Xilinx Power Estimator User Guide

UG440 (v2018.3) December 20, 2018

XILINX

Revision History

The following table shows the revision history for this document.

Section	Revision Summary
12/20/2018 Version 2018.3	
Estimating HBM Power (HBM Sheet)	Updated with default values for Read and Write rates
Using the GTM Sheet	New section added
Using the PS Sheet (Zynq-7000 SoC and Zynq UltraScale+ MPSoC)	Figure 3-20 updated
06/06/2018 Version 2018.2	
General updates	Editorial updates only. No technical content updates.
04/04/2018 Version 2018.1	
Using Soft-Decision FEC (SD-FEC) Sheet and Using RFADC-DAC Sheet	Added SD-FEC and RFADC-ADC sheets information
Setting Clocks for Zynq UltraScale+ PS Sheet	Added information on how to use the PS Tab for Zynq UltraScale+ MPSoC and RPU/APU% load
Memory Generator Wizard and the Block RAM Sheet (Block Memory)	Added notes on Block RAM Configuration Modes
Using Other Sheets (7 Series, Zynq-7000 SoC, UltraScale and UltraScale+ Devices)	Added a note on VCU power including both static and dynamic powers
Estimating HBM Power (HBM Sheet)	Added information on HBM sheet available for Virtex UltraScale+ HBM devices.

Table of Contents

Chapter 4: Exchanging Power Information

Chapter 5: Automating XPE

Chapter 6: Using Snapshots and Graph Sheets

Appendix A: Additional Resources and Legal Notices

Overview

Introduction

The Xilinx® Power Estimator (XPE) spreadsheet is a power estimation tool typically used in the pre-design and pre-implementation phases of a project. XPE assists with architecture evaluation, device selection, appropriate power supply components, and thermal management components specific for your application.

XPE considers your design resource usage, toggle rates, I/O loading, and many other factors which it combines with the device models to calculate the estimated power distribution. The device models are extracted from measurements, simulation, and/or extrapolation.

The accuracy of XPE is dependent on two primary sets of inputs:

- Device utilization, component configuration, clock, enable, and toggle rates, and other information you enter into the tool
- Device data models integrated into the tool

For accurate estimates of your application, enter realistic information which is as complete as possible. Modeling a certain aspect of the design too conservatively or without sufficient knowledge of the design can result in unrealistic estimates. Some techniques to drive the XPE to provide worst-case estimates or typical estimates are discussed in this document.

XPE is a pre-implementation tool for use in the early stages of a design cycle or when the Register Transfer Level (RTL) description is incomplete. After implementation, the XPower Analyzer (XPA) tool (in the ISE® Design Suite) or Report Power (in the Vivado® Design Suite) can be used for more accurate estimates and power analysis. For more information about XPA, see the *XPower Analyzer Help* [Ref 1]. For more information about the Vivado power analysis feature, see the *Vivado Design Suite User Guide: Power Analysis and Optimization* (UG907) [Ref 2].

XPE is a spreadsheet, so all Microsoft Excel functionality is fully retained in the writable or unprotected sections of the spreadsheet. XPE has additional functionality oriented towards ease of use. The drop-down menus and the comment-enabled cells are helpful features to guide you.

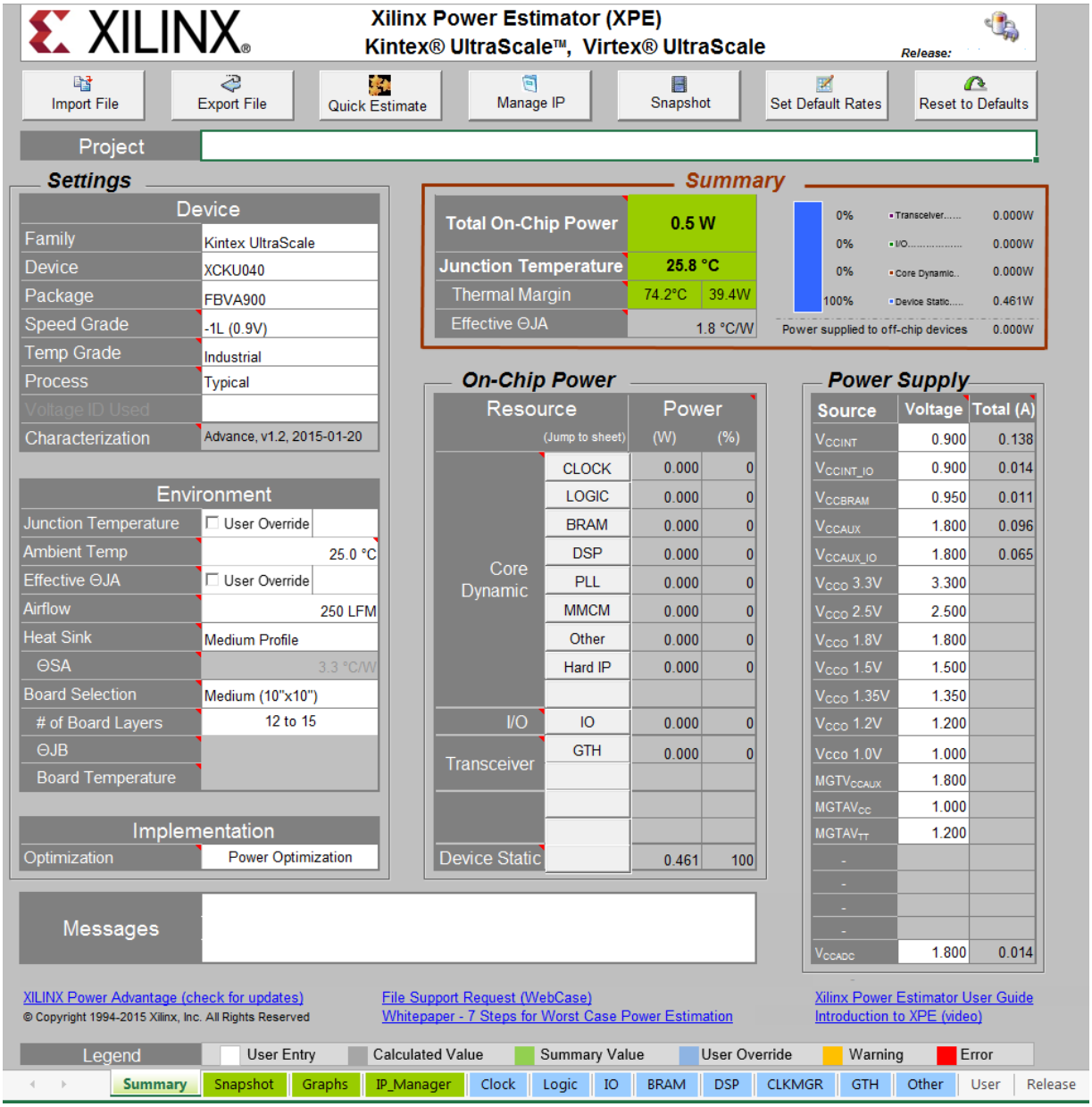


Figure 1-1: **Xilinx Power Estimator Spreadsheet**

The XPE spreadsheet also includes the Quick Estimate Wizard, the Memory Interface Configuration Wizard, the Memory Generator Wizard (for block memory and distributed memory), and the Transceiver Configuration Wizard. These wizards help novice and expert users to quickly enter the important configuration parameters, which will then generate relevant lines in the I/O, Logic, Block RAM (BRAM), Transceiver, and Other sheets, helping with accurate power estimation.

VIDEO: *The Vivado Design Suite QuickTake Video Tutorial: Using the Xilinx Power Estimator shows how the Xilinx Power Estimator can help you determine power and cooling specifications for SoC and FPGA designs early in the product's design cycle, often even before the logic within the SoC or FPGA has been designed.*

Getting Started with XPE

Opening XPE

1. XPE requires Microsoft Excel 2003 or later to be installed.

Table 1-1: **Supported versions of Microsoft Excel for XPE**

Device Family	Supported Excel version
UltraScale+™	Microsoft Excel 2007, 2010, 2013, 2016 and Office 365(.xlsm)
UltraScale™	Microsoft Excel 2007, 2010, 2013, 2016 and Office 365(.xlsm)
7series and Zynq®-7000	Microsoft Excel 2007, 2010, 2013, 2016 and Office 365(.xlsm)

   OpenOffice and Google Docs spreadsheet editors are not supported in this release of XPE.

2. Download the latest available spreadsheet for your targeted device. The XPE spreadsheets are available at the Power Efficiency web page.
3. Make sure your Microsoft Excel settings allow macro executions. XPE uses several macros built into the XPE spreadsheet.
   - **Microsoft Excel 2010 or 2013 or 2016/Office 365** - The following steps are required:

     a.From the XPE spreadsheet select **File > Options**.

     b.In the Excel Options dialog box, click **Trust Center**.

     c.In the Trust Center dialog box, click **Trust Center Settings** and select the **Macro Settings** tab.

     d.Select **Enable all macros**, then click **OK**.

     e.Reopen the XPE spreadsheet.
   - **Microsoft Excel 2007** - The following steps are required:

     a.From the Microsoft Office button select **Excel Options**.

     b.In the Options dialog box, click **Trust Center**.

     c.In the Trust Center dialog box, click **Trust Center Settings**, and select the **Macro Security** tab.

d.Select **Enable all macros**, then click **OK**.

e.Open or, if already open, reopen the XPE spreadsheet.

IMPORTANT: *If you save an Excel 2007 or later spreadsheet as an .xlsx file (Excel Workbook) you will lose the macro capability and render XPE nonfunctional. You will be warned of this if you try to save as an .xlsx file.*

- **Microsoft Excel 2003** - By default, the macro security level is set to High, which disables macros. To change the macro security level, follow these steps (actual menu names will vary with language of Microsoft Excel):

  a.On the Tools menu, point to **Macro** and click **Security**.

  b.In the Security dialog box, click the **Security Level** tab.

  c.Select **Medium**, then click **OK**.

  d.Open or, if already open, reopen the XPE spreadsheet.

  e.When prompted whether to enable or disable macros, click **Enable Macros**.

IMPORTANT: *On Windows, make sure your language is set to English. Select* ***Control Panel >Clock, Language, and Region > Region and Language****, and set Format to* ***English****.*

User Input Requirements

Power estimation for programmable devices like FPGAs is a complex process, because it is highly dependent on the amount of logic in the design and the configuration of that logic. To produce accurate estimates, the power estimation process requires accurate input values, such as resource utilization, clock rates, and toggle rates. To supply the minimum input that will allow XPE to estimate power with reasonable accuracy, you need the following:

- A target device-package-grade combination
- A good estimate of resources you expect to use in the design (for example, flip-flops, look-up tables, I/Os, block RAMs, DCMs or MMCMs, and PLLs.)
- The clock frequency or frequencies for the design
- An estimate of the data toggle rates for the design
- The external memory and transceiver based interfaces with their data rates for the design
- The thermal environment in which the design will be operating

As a general rule, input as much information about your design as available, then leave the remaining settings to default values. This strategy will allow you to determine the device power supply and heat dissipation requirements.

TIP: *Use Excel formulas to link different cells together. For example, type '=CLOCK!E10' in the* ***Clock*** *cells of the Logic sheet, which lists the resources driven by this clock domain.*

XPE Calculations and Results

XPE uses your design and environmental input, then combines this information with the device data model to compute and present an estimated distribution of the power in the targeted device.

XPE presents multiple views of the power distribution.

- **Power by Voltage Supplies** - For each required voltage source, this information is useful to select and size power supply components, such as regulators. Supply power includes both off-chip and on-chip dissipated power.
- **Power by User Logic Resources** - For each type of user logic in the design, XPE reports the expected power. This allows you to experiment with architecture, resources, and implementation trade-off choices to remain within the allotted power budget.
- **Thermal Power** - XPE lets you enter device environment settings and reports thermal properties of the device for your application, such as the expected junction temperature. With this information you can evaluate the need for passive or active cooling for your design.

The Summary sheet in XPE shows the total power for the device. Other sheets show usage-based power. Leakage within the unused portion of the considered resource (if any) is not shown.

IMPORTANT: *In XPE, the power number cells are configured to display values with three decimal places (for example, 0.000). The rounding of numbers with three precision is based on Microsoft Excel behavior. Values less than 1mW are displayed as 0.000W. You can copy a cell and paste it into the User sheet to see the actual value with precision adjusted.*

Definitions/Terminology

Supported Device Families

Separate spreadsheets are available depending on the targeted architecture. These spreadsheets are updated when new device data become available or when new features are added to XPE.

- UltraScale+™ devices
  - Kintex UltraScale+
  - Virtex UltraScale+

    - Zynq® UltraScale+
        - Zynq UltraScale+ MPSoC
        - Zynq UltraScale+ RFSoC
        - Zynq UltraScale+ MPSoC Automotive
- UltraScale™ devices
    - Kintex UltraScale
    - Virtex UltraScale
    - Kintex UltraScale Automotive
- 7 Series devices and Zynq-7000 SoCs
    - Artix®-7, Artix-7 Automotive grade, and Artix-7 Defense grade
    - Kintex-7® and Kintex-7 Defense grade
    - Virtex-7® and Virtex-7 Defense grade
    - Zynq®-7000, Zynq-7000 Automotive grade, and Zynq-7000 Defense grade
    - Spartan®-7
- Virtex-6 and Virtex-5 devices
    - Virtex-6®, Virtex-6 Low Power, and Virtex-6Q Defense grade
    - Virtex-5®, Virtex-5Q Defense grade, and Virtex-5QV Space grade
- Virtex-4
- Spartan®-6 and Spartan-3A – This spreadsheet includes all sub-families, including Spartan-6 Lower Power, Spartan-6 Automotive, Spartan-6Q Defense-grade, Spartan-3AN, and Spartan-3A DSP
- Spartan-3E
- Spartan-3

IMPORTANT: *Download the latest available spreadsheet from the* Xilinx Power Estimator (XPE) *web page.*

Device Model Accuracy

The accuracy of the characterization data existing in the tool is reflected by accuracy designations in the **Characterization** field on the Summary sheet of XPE. For most devices, the history of the accuracy designation is also displayed in the **Release** sheet. The accuracy designations are Advance, Preliminary, and Production.

Advance

These specifications are based on simulations only and are typically available soon after the device design specifications are frozen They are subject to change as silicon characterization data becomes available. Advance data accuracy is considered lower than the Preliminary and Production data.

Preliminary

The data integrated into XPE with this designation is based on complete early production silicon. Almost all the blocks in the device fabric are characterized. Data for most of the dedicated blocks like TEMAC and PCIe® block are also characterized and integrated into XPE. The accuracy of power reporting is improved compared to Advance data.

Production

The data integrated into XPE with this designation is released after enough production silicon of a particular device family member has been characterized to provide full power correlation over numerous production lots. Characterization data for all blocks in the device fabric is included.

Total Power

The total device power is calculated as follows:

Total devices power = Device Static + Design Static + Design Dynamic

The power estimates are modeled to account for temperature and voltage sensitivity. Ambient temperature and regulated voltage on the system can be keyed into the appropriate cells provided for that purpose.

Device Static Power

Also referred to as Leakage. Device static represents the transistor leakage power when the device is powered and not configured.

Design Static Power

Design static represents the additional power consumption when the device is configured but there is no switching activity. It includes static power in I/O DCI terminations, clock managers, and so forth.

For design static power calculations, XPE starts by assuming a blank bitstream. To add your design elements (for example, Logic, I/Os, BRAMs, Clock Managers) to the design static power calculations, you must enter the resource utilization and configuration in the XPE resource sheets applicable to the design. Any I/O termination should be set to match the

board and the design. For any clock managers, enter a small clock frequency to indicate usage. Enter or leave clock frequency values 0 on other resource sheets.

Note: For maximum process, the static power in a device should never exceed the reported values in the tool.

Design Dynamic Power

Design dynamic represents the additional power consumption from the user logic utilization and switching activity.

Activity Rates

XPE shows values for these types of activity rates:

- Toggle Rates
- Signal Rates

Toggle Rates

Providing accurate toggle rates in the various XPE sheets is essential to get quality power estimates. This information, however, might not be readily available at the stage in the design cycle where you enter data in XPE. Activity might be refined as the design gets more defined. Following are guidelines you can follow to help you enter design toggle activity.

- For synchronous paths, toggle rate reflects how often an output changes relative to a given clock input and can be modeled as a percentage between 0–100%. The max data toggle rate of 100% means that the output toggles every active clock edge. For example, consider a free running binary counter with a 100MHz clock. For the Least Significant Bit you would enter 100% in the **Toggle Rate** column, because this bit toggles every rising edge of the clock. For the second bit you would enter 50%, because this bit toggles every other rising edge of the clock. When data changes twice per clock cycle, enter 200% for the toggle rate.
- For non-periodic or event-driven portions of designs, toggle rates cannot be easily predicted. An effective method of estimating average toggle rates for a given design is to segregate the different sections of the design based on their functionality or hierarchy and estimate the toggle rates for each of the sub-blocks. An average toggle rate can then be arrived at by calculating the average for the entire design or hierarchy. Most logic-intensive designs work at around 12.5% average toggle rate, which is the default toggle rate setting in XPE.

  It has been observed that designs with random data patterns as input generally have toggle rates between 10%-30%. However, designs with a lot of glitch logic can have toggle rates as high as or even higher than 50%. Glitch logic is generally classified as combinatorial functions which have a high probability of the output changing when any one input changes, such as XOR gates or unregistered arithmetic logic (i.e. adders).

Functions that use large amounts of such logic, such as error detection/correction circuitry, might exhibit higher toggle rates due to this. Designs with large amounts of control path logic, such as embedded designs, on average have lower toggle rates due to large sections of logic being inactive at any given time during operation.

In summary, the primary factors that have an appreciable impact on the toggle rate of a design are:

- Input data pattern - Random data pattern versus known patterns have an impact on the toggle rate.
- Control signals - Use or lack of control signals such as reset and clock enables.
- Design logic - High glitch XOR/CARRY logic, a highly pipelined design, or an embedded design have an impact on the toggle rate.

General guidelines for the toggle-rates of Ex-OR (XOR) circuit

XOR logic cones contain more glitches and as the number of logic levels increases, the glitch count keeps increasing. However, it does have a saturation point. In a non-glitch activity, the saturation point will be at 50% toggle rate (at 3 to 4 levels of XOR tree)

In XOR logic, toggle rates depend on the circuit topology. Number of glitches depends on the exact depth and width of an XOR tree. Different XOR logic tree depth levels give different results.

Example:

Maximum XOR Toggle rate in a user combinational logic assuming 1024 wide XOR with a depth of 10 levels is as follows:

- 815% - Worst input
- 254% - Random input

Maximum XOR Toggle rate in user combinational logic assuming 32 wide XOR with a depth of 5 levels:

- 516% - Worst input
- 114% - Random input

IMPORTANT: *In all the sheets which do not have a dedicated **Clock Enable** column make sure you scale the toggle rate to account for any signal which gates this logic. For example, if the data toggle rate is modeled at 50% but the synchronizing clock is enabled 50% of the time, the resulting toggle rate should be 25% (50% x 50%).*

IMPORTANT: *To appreciate what 100% toggle rate means, think of a constantly enabled toggle flip-flop (TFF) whose data input is tied High. The T-output of this flip-flop toggles every clock edge. Very few designs could possibly have an average toggle rate that high (100%).*

Note: The I/O sheet has a column to specify signal **Data Rate**. Make sure you adjust the **Toggle Rate** and **Data Rate** columns accurately. For example, on an input signal which toggles on both edges of the clock you would enter **Toggle Rate** = **200%** and **Data Rate** = **DDR** (Dual Data Rate).

Signal Rates

Signal rate defines the number of millions of transitions per second (Mtr/s) for the element considered. This is a read-only column that appears on some of the XPE sheets (for example, the Logic, I/O, DSP, and Block RAM sheets). The general equation to calculate signal rate is:

Signal Rate (Mtr/s) = Clock Frequency (Mhz) * Effective Toggle Rate (%)

Fanout

Fanout defined in XPE is similar to the fanout reported by the synthesis tool and can differ from the fanout reported by the implementation tool. This difference is expected because fanout will vary with placement and packing of the logic.

- In XPE, fanout represents the number of individual loads or logic elements the considered element is connected to (LUTs, flip-flops, block RAM, I/O flip-flops, distributed RAM, and shift registers).
- In the Vivado IDE, fanout represents the number of SLICEs the considered net is routed to. A SLICE typically contains multiple logic elements and you generally do not control packing of the different elements into SLICEs. XPE algorithms will estimate this packing before calculating the power.

Effective ΘJA (C/W)

This coefficient defines how power is dissipated from the Xilinx device to the environment (device junction to ambient air). Typically this option is calculated by XPE, taking into account, among other things, the different environment parameters in the **Settings** panel of the Summary sheet. Entering a value in this field will override XPE calculations. Use this option if you have calculated this parameter by thermal simulations. You might also want to use this feature to factor out environmental parameters when analyzing power differences with another spreadsheet in which environment settings have been set differently.

ΘSA (C/W)

ΘSA represents the heat sink to ambient air thermal resistance. By default XPE obtains this value from a representative selection of heat sink data matched to the device package,

combined with the **Heat Sink** value you set (**Low Profile**, **Medium Profile**, or **High Profile**) and the **Airflow** value you set. The value used by XPE is shown in the **ΘSA** field on the Summary sheet.

If you have the ΘSA information for your system you can enter your specific value. First set the **Heat Sink** drop-down menu on the Summary sheet to **Custom**, then enter your **ΘSA** value.

ΘJB (C/W)

ΘJB represents the device junction to board thermal resistance. By default XPE estimates the junction to board thermal resistance based on standard JEDEC four-layer measurements. If you have done thermal simulations of your system you can enter your own specific value. First set the **Board Selection** drop-down menu on the Summary sheet to **Custom**, then enter your **ΘJB** value.

Junction Temperature (°C)

This user defined field forces the value of the device junction temperature. XPE then adjusts the ambient temperature to meet the specified junction temperature. This option could be used when you need to work backward from a known or assumed worst case junction temperature and define the environment that would ensure this temperature is not exceeded.

The Xilinx Analog to Digital Converter (XADC) component is included in many of the current devices. As the XADC measures the Junction Temperature, you should wait for the value to stabilize before and after configuring the device.

Using XPE User Interface

XPE has the following sheets:

- The Summary sheet lets you enter and edit all device and environment settings. This sheet also displays a summary of the power distribution and provides buttons to import data into XPE, export results, and globally adjust settings.
- Other sheets allow you to enter usage and activity details for the different resource types available in the targeted device, for example, I/O, Block RAM (BRAM), and Multi-Gigabit Transceivers (MGTs). These sheets report design power based on the resource usage. Resource leakage power is shown on the Summary sheet.

TIP: *XPE is intended to be intuitive to the novice spreadsheet-user. For information about a cell in the spreadsheet, move the mouse over the comment indicators (red triangle at the top right corner of the title cells) to read the relevant notes for the intended use (see Figure 1-2).*

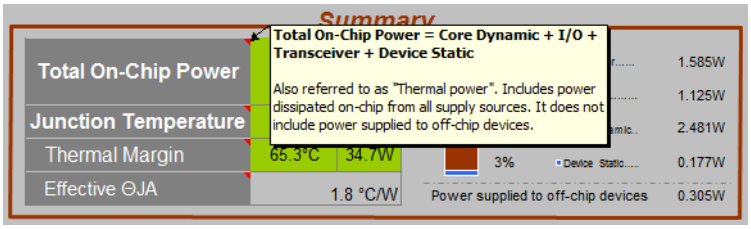


Figure 1-2: **Comment Indicators and Comment**

XPE Cell Color-Coding Scheme

To simplify data entry and review, the XPE cells are color coded. A color **Legend** appears at the bottom of the Summary sheet (see Figure 1-3).

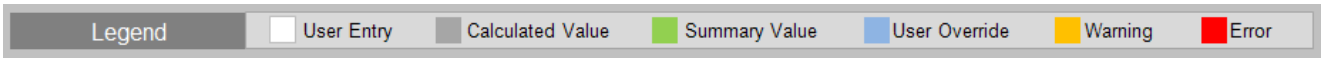


Figure 1-3: **Color Legend (Summary Sheet)**

A description of the spreadsheet color-coding scheme is provided in Table 1-2.

Table 1-2: **XPE Cell Color-Coding Scheme**

Cell Color	Cell Use	Available User Action
White	Allows user to enter data	Editable
Gray	Displays a calculated value	Read-only
Green	Displays a summary value	Read-only
Blue	User override of cells normally calculated by XPE	Editable
Orange	Flags a warning. Indicates that a resource is not available or a specification is invalid.	Editable
Red	Flags an error. Examples of errors are: • A resource limit in the device has been exceeded. • The limits of a device specification (for example, junction temperature) have been exceeded.	Read-only. Edit other cells to correct the error.

Using the Summary Sheet

The Summary sheet is the default sheet on launch and allows you to enter all device and environment settings. On this sheet the tool also reports estimated power rail-wise and

block-wise so you can quickly review thermal and supply power distribution for your design (see Figure 1-4).

You can add a description, short details about the design, or calculations related to the design in the following places:

- A **Project** field at the top of the Summary sheet allows you to add a description of the design.
- In the XPE spreadsheets for 7 Series and above devices, an area of boxes to the right of the Summary sheet allows you to add a description, details about the design, or calculations related to the design. In this area you can add links, data tables, graphics, or any other object you can enter in a regular Excel document.
- In spreadsheets for pre-7 series devices, a **Comment** field at the bottom of the Summary sheet allows you to add a description or short details about the design.
- If your data does not fit in the boxes on the Summary sheet, go to the **User** sheet. There you can add links, data tables, graphics, or any other object you can enter in a regular Excel document.

TIP: *The Spartan-3, Spartan-3E, and Virtex-4 device spreadsheets have a slightly different layout for this sheet. The description of the different user settings and data presented in this view is, however, applicable to these spreadsheets.*

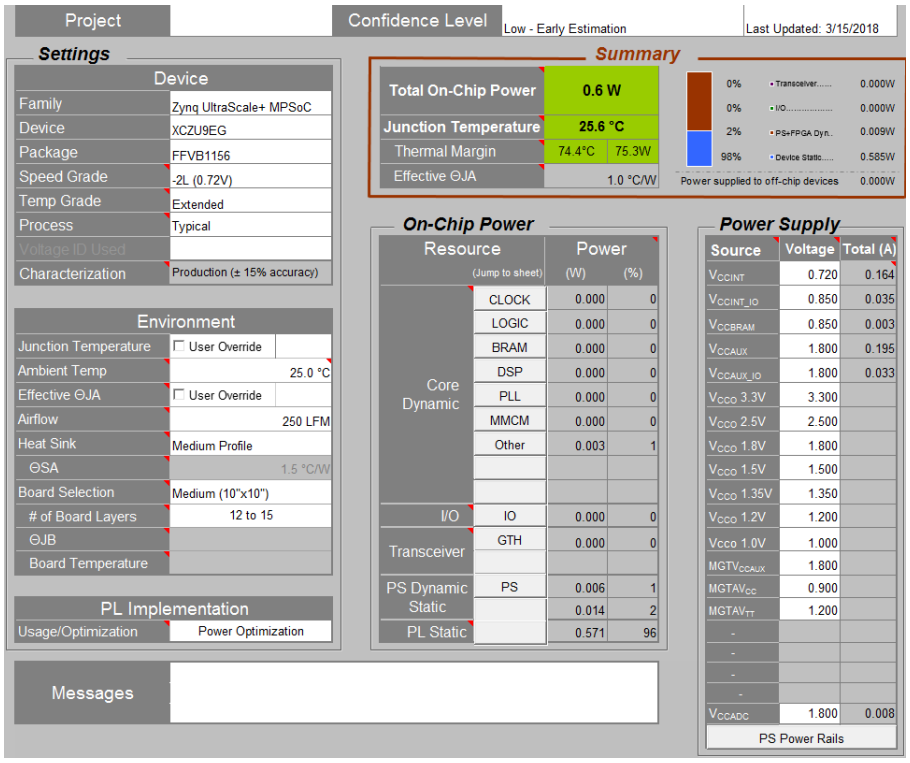


Figure 1-4: **Summary Sheet - Adjust Settings and Display Power Results**

Using the Settings Panel

Use the **Settings** panel to specify details of the device, board, cooling and ISE or Vivado Design Suite settings. This panel varies slightly depending on the targeted device. A Kintex UltraScale device example is presented in Figure 1-5.

Some settings are dependent on other settings. When this occurs the dependent cell becomes un-editable and turns to a gray background.

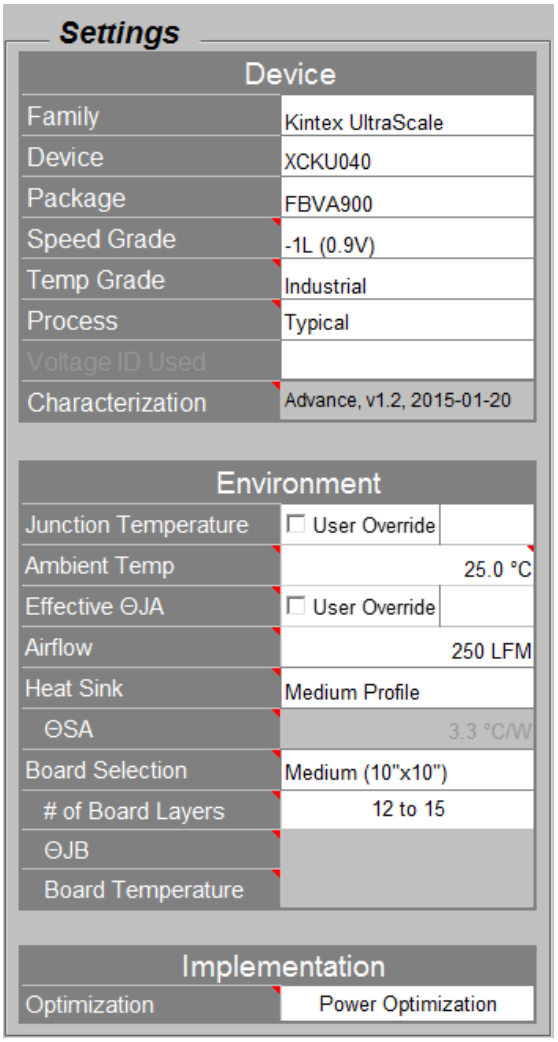


Figure 1-5: **Settings Panel**

The sections in the **Settings** panel are:

- **Device**

  Select the smallest device which meets your requirements.

IMPORTANT: *Larger devices exhibit higher device static power consumption.*

The 7 series spreadsheet has a **Voltage ID Used** entry, which applies to Virtex®-7, -1 **Speed Grade**, Commercial **Temp Grade**, and Maximum **Process** FPGAs only. If **Voltage ID Used** is set to Yes, XPE will perform all of its power calculations based on the device operating at the Voltage ID voltage. The Voltage ID (VID) voltage is the minimum possible V_{CCINT} voltage at which the Xilinx device can run and still meet its performance specifications. This voltage is tested when the Xilinx device is manufactured and the value is programmed into the DNA eFuse register on the Xilinx device. Activating the VID feature in your design to operate the Xilinx device at this VID voltage can result in a significant power savings over operating the Xilinx device at its nominal voltage.

- **Environment**

  For XPE to report the estimated junction temperature it needs to understand how the device logic is configured and activated. It also needs a description of the device environment. The information of how heat can be transferred into the surrounding air (Θ**SA**) or PCB (Θ**JB**) affects the device junction temperature. If these parameters are known enter them; otherwise, select from the different drop-down menus the environment settings closest to your specific project. This will help to indirectly determine **Effective** Θ**JA**.

IMPORTANT: *XPE uses a 2-R thermal model to calculate the junction temperature. The XPE thermal model assumes two main paths of heat flow through the top and bottom of the device into the board. The thermal model uses environment settings entered for ambient temperature, airflow, heat sink, and board selection in the effective thermal resistance and junction temperature calculations.*

Because the junction temperature estimate in XPE is based on a board setup that might vary from your actual board setup, it might not account for the effect of other heat sources on the actual board system, such as other board components close to the Xilinx device. These variations can result in differences between the XPE thermal estimate and a thermal measurement of the actual system.

For more details about the thermal parameters of the Xilinx Power Estimator, please refer to Chapter 3: Thermal Management & Thermal Characterization Methods & Conditions in the *Device Package User Guide* (UG112) [Ref 23].

- Implementation: The Implementation options are labeled as per the selected device. The labels are as follows:
  - **Implementation** (7 series, UltraScale and UltraScale+ devices)
  - **PL Implementation** (Zynq-7000 SoC and Zynq UltraScale+ MPSoC devices)
  - **ISE** (earlier device architectures)

  Settings in this section are available to focus the synthesis and implementation tools on minimizing towards different objectives. Adjust this area to best match the ISE or Vivado Design Suite settings you plan on using. This option affects the core dynamic power by an amount seen in a suite of customer designs.

IMPORTANT: *In an UltraScale/UltraScale+ device spreadsheet, this section is labeled* ***Implementation****, and only* ***Power Optimization, Default*** *and* ***None*** *settings are available. In a 7 series spreadsheet, this section is labeled* ***Implementation****, and only* ***Default*** *and* ***Power Optimization*** *settings are available. In a Zynq-7000 SoC and Zynq UltraScale+ MPSoC spreadsheets, this section is labeled* ***PL Implementation****, and only* ***Default****,* ***Power Optimization****, and* ***Powered Off*** *settings are available.*

Optimization settings are:

- **Area Reduction** - Minimizes slice usage
- **Balanced** - Default ISE Design Suite options
- **Default** - Default ISE or Vivado Design Suite options
- **Minimum Runtime** - Minimizes the runtime
- **Power Optimization** - Minimizes core dynamic power
- **Timing Performance** - Verifies the timing performance
- **Powered Off** (Zynq-7000 SoC and Zynq UltraScale+ MPSoC devices only)
- **None** (UltraScale/UltraScale+ devices only) - Turns off all power optimizations

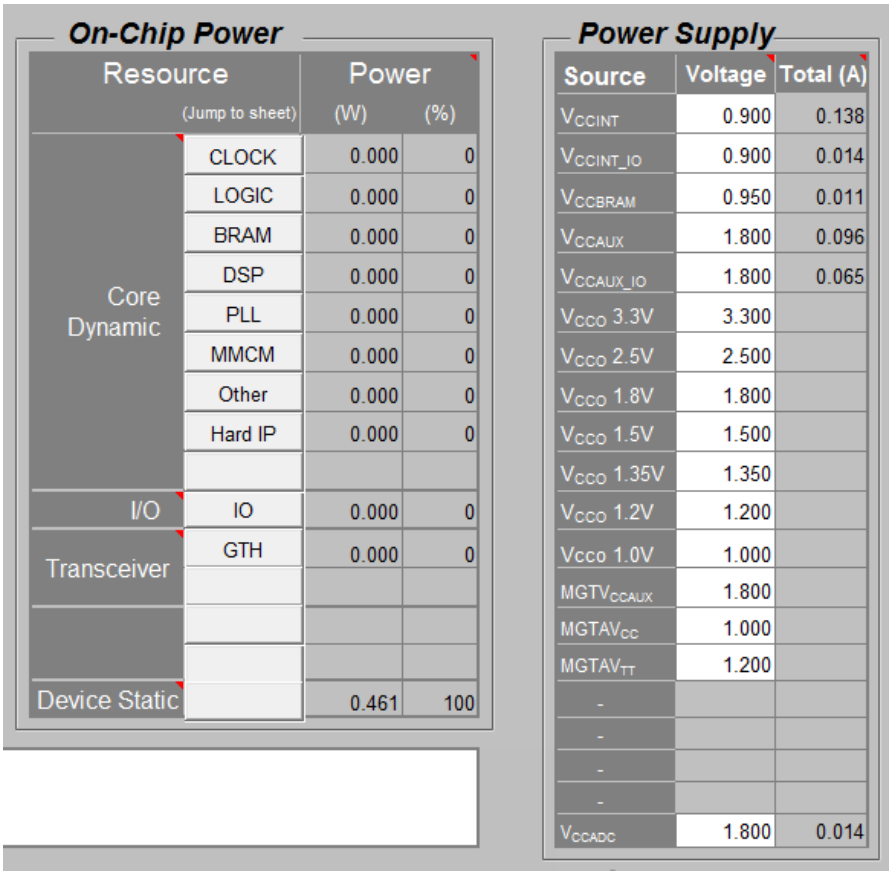
- **Power mode**

  Power Mode is available for some device families. This setting allows you to review the estimated power for the different active and power down modes of the device.

Using the Power Distribution Panels

There are two separate aspects to evaluate when integrating Xilinx devices in a system. Typically designers first evaluate the device current drawn on each voltage supply to ensure all voltage sources can provide enough power for the device to function properly. Second, designers need to know how much of that supplied power is consumed by the device itself as opposed to power supplied to off-chip components such as board termination networks. The power consumed on-chip, also referred to as thermal power, generates heat that must be transferred to the environment to maintain the device junction temperature within the normal operating range. Figure 1-6 shows the on-chip power contributing to junction temperature (**On-Chip Power** panel) and the total supply power (**Power Supply** panel).

In Zynq UltraScale+ MPSoC, power supply panel will change based on the selected device. It will display link for PS Rails as well as RF ADC-DAC Power Rails sheets as shown in Figure 1-7.



