.
toggle	boolean	Specifies the toggle. The acceptable values for this argument are thr following: • TRUE, true or 1 - The toggle is set to true. • FALSE, false or 1 - The toggle is set to false.
power_summary	boolean	Specifies whether to include the power summary, which shows the static and dynamic values in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the power summary in the report. • FALSE, false or 0 - Does not include the power summary in the report.

smartpower_report_power (continued)		
Parameter	**Type**	**Description**
rail_breakdown	boolean	Specifies whether to include the breakdown by rail summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the breakdown by rail summary in the report. • FALSE, false or 0 - Does not include the breakdown by rail summary in the report.
type_breakdown	boolean	Specifies whether to include the breakdown by type summary in the report. The acceptable values for this argument are thr following: • TRUE, true or 1 - Includes the breakdown by type summary in the report. • FALSE, false or 0 - Does not include the breakdown by type summary in the report.
clock_breakdown	boolean	Specifies whether to include the breakdown by clock domain in the report. The acceptable values for this argument are thr following: • TRUE, true or 1 - Includes the breakdown by clock domain summary in the report. • FALSE, false or 0 - Does not include the breakdown by clock domain summary in the report.
thermal_summary	boolean	Specifies whether to include the thermal summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the thermal summary in the report. • FALSE, false or 0 - Does not include the thermal summary in the report.
battery_life	boolean	Specifies whether to include the battery life summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the battery life summary in the report. • FALSE, false or 0 - Does not include the battery life summary in the report.
opcond_summary	boolean	Specifies whether to include the operating conditions summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the operating conditions summary in the report. • FALSE, false or 0 - Does not include the operating conditions summary in the report.
clock_summary	boolean	Specifies whether to include the clock domains summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the clock summary in the report. • FALSE, false or 0 - Does not include the clock summary in the report.
style	string	Specifies the format in which the report will be exported. The acceptable values for this argument are the following: • Text - The report will be exported as Text file. • CSV(by default) - The report will be exported as CSV file.
sortby	string	Specifies how to sort the values in the report. The acceptable values for this argument are the following: • power values - Sorts based on the power values. • alphabetical - Sorts in an alphabetical order.

smartpower_report_power (continued)		
Parameter	**Type**	**Description**
sortorder	string	Specifies the sort order of the values in the report. The acceptable values for this argument are the following: • ascending - Sorts the values in ascending order. • descending - Sorts the values in descending order.
instance_breakdown	boolean	Specifies whether to include the breakdown by instance in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the breakdown by instance in the report. • FALSE, false or 0 - Does not include the breakdown by instance in the report.
power_threshold	boolean	This specifies whether to include only the instances that consume power above a certain minimum value. Whenthis command is set to true, the -min_power argument must also be used to specify that only the instances thatconsume power above this minimum power value are the ones that are included in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the power threshold in the report. • FALSE, false or 0 - Does not include the power threshold in the report.
filter_instance	boolean	This specifies whether to have a limit on the number of instances to include in the Power report. When this commandis set to true, the `-max_instance` argument must also be used to specify the maximum number of instances to beincluded into the Power report. The acceptable values for this argument are the following: • TRUE, true or 1 - Indicates that you want to have a limit on the number of instances to include in the Powerreport. • FALSE, false or 0 - Indicates that you do not want to have a limit on the number of instances to include in thePower report.
min_power	decimal	Specifies which block to expand based on the minimum power value of a block.
max_instance	integer	Sets the maximum number of instances to a specified integer greater than or equal to 0 (zero). This will limit the maximum number of instances to be included in the Power report.
activity_sortorder	string	Specifies the sort order for the activity summary. The acceptable values for this argument are the following: • ascending - Sorts the values in ascending order. • descending - Sorts the values in descending order.
activity_sortby	string	Specifies how to sort the values for the activity summary. The acceptable values for this argument are the following: • pin name - Sorts based on the pin name. • net name - Sorts based on thepnet name. • domain - Sorts based on the clock domain. • frequency - Sorts based on the clock frequency. • source - Sorts based on the clock frequency source.
activity_summary	boolean	Specifies whether to include the activity summary in the report. The following acceptable values for this argument are the following: • TRUE, true or 1 - Includes the activity summary in the report. • FALSE, false or 0- Does not include the activity summary in the report.

smartpower_report_power (continued)		
Parameter	**Type**	**Description**
frequency_threshold	boolean	Specifies whether to add a frequency threshold. The acceptable values for this argument are the following: • TRUE, true or 1 - Adds a frequency threshold. • FALSE, false or 0 - Does not add a frequency threshold.
filter_pin	boolean	Specifies whether to filter by maximum number of pins. The acceptable values for this argument are the following: • TRUE, true or 1 - Filters by maximum number of pins. • FALSE, false or 0 - Des not filter by maximum number of pins.
min_frequency	decimal	Sets the minimum frequency to {decimal value [unit { Hz | KHz | MHz}]}.
max_pin	integer	Sets the maximum number of pins.
enablerates_sortorder	string	Specifies the sort order for the probabilities summary. The acceptable values for this argument are the following: • ascending - Sorts the values in ascending order. • descending - Sorts the values in descending order.
enablerates_sortby	string	Specifies how to sort the values for the probabilities summary. The acceptable values for this argument are the following: • pin name - Sorts based on the pin name. • net name - Sorts based on the net name. • domain - Sorts based on the clock domain. • frequency - Sorts based on the clock frequency. • source - Sorts based on the clock frequency source.
enablerates_summary	boolean	Specifies whether to include the probabilities summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the activity summary in the report. • FALSE, false or 0 - Does not include the activity summary in the report.
with_annotation_coverage	boolean	Specifies whether to include the annotation coverage summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the annotation coverage summary in the report. • FALSE, false or 0 - Does not include the annotation coverage summary in the report.
filename	string	Specifies the name or path of the file to be exported. This argument is mandatory. Default created under designer/<root_name> dircetory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	opcond: Invalid argument value: 'value' (expecting Best, Typical or Worst).
None	powerunit: Invalid argument value: 'value' (expecting W, mW or uW).
None	opmode: Invalid argument value: 'value' (expecting Active, Static or Flash*Freeze).
None	frequnit: Invalid argument value: 'value' (expecting Hz, KHz or MHz).
None	toggle: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).

smartpower_report_power (continued)	
Error Code	**Description**
None	power_summary: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	rail_breakdown: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	type_breakdown: Invalid argument value: ' value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	clock_breakdown: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	thermal_summary: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	battery_life: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	power_threshold: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	min_power: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are W, mW or uW)).
None	filter_instance: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	max_instance: Invalid argument value: 'integer' (expecting integer value).
None	Parameter 'max_instance' must be a positive integer value.
None	activity_sortorder: Invalid argument value: 'value' (expecting Ascending or Descending).
None	activity_summary: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	frequency_threshold: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	filter_pin: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	max_pin: Invalid argument value: 'value' (expecting integer value).
None	Parameter 'max_pin' must be a positive integer value.
None	enablerates_sortorder: Invalid argument value: 'value' (expecting Ascending or Descending).
None	enablerates_sortby: Invalid argument value: 'value' (expecting Pin Name, Net Name, Type, Polarity, Rate or Source).
None	enablerates_summary: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example generates a Power report named power_report.xml:

```
smartpower_report_power -frequnit "MHz" \
                        -opcond "Typical" \
                        -opmode "Active" \
                        -toggle "TRUE" \
                        -battery_life "TRUE" \
                        -style "TEXT" \
```

```
                        -power_summary "TRUE" \
                        -activity_sortby "Source" \
                        power_report.txt
```

5.35. smartpower_report_power_activity_map (Ask a Question)

Description

This Tcl command the activity and hazards report reads a VCD file and reports transitions and hazards for each clock cycle of the VCD file.

```
smartpower_report_power_activity_map \
              -vcd_file "VCD file" \
              [-style "Text | CSV"] \
              [-partial_parse "TRUE | FALSE"] \
              [-start_time "decimal value"] \
              [-end_time "decimal value"] \
              [-auto_detect_top_level_name "TRUE | FALSE"] \
              [-top_level_name "top level name"] \
              -report_query "Report by Cycle - summary | Report by Net - summary | Report by
Cycle - detailed | Report by Net - detailed"\
              [-report_type "activity and power | activity | power"] \
              [-sortby "functional transitions | total power | spurious transitions |
functional power | spurious power"] \
              [-sortorder "ascending | descending"] \
              [-max_cycle "integer value"] \
              [-max_net "integer value"] \
              [-clock_settings "":""] \
              [-glitch_filtering "false | auto | true"] \
              [-glitch_threshold "integer value"] \
              [-auto_construct_clock_domain "TRUE | FALSE"] \
              [-clock_period "decimal value"] \
              [-clock_offset "decimal value"] \
              [-opmode "Active | Static"] \
              [-file "file"]
```

