See all versions of this document

Xilinx Power Estimator User Guide for Versal ACAP

UG1275 (v2021.2) November 9, 2021

Xilinx is creating an environment where employees, customers, and partners feel welcome and included. To that end, we're removing non-inclusive language from our products and related collateral. We've launched an internal initiative to remove language that could exclude people or reinforce historical biases, including terms embedded in our software and IPs. You may still find examples of non-inclusive language in our older products as we work to make these changes and align with evolving industry standards. Follow this link for more information.

XILINX

Revision History

The following table shows the revision history for this document.

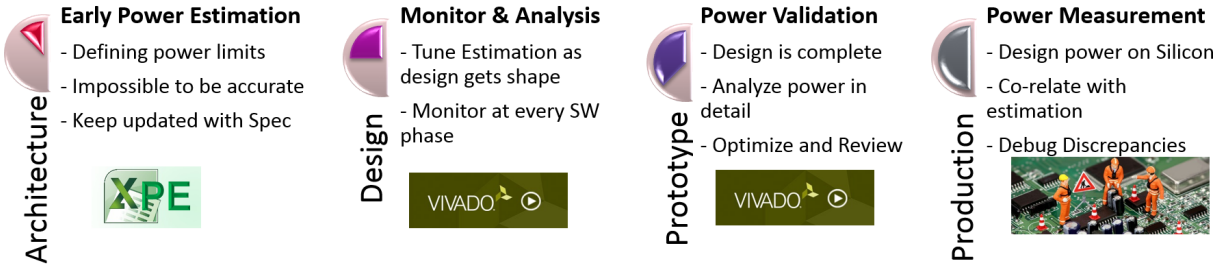
Section	Revision Summary
11/09/2021 Version 2021.2	
Production	Updated.
Fanout/Site	Added.
Slice Clock Enable	Added.
Chapter 5: Network-On-Chip and DDRMC Power	Updated figures.
DDRMC Wizard	Updated figures.
Device Estimation Overview	Updated figures.
Power Supply Design Table	Updated figures.
AI Engine Simulation Based Power Estimation	Added.
DFX Usage and NoC Clock Gating	Added.

Table of Contents

Introduction

This document describes the Xilinx® Power Estimator (XPE) usage based on Versal® ACAP architecture descriptions. Accurate power estimation during the early design cycle is the key to the lowest possible power envelope for any design. Early estimation is crucial for choosing the right device, taking advantage of architectural benefits, changing the design topology, and using different IP blocks. Thus, making better trade-offs well ahead of the design phase allows you to meet specifications and get your product to market faster. Xilinx offers two types of power estimation tools – XPE which is typically used for estimation before design implementation and Vivado Report Power which is used during design implementation with better accuracy. Both of them have many features that help you create a low-power ACAP design. Xilinx recommends the following Power Methodology to overcome any power challenges throughout the design cycle.

Figure 1: **Power Methodology**



During the conceptualization and architectural exploration phases of a project, it is crucial to assess the power budget with limited availability of architectural details of the design. XPE addresses most of the early power estimation challenges. It is typically used in the pre-design and pre-implementation phases of a project and it assists with architecture evaluation, device selection, appropriate power supply components, and thermal management solutions specific to your application. XPE considers your design resource usage, toggle rates, I/O loading, and other factors. These factors, combined with the device models, help calculate the estimated power distribution. The device models are extracted from measurements, simulation, and extrapolation. XPE is built on Microsoft Excel running on an English Windows operating system, it has a rich set of power modeling features and algorithms to offer easy access and an easy to use model.

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

- Device usage, thermal environment and solution, clock, enable, toggle rates, and other information you enter into the tool

- Device data models integrated into XPE (Power Characterization accuracy)

For accurate power estimates of your application, enter information that 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 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. During implementation, Report Power (in the Vivado® Design Suite) should be used for more accurate estimates and detailed power analysis. A common power modeling approach is adopted to ensure accurate and consistent estimation between XPE and Vivado Report Power.