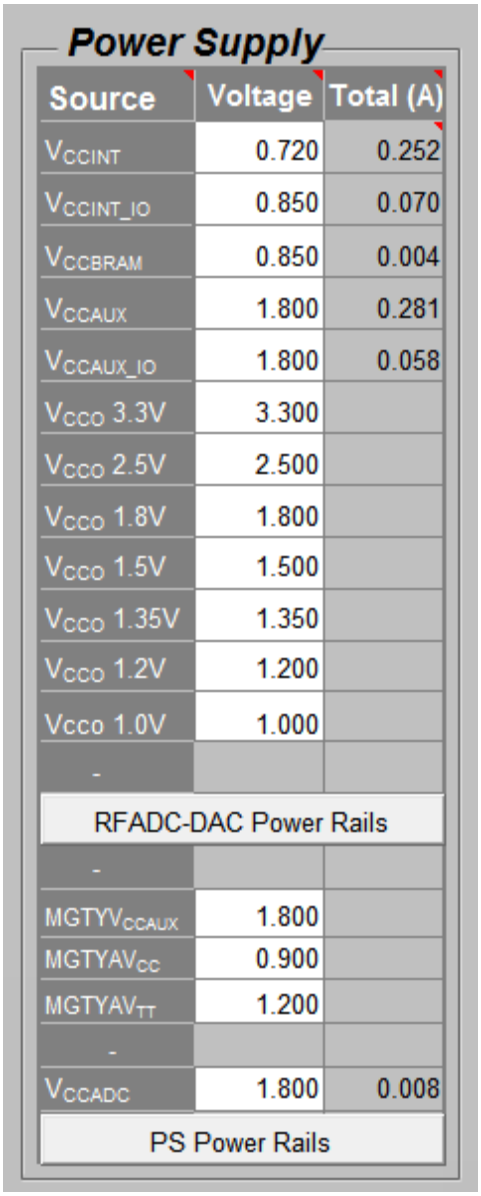
On-Chip Power

Resource		Power (W)	Power (%)
Core Dynamic	CLOCK	0.000	0
	LOGIC	0.000	0
	BRAM	0.000	0
	DSP	0.000	0
	PLL	0.000	0
	MMCM	0.000	0
	Other	0.000	0
	Hard IP	0.000	0
I/O	IO	0.000	0
Transceiver	GTH	0.000	0
Device Static		0.461	100

Power Supply

Source	Voltage	Total (A)
V_{CCINT}	0.900	0.138
V_{CCINT_IO}	0.900	0.014
V_{CCBRAM}	0.950	0.011
V_{CCAUX}	1.800	0.096
V_{CCAUX_IO}	1.800	0.065
V_{CCO} 3.3V	3.300	
V_{CCO} 2.5V	2.500	
V_{CCO} 1.8V	1.800	
V_{CCO} 1.5V	1.500	
V_{CCO} 1.35V	1.350	
V_{CCO} 1.2V	1.200	
V_{CCO} 1.0V	1.000	
$MGTV_{CCAUX}$	1.800	
$MGTAV_{CC}$	1.000	
$MGTAV_{TT}$	1.200	
-		
-		
-		
-		
V_{CCADC}	1.800	0.014

Figure 1-6: **Power Distribution Panels**



Power Supply

Source	Voltage	Total (A)
V_{CCINT}	0.720	0.252
V_{CCINT_IO}	0.850	0.070
V_{CCBRAM}	0.850	0.004
V_{CCAUX}	1.800	0.281
V_{CCAUX_IO}	1.800	0.058
V_{CCO} 3.3V	3.300	
V_{CCO} 2.5V	2.500	
V_{CCO} 1.8V	1.800	
V_{CCO} 1.5V	1.500	
V_{CCO} 1.35V	1.350	
V_{CCO} 1.2V	1.200	
V_{CCO} 1.0V	1.000	
-		
RFADC-DAC Power Rails		
-		
$MGTYV_{CCAUX}$	1.800	
$MGTYAV_{CC}$	0.900	
$MGTYAV_{TT}$	1.200	
-		
V_{CCADC}	1.800	0.008
PS Power Rails		

Figure 1-7: **Power Distribution Panels for Zynq UltraScale+ MPSoC**

Using the On-Chip Power Panel

The **On-Chip Power** panel presents the total power consumed within the device. It includes device static and user design dependent static and dynamic power. The total is broken out by resource type. This view can help determine the amount of power being consumed and dissipated by the device. It also helps identify potential areas in the user logic where trade-offs or power optimization techniques could be used to meet the targeted power budget.

In this view, you can click the resource name to directly jump to the detailed sheet for this resource.

Using the Power Supply Panel

The **Power Supply** panel displays the device estimated power across the different supply sources. For example, this information can be used to size or review voltage supply components, such as regulators. The table includes all power required by the internal logic along with power eventually sourced and consumed outside the Xilinx device, such as in external board terminations. This view includes both static and dynamic power.

You can adjust individual voltages within the supported range and XPE will calculate and display the total current required.

IMPORTANT: *When Maximum Process is selected in the Device table and any power-on supply current values exceed the estimated operating current requirements, the Power Supply panel displays the minimum power-on supply requirements, in blue. If any of the current values appear in blue, the total power indicated in the Power Supply panel will not match the Total On-Chip power in the Summary Panel.*

IMPORTANT: *For the Maximum Process, the sum of rail-wise power will not match the Total On-Chip power as there are various other factors which affect the currents.*

Alert for Maximum Package Current: This is applicable only for Virtex UltraScale+ devices. The Total Iccint current value field in power supply section turns in to red, when estimated current exceeds the maximum specification limit of a selected package.

Multiple power supplies are required to power a Xilinx device. For logic resources typically available in Xilinx devices, Table 1-3 presents the voltage source that typically powers them. This table is provided only as a guideline because these details might vary across Xilinx device families.

Table 1-3: **FPGA Resources and the Power Supply that Typically Powers Them**

Power Supply	Resources Powered
V_{CCINT}	• All CLB resources • All routing resources • Entire clock tree, including all clock buffers • Block RAM/FIFO • DSP slices • All input buffers • Logic elements in the IOB (ILOGIC/OLOGIC) • ISERDES/OSERDES • PowerPC™ processor(1) • Tri-Mode Ethernet MAC(1) • Clock Managers (MMCM, PLL, DCM, etc.)(1) • PCIe and PCS portion of MGTs
V_{CCBRAM}(3)	• Memory array of block RAMs

Table 1-3: **FPGA Resources and the Power Supply that Typically Powers Them** ***(Cont'd)***

V_{CCO}[2]	• All output buffers • Some input buffers • Input termination • Reference resistors to DCI
V_{CCAUX} **V_{CCAUX_IO}**[4]	• Clock Managers (MMCM, PLL, DCM, etc.)[1] • IODELAY/IDELAYCTRL • All output buffers • Differential Input buffers • V_{REF}-based, single-ended I/O standards, for example, HSTL18_I • Phaser[1]
$MGTAV_{CC}$ **$MGTAV_{TT}$** **$MGTV_{CCAUX}$** **V_{CCINT_GT}**	• Analog supply voltages for PMA circuits of transceivers • Transceiver termination circuits • Quad PLL • GTM Core supply
V_{CCPINT} **V_{CCPAUX}** **V_{CCPLL}** **V_{CCO_DDR}** **V_{CCO_MIO}**	• Zynq-7000 SoC: ◦ Processor ◦ Memory ◦ I/O ◦ Peripherals ◦ AXI Interfaces
$V_{CC\_PSINTFP}$ **$V_{CC\_PSINTLP}$** **V_{CC_PSAUX}** **$V_{CCPSINTFP\_DDR}$** **V_{CC_PSPLL}** **$V_{PS\_MGTRAVCC}$** **$V_{PS\_MGTRAVTT}$** **V_{CCO_PSDDR}** **$V_{CCO\_PSDDR\_PLL}$** **V_{CCO_PSIO}** **V_{CCINT_VCU}**	• Zynq-UltraScale+ MPSoC: ◦ Processor ◦ Memory ◦ I/O ◦ Peripherals

Notes:

1. These resources are available only in certain device families. Refer to the appropriate data sheets and user guides for more information.
2. V_{CCO} in bank 0 (V_{CCO_0} or V_{CCO_CONFIG}) powers all I/Os in bank 0 as well as the configuration circuitry. See the applicable Configuration User Guide.
3. Xilinx 7 series Block RAM/FIFO only.
4. Xilinx 7 series High Performance (HP) I/O banks only.

Using the Summary Panel

The Summary panel presents in a concise format the main data of interest.

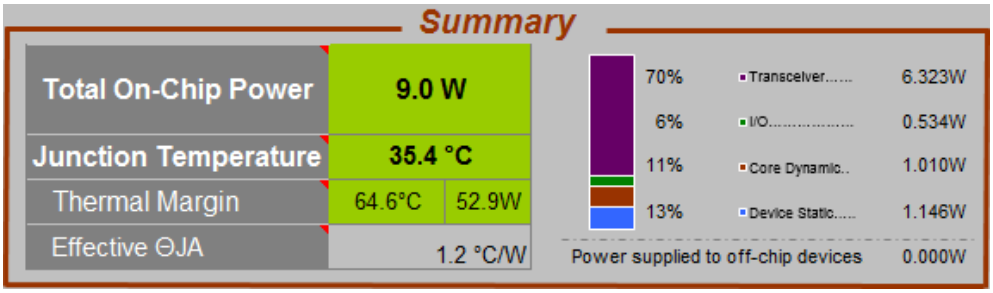


Figure 1-8: **Summary Panel**

- **Total On-Chip Power**

  Includes power consumed and dissipated by the device across all supply sources. Also referred to as thermal power. This cell follows the color scheme of the **Junction Temperature** cell described above.

- **Junction Temperature**

  Estimated junction temperature as the design operates. Each device operates within a temperature grade specified in the data sheet. The background for this cell turns orange when the value is outside the operating range (timing might be affected) and turns red when outside the absolute maximum temperature (device damage possible). The background color turns light blue when the value is set by user.

  - **Thermal Margin**

    Temperature and power margin up to or in excess of the maximum accepted range for this device Grade. Thermal margin is negative when estimated junction temperature exceeds the maximum specified value. In this case, use this information to decide how best to address the excess power consumed on-chip. Refer *7 Series FPGAs Packaging and Pinout Product Specifications User Guide* (UG475) [Ref 7] and *UltraScale and UltraScale+ FPGAs Packaging and Pinouts Product Specification User Guide* (UG575) [Ref 13]for detailed information on thermal resistance.

  - **Effective ΘJA**

    The calculated Effective Thermal Resistance (Effective ΘJA) summarizes how heat is transferred from the die to the environment. The value is calculated from the settings entered in the **Environment** panel. If you have run thermal simulations of your environment then you can also override this value (in the **Environment** panel).

Using the XPE Toolbar

To make data entry into the tool easier, XPE supports importing data from different sources and allows settings to be changed globally. The toolbar is shown in Figure 1-9.

Note: This toolbar is for the 7 series/Zynq-7000 SoC XPE spreadsheet. Toolbar buttons for earlier architecture spreadsheets might have different names.

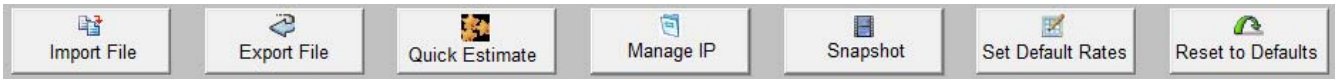


Figure 1-9: **XPE Toolbar**

Import File

Depending on what stage your design is in the device development cycle, use this dialog box to import design information and activity into the spreadsheet. In the dialog box, select the **Files of type** field to determine whether you will import an .xls or .xlsm, .mrp, or .xpe file.

For a description of the import feature, see Importing Data into XPE.

Export File

The **Export File** button lets you export the following information from the current spreadsheet:

- The current settings for your design within XPE. These settings can be imported into an XPower Analyzer session within the ISE Design Suite.
- A text power report, which allows you to analyze the power information in the XPE spreadsheet in a textual format.
- A Power Constraint file in .xdc format that can be used to setup the environment conditions, power budget etc. in Vivado Design Suite before running report power.

For a description of the export feature, see Exporting XPE Results.

Quick Estimate

The **Quick Estimate** button opens the Quick Estimate wizard. This wizard is a simple interface to allow novice and expert users to quickly enter the important parameters required for an accurate power analysis of a design implemented in a Xilinx device.

For a description of the Quick Estimate wizard, see Using the Quick Estimate Wizard.

Manage IP

The **Manage IP** button opens the IP Manager dialog box, allowing you to run IP Module wizards to easily enter various types of external memory interfaces (e.g. DDR4, DDR3, DDR3L, LPDDR2, QDR+, RLDRAM), transceiver based interfaces (e.g. 10GBASE-R, Interlaken, PCIe, Aurora, CPRI and many more), and block memory or distributed memory.

For a description of the IP_Manager wizard, see Using IP Module Wizards.

Snapshot

The **Snapshot** button takes a snapshot of the power status of the current design in the XPE spreadsheet, and places this information on the Snapshot sheet. A snapshot captures the device part, environmental information, the power consumed by your design, and the current across each of the power supply sources used in the design.

For a description of Snapshot usage, see Using the Power Comparison Snapshots Sheet.

Set Default Rates

This button opens up a dialog box which lets you change the default frequency, toggle rates or enable rates for the entire design or for specific sheets (see Figure 1-10).

In the dialog box, default values set by XPE are shown in the square brackets [].

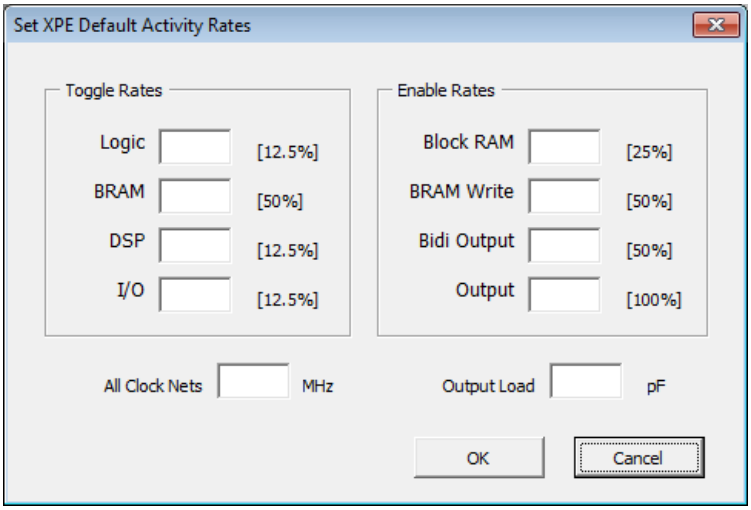


Figure 1-10: **Set Default Activity Rates Dialog Box**

The fields in the dialog box are:

- **Toggle Rates**

  Each field changes activity of the related sheet only. Acceptable range: 0 to 100%.

To learn more about toggle rates, refer to Toggle Rates.

- **Enable Rates**

  Each field changes activity of the related sheet only. Acceptable range: 0 to 100%.

- **All Clock Nets**

  The clock frequency entered here applies to CLOCK, LOGIC, IO, BRAM and DSP sheets.

- **Output Load**

  The equivalent capacitance seen by the output driver for the routing and components connected to this board trace. This setting does not affect power calculations for inputs.

Reset to Defaults

The **Reset to Defaults** button resets all user settings to their default values, except for values in the **Device** selection table on the Summary sheet, and deletes all user entered values on the block details sheets (Clock, Logic, etc.).

Using XPE Wizards

The XPE wizards, available in the 7 series/Zynq-7000 SoC or UltraScale/UltraScale+ device XPE spreadsheet, are simple interfaces to allow novice and expert users to quickly enter the important parameters required for an accurate power analysis of a design implemented in a Xilinx device.

There are two types of XPE Wizards:

- **Quick Estimate wizard** - The Quick Estimate wizard populates the XPE sheets with information about your entire design, allowing XPE to perform a rough power estimate for the design. The Quick Estimate wizard is often used as the first step in specifying your design in XPE to determine its power requirements.
- **IP Module wizards** - The IP Module wizards extend XPE to allow you to easily populate the XPE spreadsheet with information about:
  - Various types of external memory interfaces (for example, DDR3, DDR3L, LPDDR2, QDR+, and RLDRAM).
  - Transceiver based interfaces (for example, 10GBASE-R, Interlaken, PCIe, Aurora, and CPRI).
  - Block memory and distributed memory used in your design.

Using the Quick Estimate Wizard

The Quick Estimate wizard populates the XPE sheets with information about your entire design, allowing XPE to perform a rough power estimate for the design. The Quick Estimate wizard is often used as the first step in specifying your design in XPE to determine its power requirements.

After you run this rough estimate using the Quick Estimate wizard, you can view the data the wizard entered, modify the spreadsheet entries the wizard created, and add entries of your own to describe your design more completely.

If you run the Quick Estimate wizard a second time, you will replace all the spreadsheet entries from the previous run with entries from the current run.

The following manuals will help you supply information to the Quick Estimate Wizard:

- *7 Series FPGAs Configurable Logic Block User Guide* (UG474) [Ref 5]
- *7 Series FPGAs Memory Resources User Guide* (UG473) [Ref 6]
- *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 8]
- *7 Series FPGAs GTP Transceivers User Guide* (UG482) [Ref 9]
- *Zynq-7000 SoC and 7 Series Devices Memory Interface Solutions User Guide* (UG586) [Ref 10]
- *UltraScale Architecture Configurable Logic Block Advance Specification User Guide* (UG574) [Ref 11]
- *UltraScale Architecture Memory Resources Advance Specification User Guide* (UG573) [Ref 12]
- *UltraScale Architecture GTH Transceivers Advance Specification User Guide* (UG576) [Ref 14]
- *UltraScale Architecture GTY Transceivers Advance Specification User Guide* (UG578) [Ref 15]

The Quick Estimate wizard can be started from the Summary sheet by clicking the **Quick Estimate** button.

1. In the Summary sheet **Using the Settings Panel** specify the target part, including the **Speed Grade** and **Temp Grade**.
2. On the Summary sheet, click the Quick Estimate button.

3. In the XPE Quick Estimate dialog box, fill out the information in the dialog box for your design.

   The entries available in the dialog box depend on the Xilinx device in which you will implement your design.

4. When you have filled out the values for your design, click **OK**.

   After a DRC (Design Rules Check) runs, the sheets in Xilinx Power Estimator spreadsheet will be populated based on the values you entered, and XPE will estimate power for the design you specified.

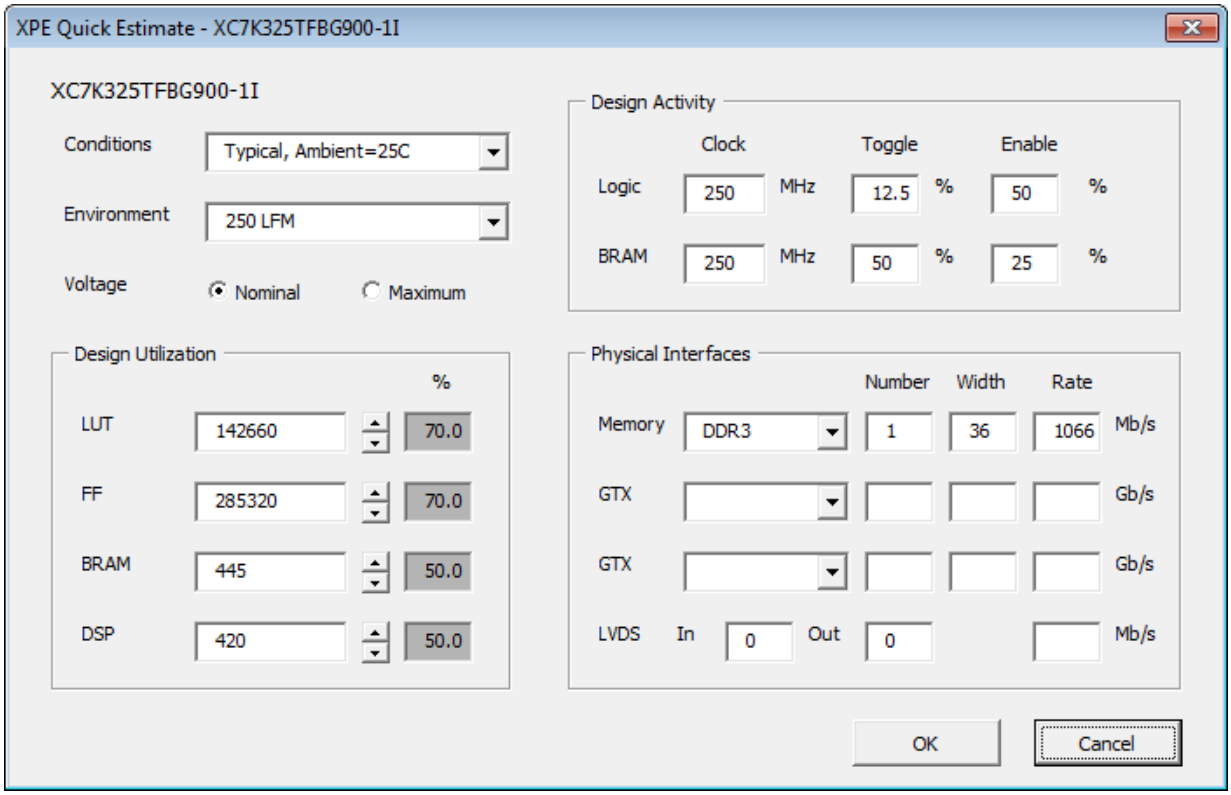


Figure 1-11: **XPE Quick Estimate Dialog Box (Kintex-7 Devices)**

The fields in the XPE Quick estimate dialog box are:

Conditions

This selection allows you to choose:

- A **Typical** process and nominal voltages at the specified **Ambient** temperature.

  OR

- A **Maximum** process and maximum voltages, with the **Junction** temperature set for a worst case power analysis at the specified temperature grade limit.

Environment

Allows you to select the airflow environment under which your device will operate (**Still Air**, **250 LFM**, or **250 LFM (w/Heatsink)**).

Voltage

Allows you to specify whether XPE will calculate power assuming the device is operating with all supplies at their **Nominal** or **Maximum** voltages.

Design Activity

- **Clock**

  Specify a single clock frequency, in MHz. The **Clock** frequency defaults to different values for the different device families (Artix-7 (including Artix-7 Automotive), Kintex-7, and Virtex-7), but you can set the **Clock** frequency to any value.

- **Toggle**

  Enter a single **Toggle** rate (in %). This toggle rate will apply to all the resources in the **Logic** or to the **BRAM**.

- **Enable**

  Enter a single **Enable** rate (in %). The **Enable** rate will apply to the slice clock enable in the **Logic** or to the **BRAM** enable.

Design Utilization

Enter the number of each resource (**LUT**, **FF**, **BRAM**, and **DSP**) you estimate your design will use.

The **%** column shows the percentage of utilization for the resource in the specified device.

You can enter a number in the box provided or use the spin buttons (the up and down arrowheads) to increase or decrease the utilization **%** by 5% each click.

If you try to enter a value greater than the total number of the resource in the device (for example, you try to enter 10,000 LUTs for a device that only contains 9600 LUTs), the value displayed will change to the total number of the resource in the device (in this example, 9600 LUTs) and the utilization **%** will be 100%.

Physical Interfaces

For the memory interface (**Memory**) you specify, enter the number of interfaces to add (**Number**), a bit width (**Width**), and a data rate (**Rate**) in Mb/s.

For the transceiver interfaces (**GTP**, **GTX**, etc.) you specify, enter the number of interfaces to add (**Number**), a bit width (**Width**), and a data rate (**Rate**) in Gb/s.

For **LVDS**, specify the number of differential pairs (**In** and **Out**), and the I/O data rate in **Mb/s**.

Using IP Module Wizards

The IP Module wizards extend XPE to allow you to easily enter various types of external memory interfaces (for example, DDR3, DDR3L, LPDDR2, QDR+, RLDRAM), transceiver based interfaces (for example, 10GBASE-R, Interlaken, PCIe, Aurora, and CPRI) and block memory or distributed memory.

The IP Module wizards are:

- Using the Memory Generator Wizard (for Distributed Memory)
- Using the Memory Generator Wizard (for Block Memory)
- Using the Memory Interface Configuration Wizard
- Using the Transceiver Configuration Wizard

Using the Memory Generator Wizard (for Distributed Memory)

In the 7 Series/Zynq-7000 SoC and UltraScale/UltraScale+ XPE spreadsheets, the Memory Generator wizard allows you to enter distributed memory information in the Logic sheet. You can access the Memory Generator Wizard by clicking the **Manage IP** button on the Summary sheet or the IP Manager sheet, or the **Add Memory** button on the Logic sheet. The XPE Memory Generator wizard provides a simplified method of populating the Logic sheet with rows related to distributed memory.

To understand the capabilities of distributed memory and the settings you will enter within XPE, see the *7 Series FPGAs Configurable Logic Block User Guide* (UG474) [Ref 5] and *UltraScale™ Architecture Configurable Logic Block Advance Specification User Guide*(UG574) [Ref 11].

To generate the Logic sheet using the XPE Memory Generator Wizard (the example shown below is from 7 series/Zynq-7000 SoC and later devices):

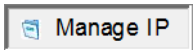
1. Open the Memory Generator wizard by doing one of the following:

   On the Logic Sheet, click the **Add Memory** button.

   OR

   On the IP Manager Sheet, click the **Manage IP** button.

   a.In the IP Manager dialog box, select the **Create IP** tab.

   b.In the dialog box IP Catalog, select **Distributed Memory**.

   c.In the dialog box, click the **Create** button.

2. In the Distributed Memory tab of the XPE Memory Generator dialog box, fill out the information in the dialog box for one distributed memory **Memory Type** in your design.

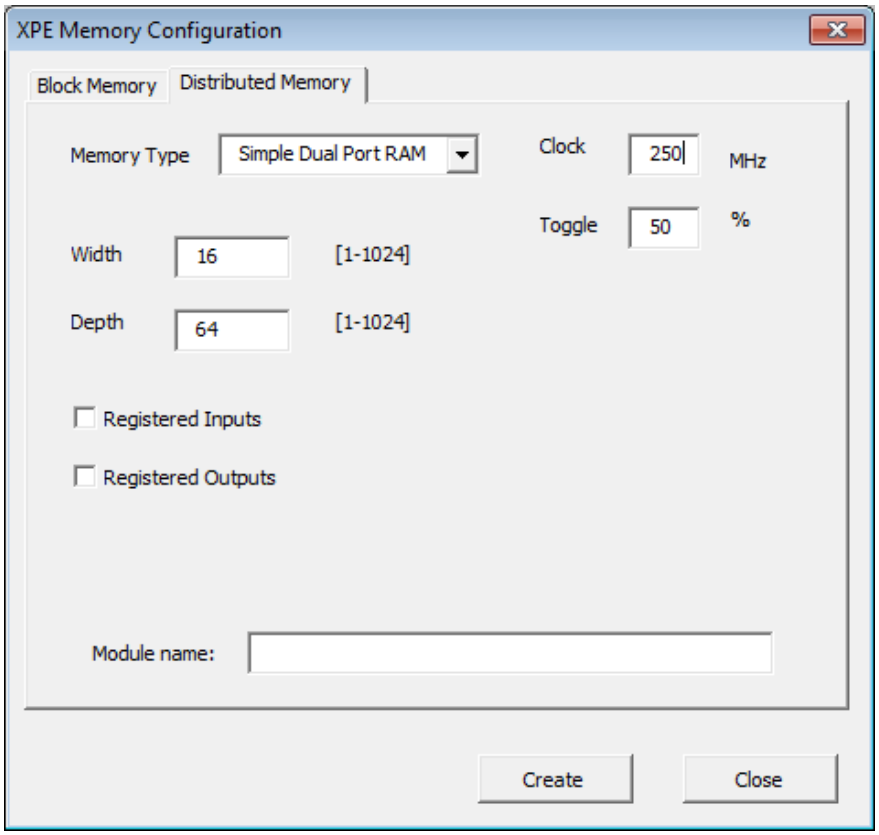


Figure 1-12: **Distributed Memory Tab - XPE Memory Generator Dialog Box (Virtex-7 Devices)**

The fields in the Distributed Memory tab are:

- **Memory Type**

  Select the type of memory your design will use.

  - **Single Port RAM**
  - **Simple Dual Port RAM**
  - **Single Port ROM**
  - **Dual Port ROM**

  For a description of these memory types, see the *7 Series FPGAs Configurable Logic Block User Guide* (UG474) [Ref 5] and *UltraScale Architecture Configurable Logic Block User Guide* (UG574)[Ref 11].

- **Clock**

  Enter the clock frequency at which the distributed memory will operate.

  For dual-port memory types, XPE assumes the same clock frequency for both ports.

- **Toggle**

  Enter the average toggle rate of the data signals. A toggle rate of 50% means that half of the data signals toggle each clock cycle.

- **Width**

  Enter the bit width for each word in the memory.

- **Depth**

  Enter the depth of the memory. **Width** × **Depth** is the total number of bits in the memory.

- **Registered Inputs**

  Specify whether the memory inputs will be registered (**Registered Inputs** selected) or not (**Registered Inputs** deselected).

  For a description of input registering, see the *7 Series FPGAs Configurable Logic Block User Guide* (UG474) [Ref 5] and *UltraScale Architecture Configurable Logic Block User Guide* (UG574)[Ref 11].

- **Registered Outputs**

  Specify whether the memory outputs will be registered (**Registered Outputs** selected) or not (**Registered Outputs** deselected).

For a description of output registering, see the *7 Series FPGAs Configurable Logic Block User Guide* (UG474) [Ref 5] and *UltraScale Architecture Configurable Logic Block User Guide* (UG574)[Ref 11].

- **Module name**

  Allows you to assign a name to the generated distributed memory configuration. This will help to distinguish multiple configurations in the XPE sheets.

3. When you have filled out the values for this distributed memory, click **Create**.

   A row in the Logic sheet will be filled in with the information you entered in the dialog box.

4. For each distributed memory type in your design, fill out the dialog box and click **Create**.

   Each time you click **Create** a row will be added to the Logic sheet.

5. When you have configured all of the distributed memory in your design, click **Close** to close the XPE Memory Generator dialog box.

Using the Memory Generator Wizard (for Block Memory)

In the 7 Series/Zynq-7000 SoC and later devices XPE spreadsheets, the Memory Generator wizard allows you to enter block memory information in the spreadsheet. You can access the Memory Generator Wizard by clicking the **Manage IP** button on the Summary sheet or the IP Manager sheet, or the **Add Memory** button on the Block RAM sheet. The XPE Memory Generator wizard provides a simplified method of filling in the Block RAM sheet in XPE.

To populate the Block RAM sheet using the XPE Memory Generator Wizard (example shown below is for 7 series devices):

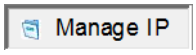
1. Open the Memory Generator wizard by doing one of the following:
   - On the Block RAM sheet, click the **Add Memory** button.

     OR

   - On the IP Manager Sheet:

     a.Click the **Manage IP** button.

b.In the IP Manager dialog box, select the **Create IP** tab.

c.In the dialog box IP Catalog, select **Block Memory**.

d.In the dialog box, click the **Create** button.

2. In the Block Memory tab of the XPE Memory Generator dialog box, fill out the information in the dialog box for one block memory **Memory Type** in your design.

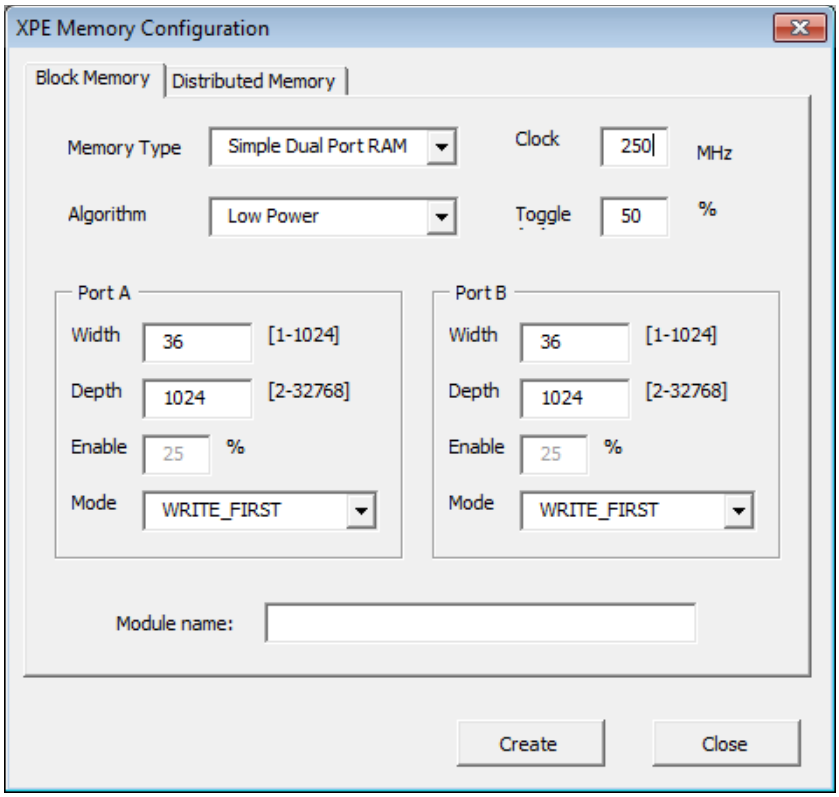


Figure 1-13: **Block Memory Tab - XPE Memory Generator Dialog Box (Virtex-7 Devices)**

The fields in the Block Memory tab are:

- **Memory Type**

  Select the type of memory your design will use.

  - Single Port RAM
  - Simple Dual Port RAM
  - True Dual Port RAM
  - Single Port ROM
  - Dual Port ROM

- **Clock**

  Enter the clock frequency at which the block RAM will operate.

  For dual-port memory types, XPE will assume the same clock frequency for both **Port A** and **Port B**.

- **Algorithm**

  Specify which of these algorithms the Xilinx design tools will use to configure block RAM primitives and connect them together:

  - **Minimum Area**

    The memory is generated using the minimum number of block RAM primitives.

  - **Low Power**

    The memory is generated such that the minimum number of block RAM primitives are enabled during a Read or Write operation.

- **Toggle**

  Enter the average toggle rate of the data signals. A toggle rate of 50% means that half of the data signals toggle each clock cycle.

- **Port A** and **Port B**

  If you have selected a single port **Memory Type**, you will enter information for **Port A** only. If you have selected a dual port **Memory Type**, you will enter information for both **Port A** and **Port B**.

  - **Width**

    Enter the bit width for each word in the port.

  - **Depth**

    Enter the depth of the port. **Width** × **Depth** is the total number of bits in the memory.

  - **Enable**

    Enter the percentage of time that the port will be enabled.

  - **Mode**

    Select the operating mode for the block RAM: **READ_FIRST**, **WRITE_FIRST**, or **NO_CHANGE**.

- **Module name**

  Allows you to assign a name to the generated block memory configuration. This will help to distinguish multiple configurations in the XPE worksheets.

3. When you have filled out the values for this block memory, click **Create**.

   A row in the Block Ram sheet and a row in the Logic sheet will be filled in with the information you entered in the dialog box.

4. For each block memory type in your design, fill out the dialog box and click **Create**.

   Each time you click **Create** a row is added to the Block RAM sheet and the Logic sheet.

5. When you have configured all of the block memory in your design, click **Close** to close the XPE Memory Generator dialog box.

See *7 Series FPGAs Memory Resources User Guide* (UG473) [Ref 6]and *UltraScale Architecture Memory Resources User Guide* (UG573) [Ref 12] for more information on Block RAM and memory settings mentioned above.

Using the Memory Interface Configuration Wizard

For the XPE spreadsheets of 7 Series and later devices, you can enter information for the I/Os involved in the interface between the Xilinx device and external memory by using the Memory Interface Configuration wizard. The Memory Interface Configuration wizard provides a simplified method of filling in the memory interface I/Os in the XPE spreadsheet.

When you configure a memory interface using the wizard, rows are added to the IP Manager sheet, and to the I/O sheet for each output line (for example, Data, Address, and Clock) from the Xilinx device that will be applied to the external memory. The wizard also places rows on the Clock sheet, on the sheet for any clock manager (for example, PLL or MMCM) that is part of the memory interface, and on the Logic sheet. Resources are added representing typical utilization to implement the physical, controller, and user interface layer.

IMPORTANT: *The Memory Interface Configuration wizard does not support all memory interface standards or all interface parameters for the supported standards. The wizard covers many of the common Memory Interface Standards. For a specific standard there could be more pins associated than configured by the wizard. In these cases you might need to modify the output of the wizard or enter the extra pins manually in the I/O sheet for your specific case. Also, if a selection is not available for a specific field, you might be able to manually override the selections in the field.*
For Better accuracy, create IP in Vivado and generate resource information to manually enter in to XPE.

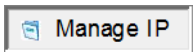
To add memory interface I/Os to the 7 series and above devices I/O sheet using the Memory Interface Configuration Wizard:

1. Open the Memory Interface Configuration wizard by doing one of the following:
   - On the I/O sheet, click the **Add Memory Interface** button.

   OR

   - On the IP Manager Sheet:

     a.Click the **Manage IP** button.



     b.In the IP Manager dialog box, select the **Create IP** tab.

     c.In the dialog box IP Catalog, select **Memory Interface**.

     d.In the dialog box, click the **Create** button.

2. In the XPE Memory Interface Configuration dialog box, fill out the information in the dialog box for one memory interface in your design.

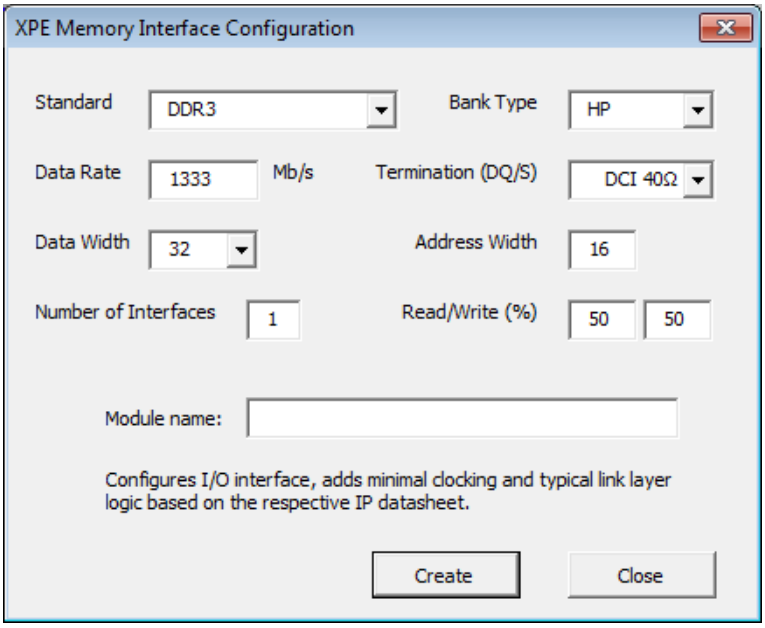


Figure 1-14: **XPE Memory Interface Configuration Dialog Box (Virtex-7 Devices)**

The following figure shows the dialog box for UltraScale devices.