Arguments

Parameter	Type	Description
vcd_file	string	Specifies the path to the *.vcd file that you want to import.
style	string	Specifies the format in which the report will be exported. The acceptable values for this argument are the following: • Text - The report will be exported as Text file. • CSV(by default) - The report will be exported as CSV file.
partial_parse	boolean	Specifies whether to partially parse the *.vcd file. The acceptable values for this argument are the following: • TRUE, true or 1 - Partially parses the *.vcd file. • FALSE, false or 0 - Does not partially parse the *.vcd file.
start_time	decimal	This option is available only if `-partially_parse` is set to true. Specifies the start time (in ns) to partially parse the *.vcd file.
end_time	decimal	This option is available only if `-partially_parse` is set to true. Specifies the start time (in ns) to partially parse the *.vcd file.
auto_detect_top_level_name	boolean	Specifies whether to automatically detect the top-level name. The acceptable values for this argument are the following: • TRUE, true or 1 - Automatically detects the top-level name. • FALSE, false or 0 - Does not automatically detect the top-level name.
top_level_name	string	Specifies the top-level name.

smartpower_report_power_activity_map (continued)

Parameter	Type	Description
report_type	string	Specifies the report query type. The acceptable values for this argument are the following: • activity - Includes activity information for each net. • power - Includes power information for each net. • activity and power - Includes activity and power information for each net.
report_query	string	Specifies the report type. The acceptable values for this argument are the following: • Report by Net - summary - Provides a summary report for each net. • Report by Net - detailed - Provides a detailed report for each net. • Report by Cycle - summary - Provides a summary report for each cycle. • Report by Cycle - detailed - Provides a detailed report for each cycle.
sortby	string	Specifies how to sort the values in the report. The acceptable values for this argument are the following: • total power - Sorts based on the power values. • spurious power - Sorts based on the spurious power. • functional power - Sorts based on the functional values. • spurious transitions - Sorts based on the spurious transitions. • functional transitions - Sorts based on the functional transitions.
sortorder	string	Specifies the sort order of the values in the report. This could be descending or ascending.
max_net	integer	Specifies the maximum number of nets to report. In a net summary or net details report, this argument limits the total number of entries. In a cycle details report, this argument limits the number of nets reported for each cycle.
max_cycle	integer	Specifies the maximum number of cycles to report. In a cycle summary or cycle details report, this argument limits the total number of entries. In a net details report, this argument limits the number of cycles reported for each net.
clock_settings	string	Specifies the settings for the clock. The format is ">clock name<:>active edge {value}<". The acceptable values for this argument are the following: • rising - Sets the clock to a rising active edge. • falling - Sets the clock to a falling active edge. • both - Sets the clock to both rising and falling active edge. • not_active - Does not use the signal as a clock.
glitch_filtering	string	Specifies whether to use glitch filtering. The acceptable values for this argument are the following: • TRUE, true or 1 - Glitch filtering is on. • FALSE, false or 0 - Enables automatic glitch filtering. This option will ignore any value specified in -glitch_threshold. • auto - Glitch filtering is off.
glitch_threshold	integer	This option is only available when `-glitch_filtering` is set to true. Specifies the glitch filtering value(in ps).

smartpower_report_power_activity_map (continued)		
Parameter	**Type**	**Description**
auto_construct_clock_domain	boolean	Specifies whether to automatically construct the clock domain. The acceptable values for this argument are the following: • TRUE, true or 1 - Automatically constructs the clock domain. • FALSE, false or 0 - Does not automatically construct the clock domain.
clock_period	decimal	Use this option to specify a virtual clock period (in ps). This should be used if `-auto_construct_clock_domain` is set to false.
clock_offset	decimal	Use this option to specify the time of the first active edge of the virtual clock (in ps). This should be used if `-auto_construct_clock_domain` is set to false.
opmode	string	Use this option to specify the mode from which the operating conditions are extracted to generate the report. • Active - The operating mode is set to Active. • Flash*Freeze - The operating mode is set to Flash*Freeze.
filename	string	Specifies the name or path of the file to be exported. This argument is mandatory. Default created under designer/<root_name> dircetory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'vcd_file' is missing.
None	Required parameter 'report_query' is missing.
None	VCD: Incorrect VCD file. "enddefinitions" not found.
None	VCD: File ./power.vcd doesn't exist: Failed to generate power report.
None	style: Invalid argument value: 'value' (expecting Text or CSV).
None	partial_parse: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	start_time: Invalid argument value: 'value' (expecting decimal value).
None	end_time: Invalid argument value: 'value' (expecting decimal value).
None	auto_detect_top_level_name: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	report_type: Invalid argument value: 'value' (expecting activity and power, activity or power).
None	report_query: Invalid argument value: 'value' (expecting Report by Cycle - summary, Report by Net - summary, Report by Cycle - detailed or Report by Net - detailed).
None	sortby: Invalid argument value: 'value' (expecting functional transitions, total power, spurious transitions, functional power or spurious power).
None	sortorder: Invalid argument value: 'value' (expecting ascending or descending).
None	max_net: Invalid argument value: 'value' (expecting integer value).
None	max_cycle: Invalid argument value: 'value' (expecting integer value).
None	glitch_filtering: Invalid argument value: 'value' (expecting false, auto or true).
None	glitch_threshold: Invalid argument value: 'value' (expecting integer value).
None	auto_construct_clock_domain: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	clock_period: Invalid argument value: 'value' (expecting decimal value).
None	clock_offset: Invalid argument value: 'value' (expecting decimal value).

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example generates an activity and hazards power report named "report_power_activity_map.txt":

```
smartpower_report_power_activity_map \
                -partial_parse "TRUE" \
                -start_time "1.0" \
                -end_time "3.0" \
                -auto_detect_top_level_name "TRUE" \
                -top_level_name "mytopmodule" \
                -glitch_filtering "auto" \
                -style "Text" \
                -opmode "Active" \
                -vcd_file power.vcd \
                -report_query "Report by Net - summary" \
                -clock_period "2000.0" \
                -clock_offset "10.0" \
                "report_power_activity_map_partial.txt"
```

See Also

- smartpower_report_power
- smartpower_report_power_peak_analyzer

5.36. smartpower_report_power_history (Ask a Question)

Description

Enter description here

```
smartpower_report_power_history
```

Arguments

Parameter	Type	Description
None	None	None

Return Type	Description
None	None

Error Codes

Error Code	Description
None	None

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

Enter example description here

```
smartpower_report_power_history
```

5.37. smartpower_report_power_peak_analyzer (Ask a Question)

Description

This Tcl command creates a cycle accurate power report, which reports a power waveform with one power value per clock period or half-period instead of an average power for the whole simulation.

```
smartpower_report_power_peak_analyzer \
                    -vcd_file "VCD file" \
                    [-style "Text | CSV"] \
                    [-partial_parse "TRUE | FALSE"] \
                    [-start_time "decimal value"] \
                    [-end_time "decimal value"] \
                    [-auto_detect_top_level_name "TRUE | FALSE"] \
                    [-top_level_name "top level name"] \
                    [-glitch_filtering "false | auto | true"] \
                    [-glitch_threshold "integer value"] \
                    [-opmode "Active | Static"] \
                    [-auto_detect_sampling_period "TRUE | FALSE"] \
                    [-sampling_clock "clk"] \
                    [-sampling_rate_per_period "TRUE | FALSE"] \
                    [-sampling_offset "decimal value"] \
                    [-sampling_period "decimal value"] \
                    [-use_only_local_extrema "TRUE | FALSE"] \
                    [-use_power_threshold "TRUE | FALSE"] \
                    [-power_threshold "decimal value"] \
                    [-file "file"]
```