Navigating Content by Design Process

Xilinx® documentation is organized around a set of standard design processes to help you find relevant content for your current development task. All Versal® ACAP design process Design Hubs and the Design Flow Assistant materials can be found on the Xilinx.com website. This document covers the following design processes:

- **System and Solution Planning:** Identifying the components, performance, I/O, and data transfer requirements at a system level. Includes application mapping for the solution to PS, PL, and AI Engine. Topics in this document that apply to this design process include:
  - Chapter 2: Device Selection and Configuration
  - Chapter 3: Processing System Power
  - Chapter 4: AI Engine Power
  - Chapter 6: Programmable Logic Power
  - Chapter 7: I/O and Transceiver Power
- **Embedded Software Development:** Creating the software platform from the hardware platform and developing the application code using the embedded CPU. Also covers XRT and Graph APIs. Topics in this document that apply to this design process include:
  - Chapter 2: Device Selection and Configuration
  - Chapter 3: Processing System Power
- **AI Engine Development:** Creating the AI Engine graph and kernels, library use, simulation debugging and profiling, and algorithm development. Also includes the integration of the PL and AI Engine kernels. Topics in this document that apply to this design process include:
  - Chapter 4: AI Engine Power

- **Hardware, IP, and Platform Development:** Creating the PL IP blocks for the hardware platform, creating PL kernels, functional simulation, and evaluating the Vivado® timing, resource use, and power closure. Also involves developing the hardware platform for system integration. Topics in this document that apply to this design process include:
  - Chapter 2: Device Selection and Configuration
  - Chapter 5: Network-On-Chip and DDRMC Power
  - Chapter 6: Programmable Logic Power
  - Chapter 7: I/O and Transceiver Power
- **System Integration and Validation:** Integrating and validating the system functional performance, including timing, resource use, and power closure. Topics in this document that apply to this design process include:
  - Chapter 2: Device Selection and Configuration
  - Chapter 8: Power Supply Design
  - Chapter 9: Snapshot, Device Duty Cycle and Graph Sheet
- **Board System Design:** Designing a PCB through schematics and board layout. Also involves power, thermal, and signal integrity considerations. Topics in this document that apply to this design process include:
  - Chapter 2: Device Selection and Configuration
  - Chapter 8: Power Supply Design

Versal ACAP Power Domains

In Versal® ACAP architecture, different functional blocks are partitioned into power domains that are powered using dedicated supply rails. These supply rails can be connected to different power sources. The power consumed within a power domain can be controlled by manipulating the supply voltage to the power domain as a trade-off with its power rails as described in Table 1. The following figure shows how the circuitry on a Versal device die is partitioned into power domains, as it appears at the device level.