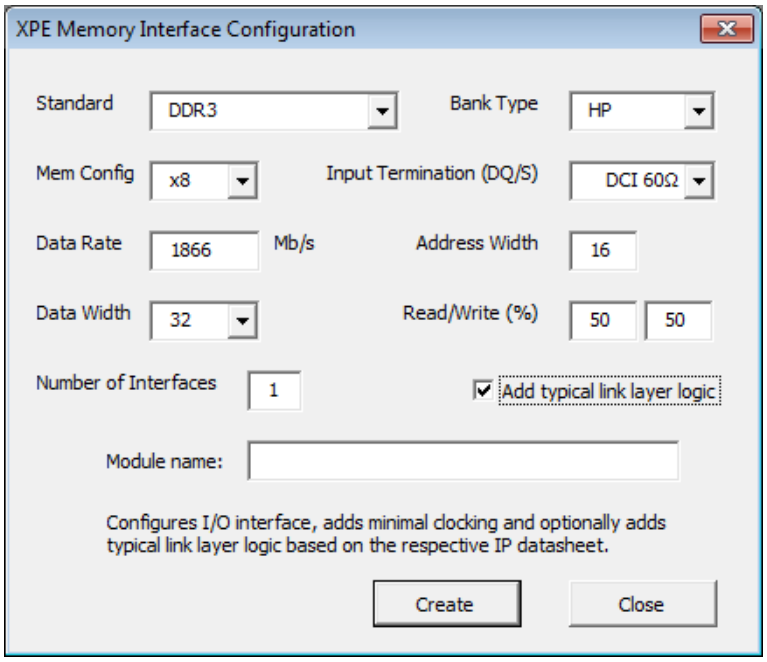


Figure 1-15: **XPE Memory Interface Configuration Dialog Box (UltraScale Devices)**

The fields in the XPE Memory Interface Configuration dialog box are:

- **Standard**

  The Memory Interface Configuration wizard supports these I/O Standards:

  - DDR2
  - DDR3
  - DDR3L
  - DDR4
  - QDR2+
  - RLDRAM2
  - RLDRAM3
  - LPDDR2

  You can also manually enter a memory interface of any other standard in the XPE spreadsheet.

For a listing of the supported I/O standards and limits for your specific device, see the appropriate data sheet:

- *Virtex-7 T and XT FPGAs Data Sheet: DC and AC Switching Characteristics* (DS183) [Ref 16]
- *Kintex-7 FPGAs Data Sheet: DC and AC Switching Characteristics* (DS182) [Ref 17]
- *Artix-7 FPGAs Data Sheet: DC and AC Switching Characteristics* (DS181) [Ref 18]
- *Virtex UltraScale Architecture Data Sheet: DC and AC Switching Characteristics* (DS893) [Ref 19]
- *Kintex UltraScale Architecture Data Sheet: DC and AC Switching Characteristics* (DS892) [Ref 20]
- *Kintex UltraScale+ FPGAs Data Sheet: DC and AC Switching Characteristics* (DS922)[Ref 38]
- *Virtex UltraScale+ FPGAs Data Sheet: DC and AC Switching Characteristics (DS923)[Ref 39]*
- Zynq UltraScale+ MPSoC Data Sheet: DC and AC Switching Characteristics (DS925)[Ref 40]

◦ **Bank Type**

Select the appropriate bank type, where the choice exists.

◦ **Mem Config (UltraScale devices only)**

Select the appropriate memory configuration.

◦ **Termination (DQ/S)/Input Termination (DQ/S)**

Refers to the DQ (data) and DQS (data strobe) pins. For memory interfaces using the HP banks, use DCI termination as appropriate. For the HR banks, select **INTERM_40**, **INTERM_50**, **INTERM_60** or external termination (no entry).

◦ **Data Rate**

Enter the target data rate for your memory device.

◦ **Address Width**

The total number of address lines used in the interface, which includes Row, Col, Bank, and, if used, Rank and CS lines.

◦ **Data Width**

Values from 8-144 in increments of 8 are supported, with memory type and device restrictions. Address, data, and control signals must be in the same I/O column so

the limit is often lower than 144. Stacked Silicon Interconnect (SSI) technology devices are limited to a width of 72 due to this restriction.

- **Read/Write (%)**

  Specify the percentage of the time the memory interface is used for reading from and writing to the external memory. The total must be less than or equal to 100% and the interface is assumed to be idle for 100% - (Read% + Write%) of the time. This is reflected in the **Output Enable**, **Term Disable** and **IBUF Disable** percentages.

- **Number of Interfaces**

  Enter the number of memory interfaces that will use the settings that you are currently entering in the dialog box. When the I/O sheet is populated with the outputs to external memory, the number of pins for each type of line (for example, Address, Data, and Clock lines) will reflect the number of **Interfaces** you specify.

- **Add typical link layer logic (Ultrascale devices only)**

  Enable this option to automatically populate the resources of the link layer logic for a specific memory interface.

- **Module Name**

  Allows you to assign a name to the generated configuration. This will help to distinguish multiple configurations on the I/O sheet.

3. When you have filled out the values for this memory interface, click **Create**.

   Rows in the I/O sheet will be populated with the information you entered in the dialog box.

4. For each memory interface in your design, fill out the information in the XPE Memory Interface Configuration dialog box and click **Create**.

   Each time you click **Create** rows will be added to the I/O sheet, and to the PHASER block on the Other sheet for 7 series devices.

5. When you have configured all of the memory interfaces in your design, click **Close** to close the XPE Memory Interface Configuration dialog box.

Using the Transceiver Configuration Wizard

For the 7 Series and above device XPE spreadsheets, you can enter transceiver information in an MGT sheet (GTP, GTH, GTX, GTY, or GTZ) by using the Transceiver Configuration wizard. The Transceiver Configuration wizard provides a simplified method of filling in the MGT sheets in the XPE spreadsheet.

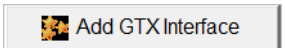
When you configure a transceiver interface using the wizard, rows will be added to the IP Manager sheet and to the sheet for the transceiver that is part of the physical transceiver interface. For some protocols the wizard also places rows on the Logic and clock sheets representing typical resources utilized to implement the data interface layer.

IMPORTANT: *The Transceiver Configuration wizard does not support all transceiver protocols or all transceiver parameters for the supported protocols. Any options not available in a dialog box field need to be entered manually in the field. Any cases where a quad has transceivers using both CPLL and QPLL, different transmit and receive rates, or different power modes, will also have to be entered manually. The wizard covers many common protocols, but you might need to modify the output of the wizard or enter the data manually in the MGT sheet for your specific case.*
For Better accuracy, create IP in Vivado and generate resource information to manually enter in to XPE.

To understand the capabilities of the 7 series, UltraScale, and UltraScale+ device MGTs and the settings you will enter within XPE, refer to the *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 8], the *7 Series FPGAs GTP Transceivers User Guide* (UG482) [Ref 9], *UltraScale Architecture GTH Transceivers Advance Specification User Guide*(UG576) [Ref 14] and *UltraScale Architecture GTY Transceivers Advance Specification User Guide* (UG578)[Ref 15].

To populate the 7 Series MGT sheet using the XPE Transceiver Configuration Wizard:

1. Open the Transceiver Configuration wizard by doing one of the following:
   - On the applicable MGT sheet, click the **Add GT Interface** button.

     OR
   - On the IP_Manager Sheet:

     a.Click the **Manage IP** button.

     b.In the IP Manager dialog box, select the **Create IP** tab.

     c.In the dialog box IP Catalog, select **Transceiver Interface**.

     d.In the dialog box, click the **Create** button.
2. In the XPE Transceivers Configuration dialog box, fill out the information in the dialog box for one set of transceivers in your design.

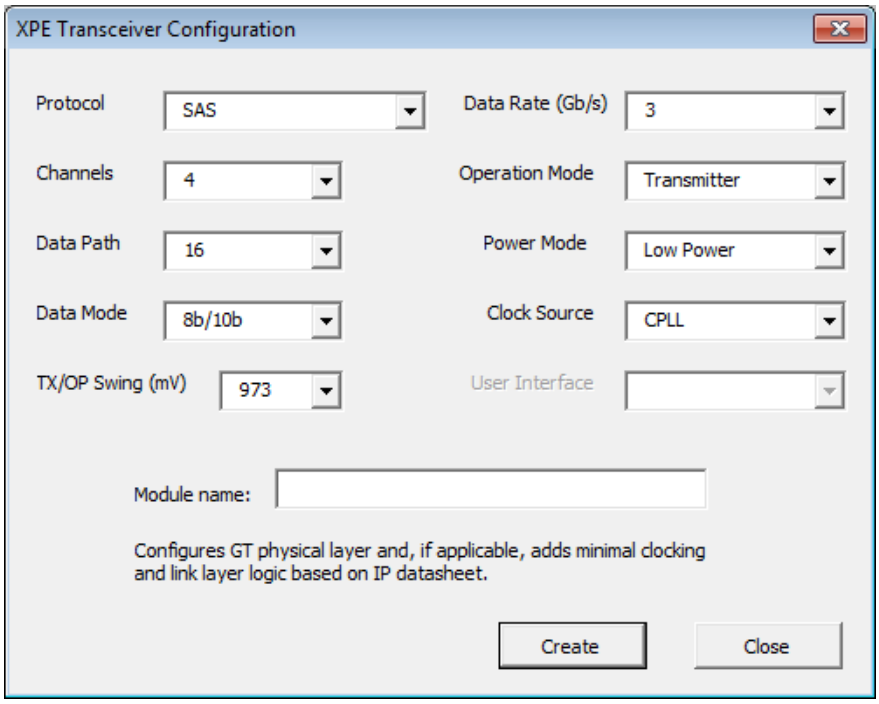


Figure 1-16: **XPE Transceiver Configuration Dialog Box (Virtex-7 Devices)**

The fields in the XPE Transceivers Configuration dialog box are:

- **Protocol**

  Allows you to select from a list of available protocols. Device, package, and speed grade limitations will limit the choices available. In some cases the number of **Channels**, **Data Mode** and **Clock Source** selections will default to values defined by the **Protocol**. The GTP configuration will not have **Power Mode** or **Clock Source** selections. The **Data Rate** and number of **Channels** will also be reflected in the PCIe information (on a GTX, GTP, GTH, or GTY sheet) as appropriate. No clocks or fabric are populated in their respective sheets.

- **Data Rate**

  After selecting the **Protocol** the **Data Rate** will either display as a fixed value defined by the **Protocol** or allow you to enter the specific **Data Rate** used in your system. Except for the rare cases where receive and transmit rates are different, both RX and TX rates will match.

- **Channels**

  Some protocols (for example, PCIe) have specific restrictions for the number of channels and others allow you to enter the number of channels used in your system.

- **Operation Mode**

  By default the **Transceiver** configuration is used, but you can select **Transmitter** or **Receiver** only operation.

- **Data Path** and **Data Mode**

  The width of the port can be configured to be two, four, or eight bytes wide. With **8b/10b** encoding used the port widths can be 16, 32 or 64 bits. With **64b/66b** encoding used the port width must be 64 bits. In **Raw** mode the port widths can be 16, 20, 32, 40, 64, or 80 bits.

- **Power Mode**

  Where the choice exists (as defined by the target transceiver) you can choose to use the power-efficient adaptive linear equalizer mode called the **Low Power** mode (LPM) or the high-performance, adaptive decision feedback equalization (**DFE**) mode.

  For a description of these modes, see the RX Equalizer (DFE and LPM) section in the *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 8].

- **Clock Source**

  Where the choice exists (as defined by the target device and data rate) you can choose to use the LC tank (**QPLL**) or ring oscillator (**CPLL**) based PLL.

- **User Interface**

  The User Interface field is applicable only to the Aurora protocols (**Aurora** and **Aurora 64b/66b**), and specifies the user interface configuration. Resources are added based on the selected **User Interface**. For more information on the Aurora user interfaces, see the *LogiCORE IP Aurora 8B/10B Product Guide* (PG046) [Ref 21] (for the **Aurora** protocol) or the *LogiCORE IP Aurora 64B/66B Product Guide* (PG074) [Ref 22] (for the **Aurora 64b/66b** protocol).

- **Module name**

  Allows you to assign a name to the generated configuration. This will help to distinguish multiple configurations in the XPE worksheets.

3. When you have filled out the values for this set of transceivers, click **Create**.

   A row in the MGT Sheet will be filled in with the information you entered in the dialog box.

4. For each set of transceivers in your design, fill out the dialog box and click **Create**.

   Each time you click **Create** a row will be added to the MGT sheet.

5. When you have configured all of the transceivers in your design, click **Close** to close the XPE Transceivers Configuration dialog box.

Summary

The ability to estimate power consumption in a design is imperative for efficient part selection, board design, and system reliability. The Xilinx Power Estimator tool with its up to date power models and ease of use features is meant to guide and simplify design utilization entry. Although gathering FPGA and SoC utilization data might seem difficult in the early design development phases, you can derive accurate power estimations with a little thought and using XPE. XPE simplifies device selection and helps parallel development of the Xilinx device logic and the Printed Circuit Board. Finally, XPE helps exploration of alternative implementation and resource configuration when supply power or thermal budgets are exceeded.

Specifying and Managing Clocks

Specifying Clocks

Important factors in dynamic power calculation are the activity and the load capacitance that needs to be switched by each net in the design. Some of the factors in determining the loading capacitance are fanout, wire length, and so forth. With clocks typically having higher activity and fanouts, the power associated with clock nets can be significant and thus is reported in a separate worksheet sheet (see Figure 2-1).

For the information needed to fill out the Clock Tree Power sheet, see the *7 Series FPGAs Clocking Resources User Guide* (UG472) [Ref 24] or the *UltraScale Architecture Clocking Resources Advance Specification User Guide* (UG572) [Ref 25].

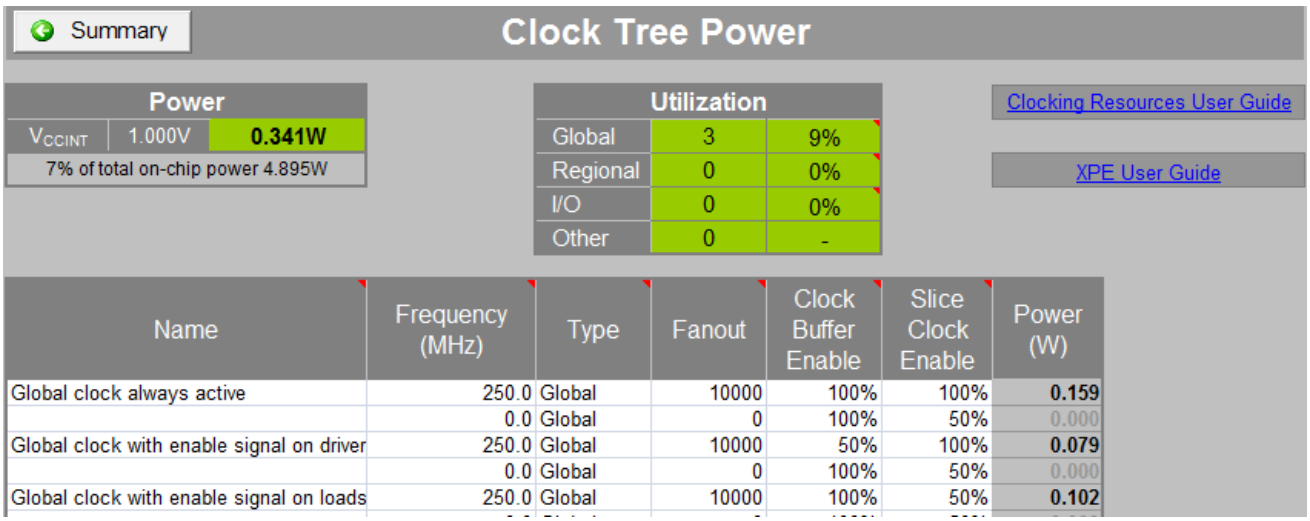


Figure 2-1: **Clock Tree Power Sheet (7 Series Devices)**

- Buffer **Type** Column

  Xilinx devices have different types of buffers capable of driving the clock routing structures and these types are modeled within XPE.

- Clock **Fanout** Column (7-series & UltraScale XPE Spreadsheets)

The number of synchronous elements driven by this clock. The best way to fill in this filed is to sum up the total number of registers, BRAMs and DSPs specified in their respective sheets for a clock.

- **Fanout/Site** Column

  Fanout/site column in UltraScale+ represents the average number of connections of the clock to a physical logic in a site such as a CLB, block RAM, or DSP block.

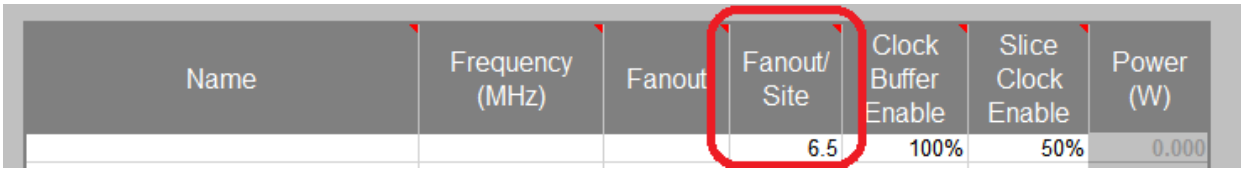


Figure 2-2: **Fanout/Site for UltraScale+ devices**

For early power estimation, it is recommended to leave the value as is. For imported .xpe files, the value is provided by Vivado, and is based on the placed and routed results to improve clock power accuracy. The value ranges from 1 (least efficient, highest power) to 16 (most efficient, lowest power). For BRAM and DSP sites, the value should be 1, as there is only one fanout in these sites.

- **Clock Buffer Enable** Column

  Gates the clock net at its source. The value is the percentage of the time in which the clock buffer is active. Reduce this percentage if you plan on disabling the clock net at the source when this portion of the design is not used. This reduces power.

- **Slice Clock Enable** Column

  Gates the clock net at its loads at CLB level. Reduce this percentage if you plan on disabling some of the clock loads with slice level Clock Enable signals. This reduces power.

  Note: Some algorithms in software such as "Intelligent Clock Gating" will remap or change the packing to minimize this number.

Using the Clock Management Resource Sheets (DCM, PMCD, PLL, MMCM, Clock Manager)

Xilinx device families have different clock generation and management capabilities. To enter information in these sheets, first review the *7 Series FPGAs Clocking Resources User Guide* (UG472) [Ref 24] or the *UltraScale Architecture Clocking Resources Advance Specification User Guide* (UG572) [Ref 25] to understand how to parameterize these resources in XPE. Depending on the step in the project development cycle you might or

might not already know all the clocking details for your design. Enter what is known or can be estimated first, then later you can always reopen and complete the spreadsheet as design details become available.

The clock management resource sheets are presented in a different way in the XPE spreadsheets that support the various FPGA and SoC architectures.

- In the 7 Series/Zynq 7000 SoC and above devices XPE spreadsheet, information for the two clock managers, MMCM and PLL, is supplied on a single sheet, the Clock Manager Power sheet. An **MMCM or PLL** column in the Clock Manager Power sheet lets you specify whether you are supplying information for the MMCM or the PLL.
- In spreadsheets for earlier devices (for example, the Virtex-5/Virtex-6 spreadsheet or the Spartan-3A/Spartan-6 spreadsheet), there will be a different sheet for each clock manager used. For example, separate DCM Power and PLL Power sheets may be displayed in these earlier spreadsheets.

Figure 2-3 shows a sample clock management resource sheet (the Clock Manager Power sheet).

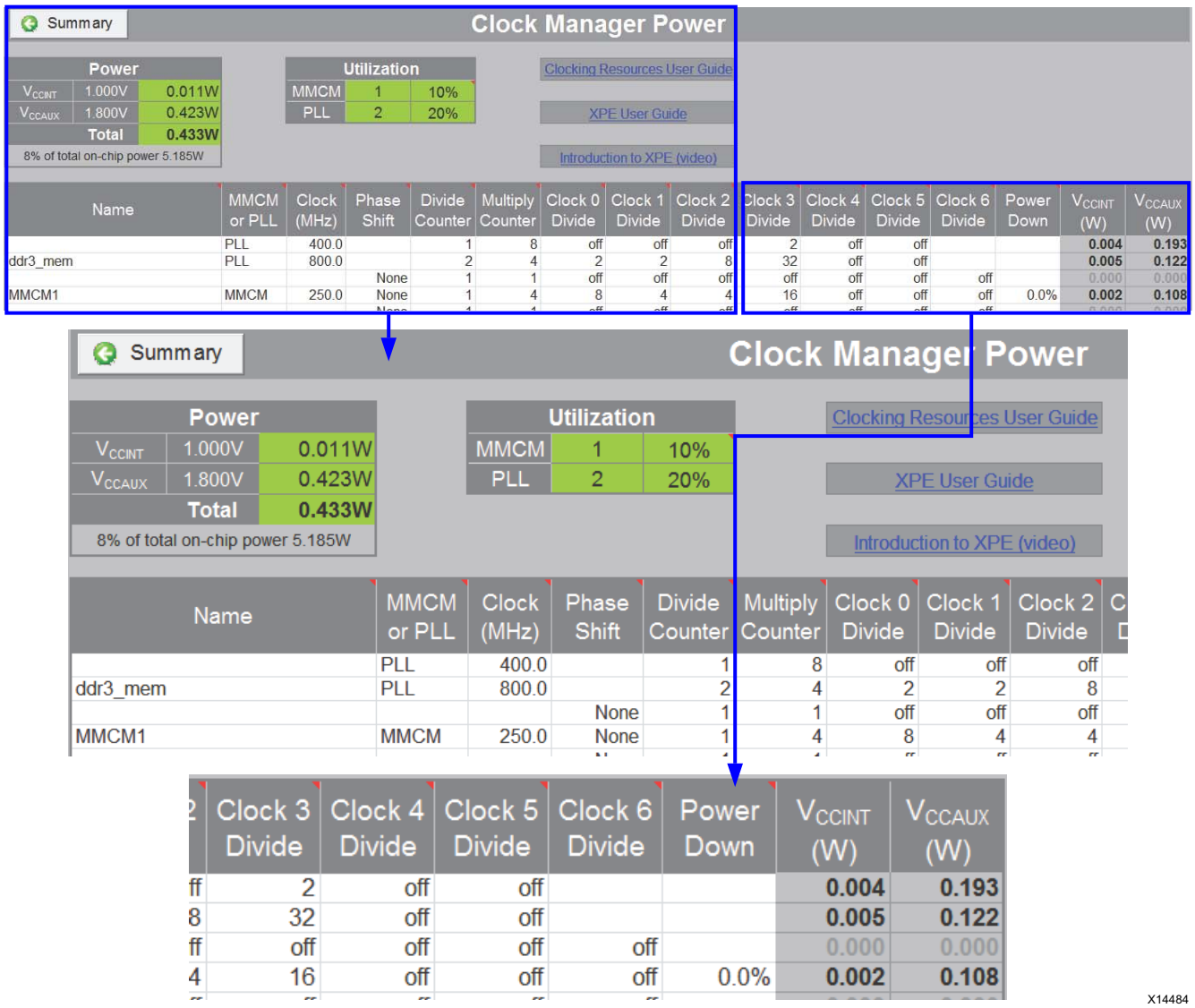


Figure 2-3: **Clock Manager Power Sheet (7 Series Devices)**

Using Xilinx Power Estimator Sheets

Overview

The following sections provide details for entering data into or interpreting results in the different available resource sheets. XPE only shows sheets available on the particular Xilinx device family and device selected. These resource sheets are organized with a center table where you enter utilization, configuration, and activity of the device resources you use. Above this main table are tables representing the total utilization and a summary of the resource's contribution to the total power per voltage supply.

These sheets represent usage based power; therefore, they include all power related to the utilization and configuration of the specified resource. The sheets do not include the leakage power contribution, because this is accounted for on the Summary sheet.

RECOMMENDED: *On sheets in which you specify a clock frequency for resources, XPE will assume that all resources on a single row in the sheet (for example, 4000 Shift Registers and 3000 FFs in a single row on the Logic sheet) are in the same clock domain. For an accurate power estimation, make sure to enter resources in different clock domains on separate rows in the spreadsheet.*

Using the Logic Sheet

The Logic sheet (see Figure 3-1) is used to estimate the power consumed in the CLB resources. The estimated power accounts for both the logic components and the routing. Two types of information should be entered:

- **Utilization** – Enter the number of LUTs configured as Logic, Shift Registers and LUT-based RAMs and ROMs. If your design or a previous generation has been implemented within ISE or the Vivado Design Suite use the **Import** button in the Summary sheet to automatically import this information. Otherwise, use your experience to estimate utilization required to implement the desired functionality.
- **Activity** – Enter the **Clock** domain this logic belongs to. Then enter the **Toggle Rate** the logic is expected to switch and the **Average Fanout**.

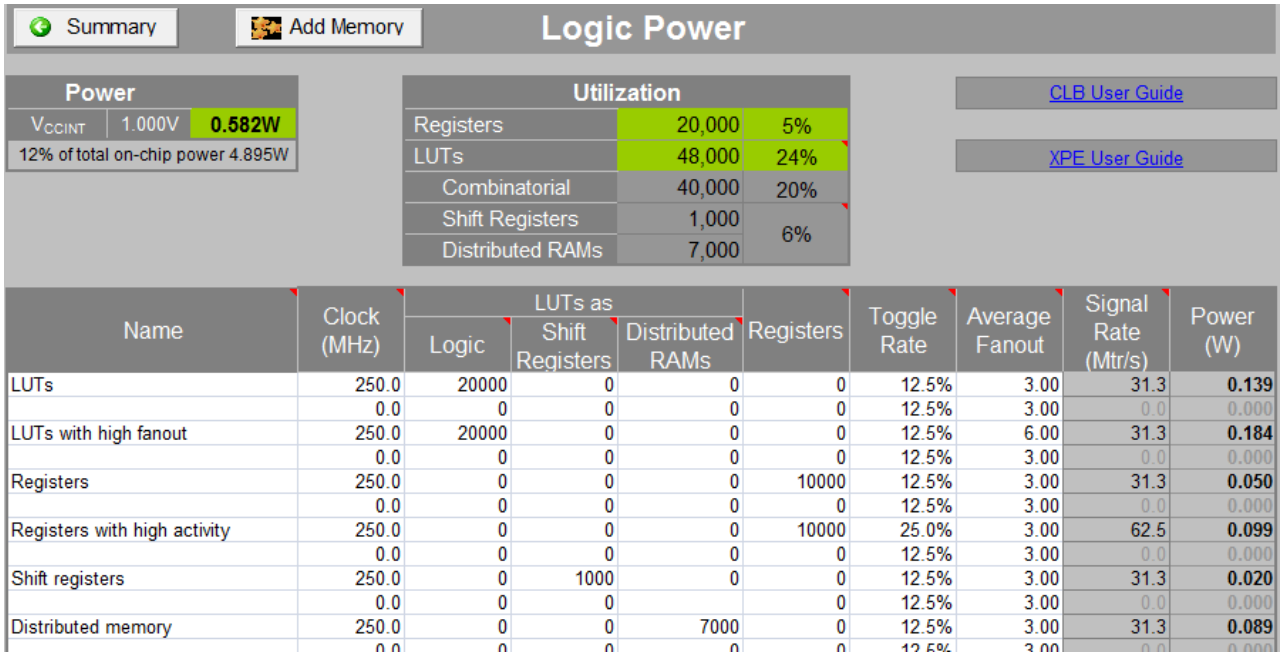
TIP: *The default setting for* ***Toggle Rate*** *(12.5%) and* ***Average Fanout*** *(3) or* ***Routing Complexity*** *(8\|10) are based on an average extracted from a suite of customer designs. In the absence of a better estimate for your specific design, Xilinx recommends using the default setting.*

Note: The **Signal Rate** column defines the number of millions of transitions per second for the considered element. This is a read-only column.

Note: The specified toggle rate represents the average switching activities on both input and outputs of total logic specified in a row. When you group multiple blocks in a single row, the average toggle for the entire group is still lower even if any specific block in a design can toggle much higher.

Signal Rate is computed in the following way:

Signal Rate (Mtr/s) = Clock Frequency (Mhz) * Toggle rate (%)



Name	Clock (MHz)	LUTs as Logic	LUTs as Shift Registers	LUTs as Distributed RAMs	Registers	Toggle Rate	Average Fanout	Signal Rate (Mtr/s)	Power (W)
LUTs	250.0	20000	0	0	0	12.5%	3.00	31.3	**0.139**
	0.0	0	0	0	0	12.5%	3.00	0.0	0.000
LUTs with high fanout	250.0	20000	0	0	0	12.5%	6.00	31.3	**0.184**
	0.0	0	0	0	0	12.5%	3.00	0.0	0.000
Registers	250.0	0	0	0	10000	12.5%	3.00	31.3	**0.050**
	0.0	0	0	0	0	12.5%	3.00	0.0	0.000
Registers with high activity	250.0	0	0	0	10000	25.0%	3.00	62.5	**0.099**
	0.0	0	0	0	0	12.5%	3.00	0.0	0.000
Shift registers	250.0	0	1000	0	0	12.5%	3.00	31.3	**0.020**
	0.0	0	0		0	12.5%	3.00	0.0	0.000
Distributed memory	250.0	0	0	7000	0	12.5%	3.00	31.3	**0.089**

Figure 3-1: **Effect of LUT Configuration, Toggle Rates, and Average Fanout on Power Estimation (7 Series Devices)**

To enter information on the Logic sheet related to distributed memory, you can use the XPE Memory Generator wizard, which appears when you click the **Add Memory** button on the Logic sheet. The XPE Memory Generator wizard provides a simplified method of adding memory-related rows to the Logic sheet.

Memory Generator Wizard and the Logic Sheet (Distributed Memory)

For the 7 series/Zynq-7000 SoC and above devices XPE spreadsheet, you can enter distributed memory information in the Logic sheet by using the Memory Generator wizard. The Memory Generator wizard provides a simplified method of populating the Logic sheet

with rows related to distributed memory, displayed as **Distributed RAMs** on the Logic sheet.

The Memory Generator wizard can be started from the Logic sheet by clicking the **Add Memory** button.

For a description of the Memory Generator wizard and how you can run the wizard from the Logic sheet, see Using the Memory Generator Wizard (for Distributed Memory).

To understand the capabilities of the 7 series/Zynq-7000 SoC distributed memory and the settings you will enter within XPE refer to the *7 Series FPGAs Configurable Logic Block User Guide* (UG474) [Ref 5] and UltraScale Architecture-Based FPGAs Memory Interface Solutions [Ref 37].

Routing Complexity for UltraScale and UltraScale+ Devices

Routing Complexity is the average number of routing resources per logic cell which includes fanout, interconnect capacitance, wire length etc. The medium complexity value is 8. The high complexity value is 10 and very high complexity value is 12. Use higher values only when experiencing a higher routing congestion especially on a high performance, high utilization designs. Xilinx recommends leaving this filed at default value for early estimation, until the design complexity is known.

During the design implementation, Vivado Report Power estimates the routing complexity accurately, which can be imported into XPE to better realize the correct values for a given design. Note that the imported values for any specific logic could be higher than the default i.e. 12, which is valid since the imported results are more elaborated to specific logic compared to manual entry based on grouping multiple blocks in a single row.

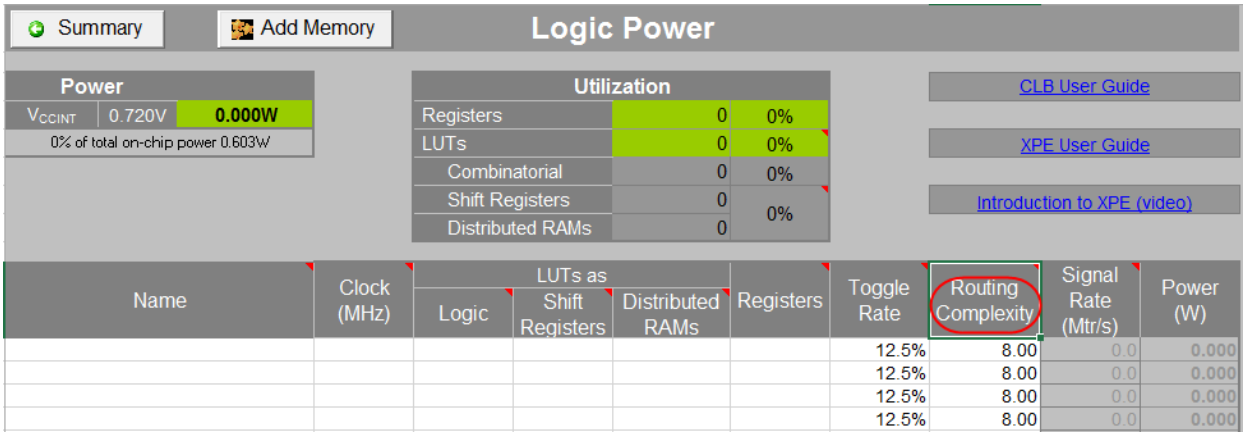


Figure 3-2: **Routing Complexity in Logic Power Sheet for UltraScale devices**

Using the IP Manager Sheet (7 Series, Zynq-7000 SoC, UltraScale and UltraScale+ Devices)

The IP Manager sheet lets you create, view, and delete IP modules created using the IP Module wizards. You can also export IP modules (to be imported into other XPE spreadsheets), and import modules from other XPE spreadsheets into your currently active spreadsheet.

Each IP module displayed in the IP Manager sheet represents the device resources used to implement one of the following in a Xilinx device:

- Block memory - Created using the Using the Memory Generator Wizard (for Block Memory).
- Distributed memory - Created using the Using the Memory Generator Wizard (for Distributed Memory).
- Memory interface - Created using the Using the Memory Interface Configuration Wizard.
- Transceiver interface - Created using the Using the Transceiver Configuration Wizard.

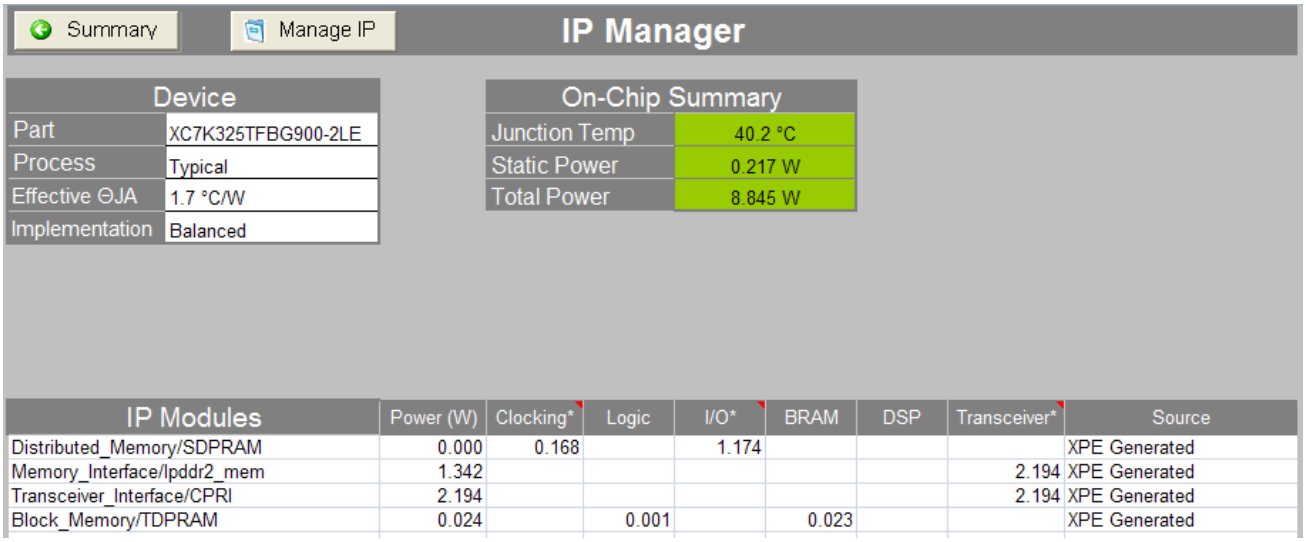


Figure 3-3: **IP Manager Sheet (7 Series Devices)**

The **IP Modules** table indicates the power associated with each IP module created, as well as the power associated with the resource sheets populated by the IP module. As shown in the **IP Modules** table in Figure 3-3, IP modules might populate more than one resource sheet. For example, a block memory IP module might place rows in both the Block RAM sheet (**BRAM** column) and the Clock sheet (**Clocking** column).

In some cases, more than one resource will be included in a single column in the **IP Modules** table. For example, the **Clocking** column might include the power associated with clock nets as well as the power associated with clock managers such as the PLL and the MMCM, and the **Transceiver** column might include the power associated with Multi-Gigabit Transceivers (MGTs) as well as the power associated with a PCIe block.

Creating an IP Module From the IP Manager Sheet

For a description of how to create IP modules from the IP Manager sheet, see the following:

- Block Memory - See Using the Memory Generator Wizard (for Block Memory).
- Distributed Memory - See Using the Memory Generator Wizard (for Distributed Memory).
- Memory Interface - See Using the Memory Interface Configuration Wizard.
- Transceiver Interface (7 Series only) - See Using the Transceiver Configuration Wizard.

Deleting an IP Module From the IP Manager Sheet

To delete an IP module from the IP Manager sheet:

1. Click the **Manage IP** button at the top of the IP Manager sheet.

   The XPE IP Manager dialog box opens.