Arguments

Parameter	Type	Description
vcd_file	string	Specifies the path to the *.vcd file that you want to import.
style	string	Specifies the format in which the report will be exported. The acceptable values for this argument are the following: • Text - The report will be exported as Text file. • CSV(by default) - The report will be exported as CSV file.
partial_parse	boolean	Specifies whether to partially parse the *.vcd file. The acceptable values for this argument are the following: • TRUE, true or 1 - Partially parses the *.vcd file. • FALSE, false or 0 - Does not partially parse the *.vcd file.
start_time	decimal	This option is available only if `-partially_parse` is set to true. Specifies the start time (in ns) to partially parse the *.vcd file.
end_time	decimal	This option is available only if `-partially_parse` is set to true. Specifies the start time (in ns) to partially parse the *.vcd file.
auto_detect_top_level_name	boolean	Specifies whether to automatically detect the top-level name. The acceptable values for this argument are the following: • TRUE, true or 1 - Automatically detects the top-level name. • FALSE, false or 0 - Does not automatically detect the top-level name.
top_level_name	string	Specifies the top-level name.

smartpower_report_power_peak_analyzer (continued)		
Parameter	**Type**	**Description**
glitch_filtering	string	Specifies whether to use glitch filtering. The acceptable values for this argument are the following: • TRUE, true or 1 - Glitch filtering is on. • FALSE, false or 0 - Enables automatic glitch filtering. This option will ignore any value specified in -glitch_threshold. • auto - Glitch filtering is off.
glitch_threshold	integer	This option is only available when `-glitch_filtering` is set to true. Specifies the glitch filtering value(in ps).
power_summary	boolean	Specifies whether to include the power summary, which shows the static and dynamic values in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the power summary in the report. • FALSE, false or 0 - Does not include the power summary in the report.
auto_detect_sampling_period	boolean	Specifies whether to automatically detect the sampling period. The acceptable values for this argument are the following: • TRUE, true or 1 - Automatically detects the sampling period. • FALSE, false or 0 - Does not automatically detect the sampling period.
sampling_clock	string	Specifies the sampling clock.
sampling_rate_per_period	boolean	Specifies whether to set the sampling rate per period. The acceptable values for this argument are the following: • TRUE, true or 1 - Specifies the sampling rate per period. • FALSE, false or 0 - Specifies the sampling rate per half period.
sampling_offset	decimal	Specifies the offset used to calculate the sampling offset (in ps).
sampling_period	decimal	Specifies the offset used to calculate the sampling period (in ps).
use_only_local_extrema	boolean	Specifies whether to limit the history size by keeping only local extrema. The acceptable values for this argument are the following: • TRUE, true or 1 - Limits the history size by keeping only local extrema. • FALSE, false or 0 - Does not limit the history size by keeping only local extrema.
use_power_threshold	boolean	Specifies whether to limit the history size by setting a power threshold. The acceptable values for this argument are the following: • TRUE, true or 1 - Limits the history size by setting a power threshold. • FALSE, false or 0 - Does not limit the history size by setting a power threshold.
power_threshold	decimal	Sets the power threshold value.
opmode	string	Use this option to specify the mode from which the operating conditions are extracted to generate the report. • Active - The operating mode is set to Active. • Flash*Freeze - The operating mode is set to Flash*Freeze.
filename	string	Specifies the name or path of the file to be exported. This argument is mandatory. Default created under designer/<root_name> dircetory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'vcd_file' is missing.
None	VCD: Incorrect VCD file. "enddefinitions" not found.
None	VCD: File ./power.vcd doesn't exist: Failed to generate power report.
None	style: Invalid argument value: 'value' (expecting Text or CSV).
None	partial_parse: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	start_time: Invalid argument value: 'value' (expecting decimal value).
None	end_time: Invalid argument value: 'value' (expecting decimal value).
None	auto_detect_top_level_name: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	glitch_filtering: Invalid argument value: 'value' (expecting false, auto or true).
None	glitch_threshold: Invalid argument value: 'value' (expecting integer value).
None	auto_detect_sampling_period: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	sampling_clock: Invalid argument value: 'value' (expecting "sampling clock pin name").
None	sampling_rate_per_period: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	sampling_offset: Invalid argument value: 'value' (expecting decimal value).
None	sampling_period: Invalid argument value: 'value' (expecting decimal value).
None	use_only_local_extrema: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	use_power_threshold: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	power_threshold: Invalid argument value: 'value' (expecting decimal value).
None	opmode: Invalid argument value: 'value' (expecting Active, Static or Flash*Freeze).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_report_power_peak_analyzer [-partial_parse "TRUE | FALSE"] \ [-start_time "decimal value"] \ [-end_time "decimal value"] \ [-auto_detect_top_level_name "TRUE | FALSE"] \ [-top_level_name "top level name"] \ [-glitch_filtering "false | auto | true"] \ [-glitch_threshold "integer value"] \ -vcd_file "VCD file" \ [-style "Text | CSV"] \ [-opmode "Active | Static"] \ [-auto_detect_sampling_period "TRUE | FALSE"] \ [-sampling_clock "clk"] \ [-sampling_rate_per_period "TRUE | FALSE"] \ [-sampling_offset "decimal value"] \ [-sampling_period "decimal value"] \ [-use_only_local_extrema "TRUE | FALSE"] \ [-use_power_threshold "TRUE | FALSE"] \ [-power_threshold "decimal value"] \ [-file "file"]'

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example generates a cycle accurate power report named "report_power_cycle_based.txt":

```
smartpower_report_power_peak_analyzer \
            -partial_parse "TRUE" \
                    -vcd_file "power.vcd" \
                    -style "Text" \
```

```
                    -start_time "0.05" \
                    -end_time "1.00" \
                    -auto_detect_top_level_name "TRUE" \
                    -glitch_filtering "FALSE" \
                    -glitch_threshold "100" \
                    -auto_detect_sampling_period "TRUE" \
                    -sampling_clock "clk" \
                    -sampling_rate_per_period "TRUE" \
                    -sampling_offset "0.00" \
                    -sampling_period "10000.00" \
                    -use_only_local_extrema "TRUE" \
                    -use_power_threshold "TRUE" \
                    -power_threshold "0.00" \
                    -opmode "Active" \
                    "report_power_cycle_based.txt"
```

See Also

- smartpower_report_power
- smartpower_report_power_activity_map

5.38. smartpower_report_power_sequencer (Ask a Question)

Description

This Tcl command creates a scenario power report for a previously defined scenario. It includes information about the global device and SmartPower preferences selection, and the average power consumption and the excepted battery life for this sequence.

Notes:

- Flash*Freeze is available only for certain families and devices(RTG4, SmartFusion 2, and IGLOO 2).
- Worst and Best are available only for certain families and devices.

```
smartpower_report_power_sequencer \
                  [-powerunit "W | mW | uW"] \
                  [-frequnit "Hz | KHz | MHz"] \
                  [-opcond "Best | Typical | Worst"] \
                  [-toggle "TRUE | FALSE"] \
                  [-scenario ""] \
                  [-style "Text | CSV | XML"] \
                  [-battery_life "TRUE | FALSE"] \
                  [-battery_capacity "decimal value"] \
                  [-rail_breakdown "TRUE | FALSE"] \
                  [-type_breakdown "TRUE | FALSE"] \
                  [-mode_breakdown "TRUE | FALSE"] \
                  [-opcond_summary "TRUE | FALSE"] \
                  {filename}
```

Arguments

Parameter	Type	Description
powerunit	string	Specifies the unit in which power is set. The acceptable values for this argument are the following: • W - The power unit is set to watts. • mW - The power unit is set to milliwatts. • uW - The power unit is set to microwatts.
frequnit	string	Specifies the unit in which frequency is set. The acceptable values for this argument are the following: • Hz - The frequency unit is set to hertz. • kHz - The frequency unit is set to kilohertz. • MHz - The frequency unit is set to megahertz.