***Note*:** Exact layout may vary based on the targeted Versal device and its size.

Figure 2: **Device Level Power Domains**

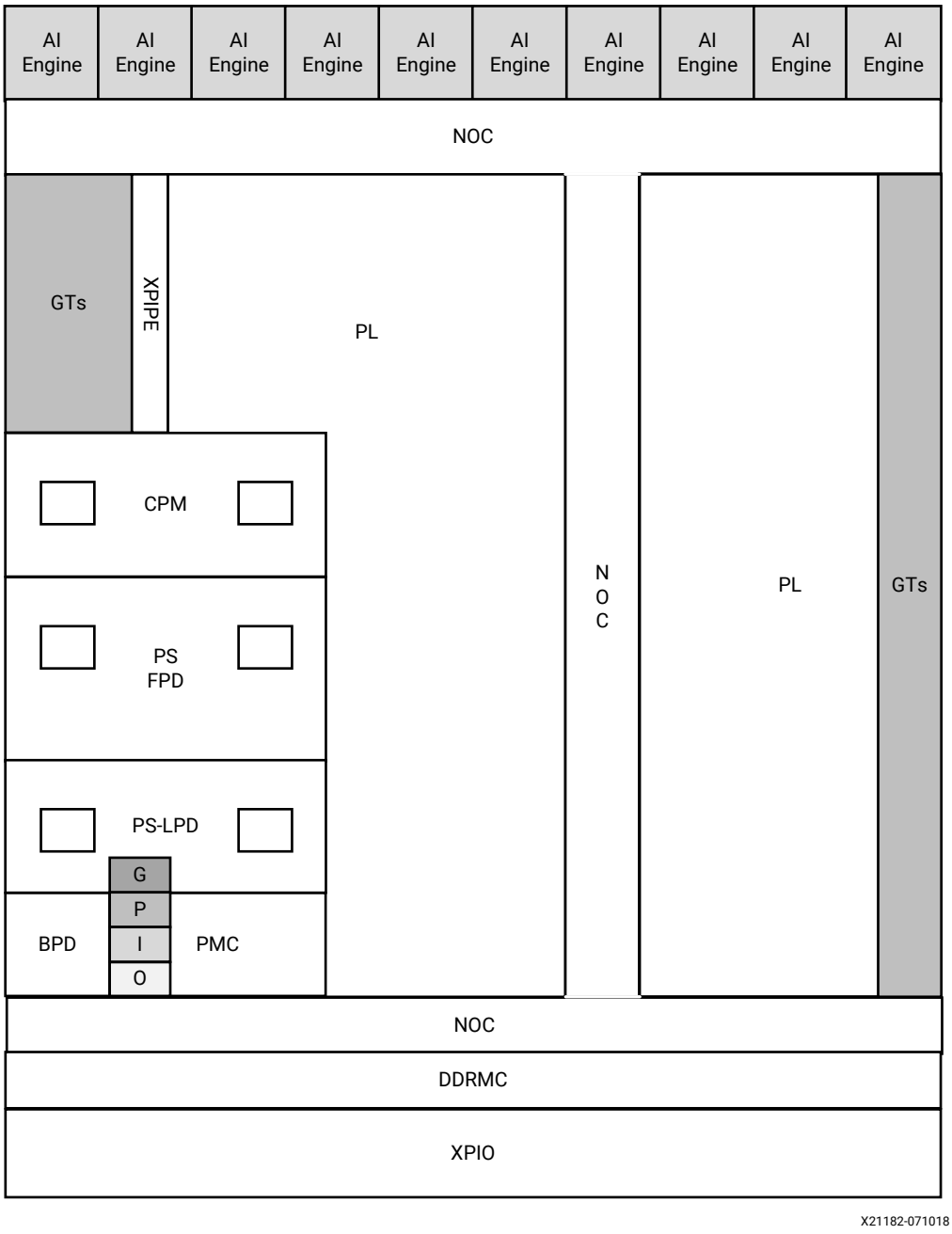


Table 1: **Power Domain Descriptions**

Power Domain	Description
PMC Domain (VCC_PMC)	This is an Always-On domain for the device and the only core domain that should be up for the device operation to start and stay maintained.

Table 1: **Power Domain Descriptions** *(cont'd)*

Power Domain	Description
PS Low-Power Domain (VCC_PSLP)	This domain needs to be up in addition to the PMC domain for the primary device configuration through USB to get started. This domain supplies power to RPUs (Realtime Processing Unit - Arm® Cortex-R5F core). Because this domain is associated with many of the low-power modes, it includes power islands to satisfy the power requirements of these modes.
PS Full-Power Domain (VCC_PSFP)	This domain supplies power to APUs (Application Processing Unit - Arm® Cortex®-A72 core) within the PS that are not required during low-power modes.
NoC and DDRMC Domain (VCC_SOC)	This power domain includes the NoC and Hardened Memory Controller. This domain should be up when the PL Domain is up.
Core and PL Domain (VCCINT)	This domain includes the internal core logic of the PL, CCIX PCIe® Module (CPM), and AI Engine.
PL RAM Domain (VCC_RAM)	This supply provides the power to the PL RAMs and the PL clocking network. This rail is expected to be up always while the VCCINT domain is up. If this supply is down, VCCINT domain incurs a power-on reset.
Battery-Powered Domain (VCC_BATT)	This is the power domain for the RTC Core and the Battery-Backed RAM (BBRAM). This domain will be on the battery supply (VCC_BATT) if the device is off; otherwise, the PMC/PS Auxiliary supply (VCCAUX_PMC) provides the power to this domain.
Analog Domain	This domain has three supplies namely GTY_AVCC, GTY_AVTT, and GTY_VCCAUX. GTY_AVCC is the analog supply for internal analog circuits of the transceivers. This includes analog circuits for the PLL, transmitters, and receivers. GTY_AVTT is the analog supply for transmitter and receiver termination circuits. GTY_VCCAUX is the auxiliary analog QPLL voltage supply for the transceivers. ***Note***: For GTP, the rails are GTP_AVCC, GTP_AVTT, and GTP_VCCAUX. For GTM, the rails are MGTM_AVCC, MGTM_AVTT, and MGTM_VCCAUX.
VCCO Domain	This domain includes all the VCCO supplies. It powers all device I/Os.