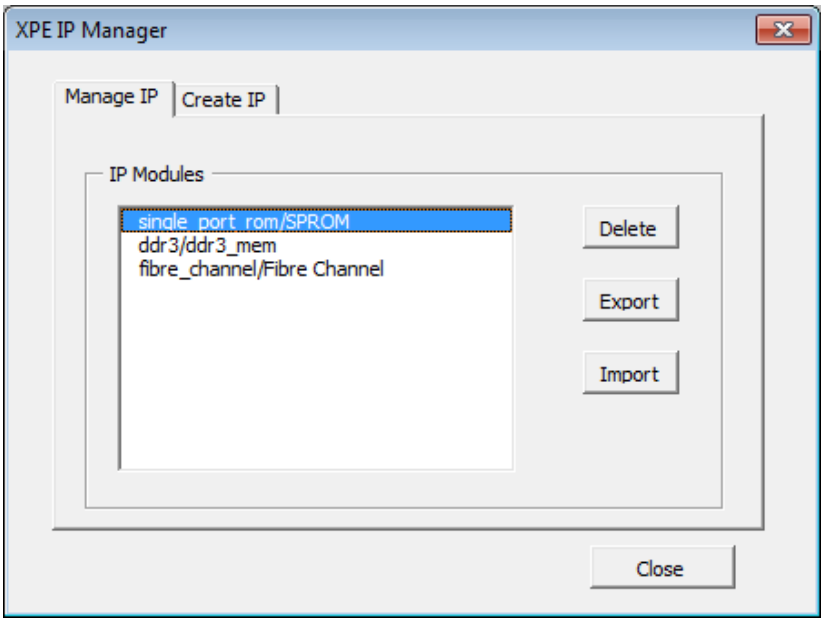


Figure 3-4: **XPE IP Manager Dialog Box**

   In the Manage IP tab of the XPE IP Manager dialog box, select the IP module you want to delete and click **Delete**.

All of the rows in the appropriate resource sheets are deleted, and the IP Module is removed from the **IP module** table on the IP Manager sheet.

Exporting an IP Module From the IP Manager Sheet

An IP module can be exported from the currently active XPE spreadsheet, for importing into another XPE spreadsheet. IP modules are exported as .xpe (XPE Exchange) files.

To export an IP module from the IP Manager sheet:

1. Click the **Manage IP** button at the top of the IP Manager sheet.

   The XPE IP Manager dialog box opens (see Figure 3-4).

2. In the Manage IP tab of the XPE IP Manager dialog box, select the IP module you want to export and click **Export**.
3. In the Export Xilinx Power Estimator IP Module dialog box, specify a **File name** for the .xpe file representing the selected IP module. Then click **Save**.

   The selected IP module is exported to an .xpe (XPE Exchange) file. This file can be imported into another XPE spreadsheet.

Importing an IP Module Into the IP Manager Sheet

An IP module can be imported into the IP Manager sheet. IP modules are imported as .xpe (XPE Exchange) files. The imported .xpe file represents an IP module exported from another XPE spreadsheet.

To import an IP module into the IP Manager sheet:

1. Click the **Manage IP** button at the top of the IP Manager sheet.

   The XPE IP Manager dialog box opens (see Figure 3-4).

2. In the Manage IP tab of the XPE IP Manager dialog box, click **Import**.
3. In the Import Xilinx Power Estimator IP Module dialog box, specify the **File name** of the .xpe file to be imported. Then click **Save**.

   The selected .xpe file is imported into the IP Manager sheet.

Using an I/O Sheet

With higher switching speeds and capacitive loads, switching I/O power can be a substantial part of the total power consumption of a Xilinx device. Because of this, it is important to accurately define all I/O related parameters. In the I/O sheet XPE helps you calculate the on-chip and, eventually, off-chip power for your I/O interfaces.

XPE provides a Memory Interface Configuration wizard to allow you to quickly enter the important parameters required for an accurate power estimate of the I/Os involved in the device's interface to external memory. For step-by-step instructions about how to use the wizard to fill out the memory interface information in the I/O sheet, see Memory Interface Configuration Wizard and the I/O Sheet.

Figure 3-5 shows the top section of the I/O sheet (for the UltraScale spreadsheet).

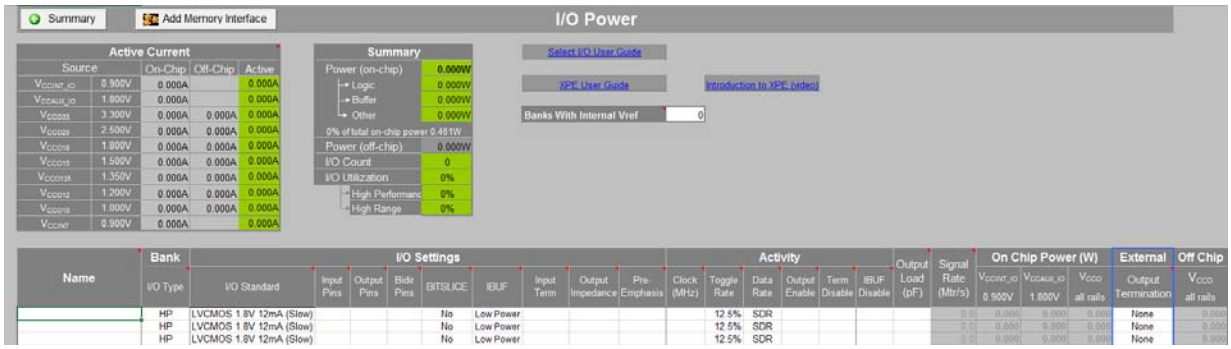


Figure 3-5: **I/O Sheet - Top Section (UltraScale Devices)**

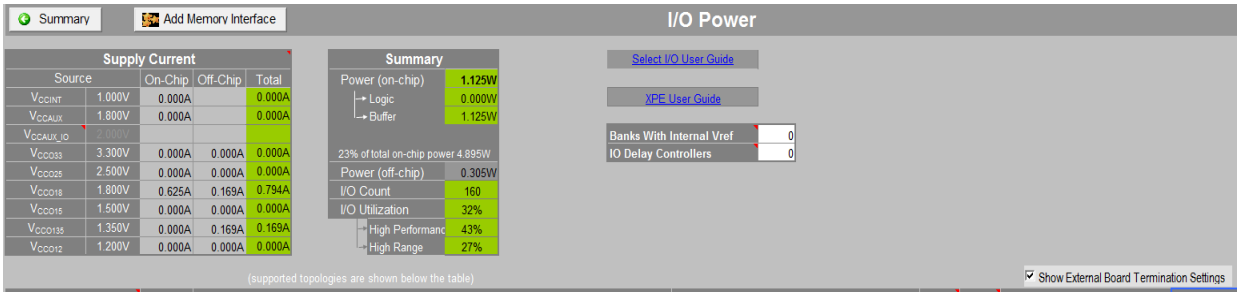


Figure 3-6: **I/O Sheet - Top Section (7 Series Devices)**

Figure 3-5 and Figure 3-6 illustrates the three main types of information entered on the I/O sheet: **IO Settings**, **Activity**, and, if needed, **External Termination**.

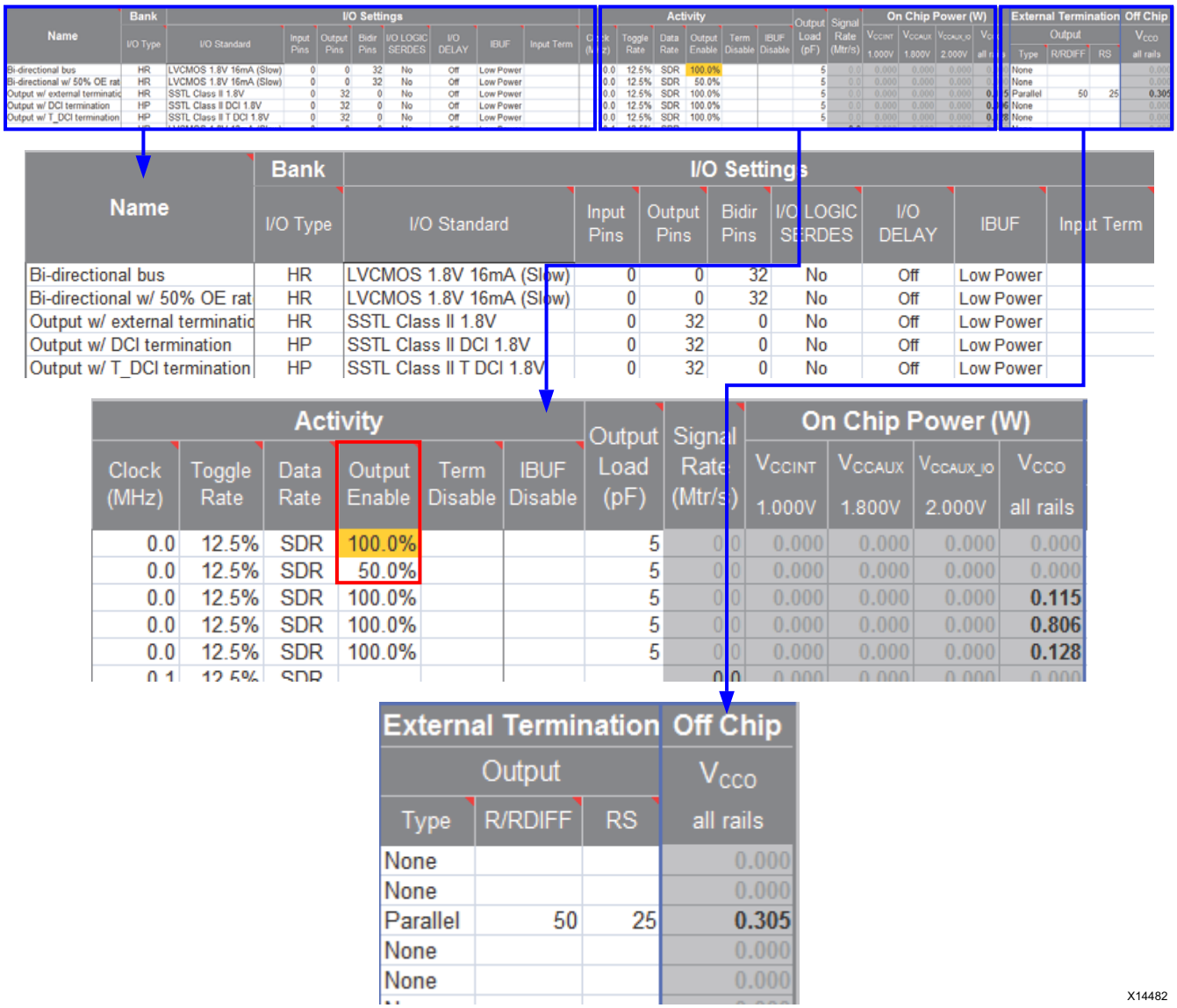


Figure 3-7: **I/O Sheet - Effect of Output Enable Rate on Power Estimates for Inputs, Outputs, and Bidirectional I/Os (7 Series Devices)**

The following paragraphs provide more information on how to fill in each of these columns.

- **I/O Settings**
  - **I/O Standard**

    Specify here the expected I/O standard you will use for this interface. Configurations which use the on-chip terminations are shown with a **DCI** suffix in this drop-down menu. Differential I/O standards have a **(pair)** suffix. For calculations, XPE assumes the standard V_{CCO} level (for example, 3.3V) that is closest to the nominal listed in the data sheet for that I/O standard.

Note: For Spartan-6 devices, the open drain standards I2C and SMBUS can use a V_{CCO} from 2.7V to 3.45V, with a 3.0V nominal voltage. In XPE these are calculated using a V_{CCO} of 3.3V.

RECOMMENDED: *To minimize power on output signals, always use the weakest driver settings that meet your performance goals (lower the drive strength and slew rate).*

TIP: *In 7 series devices, using on-chip terminated standards is a good way to improve the signal integrity of the waveforms seen by the receiver. Because the terminations are embedded inside the Xilinx device, the termination power contributes to raising the device junction temperature. To minimize this power, try using the tri-statable on-chip terminated standards (denoted **T DCI**) whenever possible.*

- **I/O Direction Columns**

  Enter the number of **Input**, **Output** and **Bidir** (bidirectional) signals for each I/O interface.

RECOMMENDED: *Because toggling activity of inputs and outputs is often very different, Xilinx recommends you place each direction on a separate row.*

TIP: *Enter one pin for each differential I/O pair. For example, if your memory has four differential DQS pairs, enter **4** on the **Input Pins** column.*

- **I/O Performance Settings**

  These performance settings, such as **I/O LOGIC SERDES** or **BITSLICE**, are family dependent. Enter the configuration in which you expect to program these I/Os.

IMPORTANT: *Typically performance settings increase power consumption. Try enabling these settings only if your I/O interface absolutely requires them.*

- **On-Chip Termination**

  For **Input Term**, select the appropriate input termination for the selected I/O standard. Select **DIFF_TERM** when using the on-chip differential termination, or select **UNTUNED_SPLIT_40, 50, or 60** Ω impedance when using the optional on-chip termination in HP banks.

  In the UltraScale device XPE spreadsheet, you can specify both input and output terminations. For **Input Term**, select **DIFF_TERM** when using the on-chip differential termination, or select **DCI** or uncalibrated termination impedance when using on-chip input termination. For **Output Term**, select **RDRV_40_40**, **RDRV_48_48** or **RDRV_60_60** when using on-chip output termination in HP banks. Set **Pre-Emphasis** to **Yes** when using the transmitter pre-emphasis feature.

- **Activity**

  Enter in the expected activity for each I/O interface in the following columns.

  - **Clock (MHz)**

    Synchronous signals: Enter the frequency of the clock capturing or generating these signals.

    Asynchronous signals: Calculate the equivalent frequency of the signal. For example, if you can determine the signal will toggle (change state) 2 million times per second then enter 1 in this column (when converting signal rate to frequency you need 2 transitions to make a period: the transition from 0 to 1 and the transition from 1 to 0).

  - **Toggle Rate**

    Synchronous elements: Enter how often compared to the clock this signal is expected to change state. For example, if the data changes every 8 clock cycles on average, enter 12.5% (1/8, converted to a percentage).

    Asynchronous elements: As explained in the **Clock (MHz)** description above, enter the equivalent frequency in the **Clock (MHz)** column and then enter 100% in this column.

  - **Data Rate**

    Synchronous elements: Enter **DDR** if the signal is sampled on both the positive and negative edges of the clock. Enter **SDR** if the signal is sampled on only one edge of the clock.

    Note: When the Data Rate is DDR, the specified toggle rate is doubled internally for power estimation. You must not calculate the toggle rate explicitly for double data rate..

    Asynchronous elements and Clocks: Enter **Async** or **Clock**.

  - **Output Enable**

    Input only signals: This column has no effect.

    Output and bidirectional signals: Specify for a long period of time how much of this time the output buffer is driving a value (compared to the time the driving buffer is disabled or tri-stated.

TIP: *As shown in Figure 3-7 (red frame) for line 1 and 2, setting **Output Enable** to **100%** is a common mistake which degrades the tool accuracy.*

- **Term Disable**

  Set DCI or IOB33 OCT to disabled (**DCITERMDISABLE**) when not in use in the fabric. Enter the percentage of time the DCI or ICT termination is disabled.

- **IBUF Disable**

  Set HSTL/SSTL IBUF to low power idle (**IBUFDISABLE**) when not in use in the fabric. Enter the percentage of time the IBUF is disabled.

- **Output Load**

  Enter the power factor for the board and other external capacitance driven by the outputs in the module.

- **External Termination**

  When not using the available on-chip termination you can use XPE to calculate the power supplied by the Xilinx device to off-chip components such as external board termination resistor networks.

  Multiple termination types are supported for I/Os configured as outputs. External input terminations are not supported, because calculations often require details of the driver side but these details are not available to XPE.

  Note: Select the **Show External Board Termination Settings** check box to display these columns and a graphic below the table. The graphic shows the supported **Output Termination Topologies**, so you can easily understand which column to fill depending on the topology you want to build.

  - **Term. Type**

    Select the appropriate topology from this drop-down menu.

  - **R/RDIFF** and **RS**

    Some termination schemes require two resistor values while others require only a single value. Refer to the termination graphic then enter the resistor value on the appropriate column. Figure 3-8 shows the supported I/O termination topologies in this release.

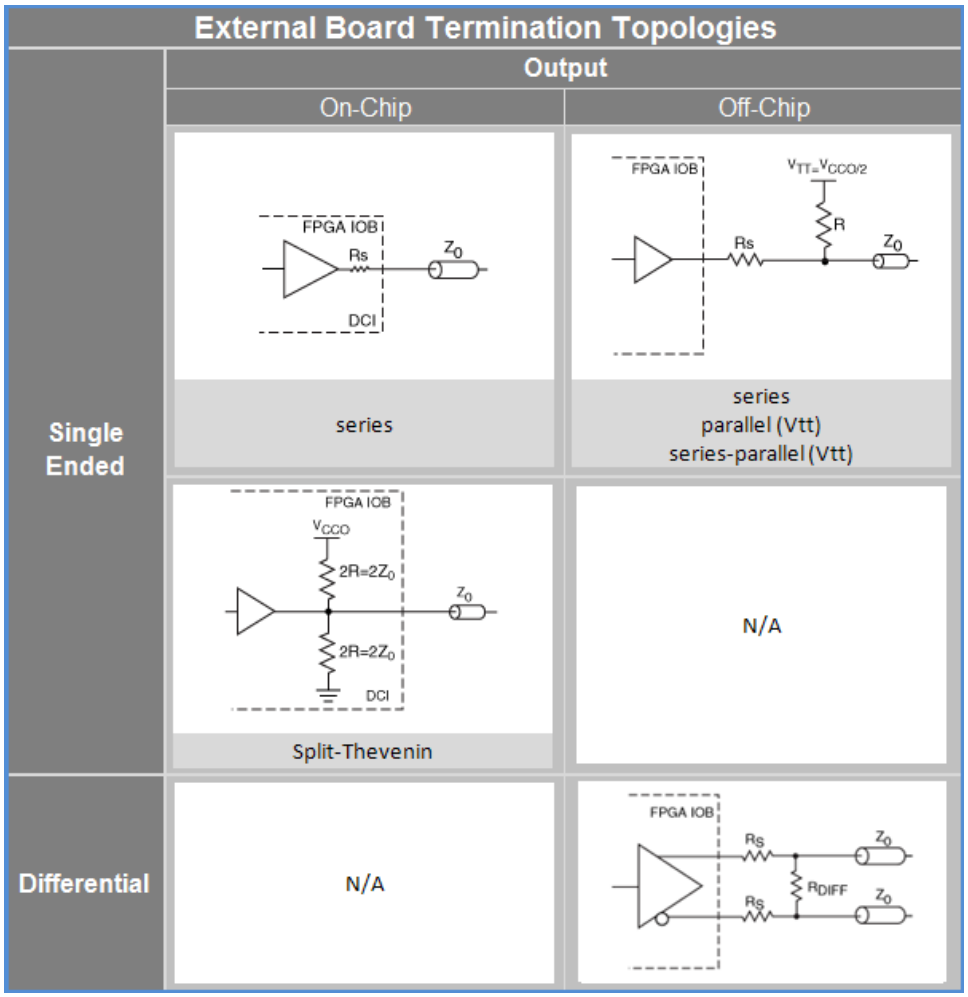


Figure 3-8: **External I/O Termination Topologies (Virtex-6 and 7 Series Devices)**

Memory Interface Configuration Wizard and the I/O Sheet

For XPE spreadsheets, you can enter information for the I/Os involved in the interface between the Xilinx device and external memory by using the Memory Interface Configuration wizard. The Memory Interface Configuration wizard provides a simplified method of filling in the memory interface I/Os in the XPE spreadsheet.

When you configure a memory interface using the wizard, rows will be added to the I/O sheet for each output line (for example, Data, Address, and Clock) from the Xilinx device that will be applied to the external memory.

The Memory Interface Configuration wizard can be started from the I/O sheet by clicking the **Add Memory Interface** button.

For a description of the Memory Interface Configuration wizard and how you can run the wizard from the I/O sheet, see Using the Memory Interface Configuration Wizard.

To understand the 7 series memory interfaces and the settings you will enter within XPE refer to the *Zynq-7000 SoC and 7 Series Devices Memory Interface Solutions User Guide* (UG586) [Ref 10].

Using the Block RAM (BRAM) Sheet

Xilinx devices have dedicated block RAM resources. To accurately set Block RAM parameters in XPE, a good understanding of device resources and configuration possibilities is recommended. If implementation details for the block RAM are known, follow the guidelines described in Setting BRAM Mode for Improved Accuracy. Otherwise, refer to Preliminary BRAM Estimates.

Note: Distributed RAM/ROM and SRL usage should be specified in the Using an I/O Sheet.

To enter information on the Block RAM sheet, you can use the XPE Memory Generator wizard, which appears when you click the **Add Memory** button on the Block RAM sheet. The XPE Memory Generator wizard provides a simplified method of adding rows to the Block RAM sheet. For information about using this wizard, see Memory Generator Wizard and the Block RAM Sheet (Block Memory).

Following are details about columns in the Block RAM sheet:

- **Enable Rate** column

  Use the **Enable Rate** to specify the percentage of time each of the block RAM ports are enabled for reading and/or writing. To save power, the RAM enable can be driven Low on clock cycles when the block RAM is not used in the design. BRAM **Enable Rate**, together with **Clock** rate, are important parameters that must be considered for power optimization.

- **Write Rate** column

  The **Write Rate** represents the percentage of time that each block RAM port performs write operations. The read rate is understood to be 100% – write rate. This is not relative to Enable Rate. For example, if the Enable Rate is 25% and write operation can be half of the time, then Write Rate is 12.5% of total time period.

- **Signal Rate** column (Read only)

  Defines the number of millions of transitions per second for the considered BRAM output port. This is a read-only column which takes into account port enable rates and a weighted average of the port widths.

Figure 3-9 illustrates the effect of block RAM configuration modes and bit widths on power estimates.

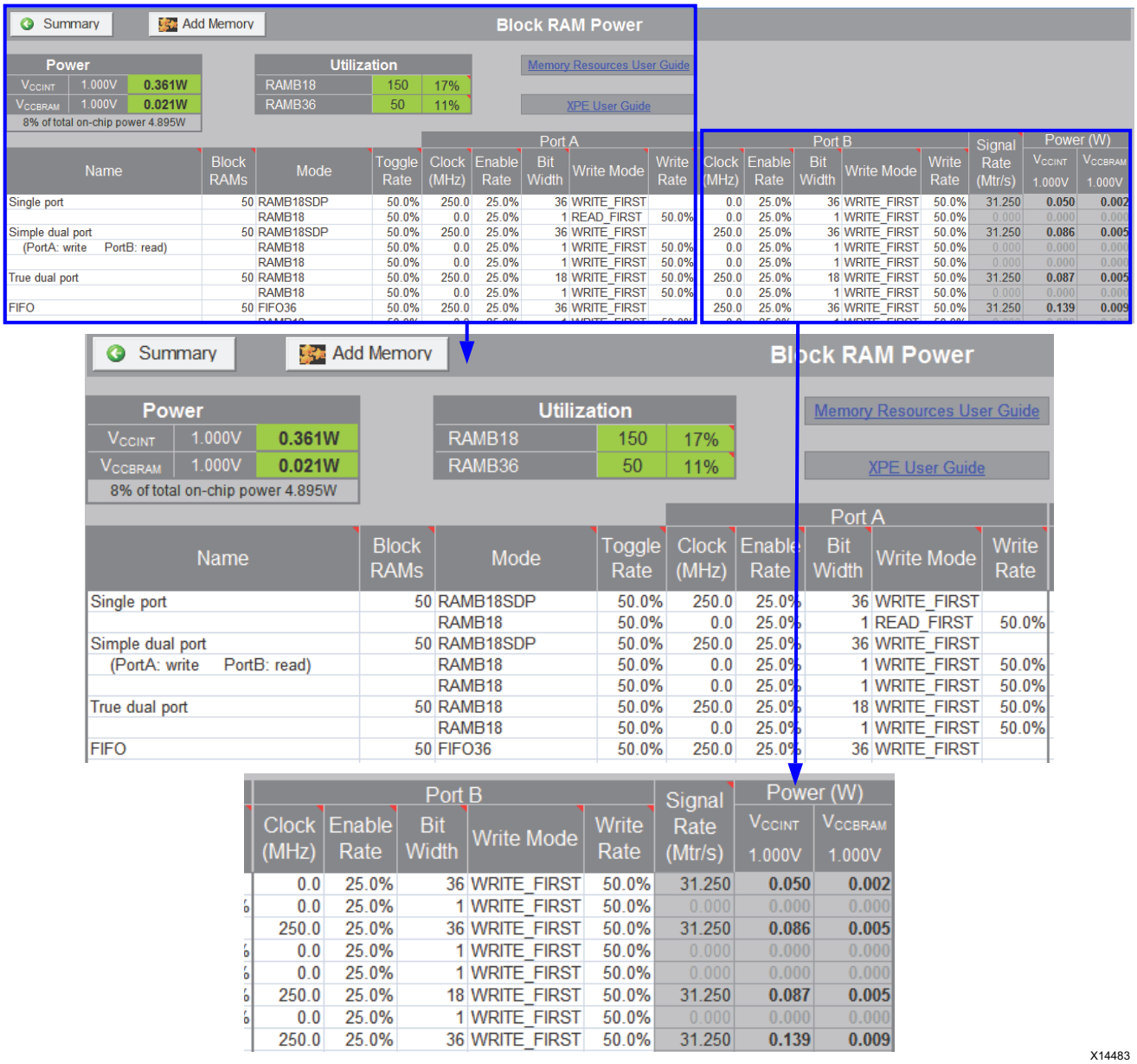


Figure 3-9: **Block RAM Sheet - Effect of Block RAM Configuration Modes and Bit Widths on Power Estimates (7 Series Devices)**

Preliminary BRAM Estimates

If the exact block RAM types and modes to be used in the design are unknown, the best approach is to determine how many kilobytes of memory are needed in the design and use the appropriate number of basic 18k True dual-port RAMs. If the data width of memory

access is known, select this from the drop-down menu for each port. Depth and width are the two most important characteristics of a memory.

Setting BRAM Mode for Improved Accuracy

If the breakdown of the memory usage of your design is known, the XPE spreadsheet allows you to specify which block RAM modes are being used. The **Mode** column has selectable values from a drop-down menu that lists the different primitive names and modes of the block RAM. Depending on the target family, this includes:

- **BRAM** - Simple dual-port or True dual-port Block RAM.
- **FIFO** - Dedicated built-in FIFO.
- **CASC (pair)** (7 series only) - Cascaded block RAM blocks (built from two RAM blocks).
- **ECC** - When the block RAM is configured in ECC mode.

In True dual-port mode the following data write mode options are available:

- **WRITE_FIRST** – The port will first write to the location and then read out the newly written data.
- **READ_FIRST** – The old data is first read out and then the new data is written in. This mode effectively allows 4 operations per clock cycle (saving power or resource utilization), because the old data can be read out and replaced with new data on the same clock cycle of each port. However, note that READ_FIRST is only more power efficient if the data in the memory is used the same time as writing out new data, and does not force separate read and write operations to get the data. If that is not the case, READ_FIRST is generally less efficient in power than NO_CHANGE in TDP mode. If the functionality of READ_FIRST is not needed, the BRAM should be configured as WRITE_FIRST or NO_CHANGE to save power.
- **NO_CHANGE** – When a Write happens the block RAM outputs remain unchanged.

In Simple dual-port mode the following data write mode options are available:

- **WRITE_FIRST** – The port will first write to the location and then read out the newly written data.
- **READ_FIRST** – The old data is first read out and then the new data is written in. This mode effectively allows 2 operations per clock cycle (saving power or resource utilization), because the old data can be read out and replaced with new data on the same clock cycle of each port. However, note that READ_FIRST is only more power efficient if the data in the memory is used the same time as writing out new data, and does not force separate read and write operations to get the data. If that is not the case, READ_FIRST is generally less efficient in power than WRITE_FIRST in SDP mode. If the functionality of READ_FIRST is not needed, the BRAM should be configured as WRITE_FIRST to save power.

- **NO_CHANGE** – Not available in the SDP mode because it is identical in behavior to WRITE_FIRST mode.

Cascade BRAM Support for UltraScale Devices

In the UltraScale and UltraScale+ device XPE spreadsheets, the BRAM sheet has additional column named **Cascade Group Size**. For cascaded BRAMs, this column shows how many BRAMs are cascaded together in each group. This number and the total number of BRAMs in the modules are used to determine the number of active BRAMs at a given time.

For example, if the module has a total of 20 BRAMs and the Cascade Group Size is 4, the tool calculates 20/4=5 groups each with 4 BRAMs cascaded together. If none of the BRAMs are cascaded, leave this column empty or set to **0**.

UltraRAM support for UltraScale+ Devices

The UltraRAM (URAM) is a high density FPGA 288Kb memory building block. URAMs coexist with BRAMs in Ultrascale+ devices. The 288 Kb blocks are cascadable to enable deeper memory implementation. The URAMs may exist with very little or no fabric resources and with no timing penalty, if pipelined appropriately.

Both of the ports share the same clock and can address all of the 4K x 72 bits. Each port can independently read from or write to the memory array. URAM supports two types of write enable schemes. The first mode is consistent with the block RAM byte write enable mode. The second mode allows gating the data and parity byte writes separately. Multiple URAM blocks can be cascaded together to create larger memory arrays. Dedicated routing in the URAM column enables the entire column height to be connected together. This makes URAM an ideal solution for replacing external memories such as SRAM.

The URAM Sheet provides columns to enter the number and configurations of the URAMs intended to be used for the design. It also displays the resulting power for the VCCINT and VCCBRAM power rails. The Utilization shows the number of URAM blocks (URAM288) used in the target device.

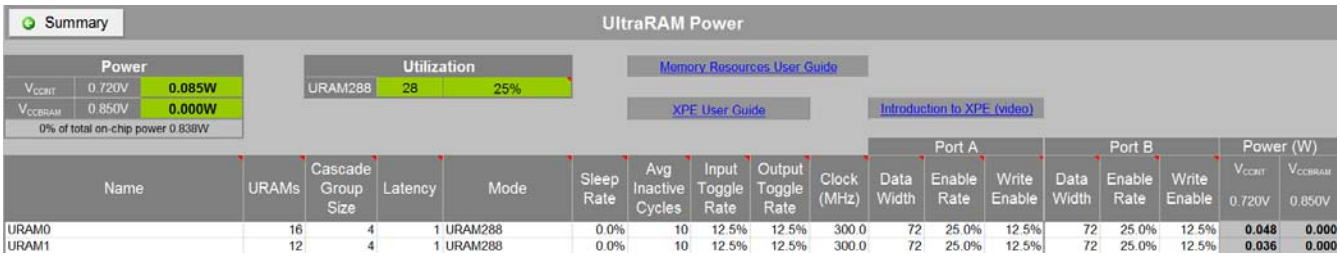


Figure 3-10: **UltraRAM Power sheet for UltraScale+ Devices**

Following are descriptions of the columns used for design entry:

Name: Enter a name to identify the URAM or URAM module.

URAMs: The Number of UltraRams in this module.

Cascade Group Size: URAM blocks support cascading to create larger memory arrays while reducing the overall power by enabling only one URAM of a cascade at a time.

Example: 20 URAM blocks with a Cascade Group Size of 4 represents 20/4 = 5 sets of cascaded URAMs of 4 blocks each. If there is no cascading, use 1 as the value of cascade group size.

Latency: The optional URAM pipeline registers are IREG_PRE (input) or REG_CAS (cascade). The default value is Cascade Group Size divided by 3. If there is no URAM cascading, only IREG_PRE can be used which corresponds to a Latency of 1.

Mode: Chooses between URAM288 (no ECC) and URAM288_with_ECC.

Sleep Rate: The percentage of time the URAM SLEEP input pin is asserted. The value of Auto is also supported for Automatic Sleep Mode.

Average Inactive Cycles: The average number of consecutive inactive cycles when in Sleep Mode. The minimum value is > 10 or the Cascade Group Size minus 2.

Input Toggle Rate: The average toggle rate of the data inputs (DIN) for both ports A and B.

Output Toggle Rate: The average toggle rate of the data outputs (DOUT) for both ports A and B.

Clock (MHz): Clock frequency of the URAM or URAM module.

Following are values specified for both URAM ports A and B:

Data Width: Specify the exact data width if less than the maximum 72 bits.

Enable Rate: The percentage of time the URAM is enabled.

Write Enable: The percentage of time the write enable input is asserted, independently of the Enable Rate. The write enable pins are the URAM RDB_WR_A and RDB_WR_B pins.

Note: The following rule is applicable when you specify **Enable Rate** and **Sleep Rate**. For each of the ports A and B, the sum of (Enable Rate / Cascade Group Size) and Sleep Rate must not exceed 100%.

Memory Generator Wizard and the Block RAM Sheet (Block Memory)

For the 7 Series/Zynq-7000 SoC and above device XPE spreadsheets, you can enter block memory information in the Block RAM sheet by using the Memory Generator wizard. The Memory Generator wizard provides a simplified method of populating the Block RAM sheet with rows related to block memory, displayed as **BRAMs** on the Block RAM sheet.

The Memory Generator wizard can be started from the Block RAM sheet by clicking the **Add Memory** button.

For a description of the Memory Generator wizard and how you can run the wizard from the Block RAM sheet, see Using the Memory Generator Wizard (for Block Memory).

IMPORTANT: *The memory type that is selected in the Vivado IDE might not propagate to the Block RAMs implemented. For 7 series devices that make use of the port aspect ratios, RAM_MODE is set to Single Dual-port RAM (SDP). For more details about these configurations, see the 7 Series FPGAs memory Resources User Guide (UG473)[Ref 6]. All other configurations for the RAM_MODE are not guaranteed to be SDP.*

Note: Asymmetric port width is not supported for 7 series XPE while using VBRAM in SDP mode.

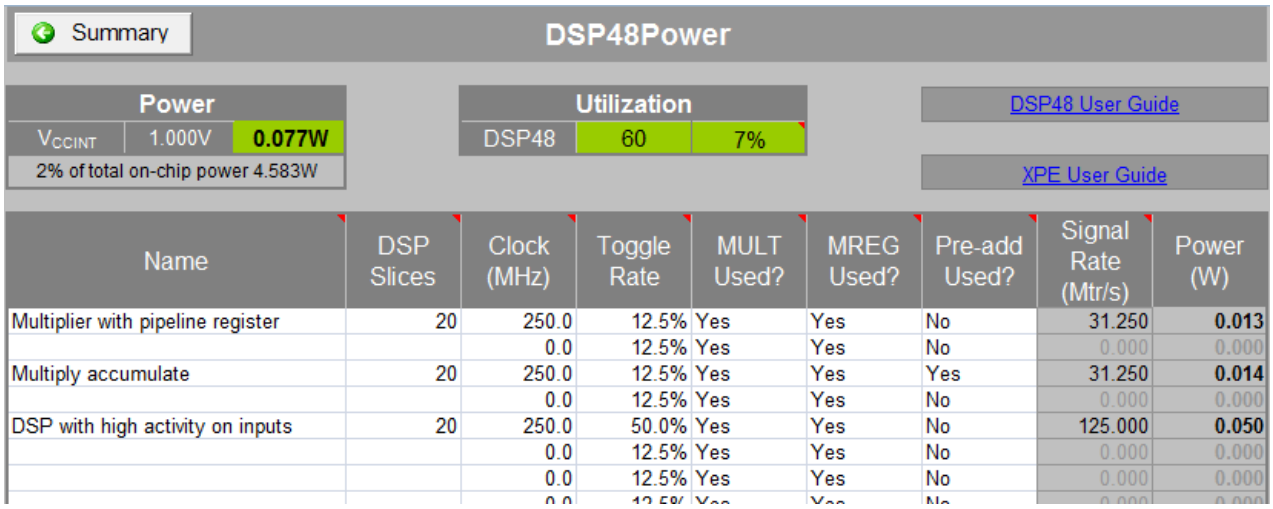
Using the DSP Sheet (MULT, DSP48)

Xilinx device families have different Digital Signal Processing (DSP) blocks with different capabilities. To enter information in these sheets first review the *7 Series DSP48E1 Slice User Guide* (UG479) [Ref 27] or the *UltraScale Architecture DSP Slice Advance Specification User Guide* (UG579) [Ref 28] to understand the parameters in the DSP sheet.

TIP: *The default DSP configuration is assumed to be 27x18 in XPE. The toggle rate must be scaled accordingly for accurate power estimation. For example, if 18x18 DSP is expected to toggle 25%, then scale it by 0.86, which is 21.5% and enter into XPE. Similarly, scale the actual toggle rate by 0.8 for 12x12 configuration.*

TIP: *DSP slices have clock enable (CE) ports. When entering data in the* ***Toggle Rate*** *column remember to multiply your data input toggle rate with the DSP slice clock enable rate. For example, if random data (typically ~38% data toggle rate) is input into the DSP slice and the slice is clock enabled only 50% of the time, then the output data toggle rate should be scaled by the CE rate such that the data toggle rate becomes 19% (38% x 50%). see Figure 3-11 for a Virtex-7 example.*

TIP: *For families that have a register within the multiplier (MREG), using this pipeline register helps lower dynamic power.*



Summary

DSP48Power

Power		
V_{CCINT}	1.000V	0.077W
2% of total on-chip power 4.583W		

Utilization		
DSP48	60	7%

DSP48 User Guide

XPE User Guide

Name	DSP Slices	Clock (MHz)	Toggle Rate	MULT Used?	MREG Used?	Pre-add Used?	Signal Rate (Mtr/s)	Power (W)
Multiplier with pipeline register	20	250.0	12.5%	Yes	Yes	No	31.250	0.013
		0.0	12.5%	Yes	Yes	No	0.000	0.000
Multiply accumulate	20	250.0	12.5%	Yes	Yes	Yes	31.250	0.014
		0.0	12.5%	Yes	Yes	No	0.000	0.000
DSP with high activity on inputs	20	250.0	50.0%	Yes	Yes	No	125.000	0.050
		0.0	12.5%	Yes	Yes	No	0.000	0.000
		0.0	12.5%	Yes	Yes	No	0.000	0.000

Figure 3-11: **DSP48E1 Power Sheet (7 Series Devices) - Effect of Clock, Toggle Rate, and MREG on Power Estimates**

Using the Transceiver Sheets (GTP, GTX, GTH, GTY, GTZ)

Different Xilinx device families have Multi-Gigabit Transceivers (MGT), which are very high performance serial I/Os. Transceivers typically use separate voltage supplies for the PCS, PMA and termination.

To understand the capabilities of the 7 series GTs and the settings within XPE refer to the *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 8] and the *7 Series FPGAs GTP Transceivers User Guide* (UG482) [Ref 9].

To understand the capabilities of the UltraScale device GTs and the settings within XPE refer to the *UltraScale Architecture GTH Transceivers Advance Specification User Guide* (UG576) [Ref 14] and the *UltraScale Architecture GTY Transceivers Advance Specification User Guide* (UG578) [Ref 15].