smartpower_report_power_sequencer (continued)		
Parameter	**Type**	**Description**
opcond	string	Specifies the operating condition. The following acceptable values for this argument are the following: • worst - The operating condition is set to worst case. • typical - The operating condition is set to typical case. • best - The operating condition is set to best case.
toggle	boolean	Specifies the toggle. The acceptable values for this argument are thr following: • TRUE, true or 1 - The toggle is set to true. • FALSE, false or 1 - The toggle is set to false.
scenario	string	Specifies a scenario that the report is generated from.
style	string	Specifies the format in which the report will be exported. The acceptable values for this argument are the following: • Text - The report will be exported as Text file. • CSV(by default) - The report will be exported as CSV file. • XML - The report will be exported as XML file.
battery_life	boolean	Specifies whether to include the battery life summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the battery life summary in the report. • FALSE, false or 0 - Does not include the battery life summary in the report.
battery_capacity	decimal value	Specifies the battery capacity(decimal positive value) in A*H.
rail_breakdown	boolean	Specifies whether to include the breakdown by rail summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the breakdown by rail summary in the report. • FALSE, false or 0 - Does not include the breakdown by rail summary in the report. This is default value.
type_breakdown	boolean	Specifies whether to include the breakdown by type summary in the report. The acceptable values for this argument are thr following: • TRUE, true or 1- Includes the breakdown by type summary in the report. • FALSE, false or 0 - Does not include the breakdown by type summary in the report. This is the default value.
mode_breakdown	boolean	Specifies whether to include the breakdown by mode in the report. The acceptable values for this argument are thr following: • TRUE, true or 1 - Includes the breakdown by mode in the report. • FALSE, false or 0 - Does not include the breakdown by mode in the report. This is the default value.
opcond_summary	boolean	Specifies whether to include the operating conditions summary in the report. The acceptable values for this argument are the following: • TRUE, true or 1 - Includes the operating conditions summary in the report. • FALSE, false or 0 - Does not include the operating conditions summary in the report.
filename	string	Specifies the name or path of the file to be exported. This argument is mandatory. Default created under designer/<root_name> dircetory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	There are no scenarios in this design, create a scenario in SmartPower, and then generate the scenario report.
None	powerunit: Invalid argument value: 'value' (expecting W, mW or uW).
None	frequnit: Invalid argument value: 'value' (expecting Hz, KHz or MHz).
None	opcond: Invalid argument value: 'value' (expecting Best, Typical or Worst).
None	toggle: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	style: Invalid argument value: 'value' (expecting Text, CSV or XML).
None	battery_life: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	battery_capacity: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'battery_capacity' must be a positive decimal value.
None	rail_breakdown: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	type_breakdown: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	mode_breakdown: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	opcond_summary: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).
None	—
None	—

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example generates a scenario power report named report.txt for "my_scenario":

```
smartpower_report_power_sequencer -scenario my_scenario \
                                  -rail_breakdown true \
                                  -type_breakdown true \
                                  -mode_breakdown true \
                                  -style text \
                                  -battery_capacity 10 \
                                  report.txt
```

See Also

- smartpower_report_power

5.39. smartpower_restore (Ask a Question)

Description

This Tcl command restores all power information previously committed in SmartPower.

```
smartpower_restore
```

Arguments

Parameter	Type	Description
None	None	None

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_restore'.

Supported Families

Supported Families	Supported Versions
PolarFire®	v12.4+
SmartFusion® 2	v12.4+
RTG4™	v12.4+
IGLOO® 2	v12.4+
PolarFire SoC	v12.6+

Example

This example restores all power information previously committed in SmartPower.

```
smartpower_restore
```

See Also

- smartpower_commit

5.40. smartpower_set_cooling (Ask a Question)

Description

This Tcl command sets the cooling style to one of the predefined types or custom value.

Note: To compute the junction temperature, set the following three commands:"smartpower_set_thermalmode, smartpower_set_tambient and smartpower_set_cooling. The junction temperature will be updated when an output command is executed, such as report(Power). Parameter 'teta' is ignored when style is set to custom.

```
smartpower_set_cooling -style "case_cooling | still_air | 1.0_mps | 2.5_mps | custom" [-teta
"positive decimal value"]
```

Arguments

Parameter	Type	Description
style	string	Specifies the cooling style to custom value or to one of the predefined types with a default thermal resistance value. The acceptable values for this argument are the following: • 1.0_mps - Predefined cooling style. • 2.5_mps - Predefined cooling style. • case_cooling - Predefined cooling style. • still_air - Predefined cooling style. • custom - Cooling style defined by user input.
teta	decimal	Specifies the thermal resistance in "C/W". This argument is only available when style is set to Custom.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'style' is missing.
None	Parameter 'style' has illegal value.
None	style: Invalid argument value: 'value' (expecting case_cooling, still_air, 1.0_mps, 2.5_mps or custom).
None	teta: Invalid argument value: 'decimal value' (expecting decimal value).
None	Parameter 'teta' must be a positive decimal value.
None	Parameter 'teta' has illegal value.
None	Parameter 'teta' is required when style is set to custom.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_cooling -style "case_cooling | still_air | 1.0_mps | 2.5_mps | custom" [-teta "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets the cooling style to still air:

```
smartpower_set_cooling -style {still_air}
```

See Also

- smartpower_set_thermalmode
- smartpower_set_tambient

5.41. smartpower_set_default_io_enable_rate (Ask a Question)

Description

Enter description here

Note: Command 'smartpower_set_default_io_enable_rate' is deprecated, but it is still executable. Use "smartpower_change_setofpin_statistics" instead.

```
smartpower_set_default_io_enable_rate -pin_enable_rate "decimal value"
```

Arguments

Parameter	Type	Description
pin_enable_rate	decimal	

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'pin_enable_rate' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_default_io_enable_rate -pin_enable_rate "decimal value"'.
None	Parameter 'pin_enable_rate' must be less than or equal to 100.000.
None	Parameter 'pin_enable_rate' must be a positive decimal value.
None	pin_enable_rate: Invalid argument value: 'value' (expecting decimal value).

Supported Families

PolarFire®

SmartFusion® 2

RTG4™

IGLOO® 2

Example

Enter example description here

```
smartpower_set_default_io_enable_rate -pin_enable_rate "4"
```

5.42. smartpower_set_mode_for_analysis (Ask a Question)

Description

This Tcl command sets the mode for cycle-accurate power analysis.

```
smartpower_set_mode_for_analysis -mode {value}
```

Arguments

Parameter	Type	Description
mode	string	Specifies the mode for cycle-accurate power analysis. This parameter is mandatory. The acceptable values for this argument are the following: • Active - The operating mode is set to Active. • Static - The operating mode is set to Static. • Flash*Freeze - The operating mode is set to Flash*Freeze(RTG4™, SmartFusion® 2, IGLOO® 2).

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'mode' is missing.
None	Parameter 'mode' has illegal value.
None	mode: Invalid argument value: 'mode_name' (expecting Active, Static or Flash*Freeze).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_mode_for_analysis -mode "Active | Static | Flash*Freeze"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example sets the mode for analysis to active:

```
smartpower_set_mode_for_analysis -mode {active}
```

5.43. smartpower_set_mode_for_pdpr (Ask a Question)

Description

This Tcl command sets the operating mode used by the Power Driven Place and Route (PDPR) tool during power optimization.

```
smartpower_set_mode_for_pdpr -opmode {operating mode name}
```

Arguments

Parameter	Type	Description
opmode	string	Sets the operating mode for your power driven place and route. Value must be a valid operating mode. This parameter is mandatory.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'opmode' is missing.
None	opmode: Invalid argument value: 'mode_name' (expecting Active, Flash*Freeze or <CustomCreatedModeName>).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_mode_for_pdpr -opmode "Active | Flash*Freeze".
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_mode_for_pdpr -opmode "Active".