Versal ACAP Auxiliary Power Domains

The Versal® devices include two major auxiliary power rails. The auxiliary power rails that are typically associated with the Analog IPs, I/O, or Battery-powered domain are listed in the following table:

Table 2: **Auxiliary Power Domain Descriptions**

Auxiliary Power Domain	Description
PMC/PS Aux Rail (VCCAUX_PMC)	This is the Auxiliary supply that is used by the circuitry in the PMC and PS PLLs, I/O, and SysMon. This supply is also used to run the Battery-Powered domain when the device is up. When the device is off, the Battery-Powered circuitry is switched to VCC_BATT. VCCAUX_PMC should be up for the Versal device to boot.
PL Aux Rail (VCCAUX)	This Aux supply is used by the PLLs, IO, and Satellite SysMons outside the PS. The core supplies associated with the modules that use this Aux supply can be VCCINT, VCC_SOC, VCC_IO, and VCC_RAM.

Multiple power supplies are required to power a Versal device. The following table describes the logic resources typically available in these devices and their corresponding power supply:

Table 3: **Versal ACAP Resources and Corresponding Power Supply**

Power Domain	Power Supply	Resources Powered
PMC Domain	VCC_PMC VCCO_500 VCCO_501 VCCO_503 VCCAUX_PMC VCCAUX_SMON VCC_FUSE	• This is an Always On domain. PMC is required always to be powered for Versal ACAP to be configured and functional during run-time. • Platform management controller • PMC IO bank, including flash, I2C, QSPI, and other interfaces • PMC and PS PLL • SYSMON • eFuse Programming
PS Domain	VCC_PSLP VCCO_502	PS Low Power domain including • Real time processing unit (RPU) • TCM • OCM • IO interface for GEM, USB, and Std performance IOs
	VCC_PSFP	PS Full Power Domain • Application processing unit (APU) • L2 Cache • CCIX

Table 3: **Versal ACAP Resources and Corresponding Power Supply** *(cont'd)*

Power Domain	Power Supply	Resources Powered
System Domain	VCC_SOC	• Network on Chip • Hardened DDR memory controller • Interconnect (HSR crossing)
	VCCAUX	• Clock managers (MMCM, PLL, and DCM) • IODELAY/IDELAYCTRL • All output buffers • Differential input buffers • VREF-based, single-ended I/O standards, for example, HSTL18_I
	VCC_IO	• IBUF • OBUF • ISERDESE • IDDR • IFF • OSERDESE • ODDR • OFF • IDELAYE • ODELAYE • RX_BITSLICE • TX_BITSLICE • TX_BITSLICE_TRI • RXTX_BITSLICE • BITSLICE_CONTROL • IDELAYCTRL • XPLL Controller

Table 3: **Versal ACAP Resources and Corresponding Power Supply** *(cont'd)*

Power Domain	Power Supply	Resources Powered
PL Domain	VCCINT	All CLB resources • All routing resources • Entire clock tree, including all clock buffers • Block RAM • DSP slices • All input buffers • AI Engines • Logic elements in the IOB (ILOGIC/OLOGIC) • Clock Managers (MMCM, DPLL, and DCM) • Hard Blocks like PCIe, MRMAC, DCMAC, and CPM
	VCC_RAM	• Memory array of block RAMs • Clock network power • DPLL
	VCC_CPM5	CPM5 ***Note*:** It is available only for some Premium series devices.
	VCCO	• All output buffers • Some input buffers • Input termination • Reference resistors to DCI
	MGTAVCC MGTAVTT MGTVCCAUX	• Analog supply voltages for PMA circuits of transceivers • Transceiver termination circuits • Quad PLL
Battery Domain	VCC_BAT	Battery domain powers RTC core and battery backed RAM

The following table shows the dependency between the mentioned power domains by providing the list of domains that have to be functionally available if a target domain is expected to be functional. A domain is functionally available if it is powered and its reset is released. The powered domains can operate at the voltage levels that are supported by that rail.

Table 4: **Dependency of Power Domains**

Target Domain	Other Core Domains that have to be up	Required Supplies
PMC	None	VCC_PMC, VCCAUX_PMC, and VCC_50x
PS-LPD	PMC	VCC_PMC, VCCAUX_PMC, VCC_50x, and VCC_PSLP
PS-FPD	PMC, PS-LPD	VCC_PMC, VCCAUX_PMC, VCC_50x, VCC_PSLP, and VCC_PSFP
NoC with DDR	PMC, VCC_AUX	VCC_PMC, VCCAUX_PMC, VCC_50x, VCC_SOC, and VCC_AUX