IMPORTANT: *In the 7 series/Zynq 7000 SoC XPE spreadsheets, PCI Express (PCIe) information is specified on a GTX, GTP, GTH, or GTZ sheet. In the UltraScale and UltraScale+ XPE spreadsheets, all of the Hard IP Blocks (PCIe - GEN1, GEN2 & GEN3) and 100G Ethernet (CMAC & ILKN) are specified on the GTH or GTY sheets. Spreadsheets for earlier Xilinx devices have a separate PCIe sheet.*

To simplify data entry, drop-down menus are provided with parameter preferred or required values. Figure 3-12 shows an example Kintex-7 XC7K325T design. The tables in the sheet header report design power and currents. Device leakage for each supply is reported on the Summary sheet.

For 7 series devices, XPE provides a Transceiver Interface Configuration wizard to allow you to quickly enter the important parameters required for an accurate transceiver power estimate. For step-by-step instructions about how to use the wizard to fill out the MGT sheets, see Using the Transceiver Configuration Wizard.

XPE calculates power for each channel including the power of all associated circuits, shared resources between channels, I/O buffers, reference clock circuitry, and so forth. You therefore do not have to enter resource usage on any other sheet (for example, Clock or I/O) to describe the transceiver resources used.

XPE presents the MGT information in an architecture-specific way. Entering 2 or any multiple of 2 channels for a GTP/GTX_DUAL entry assumes that those channels use the minimum number of DUALs. Similarly, for GTHE1 and GTXE2 4 channels share common circuitry, so XPE assumes each line uses the minimum number of quads. To use 2 channels from one quad and 2 from another, specify them on two rows in XPE.

TIP: *For Spartan-6* devices*, you can specify a GTPA1_DUAL with different settings for each channel by entering each channel on a separate row using the same base name suffixed with* `_0` *and* `_1` *(for example,* `GTP_0` *and* `GTP_1`*). A red border around the cells of two adjacent rows indicates the two GTPA1s are inferred to be in the same GTPA1_DUAL.*

The **Power Planes** field in the MGT sheet represents the number of power planes used in the design. MGT transceivers require multiple analog power supplies for the PMA (Physical Medium Attachment). The number of power planes varies by device and package. When not all available MGTs are used, it might be possible to ground unused power planes to reduce the static power.

In the UltraScale and 7 series/Zynq-7000 SoC XPE spreadsheets, the GTX, GTP, GTH, and GTY sheets have an **OOB Used** column. The OOB feature uses out-of band (OOB) signaling for PCIe and other protocols where the physical connection may be unplugged during operation. OOB is supported using high-speed amplitude detection on the inputs and squelch on the corresponding outputs. A Yes in the **OOB Used** column indicates that your design will use this feature.

TIP: *For a description of the Eye Scan feature, which you can set to On or Off in the GTX, GTH, and GTY MGT sheets, see the 7 Series FPGAs GTX/GTH Transceivers User Guide (UG476) [Ref 8], the UltraScale Architecture GTH Transceivers Advance Specification User Guide (UG576) [Ref 14], or the UltraScale Architecture GTY Transceivers Advance Specification User Guide (UG578) [Ref 15].*

TIP: *In the 7 Series GTX sheet, you can set the* ***Power Mode*** *for GTX transceivers to Low Power or DFE. For a description of the low-power mode (LPM) and the decision feedback equalization (DFE) mode, see the RX Equalizer (DFE and LPM) section in the 7 Series FPGAs GTX/GTH Transceivers User Guide (UG476) [Ref 8].*

XPE does not support all of the possible MGT configurations. See the specific *Transceiver User Guide* for more information.

Figure 3-12: **GT Power Sheet (Kintex-7 Devices) Illustrating Data Rate and Power Estimates**

Transceiver Operational Modes

Operational Modes can be assigned to Transceiver entries to estimate the power consumption of that particular mode. Each type of transceiver supports the following operational modes:

- Transceiver: the default setting - both TX and RX channels enabled
- Transmitter: TX is enabled, RX is powered down
- Receiver: TX is powered down, RX is enabled

Some transceiver types support a fourth choice for Operational Mode of Power Down. This mode reflects the power consumed when both the TX and RX channels are powered down. The Power Down setting is available for UltraScale and UltraScale+ GTH and GTY transceivers and 7-Series GTH transceivers.

Clock Source for Quad-based PLL

For UltraScale and UltraScale+ devices, there are two Quad based Phase Locked Loops (QPLL0 and QPLL1) for jitter performance or Channel based ring oscillator Phase Locked Loop (CPLL). You can select QPLL0 that runs at 16GHz with an output divider of 2 as a clock source which is more conservative. For example: 16.375Gbps/2 =8.1875 Gbps. You can also select QPLL1 running at 8Ghz. For example, 8.1875 Gbps.

Power Down for PLLs

For UltraScale+ GTH and GTY, a power down feature is also added to individual PLLs (RX clock source and TX clock source). These PLLs can be individually set to power down mode to save power when they are not in use.

Note: There is no specific PLL power down mode in UltraScale. When the Operation Mode is set to **`Power Down`**, PLLs are also assumed to be powered down by default.

Hard IP Block Support for UltraScale Devices

The Hard IP Block setting allows you to calculate the power associated with the following UltraScale device integrated IP blocks:

- **PCIe** - The integrated PCI Express core is a reliable, high-bandwidth, scalable serial interconnect. Select **PCIe** when using hard GEN1, GEN2, GEN3 or GEN4 PCIe interface with GTs. Select **PCIe_500** when using hard GEN3 PCIe interface operating with an optional 500MHz core clock frequency.
- **100G Ethernet** - The integrated block for 100 Gb/s Ethernet (100G MAC) provides a high performance, low latency 100 Gb/s Ethernet port that allows for a wide range of user customization and statistics gathering. If your design uses the integrated block for 100 Gb/s Ethernet, select **CMAC**. Select **CMAC-Low** when you use low data toggle rate

or **CMAC-High** for worse case data toggle rate. For detailed information on this IP block, see the *UltraScale Architecture Integrated Block for 100G Ethernet LogiCORE IP Product Guide* (PG165) [Ref 29].

- **Interlaken** - The integrated block for Interlaken is a scalable chip-to-chip interconnect protocol designed to enable the following:
  - The lane logic only mode allows each serial transceiver to be used to build a fully featured Interlaken interface. In devices with 48 serial transceivers, up to 600 Gb/s of total throughput can be sustained.
  - The protocol logic supported in each integrated IP core scales up to 150 Gb/s.

  If your design uses the integrated block for Interlaken, select **ILKN**. Select **ILKN-Low** when you use low Tx data toggle rate or **ILKN-High** for worse case data toggle rate.For detailed information on this IP block, see the *UltraScale Architecture Integrated IP Core for Interlaken LogiCORE IP Product Guide* (PG169) [Ref 30].

These IP blocks are designed to be combined with GTH or GTY transceivers to implement an integrated solution. You can use the Transceiver Configuration wizard to combine the appropriate GTH or GTY transceiver configuration with an integrated hard IP block.

To open the Transceiver Configuration wizard, click the **Add GTH Interface** button at the top of the GTH sheet.

Or click the **Add GTY Interface** button at the top of the GTY sheet.

For a description of the entries in the Transceiver Configuration wizard, see Using the Transceiver Configuration Wizard.

Figure 3-13 shows the Hard IP Block setting in the UltraScale device XPE spreadsheet, the transceiver power used by the Hard IP, and the utilization percentage for each type of Hard IP.

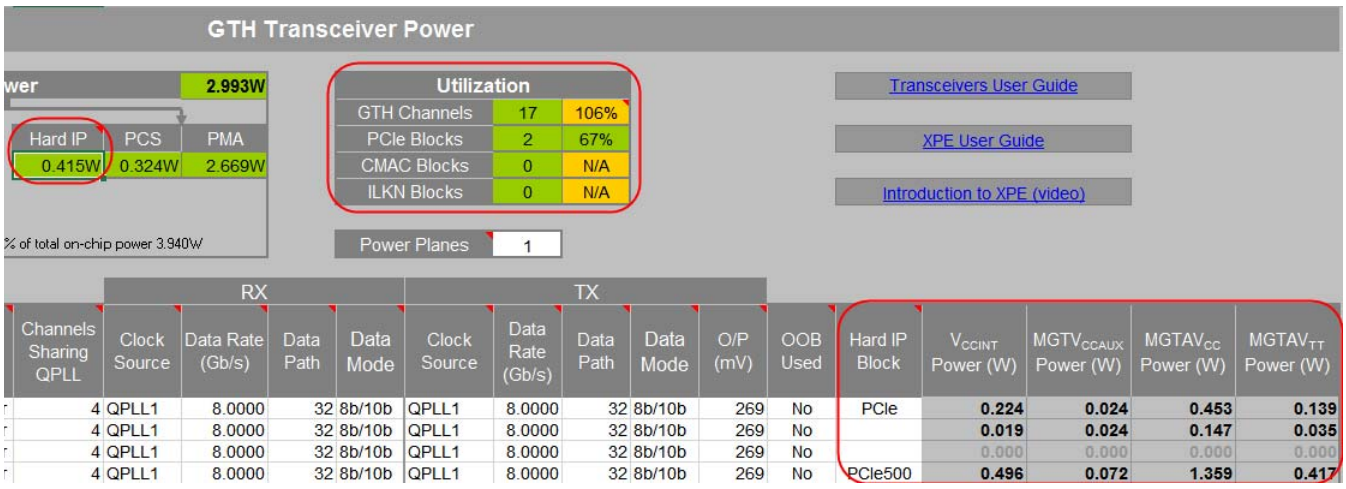


Figure 3-13: **Hard IP Settings for UltraScale Devices**

Transceiver Wizard and the MGT Sheet

For the 7 Series/Zynq-7000 SoC, UltraScale and UltraScale+ device XPE spreadsheets, you can enter transceiver information in a GT sheet (GTP, GTH, GTY, GTX, or GTZ) by using the Transceiver Configuration wizard. The Transceiver Configuration wizard provides a simplified method of filling in the GT sheets in the XPE spreadsheet.

The Transceiver Configuration wizard can be started from a GT sheet by clicking the **Add GT Interface** button (sample shown below).

For a description of the Memory Generator wizard and how you can run the wizard from the GT sheet, see Using the Transceiver Configuration Wizard.

To understand the capabilities of the 7 series GTs and the settings you will enter within XPE refer to the *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476) [Ref 8] and the *7 Series FPGAs GTP Transceivers User Guide* (UG482) [Ref 9].

To understand the capabilities of the UltraScale device GTs and the settings you will enter within XPE refer to the *UltraScale Architecture GTH Transceivers Advance Specification User Guide* (UG576) [Ref 14] and the *UltraScale Architecture GTY Transceivers Advance Specification User Guide* (UG578) [Ref 15].

GT Power Up / Power Down Sequencing

When your design has been programmed into the Xilinx device, the recommended GT transceiver power-on sequence to achieve minimum current draw is specified in the *Data Sheet: DC and Switching Characteristics* for the applicable device. The recommended

power-off sequence is the reverse of the power-on sequence to achieve minimum current draw.

If the recommended sequences are not followed, current drawn can be higher than specifications during power-up and power-down. XPE calculates GT power assuming the recommended power-up, power-down sequence is used by the design, and the power numbers reflect this.

To calculate the extra power as a result of not following the recommended power up/down sequence, consult the Xilinx Answer Records.

Using the GTM Sheet

GTM is a high speed serial transceiver available for a few Virtex UltraScale + devices and it supports PAM4 and NRZ signaling. The highest data rate supported is 58 Gb/s and lowest data rate supported is 9.8 Gb/s. For more information on GTM, see *Virtex UltraScale+ FPGAs GTM Transceivers User Guide* (UG581). Registration required. Contact your local Xilinx sales representative to request access to this lounge.

Transceiver Operational Modes

You can assign Operational Modes to Transceiver entries to estimate the power consumption of a particular mode. Each type of transceiver supports the following operational modes:

- **Transceiver**: Both TX and RX channels are enabled (default setting)
- **Transmitter**: TX is enabled, RX is powered down
- **Receiver**: TX is powered down, RX is enabled

Data Rate and Interface Width

The Data Rate in Transceiver mode is the same for both transmitter and receiver, unlike GTY which supports different Data rates for transmitter and receiver. GTM allows a PMA interface width of 64 bits for NRZ and 128 bit for PAM4 modulation. The supported data rates depend upon the speed grade. The maximum and minimum range are as follows:

- Allowed data rates for PAM4 are 19.6 to 58.00 Gb/s
- Allowed data rates for NRZ are 9.8 to 29.00 Gb/s

The GTM Transceiver Configuration Wizard can be used to create a GTM interface and the process of creating a GTM interface is similar to the process of creating the GTY and the GTH wizards.

RS-FEC

The GTM Architecture has hardened KP4 FEC support for Both transmitter and receiver. It is always enabled for PAM4 mode but can be bypassed when using the NRZ mode.

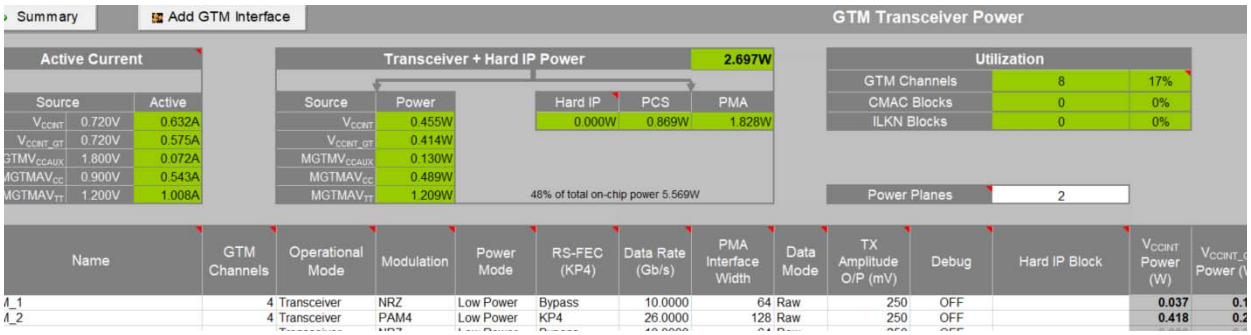


Figure 3-14: **GTM Transceiver Power**

Using the TEMAC Sheet

Different Xilinx device families contain Tri-Mode Embedded Ethernet Media Access Controller (MAC) blocks, which are used in Ethernet applications. The Ethernet MACs are paired within a TEMAC block, share a common host and DCR interface, but are independently configurable to meet all common Ethernet system connectivity needs. Refer to the applicable EMAC User Guide for a detailed description of the block capabilities and configuration.

In XPE, you need only enter the TEMAC operating clock frequency (See Figure 3-15). You typically need to know the mode and operating speed to obtain the correct clock frequency.

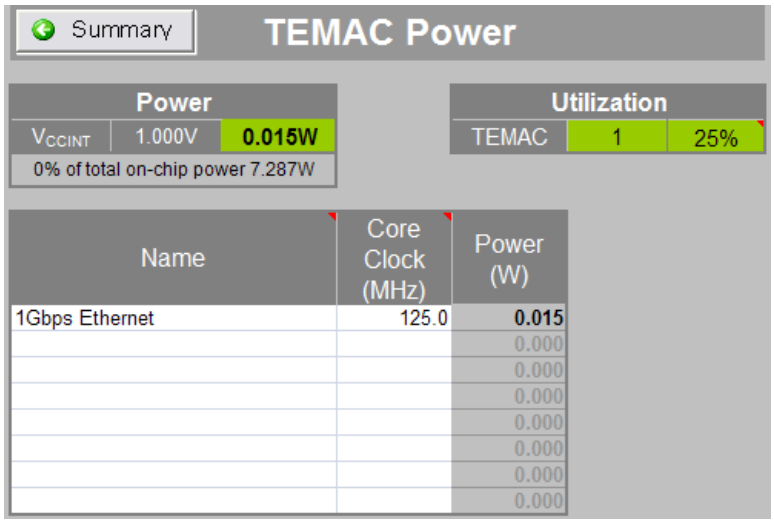


Figure 3-15: **TEMAC Power Sheet (Virtex-6 Devices)**

Using the PCIe Sheet

Different Xilinx device families have Integrated Endpoint Block for PCI Express designs (integrated Endpoint block). For detailed PCIe information, refer to the applicable PCIe User Guide and enter in XPE the settings which correspond to your application.

Note: The XPE spreadsheets of 7 series and later devices do *not* have a PCI Express sheet. For these devices, PCIe information is specified on the Multi-Gigabit Transceiver sheets. See Using the Transceiver Sheets (GTP, GTX, GTH, GTY, GTZ).

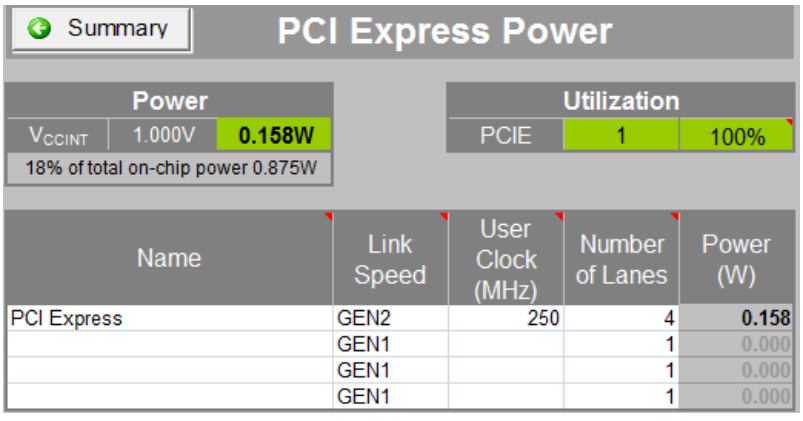


Figure 3-16: **PCIE Power Sheet (Virtex-6 Devices)**

Using PPC440 (PowerPC) Sheets

Some Xilinx device families contain high-performance PowerPC microprocessor embedded blocks.

For power estimation, these blocks are represented in a separate sheet within XPE. Details for each PowerPC setting are available in the applicable device User Guide. Typically, you can provide the processor main clock frequency along with details of the processor local bus, memory, and eventual DMA controllers. Figure 3-17 presents an example with a Virtex-5 device.

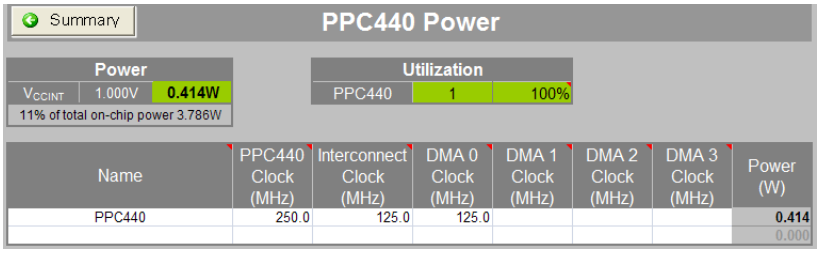


Figure 3-17: **PPC440 Power Sheet (Virtex-5 Devices)**

Using the PS Sheet (Zynq-7000 SoC and Zynq UltraScale+ MPSoC)

The PS sheet allows you to estimate power for the Processor System (PS). The PS sheet is available for Zynq-7000 SoC and Zynq UltraScale+ MPSoC Only.

Using the PS Sheet for Zynq-7000 SoC

The PS has between two and five voltage sources depending on the exact configuration. The V_{CCO_DDR} voltage is dependent on the memory interface selected and the V_{CCO_MIO0} and V_{CCO_MIO1} voltages are dependent on the I/O interfaces and standards used in the respective banks.

The PS in the Zynq-7000 SoC is described in the *Zynq-7000 SoC Technical Reference Manual* (UG585) [Ref 31].

- **Processor**

  The processor used in the PS is a dual core Cortex™-A9 processor. The number of **A9 Cores** used and their clock frequency (**Clock (MHz)**) are required information. Processor Load of 50% is for average usage and can be adjusted up or down as needed to reflect the processor loading in a specific design. The 0% setting represents the processor in WFI and the duration of Wait for Interrupt to describe the WFI. The 100% setting represents the Dhrystone benchmark. The CPU, when in a wait loop that uses every cycle should be set to 80% to represent less computation than Dhrystone.

- **PLLs**

  There are three **PLL**s in the PS that must be set to the correct frequency (**MHz**) when used. By default the **Processor** and **Memory** PLLs run at twice their associated clock frequency.

- **Memory Interface**

  DDR2, DDR3, DDR3L, and LPDDR2 memory interfaces (**Memory Type**) are supported in either 16 or 32 bit **Data Width**. The clock frequency (**Clock (MHz)**) is half the data rate, because these are all DDR interfaces. The **Read Rate** and **Write Rate** represent the usage and can be set to any values that together are less than or equal to 100%.

  The **Data Toggle Rate** is the average for the data lines with 50% being random data. The **Output Load** is the board capacitance and the external termination (**External Term**) is the far end parallel termination used for the data lines.

- **I/O Interfaces**

The PS supports a variety of standard interfaces (**I/O Standard**) and some general purpose I/O. There are two I/O banks and all interfaces on a bank must use the same voltage. Available **I/O Interface**s, **I/O Standard**s, **Number of Interfaces**, and **I/O Bank** placement are represented in the XPE tool.

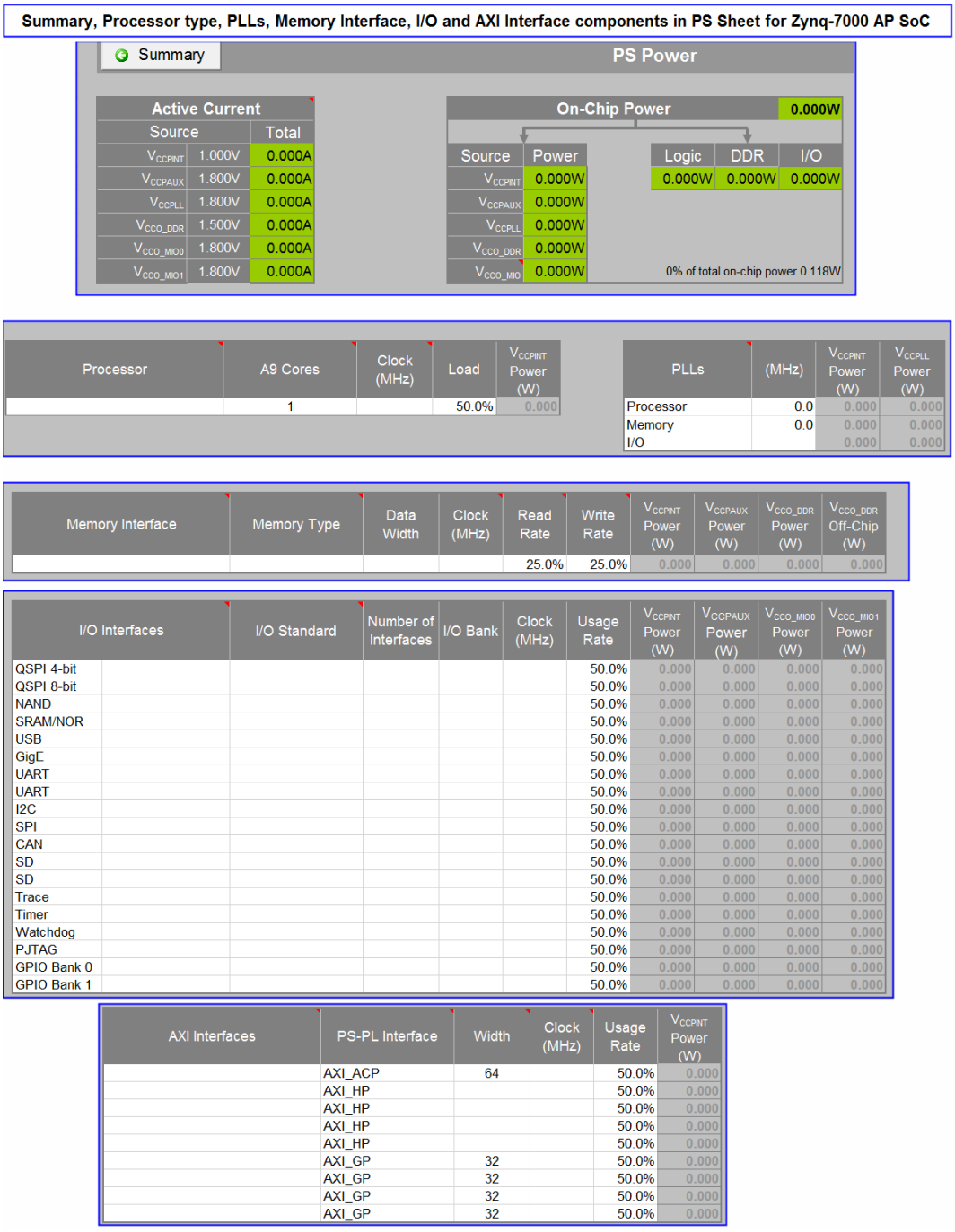


Figure 3-18: **PS Sheet for Zynq-7000 SoC**

- **AXI Interfaces**

  The PS side of the AXI interfaces are based on the AXI 3 interface specification. Each interface consists of multiple AXI channels. There are nine AXI interfaces for PS-PL interfacing.

  - AXI_ACP - One cache coherent master port for the PL.
  - AXI_HP - Four high performance/bandwidth master ports for the PL.
  - AXI_GP - Four general purpose ports (two master ports and two slave ports).

Using the PS Sheet for Zynq UltraScale+ MPSoC

The Zynq® UltraScale+™ MPSoC family is based on the Xilinx® UltraScale™ MPSoC architecture. This family of products integrates a feature-rich 64-bit quad-core Arm® Cortex™-A53 and dual-core Arm Cortex-R5 based processing system (PS) and Xilinx programmable logic (PL) UltraScale architecture in a single device.

The following configurations as shown in Figure 3-19 are available for PS Sheet:

1. Power Down Mode
2. Deep Sleep Mode
3. User config mode

Low Power and Full Power Domains

You can now select different Full Power and Low Power domain configurations from the PS panel as shown in Figure 3-20 to Figure 3-22.

- **Processor and PLLs**

The PS for UltraScale+ MPSoC integrates a feature-rich 64-bit quad-core Arm® Cortex™-A53 for full power and dual-core Arm Cortex-R5 based processing system (PS) for low power domains. It also integrates Xilinx programmable logic (PL) UltraScale architecture in a single device. APU, DDR and Video PLLs are available in full power domain while I/O and RPU PLLs are available in the low power domain.

- **Memory and I/O Interfaces**

The Arm Cortex-A53 and Cortex-R5 CPUs also include on-chip memory, external memory interfaces and a rich set of peripheral connectivity interfaces.

- **AXI FIFO (AFI) Interfaces**

AXI FIFO Interface of low power domain includes 1 slave and 1 master interfaces. However, full power domain includes 6 slave and 2 master interfaces.

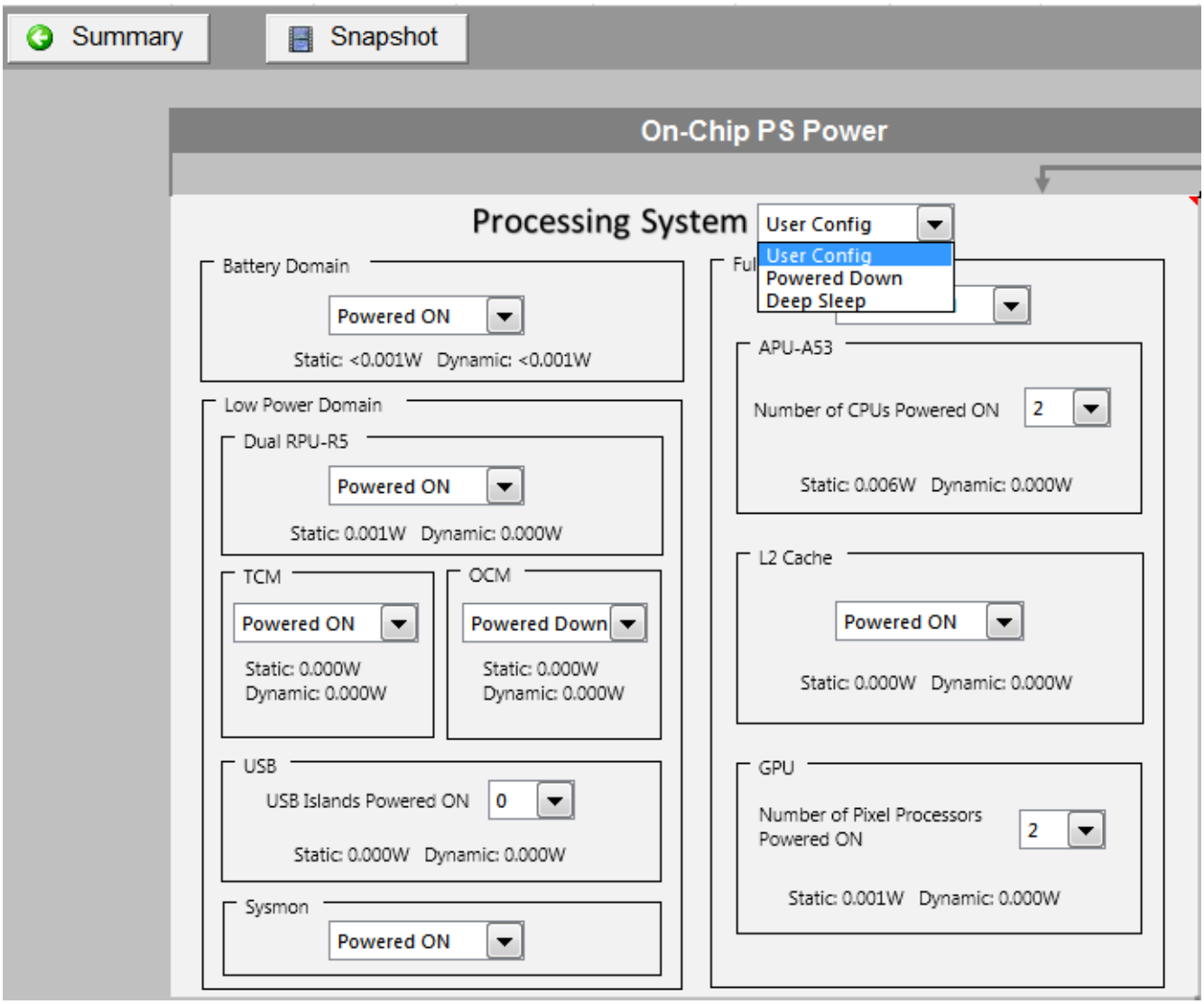
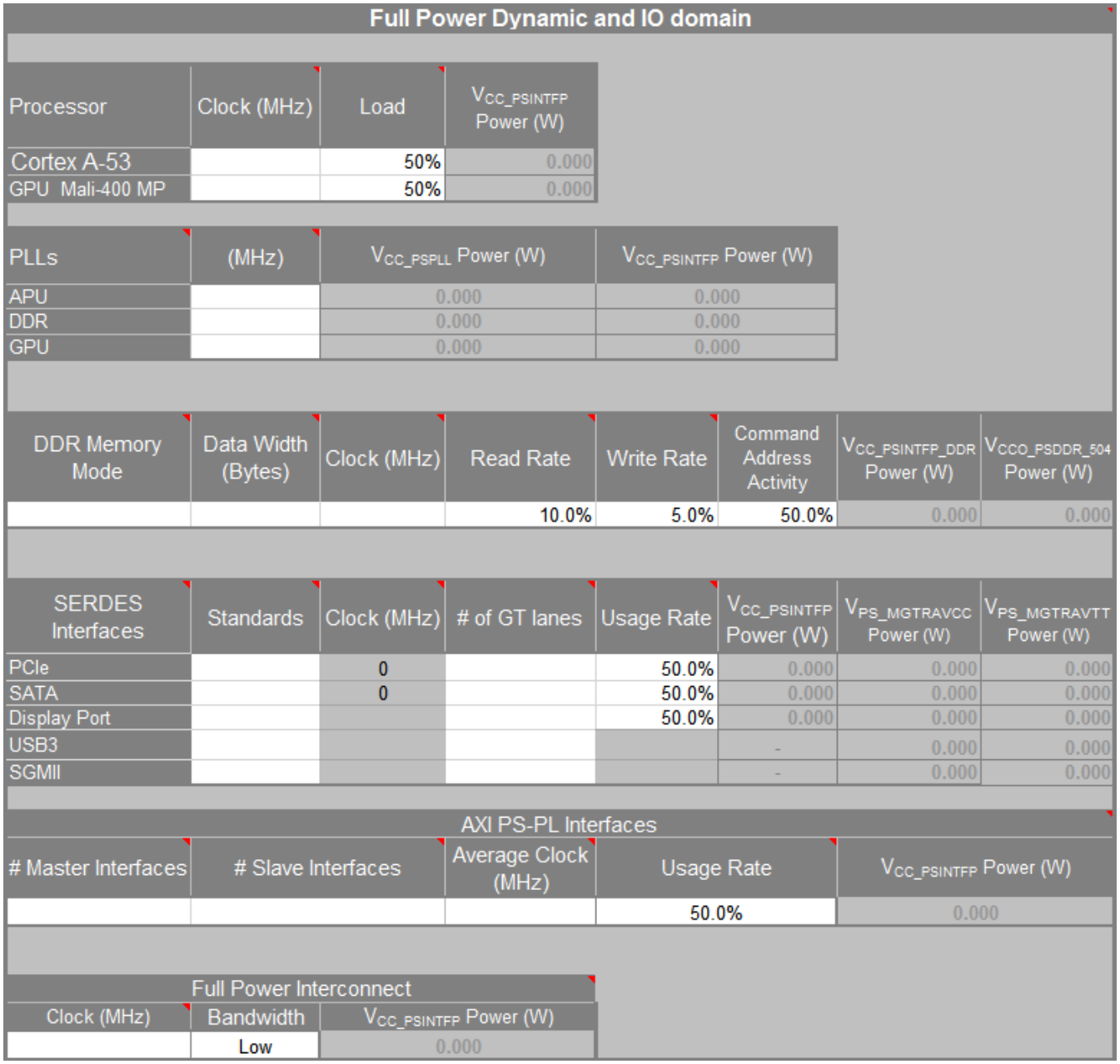


Figure 3-19: **PS Sheet for Zynq UltraScale+ MPSoC displaying Input panel**

Domain	Source	Total
Battery Power	V_{CC_PSBATT}	0.000W
Low Power (Logic + IO)	$V_{CC\_PSINTLP}$	0.009W
	$V_{CCO\_PSIO0\_500}$	0.000W
	$V_{CCO\_PSIO1\_501}$	0.000W
	$V_{CCO\_PSIO2\_502}$	0.000W
	$V_{CCO\_PSIO3\_503}$	0.002W
	V_{CC_PSPLL}	0.002W
	V_{CC_PSADC}	0.003W
	V_{CC_PSAUX}	0.004W
	LPD Power	0.020W
Full Power (Logic + IO)	$V_{CC\_PSINTFP}$	0.000W
	$V_{CCO\_PSDDR\_504}$	0.000W
	$V_{PS\_MGTRAVCC}$	0.000W
	$V_{PS\_MGTRAVTT}$	0.000W
	$V_{CC\_PSINTFP\_DDR}$	0.000W
	$V_{CC\_PSDDR\_PLL}$	0.000W
	FPD Power	0.000W

Power Supply		
Source	Voltage	Total (A)
V_{CC_PSBATT}	1.200	0.000
$V_{CC\_PSINTLP}$	0.850	0.011
$V_{CCO\_PSIO0\_500}$	3.300	0.000
$V_{CCO\_PSIO1\_501}$	3.300	0.000
$V_{CCO\_PSIO2\_502}$	3.300	0.000
$V_{CCO\_PSIO3\_503}$	3.300	0.001
$V_{CC\_PSINTFP}$	0.850	0.000
$V_{CCO\_PSDDR\_504}$	1.200	0.000
$V_{PS\_MGTRAVCC}$	0.850	0.000
$V_{PS\_MGTRAVTT}$	1.800	0.000
$V_{CC\_PSINTFP\_DDR}$	0.850	0.000
V_{CC_PSPLL}	1.200	0.002
$V_{CC\_PSDDR\_PLL}$	1.800	0.000
V_{CC_PSADC}	1.800	0.002
V_{CC_PSAUX}	1.800	0.002

Figure 3-20: **PS Sheet for Zynq UltraScale+ MPSoC displaying Sources and Power Supply**



Full Power Dynamic and IO domain

Processor	Clock (MHz)	Load	$V_{CC\_PSINTFP}$ Power (W)
Cortex A-53		50%	0.000
GPU Mali-400 MP		50%	0.000

PLLs	(MHz)	V_{CC_PSPLL} Power (W)	$V_{CC\_PSINTFP}$ Power (W)
APU		0.000	0.000
DDR		0.000	0.000
GPU		0.000	0.000

DDR Memory Mode	Data Width (Bytes)	Clock (MHz)	Read Rate	Write Rate	Command Address Activity	$V_{CC\_PSINTFP\_DDR}$ Power (W)	$V_{CCO\_PSDDR\_504}$ Power (W)
			10.0%	5.0%	50.0%	0.000	0.000

SERDES Interfaces	Standards	Clock (MHz)	# of GT lanes	Usage Rate	$V_{CC\_PSINTFP}$ Power (W)	$V_{PS\_MGTRAVCC}$ Power (W)	$V_{PS\_MGTRAVTT}$ Power (W)
PCIe		0		50.0%	0.000	0.000	0.000
SATA		0		50.0%	0.000	0.000	0.000
Display Port				50.0%	0.000	0.000	0.000
USB3					-	0.000	0.000
SGMII					-	0.000	0.000