Supported Families

PolarFire®
SmartFusion® 2
RTG4™

IGLOO® 2

Example

This example creates custom mode "MyCustomMode" based on "Active" mode and sets the "MyCustomMode" mode as the operating mode for Power Driven Place and Route(PDPR):

```
smartpower_add_new_custom_mode \
            -name "MyCustomMode" \
                    -description "for PDPR" \
            -base_mode "Active"
smartpower_set_mode_for_pdpr -opmode "MyCustomMode"
```

See Also

- smartpower_add_new_custom_mode
- smartpower_export_mpe_report

5.44. smartpower_set_operating_conditions (Ask a Question)

Description

This Tcl command sets the operating conditions used in SmartPower.

```
smartpower_set_operating_conditions -airflow "still_air "1.0_mps | 2.5_mps | custom" \
                                    -heatsink "None | custom | 10mm_Low_Profile |
15mm_Medium_Profile | 20mm_High_Profile" \
                                    -boardmodel "None_Conservative | JEDEC_2s2p" \
                                    [-teta_ja {decimal value}] \
                                    [-teta_sa {decimal value}]
```

Arguments

Parameter	Type	Description
airflow	string	Specifies the value for the still air operating condition. The acceptable values for this argument are the following: • 1.0_mps - Sets the operating conditions to best. • 2.5_mps - Sets the operating conditions to typical. • custom - Sets the operating conditions to worst. • still_air.
heatsink	string	Specifies the value of the operating condition. The acceptable values for this argument are the following: • none - No heat sink. • custom - Sets a custom heat sink size. • 10mm_Low_Profile - 10 mm heat sink. • 15mm_Low_Profile - 15 mm heat sink. • 20mm_High_Profile - 20 mm heat sink.
boardmodel	string	Specifies your board model. The acceptable values for this argument are the following: • None_Conservative - No board model, conservative routing. • JEDEC_2s2p - JEDEC 2s2p board model.
teta_ja	decimal	This is an optional parameter, that sets your teta ja value. It must be a positive decimal.
teta_sa	decimal	This is an optional parameter, that sets your teta sa value. It must be a positive decimal.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'boardmodel' has illegal value.
None	Required parameter 'boardmodel' is missing.
None	Parameter 'heatsink' has illegal value.
None	Required parameter 'heatsink' is missing.
None	Parameter 'airflow' has illegal value.
None	Required parameter 'airflow' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_operating_conditions -airflow "still_air | 1.0_mps | 2.5_mps | custom" -heatsink "None | custom | 10mm_Low_Profile | 15mm_Medium_Profile | 20mm_High_Profile" -boardmodel "None_Conservative | JEDEC_2s2p" [-teta_ja "decimal value"] [-teta_sa "decimal value"]'.

Supported Families

PolarFire®

SmartFusion® 2

RTG4™

IGLOO® 2

PolarFire SoC

Example

This example sets the operating conditions to best:

```
smartpower_set_operating_conditions -airflow "1.0_mps" \
                        -heatsink "20mm_High_Profile" \
                        -boardmodel "JEDEC_2s2p"
```

5.45. smartpower_set_operating_conditions (Ask a Question)

Description

This Tcl command sets the operating conditions used in SmartPower.

```
smartpower_set_operating_conditions -airflow "still_air | 1.0_mps | 2.5_mps | custom" \
                                    -heatsink {None | custom | 10mm_Low_Profile |
15mm_Medium_Profile | 20mm_High_Profile} \
                                    -boardmodel {None_Conservative | JEDEC_2s2p} \
                                    [-teta_ja {decimal value}] \
                                    [-teta_sa {decimal value}]
```

Arguments

Parameter	Type	Description
still_air	string	Specifies the value for the still air operating condition. The acceptable values for this argument are the following: • 1.0_mps - Sets the operating conditions to best. • 2.5_mps - Sets the operating conditions to typical. • custom - Sets the operating conditions to worst.

smartpower_set_operating_conditions (continued)

Parameter	Type	Description
heatsink	string	Specifies the value of the operating condition. The following table shows the acceptable values for this argument: • none - No heat sink. • custom - Sets a custom heat sink size. • 10mm_Low_Profile - 10 mm heat sink. • 15mm_Low_Profile - 15 mm heat sink. • 20mm_High_Profile - 20 mm heat sink.
boardmodel	string	Specifies your board model. The following table shows the acceptable values for this argument: • None_Conservative - No board model, conservative routing. • JEDEC_2s2p - JEDEC 2s2p board model.
teta_ja	decimal	This is an optional parameter, that sets your teta ja value. It must be a positive decimal.
teta_sa	decimal	This is an optional parameter, that sets your teta sa value. It must be a positive decimal.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'airflow' is missing.
None	Parameter 'airflow' has illegal value.
None	airflow: Invalid argument value: 'value' (expecting still_air, 1.0_mps, 2.5_mps or custom).
None	Parameter 'heatsink' has illegal value.
None	heatsink: Invalid argument value: 'value' (expecting None, custom, 10mm_Low_Profile, 15mm_Medium_Profile or 20mm_High_Profile).
None	Required parameter 'heatsink' is missing.
None	Parameter 'boardmodel' has illegal value.
None	boardmodel: Invalid argument value: 'value' (expecting None_Conservative or JEDEC_2s2p).
None	Required parameter 'boardmodel' is missing.
None	teta_ja: Invalid argument value: 'value' (expecting decimal value).
None	teta_sa: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_operating_conditions -airflow "still_air | 1.0_mps | 2.5_mps | custom" -heatsink "None | custom | 10mm_Low_Profile | 15mm_Medium_Profile | 20mm_High_Profile" -boardmodel "None_Conservative | JEDEC_2s2p" [-teta_ja "decimal value"] [-teta_sa "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets the operating conditions to best:

```
smartpower_set_operating_conditions -airflow "1.0_mps" \
                        -heatsink "20mm_High_Profile" \
                        -boardmodel "JEDEC_2s2p"
```

5.46. smartpower_set_pin_frequency (Ask a Question)

Description

This Tcl command sets the frequency of a pin in megahertz (MHz). If you do not use this command, each pin will have default frequency based on its domain.

Note: The pin_name must be the name of a pin that already exists in the design and already belongs to a domain. When specifying the unit, a space must be between the frequency value and the unit. Execute the "smartpower_commit" Tcl command for preserving any changes in SmartPower.

```
smartpower_set_pin_frequency -pin_name {pin_name} -pin_freq {value}
```

Arguments

Parameter	Type	Description
pin_freq	string	Specifies the value of the frequency in Hz, KHz and MHz, which can be any positive decimal number.
pin_name	string	Specifies the name of the pin for which the frequency will be set.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'pin_name' is missing.
None	Parameter 'pin_name' is missing or has invalid value.
None	Required parameter 'pin_freq' is missing.
None	Failed to set "pin name" pin frequency to 2.60 Hz.
None	pin_freq: Parameter format incorrect (expecting a positive decimal value, optionally followed by a unit (possible values are Hz, KHz or MHz)).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_pin_frequency [-pin_name "name of pin"]+ -pin_freq "decimal value [unit { Hz | KHz | MHz }]"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets the frequency of the pin named "d_in" to 100 MHz:

```
smartpower_set_pin_frequency -pin_name {d_in} -pin_freq {100}
```

See Also

- smartpower_remove_pin_frequency

5.47. smartpower_set_pin_probability (Ask a Question)

Description

This tcl command sets the probability value of a pin driving an enable pin. For I/Os, if you do not use this command, the probability of the IOEnableSet is used. For memories, if you do not use this command, the probability of the MemoriesEnableSet is used.

```
smartpower_set_pin_probability -pin_name "name of pin" -pin_proba "decimal value"
```

Arguments

Parameter	Type	Description
pin_name	string	Specifies the name of a pin for which the probability will be set. This pin must be the direct driver of an enable pin. There may be multiple `-pin_name` arguments. This is mandatory.
pin_proba	decimal value	Specifies the value of the pin probability as a percentage(%), which can be any positive decimal between 0 and 100 inclusive.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'pin_name' is missing.
None	Failed to set "value" pin frequency to 11.00.
None	pin_proba: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'pin_proba' must be less than or equal to 100.000.
None	Parameter 'pin_proba' must be a positive decimal value.
None	ERROR in setSystemDomainIdOfPin. The pin indoes not exist.
None	'smartpower_set_pin_probability [-pin_name "name of pin"]+ -pin_proba "decimal value"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example sets the probability of the pin driving the enable pin of a bidirectional I/O:

```
smartpower_set_pin_probability -pin_name {mybibuf/U0/U1:EOUT} -pin_proba {50.4}
```

The following example sets the probability of the "in" and "out" pins:

```
smartpower_set_pin_probability -pin_name {in out} -pin_proba {50.4}
```

See Also

- smartpower_remove_pin_probability

5.48. smartpower_set_preference (Ask a Question)

Description

This Tcl command sets the following preferences: power unit, frequency unit, operating mode, operating conditions, and toggle. These preferences can also be set from the preferences dialog box.