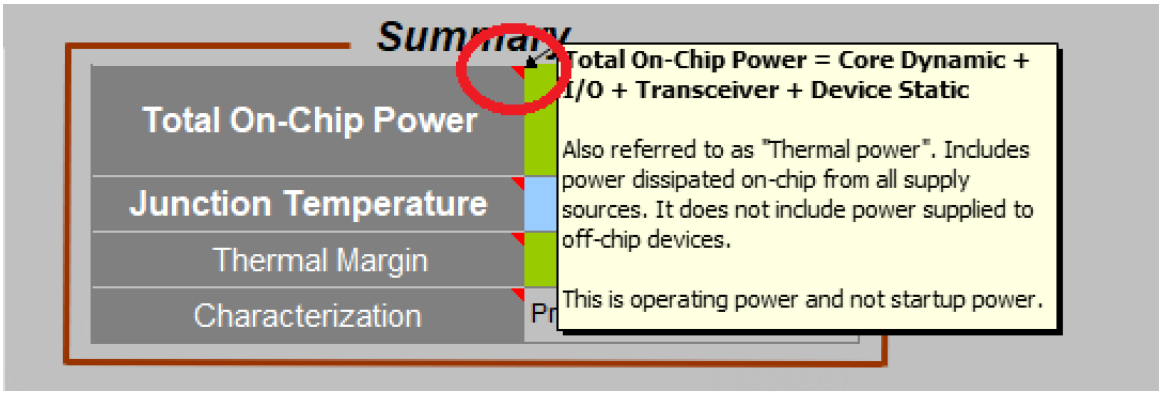
Table 4: **Dependency of Power Domains** *(cont'd)*

Target Domain	Other Core Domains that have to be up	Required Supplies
PL	PMC, NoC	VCC_PMC, VCCAUX_PMC, VCC_50x, VCC_SOC, VCCINT, VCCAUX, and VCC_RAM

XPE Layout

XPE is a spreadsheet and all Microsoft Excel functionality is fully retained in the user tab. XPE has additional functionality oriented towards ease of use. Drop-down menus and the comment-enabled cells are helpful features to guide you. Comment enabled cells have red corners, hovering over them reveals more information about the cell as shown in the following figure:

Figure 3: **Comment Enabled Cells**



The XPE spreadsheet also includes various wizards to help you to quickly enter the important configuration parameters. The wizards automatically configure the related I/O, Logic, block RAM, Transceiver and other sheets to help estimate the power accurately.

Getting Started with XPE

System Requirements

- Microsoft 365® is recommended.
- Microsoft Excel settings must allow macro executions. XPE uses several macros built into the XPE spreadsheet.
- Microsoft Excel language preference must be set to *English*.

Note: OpenOffice and Google Docs spreadsheet editors are not supported. For more information on XPE known issues and workarounds, see AR 75855.

To enable macros in the XPE:

1. Open the XPE spreadsheet.
2. From the XPE spreadsheet, select **File → Options**.
3. In the Excel Options dialog box, click **Trust Center**.
4. In the Trust Center dialog box, click **Trust Center Settings**.
5. Select the **Macro Settings** tab.
6. Select **Enable all macros**.
7. Click **OK**.

User Input Requirements

Versal® ACAP power estimation is a complex process, because it is 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 usage, clock rates, and toggle rates. To supply the minimum input that allows 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
- 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 or solution 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 allows you to determine the device power supply and heat dissipation requirements.

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 and Current of Voltage Supplies:** For each required voltage source, power and current is useful to select and size power supply components, such as regulators. Supply power includes both off-chip and on-chip dissipated power. Dynamic power is reported on an individual sheet while the current per supply is reported on the Power Design sheet.

- **On-Chip Power Per Resource Type:** 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.
- **Total On-Chip Thermal Power:** XPE lets you enter the thermal solution settings (Theta Ja) and the ambient temperature (Ta) and reports thermal properties of the device for your application, such as the expected junction temperature. It is recommended that you run the thermal simulation and not rely on XPE to get information about the thermal solution required for your system.

The Summary sheet in XPE shows the total on-chip power for the device. Other sheets show power based on resource usage.

Definitions of Terminology

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 Preview, Advance, Preliminary, and Production.

Preview

The specifications and power models are based on scaled targets and early simulation vectors.

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 MRMAC and PCIe® blocks 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. The characterization data for all blocks in the device fabric is included.