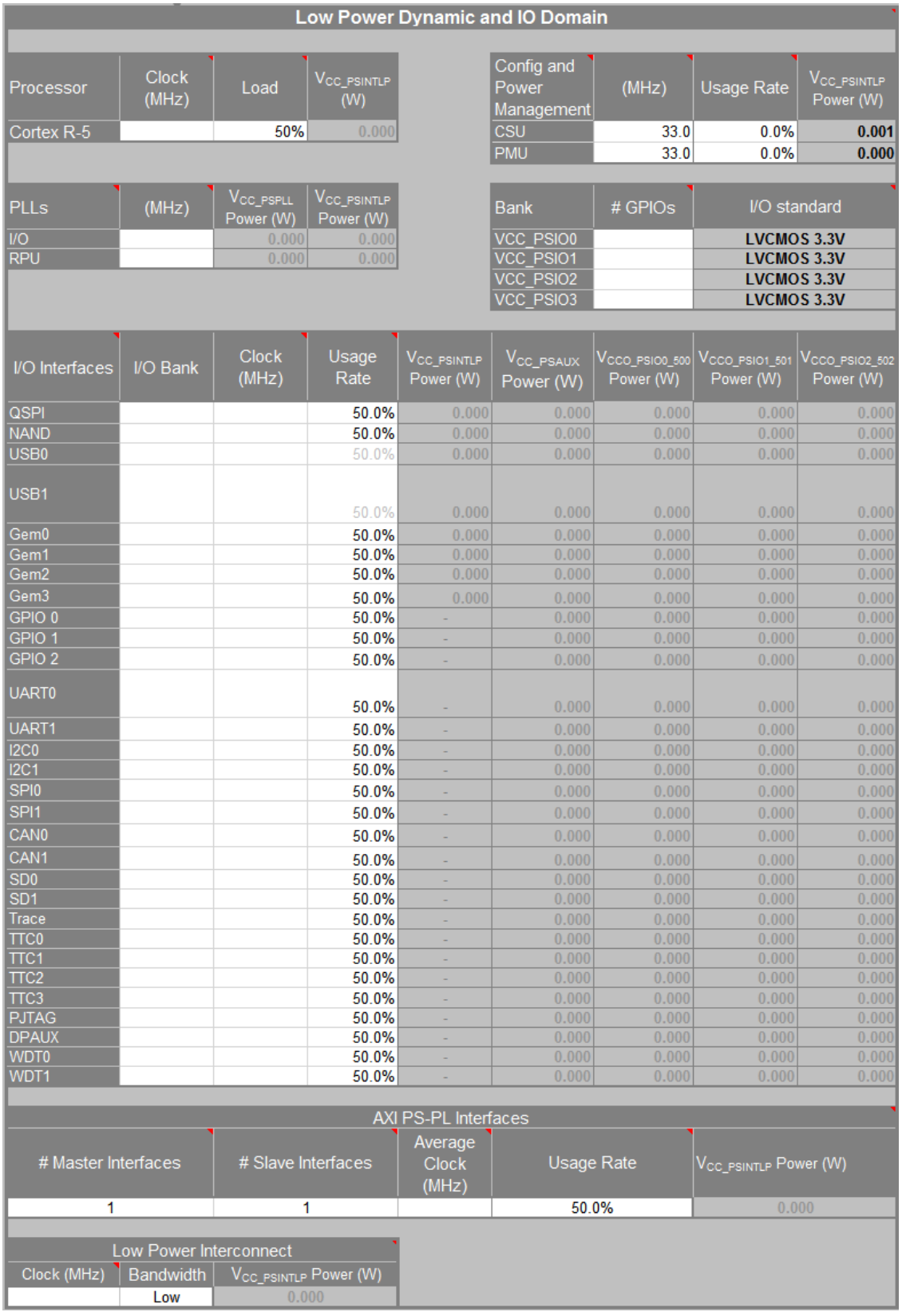
AXI PS-PL Interfaces

# Master Interfaces	# Slave Interfaces	Average Clock (MHz)	Usage Rate	$V_{CC\_PSINTFP}$ Power (W)
			50.0%	0.000

Full Power Interconnect

Clock (MHz)	Bandwidth	$V_{CC\_PSINTFP}$ Power (W)
	Low	0.000

Figure 3-21: **PS Sheet for Zynq UltraScale+ MPSoC displaying Full Power Dynamic**



Low Power Dynamic and IO Domain

Processor	Clock (MHz)	Load	$V_{CC\_PSINTLP}$ (W)
Cortex R-5		50%	0.000

Config and Power Management	(MHz)	Usage Rate	$V_{CC\_PSINTLP}$ Power (W)
CSU	33.0	0.0%	0.001
PMU	33.0	0.0%	0.000

PLLs	(MHz)	V_{CC_PSPLL} Power (W)	$V_{CC\_PSINTLP}$ Power (W)
I/O		0.000	0.000
RPU		0.000	0.000

Bank	# GPIOs	I/O standard
VCC_PSIO0		LVCMOS 3.3V
VCC_PSIO1		LVCMOS 3.3V
VCC_PSIO2		LVCMOS 3.3V
VCC_PSIO3		LVCMOS 3.3V

I/O Interfaces	I/O Bank	Clock (MHz)	Usage Rate	$V_{CC\_PSINTLP}$ Power (W)	V_{CC_PSAUX} Power (W)	$V_{CCO\_PSIO0\_500}$ Power (W)	$V_{CCO\_PSIO1\_501}$ Power (W)	$V_{CCO\_PSIO2\_502}$ Power (W)
QSPI			50.0%	0.000	0.000	0.000	0.000	0.000
NAND			50.0%	0.000	0.000	0.000	0.000	0.000
USB0			50.0%	0.000	0.000	0.000	0.000	0.000
USB1			50.0%	0.000	0.000	0.000	0.000	0.000
Gem0			50.0%	0.000	0.000	0.000	0.000	0.000
Gem1			50.0%	0.000	0.000	0.000	0.000	0.000
Gem2			50.0%	0.000	0.000	0.000	0.000	0.000
Gem3			50.0%	0.000	0.000	0.000	0.000	0.000
GPIO 0			50.0%	-	0.000	0.000	0.000	0.000
GPIO 1			50.0%	-	0.000	0.000	0.000	0.000
GPIO 2			50.0%	-	0.000	0.000	0.000	0.000
UART0			50.0%	-	0.000	0.000	0.000	0.000
UART1			50.0%	-	0.000	0.000	0.000	0.000
I2C0			50.0%	-	0.000	0.000	0.000	0.000
I2C1			50.0%	-	0.000	0.000	0.000	0.000
SPI0			50.0%	-	0.000	0.000	0.000	0.000
SPI1			50.0%	-	0.000	0.000	0.000	0.000
CAN0			50.0%	-	0.000	0.000	0.000	0.000
CAN1			50.0%	-	0.000	0.000	0.000	0.000
SD0			50.0%	-	0.000	0.000	0.000	0.000
SD1			50.0%	-	0.000	0.000	0.000	0.000
Trace			50.0%	-	0.000	0.000	0.000	0.000
TTC0			50.0%	-	0.000	0.000	0.000	0.000
TTC1			50.0%	-	0.000	0.000	0.000	0.000
TTC2			50.0%	-	0.000	0.000	0.000	0.000
TTC3			50.0%	-	0.000	0.000	0.000	0.000
PJTAG			50.0%	-	0.000	0.000	0.000	0.000
DPAUX			50.0%	-	0.000	0.000	0.000	0.000
WDT0			50.0%	-	0.000	0.000	0.000	0.000
WDT1			50.0%	-	0.000	0.000	0.000	0.000

AXI PS-PL Interfaces

# Master Interfaces	# Slave Interfaces	Average Clock (MHz)	Usage Rate	$V_{CC\_PSINTLP}$ Power (W)
1	1		50.0%	0.000

Low Power Interconnect

Clock (MHz)	Bandwidth	$V_{CC\_PSINTLP}$ Power (W)
	Low	0.000

Figure 3-22: **PS Sheet for Zynq UltraScale+ MPSoC displaying Low Power Dynamic**

Setting Clocks for Zynq UltraScale+ PS Sheet

There are many clocks present in the PS subsystem which need to be added correctly into the PS Sheet for Zynq UltraScale+ MPSoC XPE. If a Vivado design with PS IP exists, then run the report power to generate a .xpe file and import it back to XPE. This method ensures that the PS clocks are set appropriately to reflect the evaluated Vivado design.

If you want to initiate XPE PS clocks from scratch, use the following guidelines to provide accurate clock information to the tool:

1. Use the Zynq UltraScale+ MPSoC Vivado IP from IP catalog as shown in Figure 3-23.

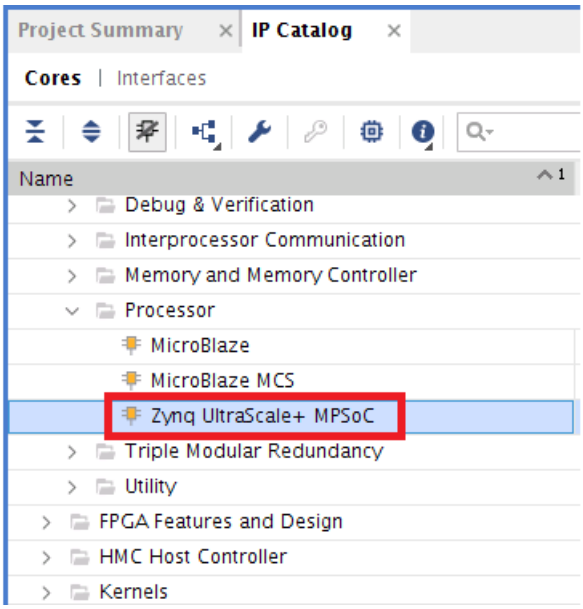


Figure 3-23: **IP Catalog**

2. Customize this IP to view all of the clocking information in the Clock Configuration - Output Clocks tab with default clock rates already populated.

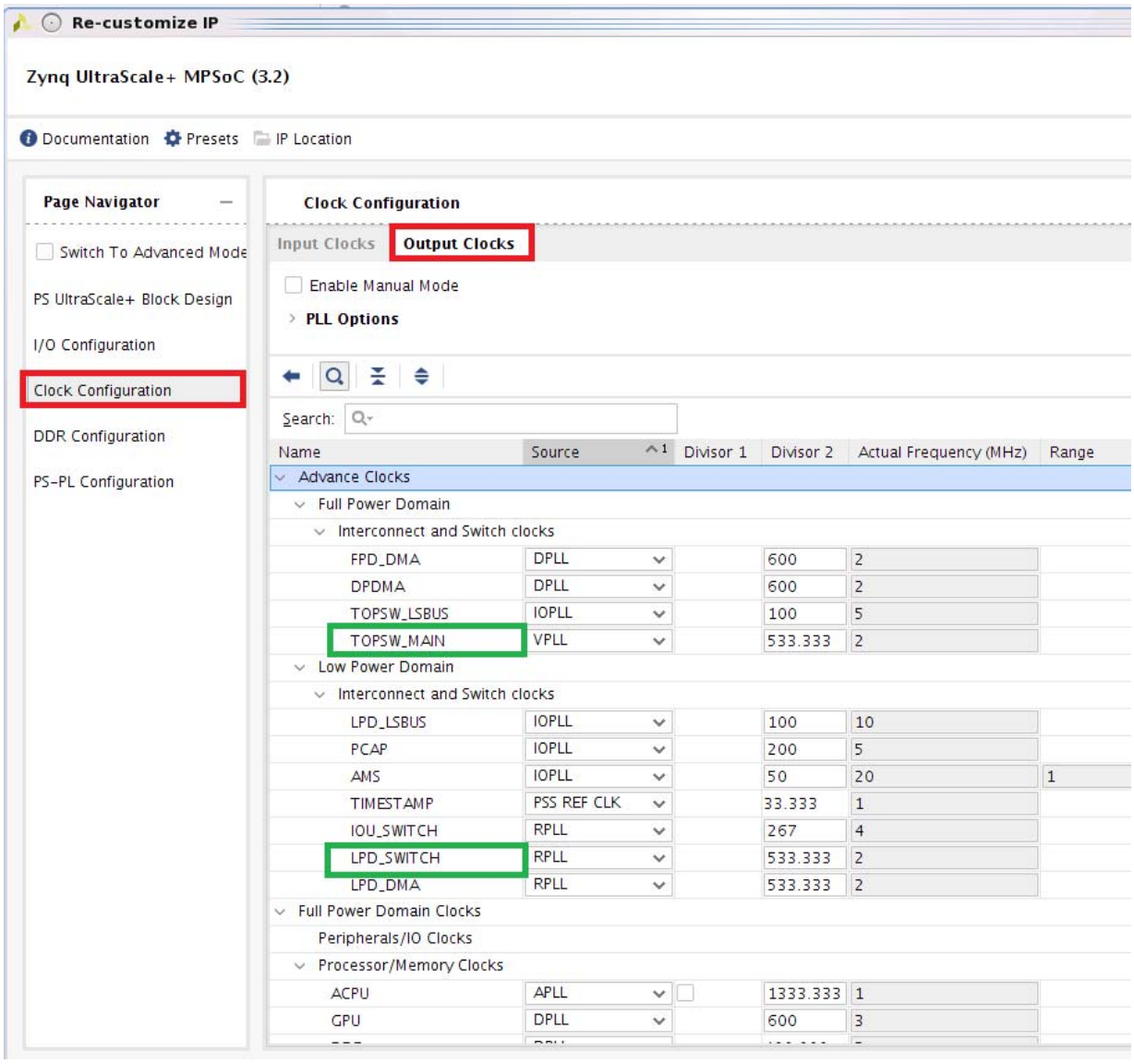
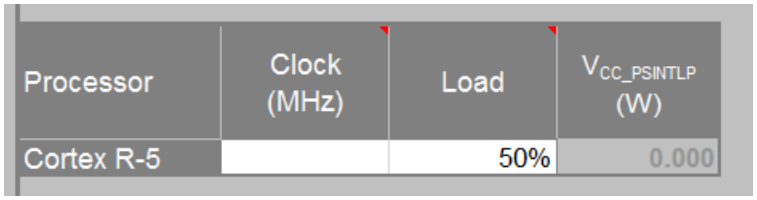


Figure 3-24: **Customize Zynq UltraScale+ MPSoC IP**

3. Low and Full power interconnect clock rates are called LPD_SWITCH (LP) and TOPSW_MAIN (FP) in the clock configuration tab as shown in Figure 3-24 as green color boxes.

4. The following block clocks can be generated with clock configuration tab of Zynq UltraScale+ MPSoC IP:

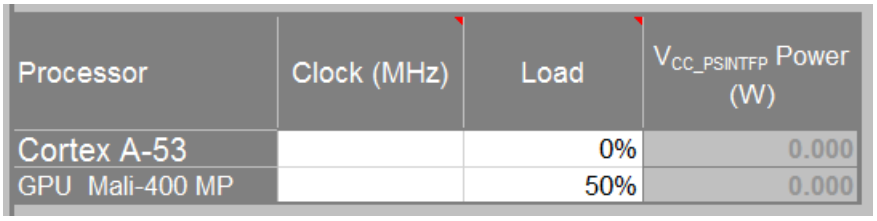
   a. R5 Clocks:



Processor	Clock (MHz)	Load	$V_{CC\_PSINTLP}$ (W)
Cortex R-5		50%	0.000

Figure 3-25: **R5 Clocks**

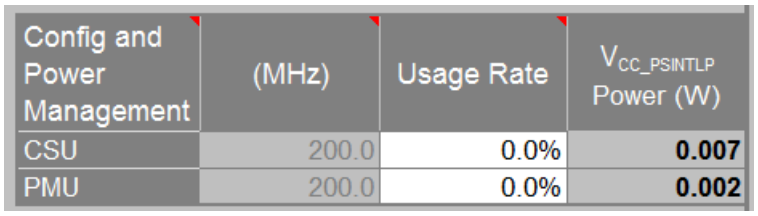
   b. A53/GPU Clocks:



Processor	Clock (MHz)	Load	$V_{CC\_PSINTFP}$ Power (W)
Cortex A-53		0%	0.000
GPU Mali-400 MP		50%	0.000

Figure 3-26: **A53 / GPU Clocks**

   c. CSU and PMU Clocks:

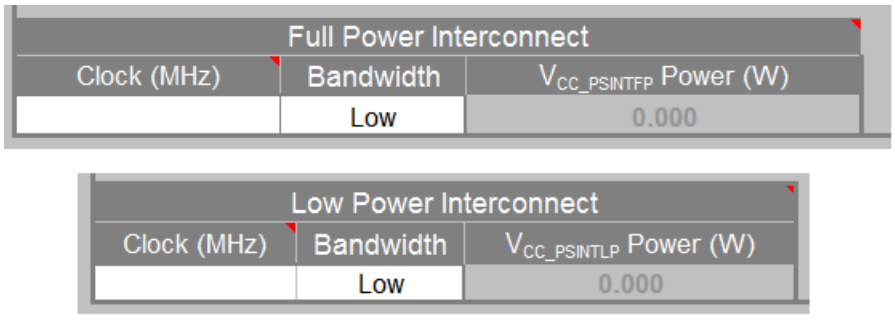


Config and Power Management	(MHz)	Usage Rate	$V_{CC\_PSINTLP}$ Power (W)
CSU	200.0	0.0%	**0.007**
PMU	200.0	0.0%	**0.002**

Figure 3-27: **CSU and PMU Clocks**

Note: The CSU value is given in the data sheet for the device but the rate is fixed . The PMU value is not in the data sheet but is the same as the CSU rate.

   d. Full power / Low power interconnect clocks:



Full Power Interconnect		
Clock (MHz)	Bandwidth	$V_{CC\_PSINTFP}$ Power (W)
	Low	0.000

Low Power Interconnect		
Clock (MHz)	Bandwidth	$V_{CC\_PSINTLP}$ Power (W)
	Low	0.000

Figure 3-28: **Full power / Low power interconnect clocks**

Note: The FP and LP interconnect clocks can be generated using the TOPSW_MAIN and LPD_SWITCH (marked as green boxes in Figure 3-24).

5. Customize the Zynq UltraScale+ IP and use the PLL Options section of Clock Configuration to generate the PLL section (marked as blue boxes in Figure 3-29).

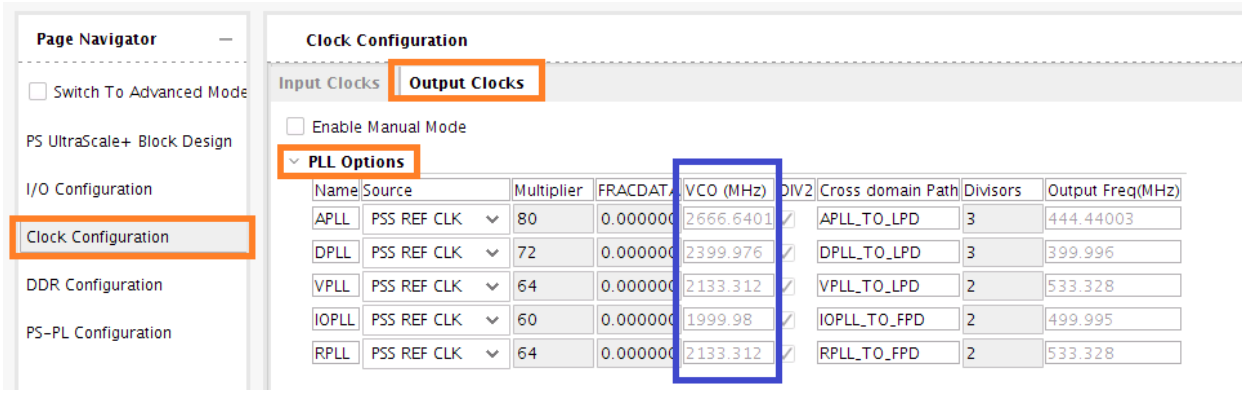


Figure 3-29: **PLL Options in Clock Configuration**

The corresponding blocks for PLL in XPE are shown below.

PLLs	(MHz)	V_{CC_PSPLL} Power (W)	$V_{CC\_PSINTFP}$ Power (W)
APU	2666.0	**0.022**	**0.004**
DDR	2400.0	**0.021**	**0.004**
GPU	2133.0	**0.020**	**0.003**

PLLs	(MHz)	V_{CC_PSPLL} Power (W)	$V_{CC\_PSINTLP}$ Power (W)
I/O	2000.0	**0.020**	**0.003**
RPU	2133.0	**0.020**	**0.003**

Figure 3-30: **Corresponding Blocks for PLL in XPE**

6. The remaining sections for DDR memory, SERDES interface, and PSIO are all design dependent.
7. After the basic setup is complete, you can manipulate the setup using the global settings box shown in Figure 3-19, page 82. This allows you to see the impact of turning some of the A53s or GPUs etc. on or off.

Note: Load (%) for the processor and IOs can be estimated based on the design. Some I/O load percentages do not affect the power. Some processor % numbers can be extracted from the profiling tools.

See Xilinx Answer: 69019 for more information.

Example: Setting Petalinux Boot IDLE state

One of the very common example is Petalinux Boot Idle state. It represents a system or state when GPU is not needed. However, the processors are immediately ready.

This mode requires the following configuration into the XPE PS sheet:

1. Set R5 Idle: Set R5 load to 0%
2. Set A53 Idle: Set A53 processor load to 0%
3. Set GPU Off:
    a. Set number of GPU processors to 0 from the drop down list
    a. Clear the clock frequency of GPU and set load to 0%
    a. Clear the GPU PLL clock frequency

Figure 3-31 shows this mode where Processors (A53 and R5) are Idle and GPU is OFF.

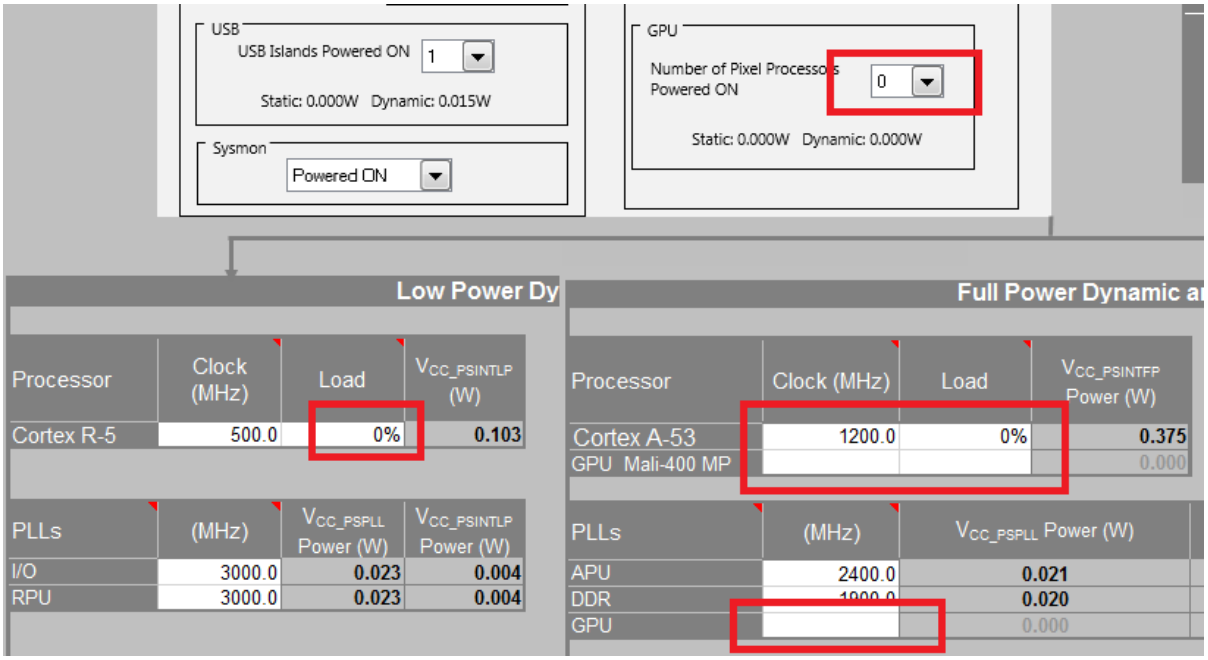


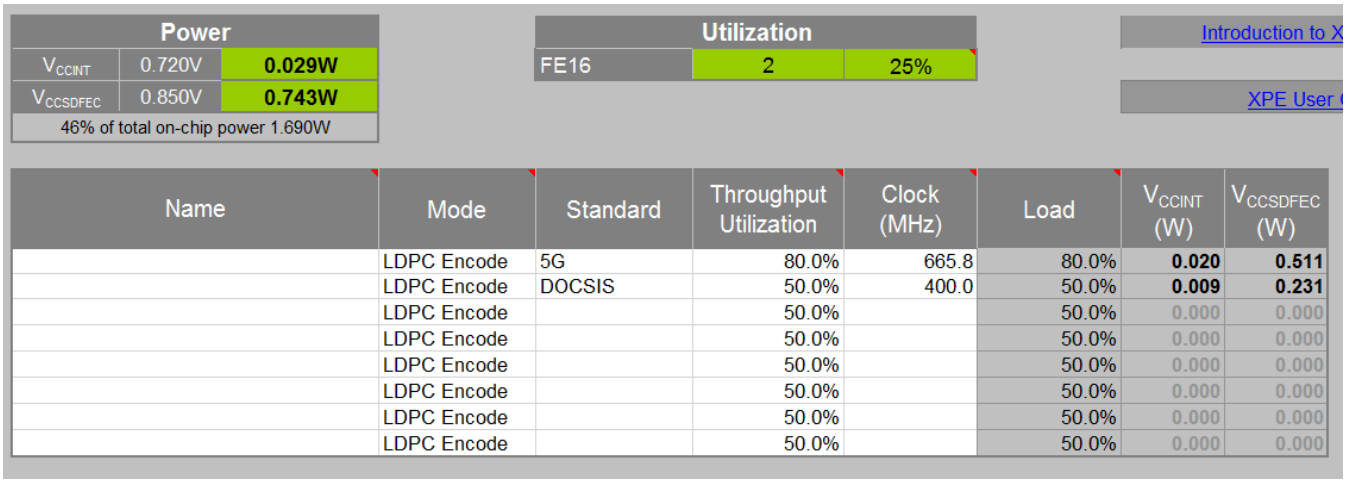
Figure 3-31: **Petalinux Boot IDLE State Example**

Using Soft-Decision FEC (SD-FEC) Sheet

Some Zynq UltraScale+ RFSoCs include highly flexible soft-decision FEC blocks for decoding and encoding data as a means to control errors in data transmission over unreliable or noisy communication channels. The SD-FEC blocks support low-density parity

check (LDPC) decode/encode and Turbo decode for use in 5G wireless, backhaul, DOCSIS, and LTE applications.

The SD-FEC is available in ZU21DR and ZU28DR devices. A total of 8 cores can be used for error correction. If you select any of the above mentioned two devices in XPE spreadsheet, a new SD-FEC tab will be added. This sheet can be used for power estimation of SD-FEC.



Power		
V_{CCINT}	0.720V	0.029W
$V_{CCSDFEC}$	0.850V	0.743W
46% of total on-chip power 1.690W		

Utilization		
FE16	2	25%

Introduction to X

XPE User

Name	Mode	Standard	Throughput Utilization	Clock (MHz)	Load	V_{CCINT} (W)	$V_{CCSDFEC}$ (W)
	LDPC Encode	5G	80.0%	665.8	80.0%	**0.020**	**0.511**
	LDPC Encode	DOCSIS	50.0%	400.0	50.0%	**0.009**	**0.231**
	LDPC Encode		50.0%		50.0%	0.000	0.000
	LDPC Encode		50.0%		50.0%	0.000	0.000
	LDPC Encode		50.0%		50.0%	0.000	0.000
	LDPC Encode		50.0%		50.0%	0.000	0.000
	LDPC Encode		50.0%		50.0%	0.000	0.000
	LDPC Encode		50.0%		50.0%	0.000	0.000

Figure 3-32: **SD-FEC sheet**

Use the following inputs to estimate the power:

Mode

The following SD-FEC configuration modes are available for selection:

- LDPC Encode
- LDPC Decode
- Turbo Decode

Standard

Specify the standard used for a particular application.You can select any of the following standards:

- 5G
- DOCSIS
- Wi-Fi
- Custom
- LTE for Turbo Decode

Throughput Utilization

Specify the throughput utilization of the SD-FEC cores used. This is instance throughput relative to maximum throughput supported by the core.

Clock

Specify the clock for the SD-FEC core. After providing these inputs, you can view the power estimation report for VCCINT rail and VCCSDFEC. On the summary sheet, VCCSDFEC rail current is also displayed under Power Supply section. See *Zynq UltraScale+ RFSoC Data Sheet: Overview* (DS898)[Ref 41] for more information on SD-FEC usage.

Using RFADC-DAC Sheet

Most Zynq UltraScale+ RFSoCs include an RF data converter subsystem, which contains multiple radio frequency analog to digital converters (RF-ADCs) and multiple radio frequency digital to analog converters (RF-DACs).

The high-precision, high-speed, power efficient RF-ADCs and RF-DACs can be individually configured for real data or can be configured in pairs for real and imaginary I/Q data. The 12-bit RF-ADCs support sample rates up to 2GSPS or 4GSPS, depending on the selected device. The 14-bit RF-DACs support sample rates up to 6.4GSPS.

The Following RFADC-DAC sheet displays the ZU28DR device containing eight 2GSPS RF-ADC and eight 6.4GSPS RF-DAC channels:

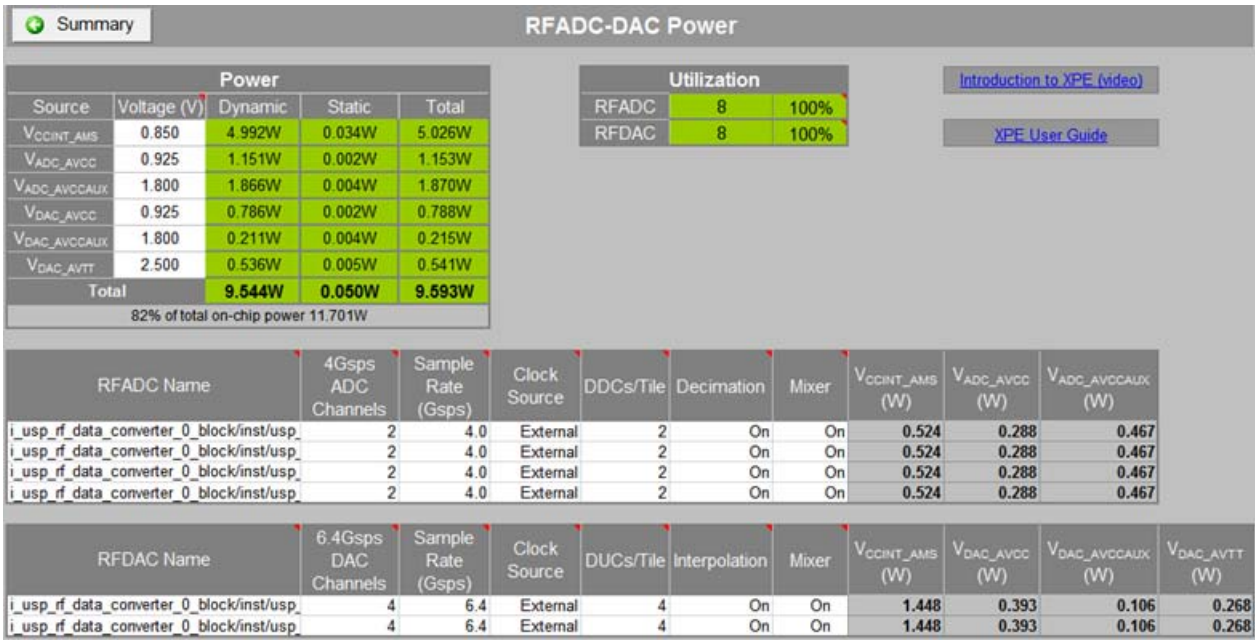


Figure 3-33: **RFADC-DAC Sheet**

Power can be estimated by selecting the valid combinations of possible configuration of ADC-DAC such as clock frequency, decimation, interpolation and mixer usage.

The XPE summary sheet does not contain the power rails of ADC-DAC block. Click RFADC-DAC Power Rails button to view the power supply section of RFADC-DAC sheet as shown in Figure 3-34.

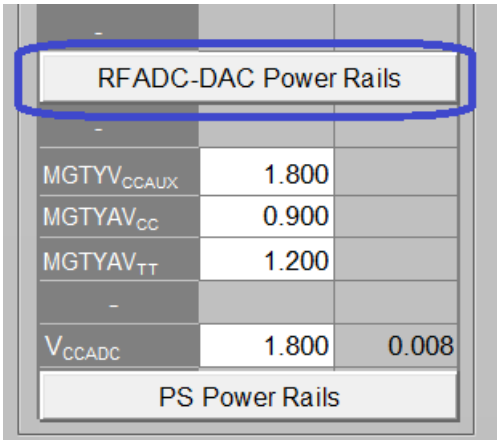


Figure 3-34: **Selection of RFADC-DAC Power Rails in Summary Sheet**

See *Zynq UltraScale+ RFSoC Data Sheet: Overview* (DS898)[Ref 41] for more information on RFADC-DAC usage.

Estimating HBM Power (HBM Sheet)

The HBM sheet is available only for Virtex UltraScale+ HBM devices. This sheet is used to estimate the power of High Bandwidth Memory. Each HBM device contains one or two 32Gb memory stacks. Figure 3-35 shows the HBM sheet for a VU37P device which contains two memory stacks.

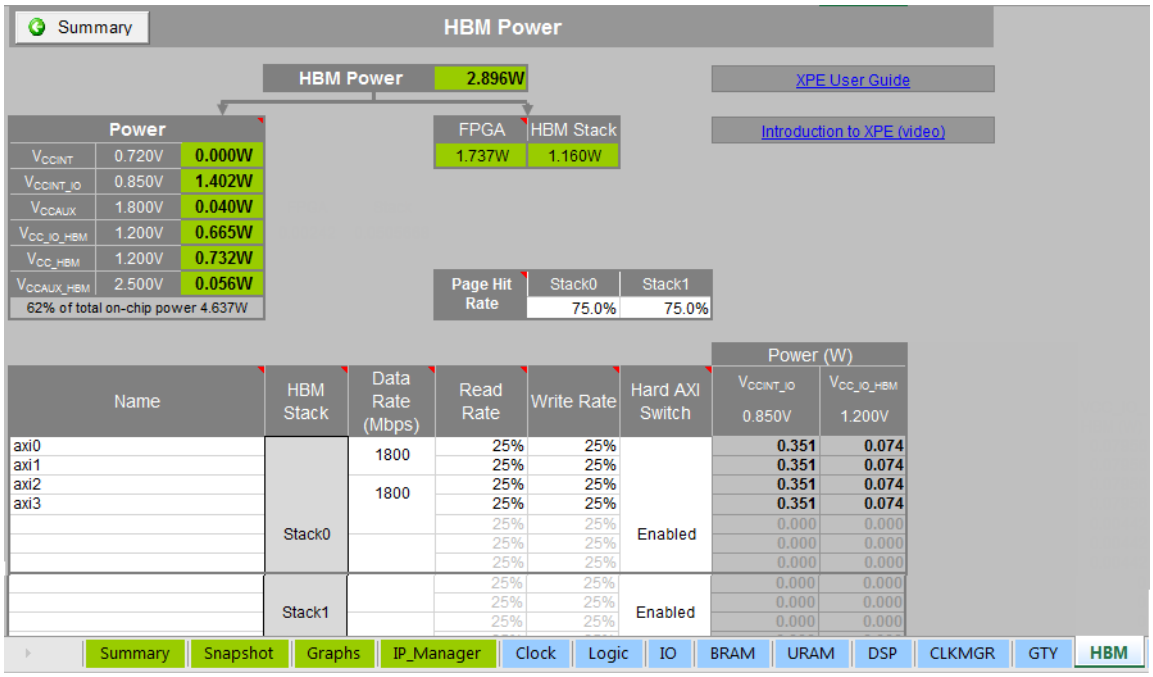


Figure 3-35: **HBM Sheet**

The HBM Power sheet displays the total dynamic power of the HBM subsystem. The total dynamic power can be further partitioned to show HBM stack power and the FPGA portion which is from the dedicated interface logic. The Power table shows the dynamic power dissipated on each power rail.

The Page Hit Rate is the estimated rate of a memory transaction to access an open page, which results in the fastest access. For example, sequential memory accesses are more likely to occur within an open page which reduces power and increases efficiency. The default value is 75% for each stack.

In the main table, each row represents a pseudo-channel associated with an AXI port capable of accessing a contiguous 2Gb section of an HBM stack. There are 16 pseudo-channels associated with Stack0 and 16 associated with Stack1. Each of the 16 psuedo-channels must access the HBM using one of the eight dedicated memory controllers and each memory controller simultaneously accesses two HBM 2Gb sections.

Data Rate for each memory controller is specified in Mbps. The valid range is from 100 to 1800 (1600 for -1 device speed). There can be different rates within the same stack. However, they must be integer multiples (if the rates are different). For example, if a stack has a memory controller with a rate of 1800, the next slower valid rate is 900 (1/2) and the next slower rate after that is 450 (1/4) and so on.

The default value of Read Rate and Write Rate are set to 25%. However, the maximum rate also depends on the page hit rate. The valid condition for Read and Write rates are as follows:

- If the Page Hit Rate is less than 75%, the (Read Rate + Write Rate) is less than or equal to 50%
- If the Page Hit Rate is greater than or equal to 75%, the (Read Rate + Write Rate) is less than or equal to 90%

If the above conditions are not met, the Read Rate and Write Rate cells are marked with a dark yellow warning color. However, the power numbers displayed are still valid.

HBM in Summary Sheet

Virtex UltraScale+ HBM total on-chip power is partitioned into two separate totals: one for the FPGA portion and another for the HBM Stack portion. The reason for this partition is that each portion has its own junction temperature. When targeting a device with HBM, this information appears in an additional summary located to the right side of the main summary sheet as shown in Figure 3-36.