Note: Running the script with invalid argument it will trigger a warning message: Ignoring invalid argument 'argname'.

```
smartpower_set_preference [-powerunit "W | mW | uW"] \
                          [-frequnit "Hz | KHz | MHz"] \
                          [-opcond "Best | Typical | Worst"] \
                          [-opmode "Active | Static"] \
                          [-toggle "TRUE | FALSE"]
```

Arguments

Parameter	Type	Description
powerunit	string	Specifies the unit in which power is set. The acceptable values for this argument are the following: • W - The power unit is set to watts. • mW - The power unit is set to milliwatts. • uW - The power unit is set to microwatts.
frequnit	string	Specifies the unit in which frequency is set. The acceptable values for this argument are the following: • H - The frequency unit is set to hertz. • kHz - The frequency unit is set to kilohertz. • MHz - The frequency unit is set to megahertz.
opmode	string	Specifies the operating mode. The acceptable values for this argument are the following: • active - The operating mode is set to active. • static - The operating mode is set to static. • Flash*Freeze - The operating mode is set to Flash*Freeze (RTG4, SmartFusion 2 and IGLOO 2).
opcond	string	Specifies the operating condition. The acceptable values for this argument are the following: • worst - The operating condition is set to worst-case. • typical - The operating condition is set to typical case. • best - The operating condition is set to best case.
toggle	boolean	Specifies the toggle. The acceptable values for this argument are the following: • TRUE, true or 1 - The toggle is set to true. • FALSE, false or 0 - The toggle is set to false.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'powerunit' has illegal value.
None	powerunit: Invalid argument value: 'value' (expecting W, mW or uW).
None	frequnit: Invalid argument value: 'value' (expecting Hz, KHz or MHz).

smartpower_set_preference (continued)	
Error Code	**Description**
None	opmode: Invalid argument value: 'value' (expecting Active, Static or Flash*Freeze).
None	opcond: Invalid argument value: 'value' (expecting Best, Typical or Worst).
None	toggle: Invalid argument value: 'value' (expecting TRUE, 1, true, FALSE, 0 or false).

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2

Example

This example sets the frequency of the power unit to "watts", the frequency unit to "Hz", the operating mode to "active", the operating condition to "typical", and the toggle to "true":

```
smartpower_set_preference -powerunit {w} \
                          -frequnit {hz} \
                          -opmode {active} \
                          -opcond{typical} \
                          -toggle {true}
```

5.49. smartpower_set_process (Ask a Question)

Description

This Tcl command sets the process used in SmartPower to one of the pre-defined types.

```
smartpower_set_process -process {value}
```

Arguments

Parameter	Type	Description
process	string	Specifies the value of the operating condition. The acceptable values for this argument are the following: • Typical - Sets the process for SmartPower to typical. • Maximum - Sets the process for SmartPower to maximum.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'process' is missing.
None	Parameter 'process' has illegal value.
None	process: Invalid argument value:""(expecting Typical or Maximum).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_process -process "Typical | Maximum"'

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2

PolarFire SoC

Example
This example sets the operating conditions to typical:

```
smartpower_set_process -process {Typical}
```

5.50. smartpower_set_scenario_for_analysis (Ask a Question)

Description
This Tcl command sets the scenario for cycle-accurate power analysis.

```
smartpower_set_scenario_for_analysis -scenario {value}
```

Arguments

Parameter	Type	Description
scenario	string	Specifies the mode for cycle-accurate power analysis.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'scenario' is missing.
None	scenario: Invalid argument value: 'ScenarioName' (expecting existing_scenario_name).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_scenario_for_analysis -scenario "MyScenario."

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example
The following example sets the scenario for analysis to "my_scenario" custom scenario:

```
smartpower_set_scenario_for_analysis -scenario {my_scenario}
```

See Also
- smartpower_add_new_scenario
- smartpower_remove_scenario

5.51. smartpower_set_tambient (Ask a Question)

Description
Enter description here

Note: Command 'smartpower_set_tambient' is deprecated, but it is still executable. Use "smartpower_temperature_opcond_set_design_wide" instead.

```
smartpower_set_tambient -temp "decimal value"
```

Arguments

Parameter	Type	Description
temp	decimal	—

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'temp' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_tambient -temp "decimal value"'.
None	Parameter 'temp' must be less than or equal to 125.000.
None	temp: Invalid argument value: 'value' (expecting decimal value).

Supported Families

PolarFire®

SmartFusion®2

RTG4™

IGLOO® 2

Example

Enter example description here

```
smartpower_set_tambient -temp "5"
```

5.52. smartpower_set_temperature_opcond (Ask a Question)

Description

This tcl command sets the temperature in the operating conditions to one of the predefined types.

```
smartpower_set_temperature_opcond -use {value}
```

Arguments

Parameter	Type	Description
use	string	Specifies the temperature in the operating conditions. The acceptable values for this argument are the following: • oprange - Sets the temperature in the operating conditions as specified in your Project Settings. • design - Sets the temperature in the operating conditions as specified in the SmartPower design-wide operating range. Applies to SmartPower only. • mode - Sets the temperature in the operating conditions as specified in the SmartPower mode-specific operating range. Applies to SmartPower only.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'use' has illegal value.
None	Required parameter 'use' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_temperature_opcond -use "oprange | design | mode"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets the temperature in the operating conditions as specified in the custom mode-settings:

```
smartpower_set_temperature_opcond -use {mode}
```

See Also

- smartpower_temperature_opcond_set_design_wide
- smartpower_temperature_opcond_set_mode_specific

5.53. smartpower_set_thermalmode (Ask a Question)

Description

This Tcl command sets the mode of computing junction temperature.

Note: To compute the junction temperature, set the smartpower_set_thermalmode, smartpower_set_tambientand smartpower_set_cooling commands. The junction temperature will be updated when an output command is executed, such as report(Power).

```
smartpower_set_thermalmode [-mode {value}]
```

Arguments

Parameter	Type	Description
mode	string	Specifies the mode in which the junction temperature is computed. The acceptable values for this argument are the following: • ambient - The junction temperature will be iteratively computed with total static power. • opcond - The junction temperature will be given as one of the operating condition range valuesspecified in the device selection.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Required parameter 'mode' is missing.
None	Parameter 'mode' has illegal value.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_thermalmode -mode "opcond | ambient"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

The following example sets the computing of the junction temperature to ambient mode:

```
smartpower_set_thermalmode -mode {ambient}
```

5.54. smartpower_set_voltage_opcond (Ask a Question)

Description

This Tcl command sets the voltage in the operating conditions.

```
smartpower_set_voltage_opcond -voltage {value} -use {value}
```

Arguments

Parameter	Type	Description
voltage	string	Specifies the voltage supply in the operating conditions. The acceptable values for this argument are the following: • VDD - Sets the voltage operating conditions for VDD. • VDD18 - Sets the voltage operating conditions for VDD18. • VDDAUX - Sets the voltage operating conditions for VDDAUX. • VDDI 1.1 - Sets the voltage operating conditions for VDD 1.1. • VDDI 1.2 - Sets the voltage operating conditions for VDDI 1.2. • VDDI 1.35 - Sets the voltage operating conditions for VDDI 1.35. • VDDI 1.5 - Sets the voltage operating conditions for VDDI 1.5. • VDDI 1.8 - Sets the voltage operating conditions for VDDI 1.8. • VDDI 2.5 - Sets the voltage operating conditions for VDDI 2.5. • VDDI 3.3 - Sets the voltage operating conditions for VDDI 3.3. • VDD25 - Sets the voltage operating conditions for VDD25. • VDDA - Sets the voltage operating conditions for VDDA. • VDDA25 - Sets the voltage operating conditions for VDDA25. • VPP - Sets the voltage operating conditions for VPP.

smartpower_set_voltage_opcond (continued)		
Parameter	**Type**	**Description**
use	string	Specifies the voltage in the operating conditions for each voltage supply. The acceptable values for this argument are the following: • oprange - Sets the voltage in the operating conditions as specified in your Project Settings. • design - Sets the voltage in the operating conditions as specified in the SmartPower design-wideoperating range. Applies to SmartPower only. • mode - Sets the voltage in the operating conditions as specified in the SmartPower mode-specificoperating range. Applies to SmartPower only.