The accuracy of any power estimation is derived from the information input to the models. XPE uses the following models based on the device characterization:

- **PREVIEW:** +/-30%
- **ADVANCE:** +/-25%
- **PRELIMINARY:** +/-20%
- **PRODUCTION:** +/-15%

XPE and Report Power use the same models but Report Power has more details on the design being implemented such as resource settings and usage, net fanout, and net lengths which allows Report Power to give a more accurate estimation.

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

Total Power

The total device power is calculated as follows:

- Total device power = Device Static + Design Static + Design Dynamic.

  ***Note*:** XPE reports Device and Design Static as a single static value.

- The power estimates are modeled to account for temperature and voltage sensitivity. Ambient temperature (Theta Ja) and regulated voltage on the system can be keyed into the appropriate cells provided for that purpose. Additional inputs include Process (Typical or Maximum) and an environment where you can choose either a forced Junction Temperature or specify Ambient Temperature with the Effective package Theta JA.

Device Static Power

Also referred to as Leakage. Device static represents the transistor leakage power when the device is powered and not configured. Device static is directly related to the Tj of the device and will increase as the temperature of the device increases.

Design Static Power

Design static power represents the additional power consumption for power gated blocks when the device is configured but there is no switching activity. There are certain resources that have zero static power contribution unless they are used in the design. When their usage totals are entered in XPE, they contribute to the design static power. Block RAM, UltraRAM, DDRMC, GT, I/O, and clock managers are power gated and contribute to design static power. On the other hand, there are certain resources that are always powered and do not contribute to the design static power because they are already included in the device static. CLOCK, LOGIC, DSP, AIE, and NoC are always powered and do contribute to the design static power.

TIP: *To add your design elements (for example, I/Os, block RAMs, UltraRAMs, DDRMC, GT, and Clock Managers) to the design static power calculations, you must enter the resource usage 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 to 0 on other resource sheets.*

Design Dynamic Power

Design dynamic represents additional power consumption from the user logic resources use and clocking, routing, switching activity, and load. Design Dynamic power is constant and does not change with changes in device temperature.

Clock

Important factors in dynamic power calculation are the activity and the load capacitance that needs to be switched by each physical driver in the design. Some of the factors in determining the loading capacitance are fanout and interconnect distance. With clocks typically having higher activity and fanouts, the power associated with clock nets can be significant.

Load

This is the average percentage of time for a given processor to execute instructions. It has a significant impact on power.

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 100 MHz 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.

Junction Temperature

This user defined field forces the value of the device junction temperature (Tj) when the Override box is checked. This option could be used when you need to work backward from a known junction temperature (measured using SysMon) or the assumed worst case junction temperature and define the environment that would ensure this temperature is not exceeded. You should specify the ambient temperature and XPE then adjusts the Effective Theta-JA appropriately to get to the specified junction temperature.

Effective Θ JA (°C/W)

This coefficient defines how power is dissipated from the device silicon to the environment (device junction to ambient air). It includes contributions from all elements, from the silicon chip dimensions to the surrounding air, plus any material in between, such as the package, the PCB, any heat sink, and airflow.

$$\Theta Ja = (Tj - Ta)/ Pd$$

This value is best obtained from thermal simulation using Xilinx provided thermal models. These are accessible via Power Design sheet and Thermal Model tab.

XILINX

Chapter 2

Device Selection and Configuration

When you launch XPE, the Summary sheet is selected by default. This sheet plays an essential role in effective power estimation of Versal® devices. The summary sheet provides the following:

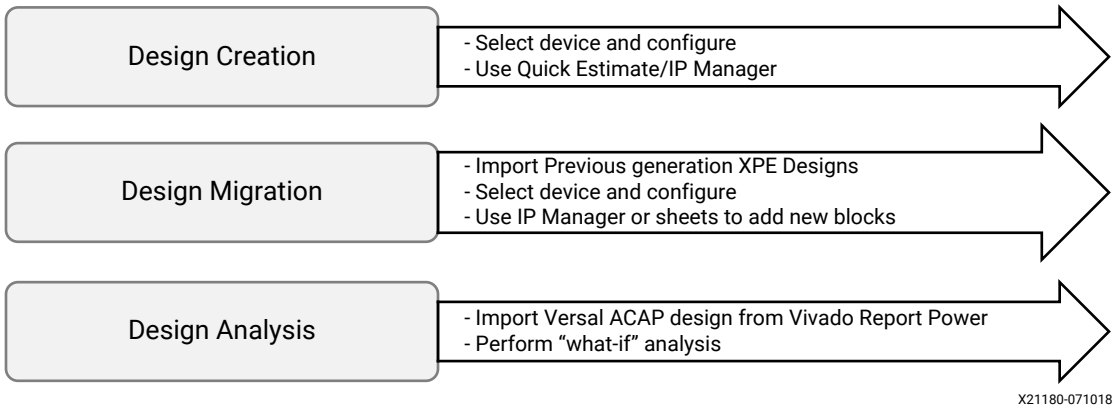
- Device setup, which includes device selection, thermal settings, and configuration of the PMC (Platform Management Controller).
- High-level design entry, which includes a Quick Estimate wizard under Create Design, IP block creation and management, Design migration (import from previous architecture), and Vivado® Report Power post-processing.
- A dashboard, which includes a summary of estimated Total On-Chip Power, Junction Temperature, Thermal Margin, and On-Chip Power broken down by Resource types.