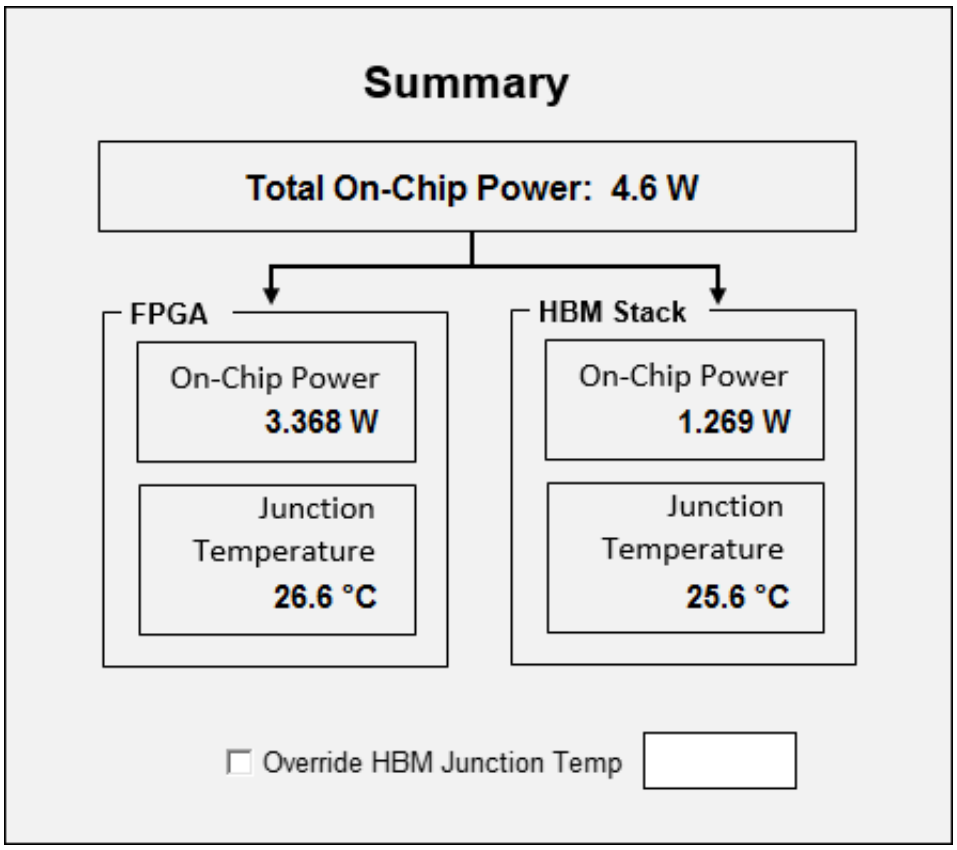
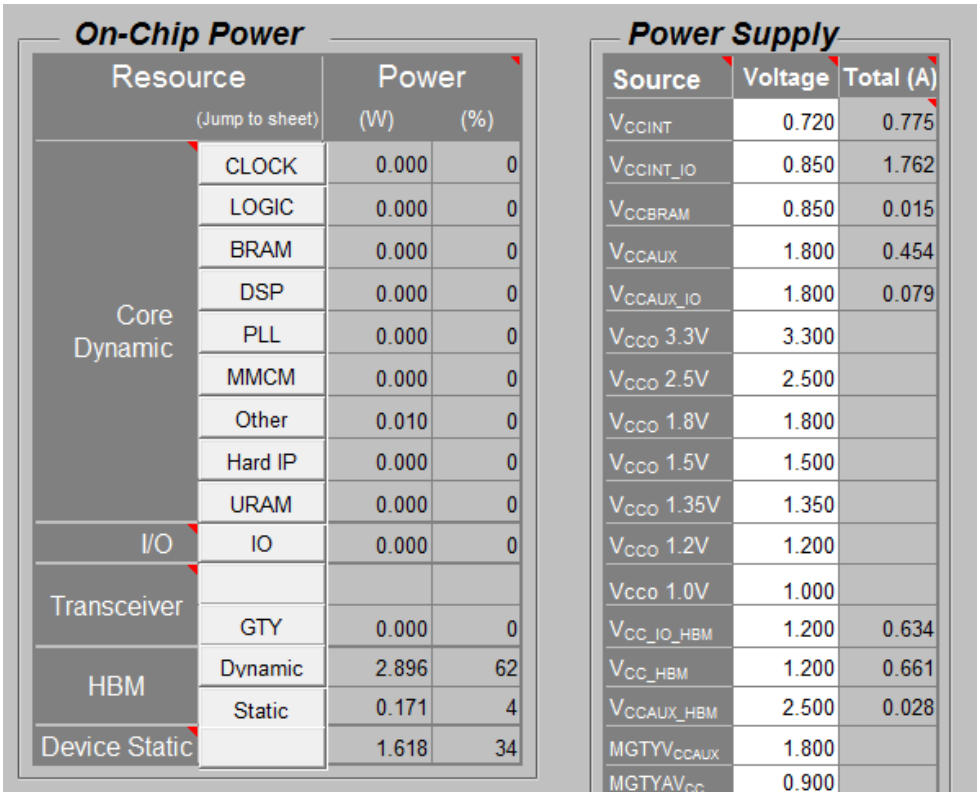


Figure 3-36: **HBM in Summary Sheet**

The summary contains the division between FPGA and HBM Stack for both On-Chip Power and Junction Temperature. The HBM Stack On-Chip Power is the HBM Static power (reported in the On-Chip Power table) and the HBM Dynamic power (the HBM portion of dynamic power shown on the HBM sheet). There is also the **Override HBM Junction Temperature** option, which works similar to the override option for the device. When you select this checkbox, the junction temperature can be entered in the box and it will be used to calculate the new HBM On-Chip Power. While it is possible to override only the HBM Junction Temperature, the HBM junction temperature is forced to be overridden if the device junction temperature is overridden in the Environment section.

The On-Chip Power table displays the totals for HBM Dynamic and Static power. The Dynamic power reported in the table is the same total reported on the HBM sheet which includes FPGA dedicated interface logic. The Power Supply table shows the current per HBM rail as shown in Figure 3-37.



On-Chip Power

Resource		Power (W)	(%)
Core Dynamic	CLOCK	0.000	0
	LOGIC	0.000	0
	BRAM	0.000	0
	DSP	0.000	0
	PLL	0.000	0
	MMCM	0.000	0
	Other	0.010	0
	Hard IP	0.000	0
	URAM	0.000	0
I/O	IO	0.000	0
Transceiver			
	GTY	0.000	0
HBM	Dynamic	2.896	62
	Static	0.171	4
Device Static		1.618	34

Power Supply

Source	Voltage	Total (A)
V_{CCINT}	0.720	0.775
V_{CCINT_IO}	0.850	1.762
V_{CCBRAM}	0.850	0.015
V_{CCAUX}	1.800	0.454
V_{CCAUX_IO}	1.800	0.079
V_{CCO} 3.3V	3.300	
V_{CCO} 2.5V	2.500	
V_{CCO} 1.8V	1.800	
V_{CCO} 1.5V	1.500	
V_{CCO} 1.35V	1.350	
V_{CCO} 1.2V	1.200	
Vcco 1.0V	1.000	
$V_{CC\_IO\_HBM}$	1.200	0.634
V_{CC_HBM}	1.200	0.661
V_{CCAUX_HBM}	2.500	0.028
$MGTYV_{CCAUX}$	1.800	
$MGTYAV_{CC}$	0.900	

Figure 3-37: **Power Supply Table**

Using Other Sheets (7 Series, Zynq-7000 SoC, UltraScale and UltraScale+ Devices)

The Other sheet allows you to calculate the power associated with these device features:

- **XADC** - (7 Series and Zynq-7000 SoC only) The XADC (Xilinx Analog-to-Digital Converter) is the basic building block that enables agile mixed signal functionality in Xilinx 7 series devices. The XADC includes a dual 12-bit, 1 Mega sample per second (MSPS) ADC and on-chip sensors.

  In the 7 series or Zynq-7000 SoC, the XADC can be powered down if unused, to save power. In the **XADC** table, set **Powered Down** to Yes if the XADC will be powered down by setting the power down bits in the device's configuration register or by disconnecting the V_{CCADC} supply. Set **Powered Down** to No if the XADC will not be powered down.

  XADC Clock (MHz) specifies the frequency of the DRP clock if your design uses the XADC. Leave this blank if your design does not instantiate the XADC or the XADC is powered down.

Note: Since XADC is on by default in the FPGA, the XADC power is reported as part of the static power.

The XADC is described in the *7 Series FPGAs and Zynq-7000 SoC XADC Dual 12-Bit 1MSPS Analog-to-Digital Converter User Guide* (UG480) [Ref 32].

- **SYSMON** - (UltraScale and UltraScale+ devices only) The System Monitor (SYSMON) monitors the UltraScale device physical environment using on-chip temperature and supply sensors, external analog inputs, and an integrated analog to digital converter (ADC).

  In the UltraScale device, the SYSMON can be powered down if unused, to save power. In the **SYSMON** table, set **Powered Down** to Yes if the SYSMON will be powered down by setting the power down bits in the device configuration register or by disconnecting the V_{CCADC} supply. Set **Powered Down** to No if the SYSMON will not be powered down.

  Note: Since SYSMON is on by default in the FPGA, the SYSMON power is reported as part of the static power

  Clock (MHz) specifies the frequency of the DRP clock if your design uses the SYSMON. Leave this blank if your design does not instantiate the SYSMON or the SYSMON is powered down.

  The SYSMON is described in the *UltraScale Architecture System Monitor Advance Specification User Guide* (UG580) [Ref 33].

- **Config** - The **Config** (Configuration) table allows you to specify these device configuration features:

  - **Readback CRC Clock (MHz)** - Xilinx 7 series and UltraScale devices include a feature to do continuous readback of configuration data in the background of a user design. This feature is aimed at simplifying detection of Single Event Upsets (SEUs) that cause a configuration memory bit to flip and can be used in conjunction with the FRAME ECC feature for advanced operations such as SEU corrections. In the **Config** table, enter the **ReadBack CRC Clock** frequency to include this feature in the XPE power estimate. Leave this blank if your design does not use the Readback CRC feature.

    Readback CRC is described in the *7 Series FPGAs Configuration User Guide* (UG470) [Ref 34] or the *UltraScale Architecture Configuration Advance Specification User Guide* (UG570) [Ref 35].

  - **Config Bank Voltage** - Specifies the setting of the Configuration Bank Voltage Select (CFGBVS), which determines the I/O voltage operating range and voltage tolerance for the configuration-related I/O banks in the device.

    Configuration bank voltage is described in the *7 Series FPGAs Configuration User Guide* (UG470) [Ref 34] or the *UltraScale Architecture Configuration Advance Specification User Guide* (UG570) [Ref 35].

- **PHASER** - Phaser blocks are available in 7 Series devices to simplify the interface with high-speed memory devices. For power estimation, these blocks are represented in a table on the Other sheet. Details for each Phaser setting are available in the *Zynq-7000 SoC and 7 Series Devices Memory Interface Solutions User Guide* (UG586) [Ref 10].

  In the Phaser table, the **Phaser INs** column is used to specify the number of PHASER_IN and PHASER_IN_PHY blocks used. Similarly, the **Phaser OUTs** column is used for both PHASER_OUT and PHASER_OUT_PHY blocks.

- **VCU** - A VCU (Video Codec Unit) exists in Zynq UltraScale+ EV devices. The VCU provides a multi-standard video encoding and decoding capabilities, that includes High Efficiency Video Coding (HEVC), i.e., H.265, and Advanced Video Coding (AVC), i.e., H.264 standards. The VCU is capable of simultaneous encoding and decoding video streams at rates up to 4Kx2K at 60 frames per second (fps).

  Other Sheet in XPE populates the VCU parameters to estimate the power consumption for the VCU block. This VCU estimator will be available when you select any Zynq UltraScale+ EV device in Summary sheet. Figure 3-38 displays the VCU power estimator.

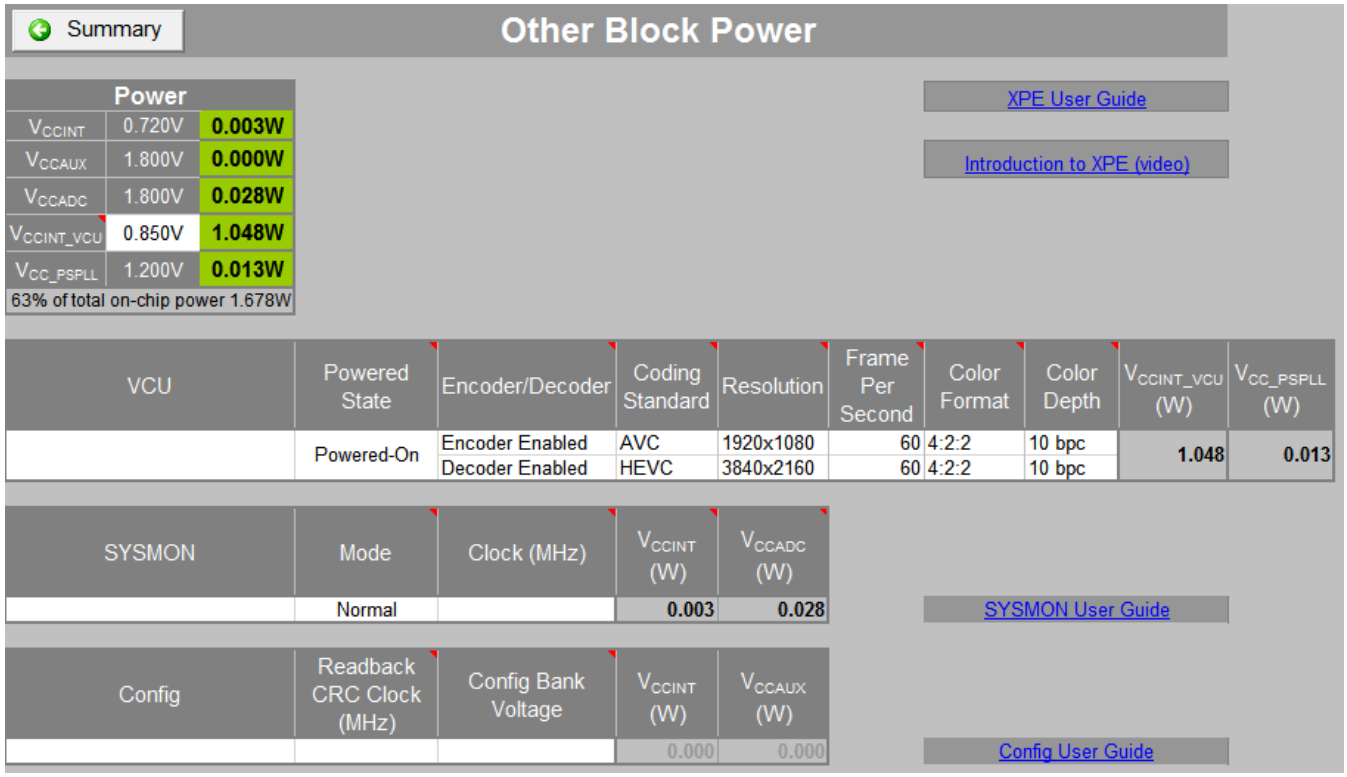


Figure 3-38: **Other Sheet Displaying the VCU Power Estimator**

Power can be estimated for any combination of VCU configurations by selecting the desired parameters for the VCU block for encoder and decoder such as resolution, color format etc.

Note: On the Other Sheet, the reported VCU power is only the dynamic power while on the Summary Sheet, the VCU power will include both the static and dynamic powers.

- **eFUSE** - UltraScale and UltraScale+ devices have a security feature that allows you to program eFUSE bits that force the FPGA device to only allow encrypted bitstreams during runtime. XPE allows you to estimate the power consumption when the eFUSE is being programmed during runtime. You must analyze the power in conjunction with the eFUSE programming image.

  Note: This power will not be considered for On-Chip Power computation since it is specific to eFUSE state, where the application design is not programmed.

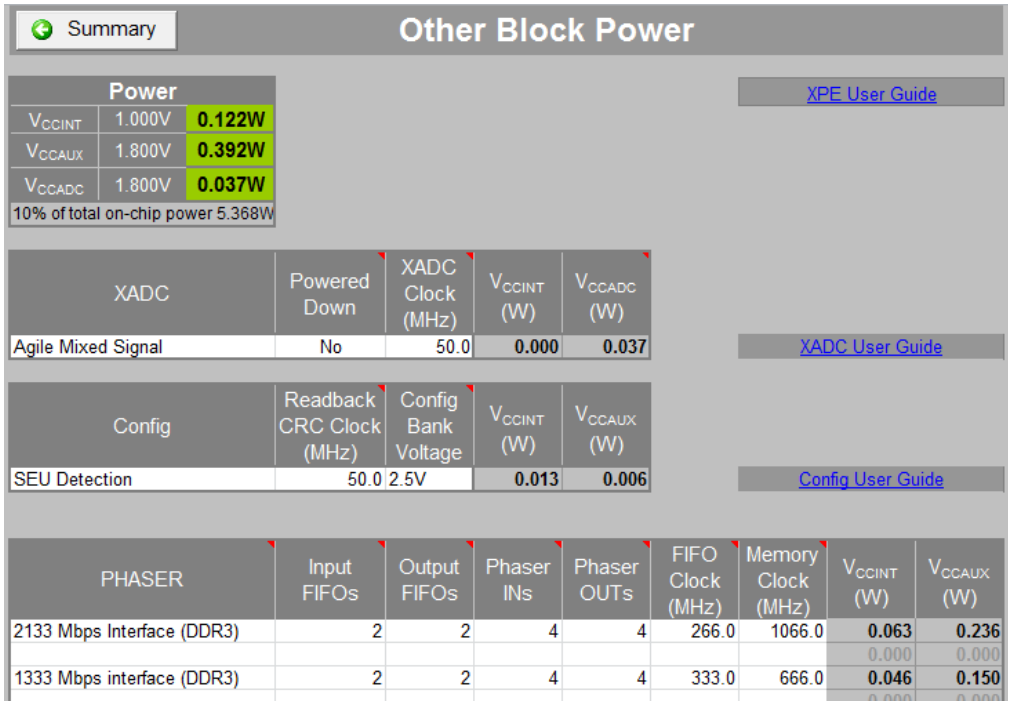


Figure 3-39: **Other Power Sheet (7 Series Devices)**

User Sheet

This sheet is intentionally left blank and user editable. On this sheet you can provide any documentation (text, image, or hyperlinks), details about the project, assumed conditions, or collect the results important to your application.

XILINX

Chapter 4

Exchanging Power Information

Overview

To determine device power supply requirements and estimate thermal dissipation throughout the design process, data exchange mechanisms are available between different power estimation tools such as Xilinx Power Estimator (XPE) and XPower Analyzer (XPA) in the ISE Design Suite and the Vivado® power analysis feature in the Vivado Design Suite. Details on the methodology and user flow are presented in the *Power Methodology Guide* (UG786) [Ref 3].

Exporting Settings from XPE to XPower Analyzer

In a typical development process you will first perform power estimation in XPE to size the voltage supply sources, evaluate thermal power dissipation paths, and allocate the total power budget to the different blocks in the FPGA system. Later in the development cycle you will want to perform post implementation power analysis in XPower Analyzer to validate against your power and thermal goals. Instead of manually re-entering this environmental data into XPA you can export to a file and have XPA read it for your next analysis. This process exports all environment, thermal, and voltage settings which in turn helps getting realistic power estimations in XPA that can easily be compared between the two tools.

For step-by-step export instructions, see Exporting XPE Results.

Importing Results from XPower Analyzer

This flow is useful in the following cases:

- The power reported in XPower Analyzer exceeds your requirements and you want to evaluate different scenarios, adjusting resources used, count, and configuration. You can also estimate power gains from techniques such as logic gating or resource time sharing, without modifying your code.
- Your project uses (or reuses) IP blocks already implemented in a previous design or acquired. You can import these existing blocks into XPE to quickly get resource and power usage for these blocks. You can then focus your efforts in XPE to enter data for the new pieces of logic not yet defined.
- Team-based design – A project manager can regularly monitor power for the entire design by integrating resource usage and power consumption for modules developed by the different teams.

For step-by-step import instructions, see Importing Data into XPE.

Importing Results from Vivado Power Analysis

A data exchange mechanism is available to import data from the Vivado® power analysis feature into Xilinx Power Estimator (XPE). This data exchange mechanism is available for the 7 Series, UltraScale and UltraScale+ device families.

This flow is useful in the following cases:

- The power reported in the Vivado Design Suite exceeds your requirements and you want to evaluate different scenarios, adjusting resources used, count, and configuration. You can also estimate power gains from techniques such as logic gating or resource time sharing, without modifying your code.
- Your project uses (or reuses) IP blocks already implemented in a previous design or acquired. You can import these existing blocks into XPE to quickly get resource and power usage for these blocks. You can then focus your efforts in XPE to enter data for the new pieces of logic not yet defined.
- Team-based design - A project manager can regularly monitor power for the entire design by integrating resource usage and power consumption for modules developed by the different teams.

For step-by-step import instructions, see Importing Data into XPE.

Importing and Exporting the Data

Depending on the stage in the device development cycle your design is in, XPE provides multiple mechanisms to simplify data entry and manage output data. These mechanisms use the data import and data export features of XPE.

The XPE import and export features are useful for exchanging power information with XPower Analyzer in the ISE® Design Suite and the Vivado® power analysis feature in the Vivado Design Suite (see Importing Results from Vivado Power Analysis).

For step-by step instructions for importing or exporting XPE data, see the following sections:

- Importing Data into XPE
- Exporting XPE Results

Importing Data into XPE

In the Summary sheet, click the **Import** button to open the dialog box shown in Figure 4-1. This dialog box varies slightly depending on device architecture, because newer family spreadsheets offer more import capabilities.

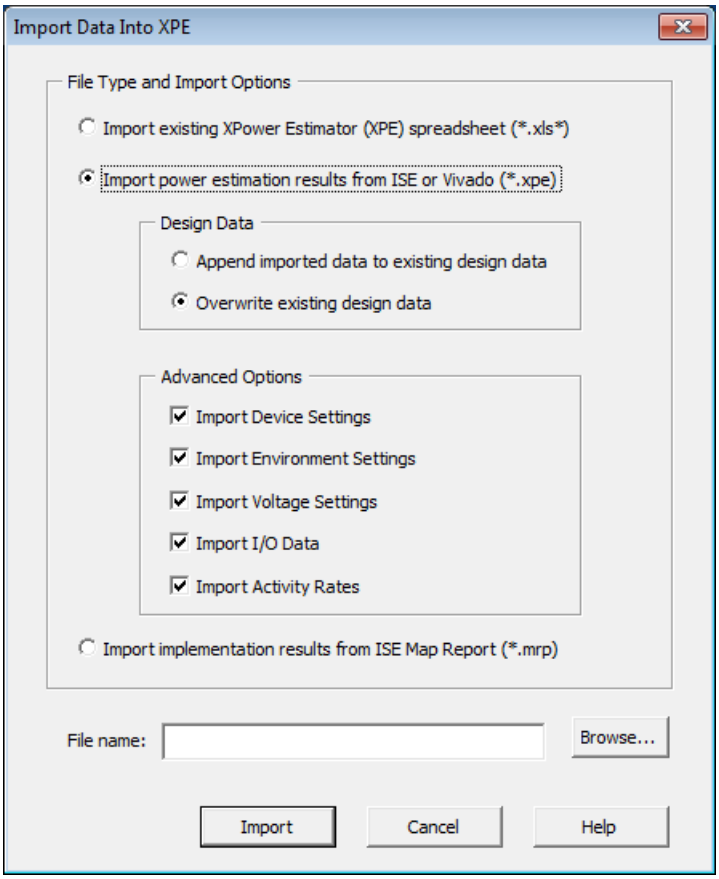


Figure 4-1: **Import Dialog Box (7 Series Devices)**

This dialog box lets you select among the following import options:

Importing the Existing Xilinx Power Estimator spreadsheet (*.xls)

Use this option to import an existing XPE spreadsheet (.xls or .xlsm file). This option is useful when starting a new design which reuses previous IP blocks or when updating the design information into the latest spreadsheet version. This action deletes all data in the current spreadsheet, then imports all data from the selected spreadsheet.

IMPORTANT: *When the import is complete, make sure to verify and adjust the imported data where appropriate. For example, adjust utilization and resources count columns when porting a design to a new architecture.*

Importing the Power Estimation Results from ISE or Vivado (*.xpe)

Use this option to further analyze your design by importing complete designs or IP blocks. The .xpe file you are importing can come from either the ISE Design Suite or the Vivado Design Suite.

- ISE Design Suite - The .xpe file was produced by XPower Analyzer or by the `-xpe` option to the `xpwr` command. See the XPower Analyzer Help or the description of the `-xpe` option to the `xpwr` command line in the *Command Line Tools User Guide* (UG628) [Ref 4] for details on how to generate this interoperability file.
- Vivado Design Suite - The .xpe file was produced by the report power tool or by the `-xpe` option to the `report_power` Tcl command. See the *Vivado User Guide: Power Analysis and Optimization* (UG907) [Ref 2] for details on how to generate this interoperability file.

To determine device power supply requirements and estimate thermal dissipation throughout the design process, data exchange mechanisms are available between the different power estimation tools, Xilinx Power Estimator (XPE) and XPower Analyzer (XPA), which is in the ISE Design Suite. Details on the methodology and user flow are presented in the *Power Methodology Guide* (UG786) [Ref 3]. This data exchange mechanism is available for the Spartan-6, Virtex-6, Artix-7, Kintex-7, and Virtex-7 device families.

Benefits and use model for this flow are presented in Importing Results from Vivado Power Analysis.

To import this data into the spreadsheet:

a. In the Summary sheet of the XPE spreadsheet, click **Import File**.

b. In the import dialog box, browse and select the `.xpe` file to import.

c. (Optional. 7 series devices and Zynq-7000 SoC only) In the **Design Data** section of the dialog box, select whether you want the imported data to override any previously entered data in the spreadsheet or rather append to the existing results.

d. (Optional. 7 series devices and Zynq-7000 SoC only) In the **Advanced Options** section of the dialog box, specify data to include during the import (**Device Settings**, **Environment Settings**, **Voltage Settings**, **I/O Data**, and **Activity Rates**).

Note: When you import an XPE file from the Vivado Design Suite, the Optimization field in the Implementation section of the Summary Tab is always set to Default for 7 series devices and to None for UltraScale devices. XPE does not apply any default optimization to imported data, because it assumes that the data from Vivado Report Power already accounts for power optimization from the Vivado Design Suite design flow.

TIP: *When you import a .xpe file from the Vivado Design Suite, the imported data will be displayed hierarchically in the Logic sheet, BRAM sheet, and DSP sheet. In Figure 4-2, the* ***Name*** *column in the Logic sheet contains a row for a parent module followed by rows with names indented, representing modules within the parent.*

Name	Clock (MHz)	LUTs as			Registers
		Logic	Shift Registers	Distributed RAMs	
top/cpuEngine					
Combinatorial	2.8	6595	0	0	0
clkgen/cpuClk	50.0	0	0	0	3333
clkgen/wbClk	50.0	0	0	0	506
top/fftEngine					
Combinatorial	24.2	1797	0	0	0
clkgen/fftClk	100.0	0	1	0	1339
clkgen/wbClk	50.0	0	0	0	116
top/mgtEngine					
Combinatorial	13.2	403	0	0	0
clkgen/wbClk	50.0	0	0	0	66
mgtEngine/gt_usrclk_source/GT0_TXUSR	78.1	0	0	0	144
mgtEngine/gt_usrclk_source/GT2_TXUSR	78.1	0	0	0	144
mgtEngine/gt_usrclk_source/GT4_TXUSR	78.1	0	0	0	144
mgtEngine/gt_usrclk_source/GT6_TXUSR	78.1	0	0	0	144

Figure 4-2: **Hierarchical Display of Imported Data (7 Series Devices)**

Importing Implementation Results from ISE Map Report (*.mrp)

Select the import from Map Report (.mrp file) when portions of the design have been implemented in the ISE Design Suite. You can import the exact resource count from a Map Report to get a more accurate power estimation after the design is placed. This flow is also used when portions of the design are implemented while others are still being designed, so you can add details for the expected remaining logic and evaluate the total design power distribution.

Note: This process overwrites any utilization data, but preserves environment settings.

XPE Import: Project, Confidence Level and Date

When you import .xpe file generated by Vivado in UltraScale+ XPE, it will contain the following information about the Vivado design:

1. **Project**: This field will list the following information:
   a. Top level design name of the Vivado project.
   b. Vivado version for which design is implemented.
2. **Confidence level**: This field will contain following information:

a. Confidence about the accuracy of power report. It will be Medium for the vector-less power estimation and High for the SAIF based power estimation.

b. Design state which will be either Synthesized or Implemented. This is the state of design on which the power report was generated.

3. **Last updated**: Displays the date of report power generation in Vivado.

Figure 4-3 from XPE shows the above import information.

Figure 4-3: **Import Information of Vivado Project in XPE**

Importing the Power Estimation Results (*.xpe) for UltraScale+ XPE - Power Optimization Status

For UltraScale+ XPE, when you import a .xpe file from the Vivado Design Suite, the optimization field in the Implementation section of the Summary Tab will display the following status details of power optimization:

- **Imported-BramPwrOpt**: Indicates that only the `bram_power_opt` from `opt_design` was run for the imported data.
- **Imported-NoPwrOpt**: Indicates that the `power_opt` was not run in Vivado for the imported data (not even in `opt_design`). This may occur if the `bram_power_opt` is explicitly skipped in the implementation run.
- **Imported-FullPwrOpt**: Indicates that the `power_opt_design` was run at some stage in the implementation run for the imported data.

Figure 4-4 displays the status details of power optimization.

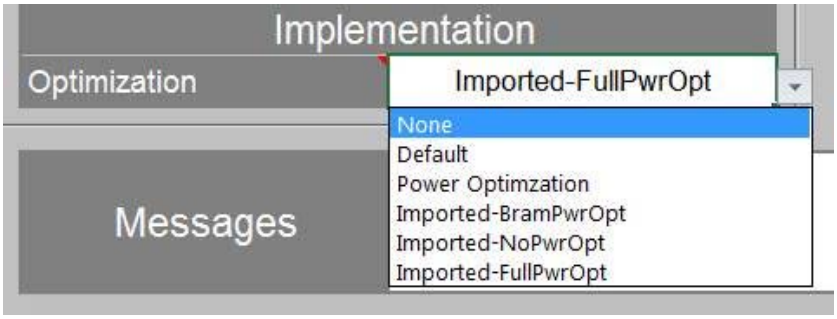


Figure 4-4: **UltraScale+ XPE - Importing Power Estimation (*.xpe) Results**

TIP: *In 7 Series and earlier device families, you will notice resources used are grouped into a minimum set of lines after import. The map report only contains the counts of the various blocks and you will*

need to set the bit width, data rate, clock, mode, enable, and other configurations on each XPE sheet to match your design.

TIP: *The I/O and BRAM sheets are populated based on unique configuration. I/Os are grouped by bus and all BRAMs with the same configuration appear on a single line. If needed, you might therefore need to add additional rows and adjust the counts to group by clock domain, module, or functionality.*

Exporting XPE Results

In the Summary sheet click the **Export File** button to open the dialog box shown in Figure 4-5.

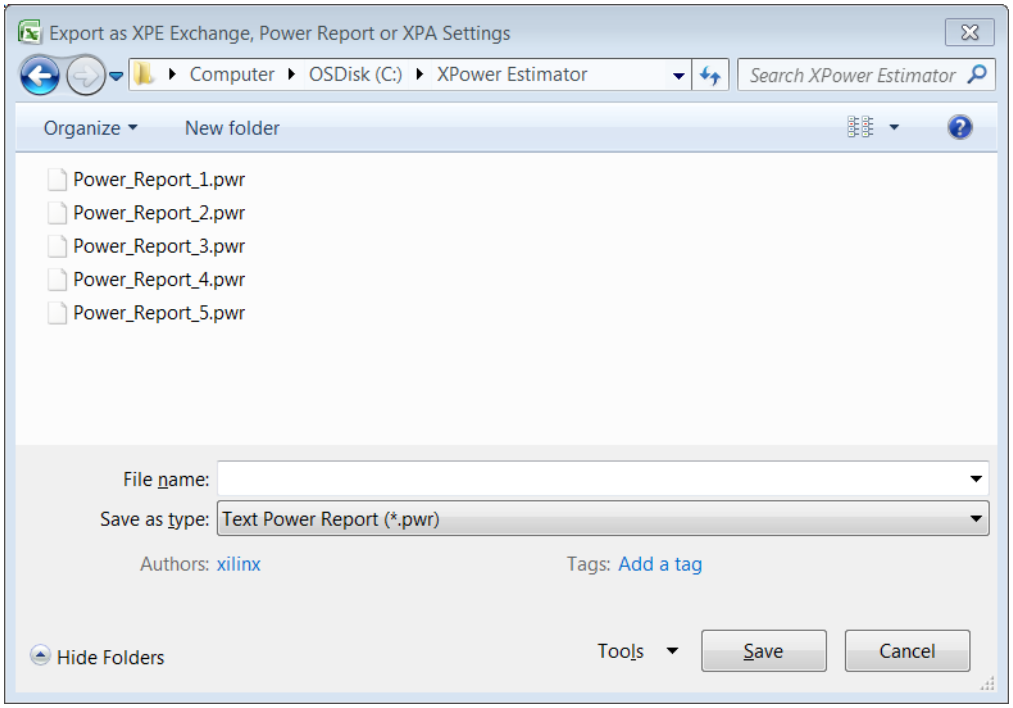


Figure 4-5: **Export Dialog Box**

In the dialog box the **Save as type** field lets you select among the following data formats:

Exporting as XPA Settings (*.xpa) File

Use this format to export XPE settings so they can then be applied to an XPower Analyzer session. This tool is typically used later in the design cycle when you are ready to perform a post place and route power analysis. The tool will create an .xpa file which contains all the environment settings, such as thermal, board and voltage properties.

This simplifies the analysis setup in XPower Analyzer and ensures power data can be compared between the two tools.

Note: To read the data exported from XPE into XPower Analyzer, enter the **Settings file** name (`*.xpa`) in the dialog box that appears when you open a design in XPower Analyzer (**File > Open Design**).

Note: In the XPower (XPWR) command line tool, which performs a power analysis on your design within the ISE Design Suite, use the `-x <file_name>` switch to read in the XPE exported data.

Exporting as Text Power Report (*.pwr)

Use this format to export XPE Summary sheet results in a text format. XPE will save all the information on the Summary sheet in a sequence of tables so the information is easy to read. This feature can be used to archive or compare multiple scenarios. It can also help if your design flow uses scripts to parse and use XPE results.

Exporting as an XPE Exchange (*.xpe) File

Use this format to export the contents of an XPE spreadsheet in a smaller file, and then restore it by importing it into another spreadsheet.

Exporting as an XDC Constraint (*.xdc) File

Use this format to export the XPE environmental, thermal and budget information in the form of Xilinx design constrains (*.xdc) file using the `set_operating_condition` commands. The exported file can be sourced in Vivado project to get the same design constraints as set in XPE. This file will have the following information in the exported *.xdc file:

1. Device Process
2. Junction Temperature
3. Ambient Temperature
4. Airflow
5. Heat sink
6. Board and board layers
7. Voltage
8. Design Power Budget

Note: *.xdc export feature is available only for UltraScale+ XPE spreadsheet.

Automating XPE

Overview

To simplify data entry and export or to assist with data manipulation Microsoft Excel offers a variety of mechanisms which you can use to increase your productivity or the breadth of your power estimation and analysis. The following section provides reference material and examples to help you get started quickly with Excel internal automation features and interface with some of the most common external scripting languages.

Using Named Cells

Excel provides a mechanism to name a cell or a range of cells so these can be used within formulae or scripts without referring to them as cell XY coordinates. Because the XPE spreadsheet is protected you cannot see 'named' cells defined on the protected areas. You can however name cells in the unprotected area (User sheet). The following tables and examples show the named cells within XPE that are available to facilitate user formulas and scripting.

Get Available Resource Counts

The following named cells represent the maximum available resources available for the considered device and package. None of these cells are visible in the spreadsheet, however you can use these read only values in your calculations.

Table 5-1: **Resource Counts - Named Cells**

Resource	Named Cells	Description
LUTs	NUM_LUTS	Includes all LUTS
	NUM_LUTRAM	Shift Registers and Distributed Memories LUTs
Registers	NUM_FFS	
DSP blocks	NUM_DSPS	
Block RAMs	NUM_BRAMS	
PLLs	NUM_PLLS	

Table 5-1: **Resource Counts - Named Cells *(Cont'd)***

Resource	Named Cells	Description
MMCMs	NUM_MMCMS	
DCMs	NUM_DCMS	
Transceivers	NUM_GTPS	Lowest speed blocks
	NUM_GTS	Lower speed blocks
	NUM_GTHS	High Speed blocks
	NUM_GTZS	Highest Speed blocks

Examples:

Formulas to quickly set device utilization and evaluate thermal effects when varying device, package or cooling parameters:

= INT(NUM_LUTS * 0.75) Sets total LUT utilization to 75% of device capacity (if entered on the Logic sheet)

= INT(NUM_DSPS * 0.90) Sets DSP block utilization to 90% of device capacity (if entered in DSP sheet)

Get Device Operating Limits

The following named cells represent operating limits for the considered device, package, speed grade and temperature grade. None of these cells are visible in the spreadsheet however you can use these read only values in your calculations.

Table 5-2: **Operating Limits - Named Cells**

Resource	Named Cells	Description
Temperature	TJ_MAX	Maximum operating junction temperature (°C)
	TJ_MIN	Minimum operating junction temperature (°C)
Voltages	VCC_MAX	Maximum operating V_{CCINT} voltage (V)
	VCC_MIN	Minimum operating V_{CCINT} voltage (V)
Transceivers	GTP_MAXRATE	Maximum data rate of lowest speed blocks (Gb/s)
	GTX_MAXRATE	Maximum data rate of lower speed blocks (Gb/s)
	GTH_MAXRATE	Maximum data rate of high speed blocks (Gb/s)
	GTZ_MAXRATE	Maximum data rate of highest speed blocks (Gb/s)

Example:

Formula to enter into the user **Junction Temperature** cell on the Summary sheet to force the device junction temperature to the maximum allowed while evaluating different temperature or device and package combination:

```
= TJ_MAX
```

Get and Edit Summary Information

Many cells in the Summary sheet or tables at the top of the other sheets are named. To find these names in Excel you can select the cell then if it is named the 'name box' area of the formula bar will show that name. The following paragraph highlights some of the most commonly used cells on the Summary sheet.