Return Type	Description
None	None

Error Codes

Error Code	Description
	Parameter 'use' has illegal value.
	Required parameter 'use' is missing.
	Parameter 'voltage' has illegal value.
	Required parameter 'voltage' is missing.
	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_set_voltage_opcond -voltage "VDD | VDD18 | VDDI 1.8 | VDD25" -use "oprange | design | mode"'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets the VCCA as specified in the SmartPower mode-specific settings:

```
smartpower_set_voltage_opcond -voltage {VDD} -use {mode}
```

See Also

- smartpower_voltage_opcond_set_mode_specific
- smartpower_voltage_opcond_set_design_wide

5.55. smartpower_temperature_opcond_set_design_wide (Ask a Question)

Description

This Tcl command sets the temperature for SmartPower design-wide operating conditions.

```
smartpower_temperature_opcond_set_design_wide \
                        [-best {value}] \
                                [-typical {value}] \
                                [-worst {value}] \
                                [-thermal_mode {value}]
```

Arguments

Parameter	Type	Description
best	decimal	Specifies the best temperature (in degrees Celsius) used for design-wide operating conditions.
typical	decimal	Specifies the typical temperature (in degrees Celsius) used for design-wide operating conditions.
worst	decimal	Specifies the worst temperature (in degrees Celsius) used for design-wide operating conditions.
thermal_mode	string	Specifies the mode in which the junction temperature is computed. The acceptable values for this argument are the following: • ambient - The junction temperature will be iteratively computed with total static power. • opcond - The junction temperature will be given as one of the operating condition rangevalues specified in the device selection.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'thermal_mode' has illegal value.
None	Parameter 'worst' has illegal value.
None	Parameter 'typical' has illegal value.
None	Parameter 'best' has illegal value.
None	Invalid best voltage value (00) for VDD. VDD best voltage must be between 0.970 and 1.030.
None	Invalid typical voltage value (00) for VDD. VDD typical voltage must be between 0.970 and 1.030.
None	Invalid argument value: 'voltage_value' (expecting VDD, VDDAUX, VDDI 1.8, VDD25 or VDD18).
None	Invalid voltage sequence. The value for best cannot exceed typical and typical cannot exceed worst.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_temperature_opcond_set_design_wide [-thermal_mode "opcond | ambient"] [-best "decimal value"] [-typical "decimal value"] [-worst "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets the temperature for design-wide operating conditions to "Best 20", "Typical 30", and "Worst 60":

```
smartpower_temperature_opcond_set_design_wide -best {20} -typical {30} -worst {60}
```

See Also

- smartpower_temperature_opcond_set_mode_specific

- smartpower_set_temperature_opcond

5.56. smartpower_temperature_opcond_set_mode_specific (Ask a Question)

Description

This Tcl command sets the temperature for SmartPower mode-specific operating conditions.

```
smartpower_temperature_opcond_set_mode_specific \
                    -opmode "Active | Static" \
                    [-thermal_mode "opcond | ambient"] \
                    [-best "decimal value"] \
                    [-typical "decimal value"] \
                    [-worst "decimal value"]
```

Arguments

Parameter	Type	Description
opmode	string	Specifies the operating mode. The acceptable values for this argument are the following: • Active - The operating mode is set to Active. • Static - The operating mode is set to Static. • Flash*Freeze - The operating mode is set to Flash*Freeze(RTG4™, SmartFusion® 2 and IGLOO® 2).
thermal_mode	string	Specifies the mode in which the junction temperature is computed. This parameter is optional. The acceptable values for this argument are the following: • ambient - The junction temperature will be iteratively computed with total static power. • opcond - The junction temperature will be given as one of the operating condition rangevalues specified in the device selection.
best	decimal	Specifies the best temperature (in degrees Celsius) for the selected mode. This parameter is optional.
typical	decimal	Specifies the typical temperature (in degrees Celsius) for the selected mode. This parameter is optional.
worst	decimal	Specifies the worst temperature (in degrees Celsius) for the selected mode. This parameter is optional.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'worst' has illegal value.
None	Parameter 'typical' has illegal value.
None	typical: Invalid argument value: 'value' (expecting decimal value).
None	Parameter 'best' has illegal value.
None	best: Invalid argument value: 'best_value' (expecting decimal value).
None	Parameter 'thermal_mode' has illegal value.
None	thermal_mode: Invalid argument value: 'value' (expecting opcond or ambient).
None	Required parameter 'opmode' is missing.
None	Parameter 'opmode' has illegal value.
None	opmode: Invalid argument value: 'value' (expecting Active or Static).
None	Invalid temperature sequence for 'Active' mode. The value for best cannot exceed typical and typical cannot exceed worst.

smartpower_temperature_opcond_set_mode_specific (continued)	
Error Code	**Description**
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_temperature_opcond_set_mode_specific -opmode "Active | Static" [-thermal_mode "opcond | ambient"] [-best "decimal value"] [-typical "decimal value"] [-worst "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets the temperature for mode-specific operating conditions for mode1:

```
smartpower_temperature_opcond_set_mode_specific \
                              -opmode {Active} \
                    -thermal_mode {ambient} \
                              -best {20} \
                              -typical {30} \
                              -worst {60}
```

See Also

- smartpower_temperature_opcond_set_design_wide
- smartpower_set_temperature_opcond

5.57. smartpower_voltage_opcond_set_design_wide (Ask a Question)

Description

This tcl command sets the voltage settings for SmartPower design-wide operating conditions.

```
smartpower_voltage_opcond_set_design_wide \
          -voltage "VDD | VDD18 | VDDI 1.8 | VDD25" \
          [-best "decimal value"] \
          [-typical "decimal value"] \
          [-worst "decimal value"]
```

Arguments

Parameter	Type	Description
voltage	string	Specifies the voltage supply in the operating conditions. The acceptable values for this argument are the following: • VDD - Sets the voltage operating conditions for VDD. • VDDAUX - Sets the voltage operating conditions for VDDAUX. • VDD25 - Sets the voltage operating conditions for VDD25. • VDDI 2.5 - Sets the voltage operating conditions for VDDI 2.5. • VDD18 - Sets the voltage operating conditions for VDD18. • VPP - Sets the voltage operating conditions for VPP. • VCCA - Sets the voltage operating conditions for VCCA. • VCCI 3.3 - Sets the voltage operating conditions for VCCI 3.3. • VCCI 2.5 - Sets the voltage operating conditions for VCCI 2.5. • VCCI 1.8 - Sets the voltage operating conditions for VCCI 1.8. • VCCI 1.5 - Sets the voltage operating conditions for VCCI 1.5. • VCC33A - Sets the voltage operating conditions for VCC33A. • VCCDA - Sets the voltage operating conditions for VCCDA. • VPP - Sets the voltage operating conditions for VPP.
best	decimal	Specifies the best voltage used for design-wide operating conditions.
typical	decimal	Specifies the typical voltage used for design-wide operating conditions.
worst	decimal	Specifies the worst voltage used for design-wide operating conditions.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'worst' has illegal value.
None	Parameter 'typical' has illegal value.
None	Parameter 'best' has illegal value.
None	Invalid best voltage value (0.000) for VDDI 1.8. VDDI 1.8 best voltage must be between 1.710 and 1.890.
None	Parameter 'voltage' has illegal value.
None	Required parameter 'voltage' is missing.
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_voltage_opcond_set_design_wide -voltage "VDD | VDD18 | VDDI 1.8 | VDD25" [-best "decimal value"] [-typical "decimal value"] [-worst "decimal value"]'

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets VDDI 1.8 for design-wide to best 1.72, typical 1.85 and worst 1.88:

```
smartpower_voltage_opcond_set_design_wide \
                              -voltage {VDDI 1.8} \
                              -best {1.72} \
                              -typical {1.85} \
                              -worst {1.88}
```

See Also

- smartpower_set_voltage_opcond
- smartpower_voltage_opcond_set_mode_specific

5.58. smartpower_voltage_opcond_set_mode_specific (Ask a Question)

Description

This tcl command sets the voltage settings for SmartPower mode-specific use operating conditions.

```
smartpower_voltage_opcond_set_mode_specific \
                  -opmode "Active | Static" \
                  -voltage "VDD | VDD18 | VDDI 1.8 | VDD25" \
                  [-best "decimal value"] \
                  [-typical "decimal value"] \
                  [-worst "decimal value"]
```