Design Considerations for Effective Power Estimation

The design entry in XPE is classified into the following three categories:

- **Design creation:** Design creation represents a complete manual entry for power estimation and it starts with proper device selection followed by thermal condition specification and more importantly configuring the platform management controller. After the design is configured, use Quick Estimate and IP Manager to create designs followed by refinement on individual block sheets.
- **Design migration:** Design migration starts by importing the previous generation XPE design (`.xpe`) into Summary sheet. The import flow only retrieves the data that is relevant to the Versal® architecture – mostly the programmable logic (PL) and Processing system (PS) configuration. Thus, it requires device selection and configuration to complete the design creation in addition to adding the required new blocks through IP Manager or through manual entry in block sheets.
- **Design analysis:** Design analysis is specific to Versal ACAP designs created in Vivado® and imported into XPE for further analysis or post-processing.

Figure 4: **XPE Design Flow in Versal Devices**



***Note*:** For more information, see Importing XPE Data from a Previous Device Family.

The following sections describe the recommended design flow in detail:

Device Configuration

Device selection is a crucial step for an application based on resource, performance, power, and package requirements. The recommendation is to select the smallest device that meets your requirements.

IMPORTANT! *Larger devices exhibit higher device static power consumption, however Versal with its hard IPs, improved device architecture such as the DSP58, processing subsystem, and network on chip can reduce resource requirement and potentially mean a smaller device is required therefore lowering the static power associated with a design.*

Currently Versal AI Core, Versal Prime, and Versal Premium series are supported. They are followed by the device/part name itself which is an unique identifier to represent a device under a specific family. Similarly, there are different package offerings under a device according to type, assembly, characteristics and transceiver/IO resource combinations. Refer to the product table for the detailed product naming and resource information. Device Grade includes both temperature and speed grade of the device. Versal devices support the following temperature grades, each having an impact on power.

- Industrial-operating temperature range is -40°C to 100°C.
- Extended-operating temperature range is 0°C to 100°C.

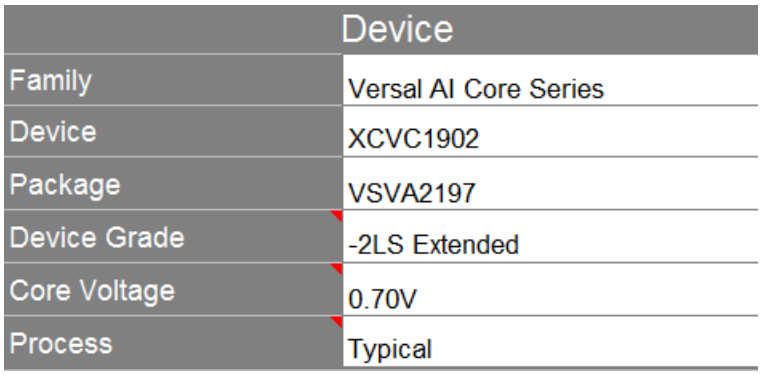
  TIP: *Certain speedgrades of Versal devices may operate between 100 and 110°C for 3% of their lifetime. For more information, see the Extending the Thermal Solution by Utilizing Excursion Temperatures (WP517) and Versal ACAP Data Sheet.*

- Q-grade operating temperature range is -40°C to 125°C.