Table 5-3: **Summary Panel - Named Cells (See Figure 1-8)**

Named Cell	Description
JUNCTION_TEMP	Estimated or forced **Junction Temperature** (°C)
THERMAL_MARGIN_C	Temperature margin for the device temperature grade (°C)
TJA	Estimated or specified **Effective ΘJA** (°C/W)
TOTAL_POWER	**Total On-Chip Power** (W)
THERMAL_MARGIN_W	Power margin for the device temperature grade (W)
OFFCHIP_POWER	Total power supplied to off-chip devices (W)

Table 5-4: **On-Chip Power Panel - Named Cells (See Figure 1-6)**

Named Cell	Description
CLOCK_POWER	Clock tree power (W)
LOGIC_POWER	CLB Logic power (W)
BRAM_POWER	Block RAM power
DSP_POWER	DSP blocks power (W)
PLL_POWER	PLL blocks power (W)
MMCM_POWER	MMCM blocks power (W)
PHASER_POWER	PHASER blocks power (W)
PCIE_POWER	PCIE blocks power (W)
IO_POWER	SelectIO blocks power (W)
GTP_POWER	Lowest speed transceiver blocks power (W)
GTX_POWER	Lower speed transceiver blocks power (W)
GTH_POWER	High speed transceiver blocks power (W)
GTZ_POWER	Highest speed transceiver blocks power (W)
STATIC_POWER	Device static power (W)
PS_POWER	Zynq-7000 SoC/Zynq-7000 MPSoC processing system (PS) power (W)
PS_STATIC	Zynq-7000 SoC/Zynq-7000 MPSoC PS device static power (W)
PL_STATIC	Zynq-7000 SoC/Zynq-7000 MPSoC programmable logic (PL) device static power (W)

Table 5-5: **Power Supply Panel - Named Cells**

Named Cell	Description
VCCINT	V_{CCINT} core voltage level (V)
VCCBRAM	V_{CCBRAM} voltage level (V)
VCCAUX	V_{CCAUX} voltage level (V)
VCCAUX_IO	V_{CCAUX_IO} voltage level (V)
VCCO33	V_{CCO} 3.3V voltage level (V)
VCCO25	V_{CCO} 2.5V voltage level (V)
VCCO18	V_{CCO} 1.8V voltage level (V)
VCCO15	V_{CCO} 1.5V voltage level (V)
VCCO135	V_{CCO} 1.35V voltage level (V)
VCCO12	V_{CCO} 1.2V voltage level (V)

Table 5-6: **Environment Table - Named Cells**

Named Cell	Description
AMBIENT_TEMP	Ambient temperature (°C)
BOARD_TEMP	Board temperature (°C)
CUSTOMTSA	User specified Theta SA thermal resistance (°C/W)
CUSTOMTJB	User specified Theta JB thermal resistance (°C/W)

Table 5-7: **Miscellaneous Named Cells**

Named Cell	Description
PROJECT	User description of the spreadsheet
VERSION	Spreadsheet revision
RELEASE_DATE	Spreadsheet release date

Using Formulas

With Excel formulas you can simplify data entry, spreadsheet parameterization or create customer reports as explained in the following examples

Example1: Set clock frequency of all attached synchronous loads in a single place.

Typically a clock net may reach multiple types of resources. Instead of entering the clock frequency on each sheet the following formula can be used on the resource sheets while the clock frequency is only defined once in the Clock sheet. Any change of the clock frequency would immediately be reflected on all the linked resource sheets

```
=CLOCK!E19
```

Example 2: Calculate the fanout sum of all the different loads driven by a clock.

On the clock sheet you might find it useful to enter formulas similar to:

```
=SUM(LOGIC!G12:I12,BRAM!E10,DSP!E8)
=SUM(IO!I19:K19)
```

Example 3: Select the GTX data rates to the PCIe interface speed and number of lanes. Entering the following formulae for GTX line rate and number of channels will track the PCIe interface.

- Set channel data rate based on the PCIE bock configuration (if entered on the GTX sheet)

  ```
  =IF(PCIE!E8="GEN3",8,IF(PCIE!E8="GEN2",5,2.5))
  ```

- Set the number of GTX channels to reflect the number of PCIE lanes (if entered on the GTX sheet)

  ```
  =PCIE!G8
  ```

Example 4: Parameterize the spreadsheet entry using formulas and the **User** sheet. Figure 5-1 illustrates how to evaluate power when a module is replicated more or fewer times in the design. By varying the number of instances, the quantity of resources for the base blocks, or clock frequency, an Excel formula can automatically recalculate the values which need to be entered in other sheets. In Figure 5-1, the value for **Number instance** (named **num_inst**) in the **User** sheet automatically calculates utilization and activity for cells that appear in the **Logic** sheet.

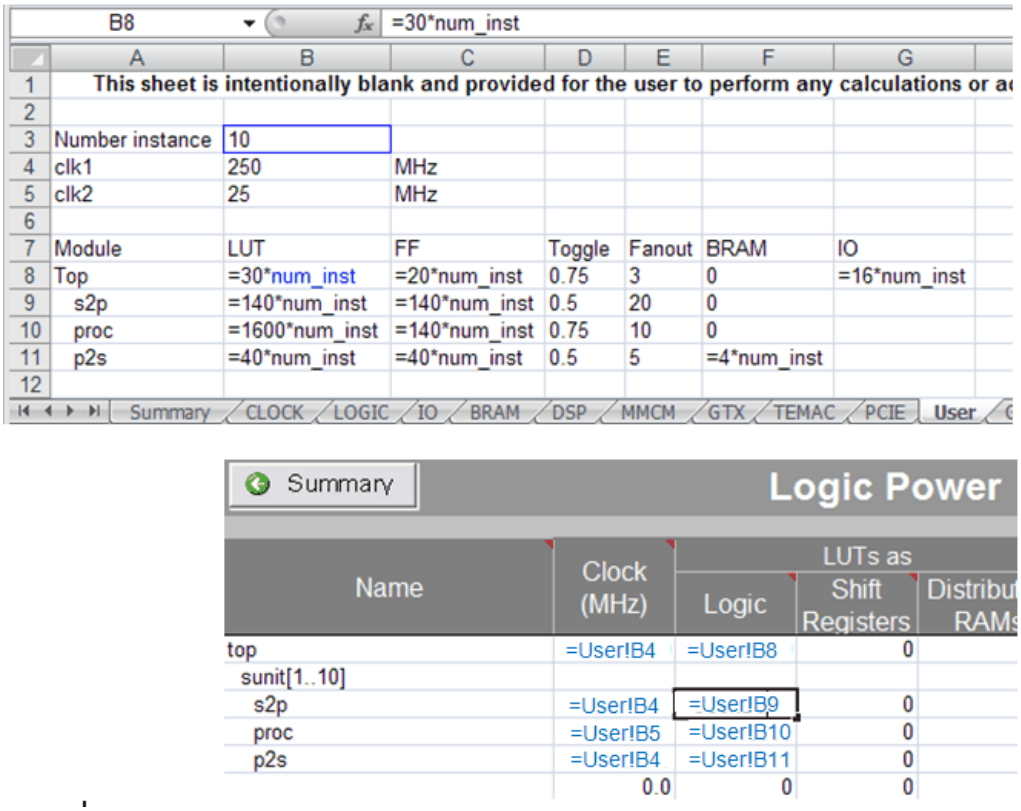


Figure 5-1: **Parameterizing Data Entry Using Formulas on the User Sheet**

Using Visual Basic Macros

The following examples define the public Visual Basic functions defined in the Xilinx 7 series XPE spreadsheet to help you with your automation needs. They provide convenient ways to load files, create power reports, change parts, packages and environment settings from Excel or another program.

- Create a text power report and save with name specified as argument.

```
Public Sub GeneratePowerReportFile(FileName As String)
```

- Create a settings file and save with name specified as argument. This file can later be used in XPower Analyzer.

```
Public Sub GenerateXPAFile(FileName As String)
```

- Create an XPE file and save with name specified as argument. This file can later be used to restore the current settings in XPE.

```
Public Sub GenerateXPEFile(FileName As String)
```

- Import an existing XPE spreadsheet (.xls* path/file specified as argument).

```
Public Sub ImportXPEFile(path As String)
```

- Import a place and route map report (.mrp path/file specified as argument).

```
Public Sub ImportMapReportFile(FileName As String)
```

- Import implementation results in .xpe format. Review the Import dialog options for details and format of the different arguments.

```
Public Sub ImportXmlFile(FileName As String, Append As Boolean,DevSettings As
Boolean, EnvSettings As Boolean, VoltSettings As Boolean, IOSettings As Boolean)
```

- Take a snapshot of the currently loaded Power information or load a snapshot of another XPE spreadsheet:

```
Public Sub TakeSnapshot( FileName As String)
```

  Usage: Pass an empty string for `FileName` to take a snapshot of the active workbook, or pass the file name of another workbook that will import as a snapshot.

  Example:

```
TakeSnapshot("")
```

- Set device information and check whether the device is valid. Returns True if valid device or False if not valid.

```
Public Function SetDeviceInfo(Device As String, Package As String, TempGrade As
String, SpeedGrade As String) As Boolean
```

  Example:

```
SetDeviceInfo("XC7K325T", "FBG900", "Industrial", "-1")
```

- Read resource utilization % of the specific resource by a pre-defined name passed as `ResourceName`.

```
Public Function GetUtilization(ResourceName As String) As Double
```

  Sample values for `ResourceName`: `CLOCK_GLOBAL`, `CLOCK_REGIONAL`, `LOGIC_LUTS`, `IO_TOTAL`, `IO_HP`, `BRAM_RAMB18`

  Example:

```
my $util = $Book->GetUtilization("BRAM_RAMB18");
```

  This returns the value of the BRAM sheet RAMB18 utilization, e.g. 75 for 75%.

- Set the default voltages for all supply voltages. Set argument to False for Nominal voltages and to True for Maximum voltage levels.

```
Public Sub SetDefaultVoltages(Maximum As Boolean)
```

- Set the **Device** field on the Summary sheet (will automatically adjust the **Family** field if required).

```
Public Function SetDevice(Device As String) As Boolean
```

- Set the **Package** field on the Summary sheet.

```
Public Function SetPackage(Package As String) As Boolean
```

- Set the **Process** field on the Summary sheet. Set argument to False for Typical process and True for Maximum process.

```
Public Sub SetProcess (Maximum As Boolean)
```

- Set the **Temp Grade** field on the Summary sheet. Options are "Commercial", "Industrial", "Q-Grade", "Extended", and so forth.

```
Public Function SetTemperatureGrade (Grade as String) as Boolean
```

- Set the **Speed Grade** field on the Summary sheet. Options are "-1", "-1L", and so forth.

```
Public Function SetSpeedGrade (Grade as String) as Boolean
```

- Set the **Heat Sink** field on the Summary sheet. Options are "Custom", "None", "Low Profile"

```
Public Function SetHeatSink (Sink as String) as Boolean
```

- Set the **Board Selection** field on the Summary sheet. Options are "Custom", "JEDEC", "Small", "Medium", "Large".

```
Public Function SetBoard (BoardSize as String, BoardLayers as Integer) as Boolean
```

- Set the **User Override** for the **Junction Temperature**, and value.

```
Public Function SetJunctionTemperature(Temperature As Double, OverRide As
Boolean) As Boolean
```

- Set the **User Override** for the **Effective ThetaJA**, and value.

```
Public Function SetEffectiveThetaJA(ThetaJA As Double, OverRide As Boolean) As
Boolean
```

Scripting XPE

Microsoft Excel capabilities described in the previous paragraphs can be accessed from any framework with access to the COM interface. This Component Object Model (COM) is a binary interface standard for software that enable interprocess communications in a large range of programming languages (for example, Visual Basic, Perl, Java). The following examples illustrate how you can set XPE environment parameters, run calculations and read or export results from different languages.

Visual Basic Scripting Example

This simple example opens XPE, then export results into a text power report using the Visual Basic scripting language.

```
Dim XPE As Workbook
XPEfilename = "C:\\Power\\7_Series_XPE_13_1.xls"

On Error Resume Next
Set XPE = Workbooks(XPEfilename)
   ' Opening XPE
   On Error GoTo 0
   If (XPE Is Nothing) Then
      Set XPE = Application.Workbooks.Open(XPEfilename, UpdateLinks:=vbFalse,
ReadOnly:=vbTrue)
      If XPE Is Nothing Then ' Open failed
         MsgBox ("XPE Open Failed: " & XPEfilename & "Err=" & Err)
         Exit Function
      End If
   End If
' Set Vccint voltage
XPE.Sheets("Summary").Range("VCCINT").Value = myVccint
TotalPower = XPE.Sheets("Summary").Range("TOTAL_POWER").Value
' Export XPE results into a text power report
XPESub = "'" & XPE.Name & "'!" & "ThisWorkBook.GeneratePowerReportFile"
Application.Run(XPESub, FileName)
```

Perl Scripting Example

This simple example opens XPE then export results into a text power report using Perl scripting language.

```
use Win32::OLE;
use Win32::OLE::Const 'Microsoft Excel';

my $myXPEfilename = "C:\\Power\\7_Series_XPE_13_1.xls";

# Opening XPE
my $Excel = Win32::OLE->GetActiveObject('Excel.Application')
    || Win32::OLE->new('Excel.Application', 'Quit');
my $Book = $Excel->Workbooks->Open($myXPEfilename);
# Export XPE results into a text power report
$Excel->Run("ThisWorkBook.GeneratePowerReportFile", "$path/${design}.pwr");
```

XILINX

Chapter 6

Using Snapshots and Graph Sheets

Using the Power Comparison Snapshots Sheet

The Power Comparison Snapshots sheet allows you to capture a series of snapshots of the power status of your design under varying conditions or at different points in its design cycle. Each snapshot displays device part, environmental information, the power consumed by your design, and the voltage and current across each of the power supply sources used in the design. You can use the Power Comparison Snapshots sheet to compare the power consumed under different conditions and the power calculated at different points in the design cycle.

Summary | Snapshot | Import | Clear All — **Power Comparison Snapshots**

Settings	Baseline	Power Optimization	Voltage ID	Industrial	2L	0.97V
	XPE: 2014.1	XPE: 2014.1	XPE: 2014.1	XPE: 2014.1	XPE: 2014.1	XPE: 2014.1
	2 Apr 2014 @ 17:04	2 Apr 2014 @ 17:05	2 Apr 2014 @ 17:10	2 Apr 2014 @ 17:12	2 Apr 2014 @ 17:16	2 Apr 2014 @ 17:23
	7_Series_XPE_2014_1_fc	7_Series_XPE_2014_1_f	7_Series_XPE_2014_1_f	7_Series_XPE_2014_1_f	7_Series_XPE_2014_1_f	7_Series_XPE_2014_1_f
Part	XC7VX330TFFG1761-1C	XC7VX330TFFG1761-1C	XC7VX330TFFG1761-1C	XC7VX330TFFG1761-1I	XC7VX330TFFG1761-2LI	XC7VX330TFFG1761-2LI
Ambient Temperature	25.0 °C	25.0 °C	25.0 °C	25.0 °C	25.0 °C	25.0 °C
Process	Maximum	Maximum	Maximum	Maximum	Maximum	Maximum
Implementation	Default	Power Optimization	Power Optimization+VID	Power Optimization	Power Optimization	Power Optimization
Summary						
Total On-Chip Power	5.004 W	4.923 W	4.814 W	4.741 W	4.677 W	4.546 W
Junction Temperature	30.8 °C	30.7 °C	30.6 °C	30.5 °C	30.4 °C	30.3 °C
Effective ΘJA	1.2 °C/W	1.2 °C/W	1.2 °C/W	1.2 °C/W	1.2 °C/W	1.2 °C/W
On-Chip Power						
Clocking	0.697 W	0.666 W	0.666 W	0.666 W	0.666 W	0.645 W
Logic	0.564 W	0.516 W	0.516 W	0.516 W	0.516 W	0.486 W
I/O	1.125 W	1.125 W	1.125 W	1.125 W	1.125 W	1.125 W
BRAM	0.459 W	0.459 W	0.459 W	0.459 W	0.459 W	0.433 W
DSP	0.437 W	0.437 W	0.437 W	0.437 W	0.437 W	0.413 W
Transceiver	1.149 W	1.149 W	1.149 W	1.149 W	1.149 W	1.136 W
PS						
Device Static	0.573 W	0.571 W	0.463 W	0.390 W	0.325 W	0.309 W

Supply Summary	Voltage	Current	Voltage	Current	Voltage	Current	Voltage	Current	Voltage	Current	Voltage	Current
V_{CCINT}	1.000 V	2.385 A	1.000 V	2.304 A	1.000 V	2.199 A	1.000 V	2.159 A	1.000 V	2.114 A	0.970 V	2.046 A
V_{CCBRAM}	1.000 V	0.099 A	1.000 V	0.099 A	1.000 V	0.099 A	1.000 V	0.098 A	1.000 V	0.097 A	1.000 V	0.097 A
V_{CCAUX}	1.800 V	0.256 A	1.800 V	0.255 A	1.800 V	0.255 A	1.800 V	0.238 A	1.800 V	0.228 A	1.800 V	0.228 A
V_{CCAUX_IO}												
V_{CCO} 3.3V												
V_{CCO} 2.5V												
V_{CCO} 1.8V	1.800 V	0.798 A	1.800 V	0.798 A	1.800 V	0.798 A	1.800 V	0.801 A	1.800 V	0.801 A	1.800 V	0.801 A
V_{CCO} 1.5V												
V_{CCO} 1.35V												
V_{CCO} 1.2V												
$MGTV_{CCAUX}$	1.800 V	0.040 A	1.800 V	0.040 A	1.800 V	0.040 A	1.800 V	0.040 A	1.800 V	0.040 A	1.800 V	0.040 A
$MGTAV_{CC}$	1.000 V	0.589 A	1.000 V	0.589 A	1.000 V	0.589 A	1.000 V	0.584 A	1.000 V	0.582 A	1.000 V	0.582 A
$MGTAV_{TT}$	1.200 V	0.255 A	1.200 V	0.255 A	1.200 V	0.255 A	1.200 V	0.255 A	1.200 V	0.255 A	1.200 V	0.255 A
$MGTZV_{CCH}$												
$MGTZAV_{CC}$												
$MGTZV_{CCL}$												
V_{CCPINT}												
V_{CCPAUX}												
V_{CCPLL}												
V_{CCO_DDR}												
V_{CCO_MIO0}												
V_{CCO_MIO1}												
V_{CCADC}	1.800 V	0.025 A	1.800 V	0.025 A	1.800 V	0.025 A	1.800 V	0.025 A	1.800 V	0.025 A	1.800 V	0.025 A

Figure 6-1: **Power Comparison Snapshots Sheet (7 Series Devices)**

A snapshot can represent:

- Power information for this XPE spreadsheet, captured at a certain time. When you create the snapshot, all of this information is copied from the Summary sheet of this spreadsheet to the Power Comparison Snapshots sheet.
- Power information for a different XPE spreadsheet, captured at a certain time. When you create the snapshot, all of this information is copied from the Summary sheet of the other spreadsheet to this Power Comparison Snapshots sheet.

- Power information for a design implemented in the ISE tools, captured at a certain time. When you create the snapshot, the power information is imported from the ISE Power Report into the Power Comparison Snapshots sheet.

The Power Comparison Snapshots sheet allows you to explore What If? scenarios, changing the part or the environmental conditions under which the part will operate and observing the effect on power the changes will have. You can also create a snapshot of the power calculated when your design is implemented in the ISE tools, to see how the power calculated for the implemented design compares to the power calculated before the design was implemented.

Using snapshots, you can explore What If? scenarios such as:

- How will power consumption change if I implement a design in different Xilinx architectures? What is the difference in power consumption when the same design is implemented in a Kintex device versus an Artix device?
- How does a design's power consumption vary as a function of junction temperature?
- How does a design's GT Power consumption change as a function of device type?
- How much power could be saved by using power optimization, or by choosing a -2L (low power) or a 0.9V part?
- How does the power and temperature vary under nominal versus maximum operating conditions?
- How does the design's power consumption vary as the design undergoes revision with respect to power, features, and performance?
- How does the power consumption vary between the Xilinx Power Estimator (XPE), XPower Analyzer (XPA), and Vivado Report Power estimations?
- How does the power consumption vary with changing clock frequencies?
- How did my pre-design estimate compare with post-design results and imported design data?

The four sections in the Power Comparison Snapshots sheet are:

- **Settings** - Displays the following:
  - The name of the snapshot (top line in the table)
  - Source and version of the snapshot data creator (for example, "ISE: 13.4" for an imported snapshot)
  - Data and Time the snapshot was created
  - The file name of XPE or the imported source
  - The **Part** (device, package, and speed grade) for which the power values were calculated.

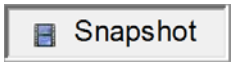
  - The value for **Ambient Temperature** under which the device will operate, as specified when the snapshot was taken.
  - The **Process** (Typical or Maximum) specified when the snapshot was taken. The **Process** setting accounts for the power dissipation caused by the manufacturing process.
  - The **Implementation** (Default or Power Optimization) specified when the snapshot was taken. This setting focuses the synthesis and implementation tools in the ISE Design Suite or the Vivado Design Suite on minimizing power towards different objectives when the design is implemented.
- **Summary** - Displays the following:
  - **Total On-Chip Power** - The total power consumed within the device for each snapshot. It includes device static and design dependent static and dynamic power.
  - The values for **Junction Temperature** and **Effective ΘJA** under which the device will operate, as specified when the snapshot was taken.
- **On-Chip Power** - The **On-Chip Power** section presents the total power consumed within the device by each resource type.

  In some cases, more than one resource will be included in a single row. For example, the **Clocking** row might include the power associated with clock nets as well as the power associated with clock managers such as the PLL and the MMCM, and the **Transceiver** row might include the power associated with Multi-Gigabit Transceivers (MGTs) as well as the power associated with a Hard IP block.
- **Supply Summary** - Displays the voltage and estimated current across the different supply sources. The table includes all power required by the internal logic along with power eventually sourced and consumed outside the Xilinx device, such as in external board terminations. This view includes both static and dynamic power.

Adding a Snapshot of the Current Spreadsheet

To add a snapshot of the current XPE spreadsheet to the Power Comparison Snapshots sheet:

1. Click the **Snapshot** button on the Power Comparison Snapshots sheet.



   A snapshot for the current XPE spreadsheet appears in the far right column of the table in the sheet.

2. If desired, rename the snapshot at the top row of the table.

Importing a Snapshot

You can import a snapshot containing power values calculated from a different XPE spreadsheet or power values calculated when the design is implemented in the ISE Design Suite or the Vivado Design Suite.

Note: When you import power information into the Power Comparison Snapshots sheet, the FPGA or SoC represented in the imported data does not have to match the device specified in the current XPE spreadsheet.

To import a snapshot into the Power Comparison Snapshots sheet:

1. Click the **Import** button at the top of the Power Comparison Snapshots sheet.

   A snapshot for the current XPE spreadsheet will appear in the far right column of the table in the sheet.

2. In the Select XPE File to Import dialog box, select the following in the **Files of type** box:
   - **XPE Workbook (*.xls*)**, if you are importing information from a different XPE spreadsheet.
   - **Power Report (*.pwr)**, if you are importing information from a Power Report generated within the ISE Design Suite or the Vivado Design Suite.
3. Browse to the file you will import and click **Open**.

A snapshot appears in the far right column of the table in the Power Comparison Snapshots sheet. If desired, rename the snapshot at the top row of the table.

Deleting Snapshots from the Power Comparison Snapshots Sheet

To delete a single snapshot, click the box with the red "**X**" at the top of the snapshot.

To delete all of the snapshots on the Power Comparison Snapshots sheet, click the **Clear All** button at the top of the sheet.

Using Graph Sheets

The following power graphs display the graphical representation of your power estimates.

On-Chip Power by Function

This graph displays the variation of power for each functional block.

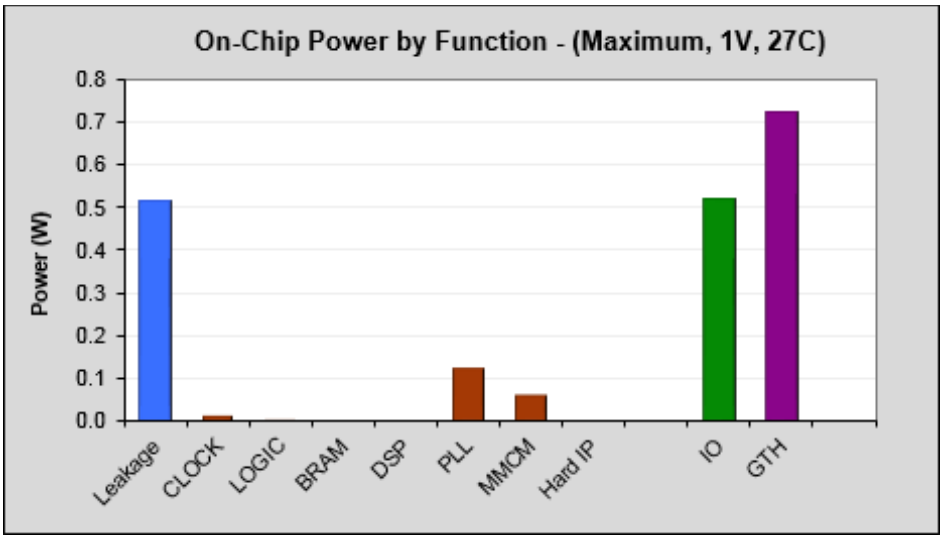


Figure 6-2: **Graph displaying On-Chip Power by Function**

On-Chip Power over Vccint

This graph displays the variation of power with respect to Vccint (core voltage).

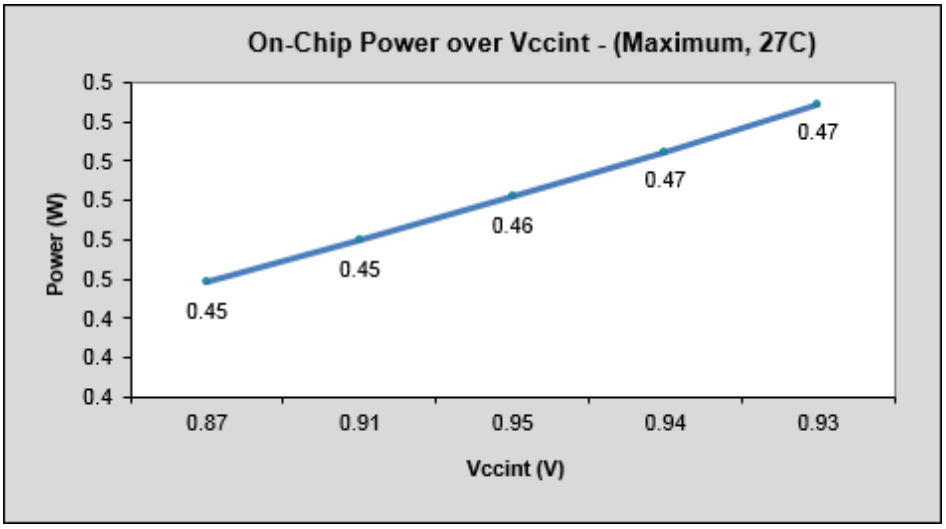


Figure 6-3: **Graph displaying On-Chip Power Over Vccint**

Static Current by Supply

This graph displays the power for each supply rails.

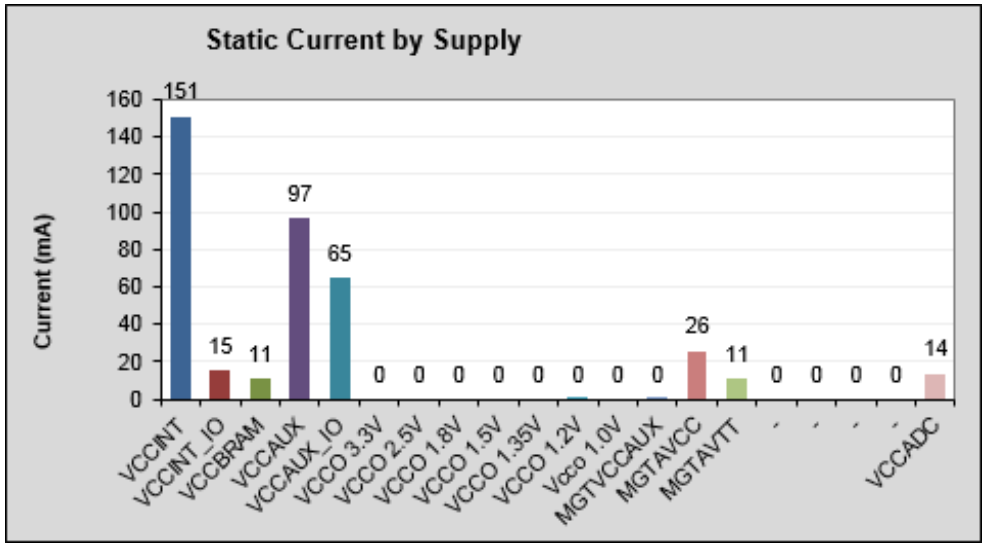


Figure 6-4: **Graph displaying the Static Current by Supply**

On-Chip Typical vs Maximum Power

This graph displays the power with respect to PVT (Power, Voltage, and Temperature) changes.

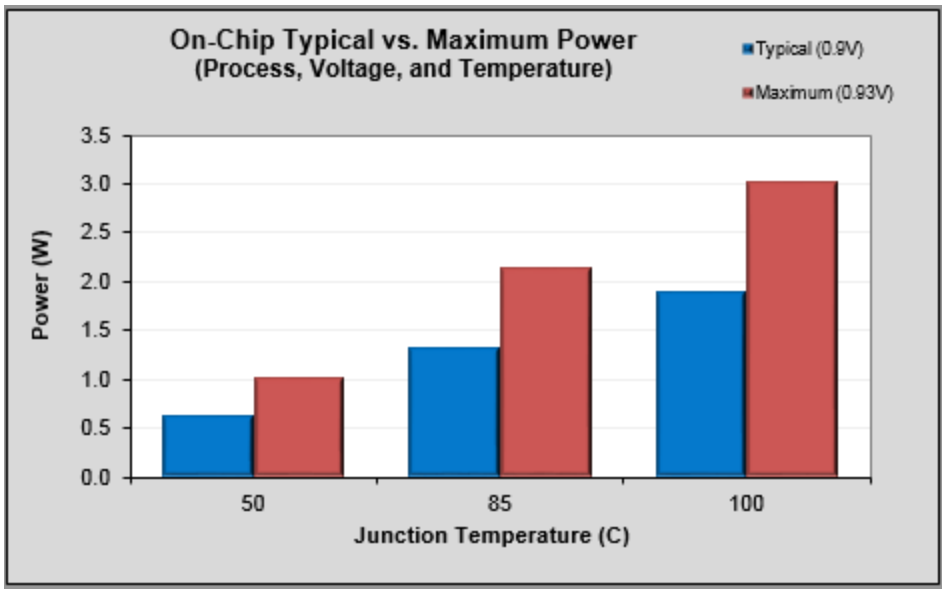


Figure 6-5: **Graph displaying the On-Chip Typical Vs Maximum Power variance**

On-Chip Power

This graph displays the very fine variation of power (static and maximum) with respect to Junction temperatures.

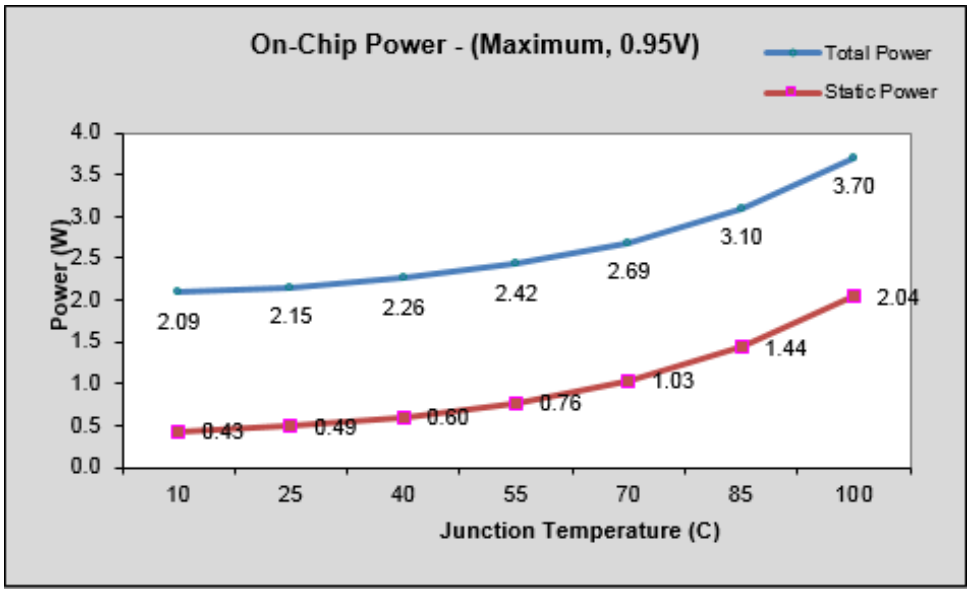


Figure 6-6: **Graph displaying the On-Chip Power variance with respect to Junction Temperatures**

Appendix A

Additional Resources and Legal Notices

Xilinx Resources

For support resources such as Answers, Documentation, Downloads, and Forums, see Xilinx Support.

Solution Centers

See the Xilinx Solution Centers for support on devices, software tools, and intellectual property at all stages of the design cycle. Topics include design assistance, advisories, and troubleshooting tips.

Documentation Navigator and Design Hubs

Xilinx Documentation Navigator provides access to Xilinx documents, videos, and support resources, which you can filter and search to find information. To open the Xilinx Documentation Navigator (DocNav):

- From the Vivado IDE, select **Help > Documentation and Tutorials**.
- On Windows, select **Start > All Programs > Xilinx Design Tools > DocNav**.
- At the Linux command prompt, enter `docnav`.

Xilinx Design Hubs provide links to documentation organized by design tasks and other topics, which you can use to learn key concepts and address frequently asked questions. To access the Design Hubs:

- In the Xilinx Documentation Navigator, click the **Design Hubs View** tab.
- On the Xilinx website, see the Design Hubs page.

Note: For more information on Documentation Navigator, see the Documentation Navigator page on the Xilinx website.

References

1. XPower Analyzer Help
2. *Vivado® Design Suite User Guide: Power Analysis and Optimization* (UG907)
3. *Power Methodology Guide* (UG786) - for ISE tools
4. *Command Line Tools User Guide* (UG628) - for ISE tools
5. *7 Series FPGAs Configurable Logic Block User Guide* (UG474)
6. *7 Series FPGAs Memory Resources User Guide* (UG473)
7. *7Series FPGAs Packaging and Pinout Product Specifications User Guide*(UG475)
8. *7 Series FPGAs GTX/GTH Transceivers User Guide* (UG476)
9. *7 Series FPGAs GTP Transceivers User Guide* (UG482)
10. *Zynq®-7000 SoC and 7 Series Devices Memory Interface Solutions User Guide* (UG586)
11. *UltraScale™ Architecture Configurable Logic Block Advance Specification User Guide* (UG574)
12. *UltraScale Architecture Memory Resources Advance Specification User Guide* (UG573)
13. *Kintex UltraScale and Virtex FPGAs Packaging and Pinout Product Specifications User Guide* (UG575)
14. *UltraScale Architecture GTH Transceivers Advance Specification User Guide* (UG576)
15. *UltraScale Architecture GTY Transceivers Advance Specification User Guide* (UG578)
16. *Virtex®-7 T and XT FPGAs Data Sheet: DC and AC Switching Characteristics* (DS183)
17. *Kintex®-7 FPGAs Data Sheet: DC and AC Switching Characteristics* (DS182)
18. *Artix®-7 FPGAs Data Sheet: DC and AC Switching Characteristics* (DS181)
19. *Virtex UltraScale Architecture Data Sheet: DC and AC Switching Characteristics* (DS893)
20. *Kintex UltraScale Architecture Data Sheet: DC and AC Switching Characteristics* (DS892)
21. *LogiCORE™ IP Aurora 8B/10B Product Guide* (PG046)
22. *LogiCORE IP Aurora 64B/66B Product Guide* (PG074)
23. *Device Package User Guide* (UG112)
24. *7 Series FPGAs Clocking Resources User Guide* (UG472)
25. *UltraScale Architecture Clocking Resources Advance Specification User Guide* (UG572)
26. *7 Series FPGAs SelectIO Resources User Guide* (UG471)
27. *7 Series DSP48E1 Slice User Guide* (UG479)

28. *UltraScale Architecture DSP Slice Advance Specification User Guide* (UG579)
29. *UltraScale Architecture Integrated Block for 100G Ethernet LogiCORE IP Product Guide* (PG165)
30. *UltraScale Architecture Integrated IP Core for Interlaken LogiCORE IP Product Guide* (PG169)
31. *Zynq-7000 SoC Technical Reference Manual* (UG585)
32. *7 Series FPGAs and Zynq-7000 SoC XADC Dual 12-Bit 1MSPS Analog-to-Digital Converter User Guide* (UG480)
33. *UltraScale Architecture System Monitor Advance Specification User Guide* (UG580)
34. *7 Series FPGAs Configuration User Guide* (UG470)
35. *UltraScale Architecture Configuration Advance Specification User Guide* (UG570)
36. Descriptions of the resources available in an FPGA can be found under **Silicon Devices** on the Xilinx Support web page.
37. *UltraScale Architecture FPGAs Memory IP Product Guide* (PG150)
38. *Kintex UltraScale+ FPGAs Data Sheet: DC and AC Switching Characteristics* (DS922)
39. *Virtex UltraScale+ FPGAs Data Sheet: DC and AC Switching Characteristics* (DS923)
40. Zynq UltraScale+ MPSoC Data Sheet: DC and AC Switching Characteristics (DS925)
41. *Zynq UltraScale+ RFSoC Data Sheet: Overview* (DS898)
42. Vivado Design Suite Documentation

Training Resources

1. Vivado Design Suite QuickTake Video Tutorial: Using the Xilinx Estimator
2. Vivado Design Suite QuickTake Video: How to Estimate UltraScale Device Power using XPE
3. Vivado Design Suite QuickTake Video Tutorials

Please Read: Important Legal Notices

The information disclosed to you hereunder (the "Materials") is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available "AS IS" and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx's limited warranty, please refer to Xilinx's Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx's Terms of Sale which can be viewed at https://www.xilinx.com/legal.htm#tos.

AUTOMOTIVE APPLICATIONS DISCLAIMER

AUTOMOTIVE PRODUCTS (IDENTIFIED AS "XA" IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE ("SAFETY APPLICATION") UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD ("SAFETY DESIGN"). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.