Arguments

Parameter	Type	Description
opmode	string	Specifies the mode from which the operating conditions are extracted to generate the report. • Active - The operating mode is set to Active. • Static - The operating mode is set to Static. • Flash*Freeze - The operating mode is set to Flash*Freeze(RTG4™, SmartFusion® 2 and IGLOO® 2).
voltage	string	Specifies the voltage in the operating conditions. The acceptable values for this argument are the following: • VDD - Sets the voltage operating conditions for VDD. • VDD18 - Sets the voltage operating conditions for VDD18. • VDDAUX - Sets the voltage operating conditions for VDDAUX. • VDDI 1.1 - Sets the voltage operating conditions for VDD 1.1. • VDDI 1.2 - Sets the voltage operating conditions for VDDI 1.2. • VDDI 1.35 - Sets the voltage operating conditions for VDDI 1.35. • VDDI 1.5 - Sets the voltage operating conditions for VDDI 1.5. • VDDI 1.8 - Sets the voltage operating conditions for VDDI 1.8. • VDDI 2.5 - Sets the voltage operating conditions for VDDI 2.5. • VDDI 3.3 - Sets the voltage operating conditions for VDDI 3.3. • VDD25 - Sets the voltage operating conditions for VDD25. • VDDA - Sets the voltage operating conditions for VDDA. • VDDA25 - Sets the voltage operating conditions for VDDA25.
best	decimal	Specifies the best voltage used for mode-specific operating conditions.
typical	decimal	Specifies the typical voltage used for mode-specific operating conditions.
worst	decimal	Specifies the worst voltage used for mode-specific operating conditions.

Return Type	Description
None	None

Error Codes

Error Code	Description
None	Parameter 'worst' has illegal value.
None	Parameter 'typical' has illegal value.
None	Parameter 'best' has illegal value.
None	Parameter 'voltage' has illegal value.
None	Parameter 'opmode' has illegal value.
None	Required parameter 'opmode' is missing.
None	Required parameter 'voltage' is missing.
None	Invalid best voltage value (1.200) for VDD18. VDD18 best voltage must be between 1.710 and 1.890.
None	Invalid best voltage value (1.200) for VDDI 1.8. VDDI 1.8 best voltage must be between 1.710 and 1.890.
None	Invalid best voltage value (1.200) for VDDAUX. VDDAUX best voltage must be between 2.375 and 2.625.
None	Invalid best voltage value (1.200) for VDD25. VDD25 best voltage must be between 2.375 and 2.625.
None	Invalid best voltage value (1.111) for VDDI 2.5. VDDI 2.5 best voltage must be between 2.375 and 2.625.
None	Invalid best voltage value (1.200) for VDD. VDD best voltage must be between 0.970 and 1.030.
None	Invalid best voltage value (1.111) for VDD. VDD best voltage must be between 0.970 and 1.030.
None	Invalid best voltage value (1.111) for VDD18. VDD18 best voltage must be between 1.710 and 1.890.
None	Invalid worst voltage value (1.111) for VDD. VDD worst voltage must be between 0.970 and 1.030.
None	voltage: Invalid argument value: 'voltage_value' (expecting VDD, VDDI 2.5 or VPP).
None	voltage: Invalid argument value: 'voltage_value' (expecting VDD, VDDAUX, VDDI 1.8, VDD25 or VDD18).
None	voltage: Invalid argument value: 'voltage_value' (expecting VDD, VDD18, VDDI 1.8, VDDAUX or VDD25).
None	Parameter 'param_name' is not defined. Valid command formatting is 'smartpower_voltage_opcond_set_mode_specific -opmode "Active | Static" -voltage "VDD | VDD18 | VDDI 1.8 | VDD25" [-best "decimal value"] [-typical "decimal value"] [-worst "decimal value"]'.

Supported Families

PolarFire®
SmartFusion® 2
RTG4™
IGLOO® 2
PolarFire SoC

Example

This example sets the voltage for the static mode and sets "best to 0.98", "typical to 1" and "worst to 1.01":

```
smartpower_voltage_opcond_set_mode_specific \
                        -opmode {active} \
```

```
                    -voltage {VDD} \
                    -best {0.98} \
                    -typical {1} \
                    -worst {1.01}
```

See Also

- smartpower_set_voltage_opcond
- smartpower_voltage_opcond_set_design_wide

6. Revision History (Ask a Question)

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

Revision	Date	Description
M	05/2025	This document is released with Libero SoC Design Suite v2025.1 without changes from v2024.2.
L	08/2024	This document is released with Libero SoC Design Suite v2024.2 without changes from v2024.1.
K	02/2024	This document is released with Libero SoC Design Suite v2024.1 without changes from v2023.2.
J	08/2023	Updated section Operating Conditions to describe the procedure for changing the VPP voltage operating conditions.
H	04/2023	This document is released with Libero 2023.1 SoC Design Suite without changes from v2022.3.
G	12/2022	This document is released with Libero 2022.3 SoC Design Suite without changes from v2022.2.
F	08/2022	The following changes are made in this revision: • Updated section Importing a VCD File.
E	04/2022	The following changes are made in this revision: • Added SmartPower Tcl Commands from the Libero SoC Tcl Command Reference Guide. • Updated section Importing a VCD File.
D	12/2021	This document is released with Libero 2021.3 SoC Design Suite without changes from v2021.2.
C	08/2021	This document is released with Libero 2021.2 SoC Design Suite without changes from v2021.1.
B	04/2021	The following changes are made in this revision: • Updated section Export Report for Microchip Power Estimator (MPE). • Added section Freeze/Unfreeze Calculations.
A	11/2020	Document converted to Microchip template. Initial Revision.

Microchip FPGA Support

Microchip FPGA products group backs its products with various support services, including Customer Service, Customer Technical Support Center, a website, and worldwide sales offices. Customers are suggested to visit Microchip online resources prior to contacting support as it is very likely that their queries have been already answered.

Contact Technical Support Center through the website at www.microchip.com/support. Mention the FPGA Device Part number, select appropriate case category, and upload design files while creating a technical support case.

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

- From North America, call **800.262.1060**
- From the rest of the world, call **650.318.4460**
- Fax, from anywhere in the world, **650.318.8044**

Microchip Information

Trademarks

The “Microchip” name and logo, the “M” logo, and other names, logos, and brands are registered and unregistered trademarks of Microchip Technology Incorporated or its affiliates and/or subsidiaries in the United States and/or other countries (“Microchip Trademarks”). Information regarding Microchip Trademarks can be found at https://www.microchip.com/en-us/about/legal-information/microchip-trademarks.

ISBN: 979-8-3371-1258-9

Legal Notice

This publication and the information herein may be used only with Microchip products, including to design, test, and integrate Microchip products with your application. Use of this information in any other manner violates these terms. Information regarding device applications is provided only for your convenience and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. Contact your local Microchip sales office for additional support or, obtain additional support at www.microchip.com/en-us/support/design-help/client-support-services.

THIS INFORMATION IS PROVIDED BY MICROCHIP “AS IS”. MICROCHIP MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WHETHER EXPRESS OR IMPLIED, WRITTEN OR ORAL, STATUTORY OR OTHERWISE, RELATED TO THE INFORMATION INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE, OR WARRANTIES RELATED TO ITS CONDITION, QUALITY, OR PERFORMANCE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, OR CONSEQUENTIAL LOSS, DAMAGE, COST, OR EXPENSE OF ANY KIND WHATSOEVER RELATED TO THE INFORMATION OR ITS USE, HOWEVER CAUSED, EVEN IF MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP’S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THE INFORMATION OR ITS USE WILL NOT EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO MICROCHIP FOR THE INFORMATION.

Use of Microchip devices in life support and/or safety applications is entirely at the buyer’s risk, and the buyer agrees to defend, indemnify and hold harmless Microchip from any and all damages, claims, suits, or expenses resulting from such use. No licenses are conveyed, implicitly or otherwise, under any Microchip intellectual property rights unless otherwise stated.

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip products:

- Microchip products meet the specifications contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is secure when used in the intended manner, within operating specifications, and under normal conditions.
- Microchip values and aggressively protects its intellectual property rights. Attempts to breach the code protection features of Microchip products are strictly prohibited and may violate the Digital Millennium Copyright Act.
- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of its code. Code protection does not mean that we are guaranteeing the product is “unbreakable”. Code protection is constantly evolving. Microchip is committed to continuously improving the code protection features of our products.