The unique power binning strategy of Xilinx enables lower static power for Industrial grade compared to Extended grade devices. Versal devices offer enhanced voltage scaling options, which is key for highest performance/watt on these devices. Versal devices support the following operating modes:

- High Performance-Core operating voltage is 0.88V (V_{HP})
- Mid/Balanced Power/Performance-Core operating voltage is 0.80V (V_{MP})
- Low Power-Core operating voltage is 0.70V (V_{LP})

Figure 5: **Device Configuration in Summary Sheet**



The Process represents the manufacturing process variation of the device, which significantly influences leakage power. The Process variation model is a common industry practice to meet the yield requirements without violating the device specification. During device manufacturing, the silicon process (doping, lithography) alters the transistor characteristics a bit, resulting in variations between devices. Typical is the median boundary where the device meets both performance and power specification. Then move towards higher performance distribution and set the maximum limit where it still meets specific temperature grade specification on both power and performance.

RECOMMENDED: *Set **Typical** for device selection or comparison or battery powered low operating temperature applications. For all other use cases, set Process to **Maximum** for worst case power estimation which is important for power supply (board) and thermal design for higher power consuming applications.*

Thermal Settings

Leakage power increases exponentially with respect to junction temperature. Thus, it is very important to accurately specify the environment conditions in XPE. Environment settings allow the following flows:

- **User specified junction temperature:** Enter the targeted junction temperature or the maximum junction temperature of the device and ambient temperature.
  - XPE calculates maximum Effective ΘJA to meet requirements (the Effective ΘJA field is read-only).

- **XPE estimated junction temperature:** Enter ambient temperature and effective ΘJA, from which XPE calculates the junction temperature.
  - Derive Effective ΘJA from thermal simulation

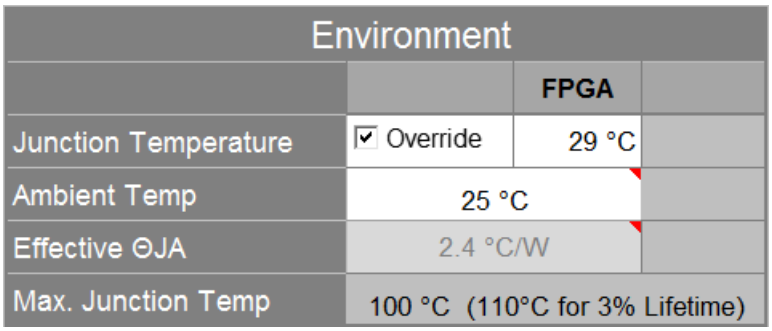
Note: XPE uses the thermal settings only to calculate the junction temperature based on the following formula:

```
Junction Temperature = Ambient Temperature + (Effective ΘJA * Total On-chip Power)
```

IMPORTANT! *XPE does not perform thermal analysis to calculate the effective ΘJA. It reports the requirements based on given junction temperature and ambient temperature values. Xilinx suggests that you replace the XPE populated maximum Effective ΘJA with the determined Effective ΘJA from thermal analysis of the system, preferably from thermal simulation or actual system measurement.*

XPE alerts you if the specified or calculated junction temperature exceeds the device temperature grade margin.

Figure 6: **Thermal Settings in XPE**



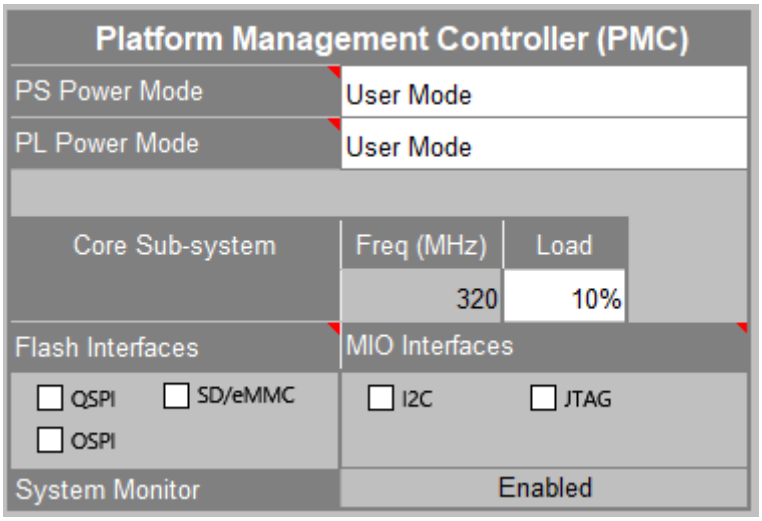
Note: Xilinx provides thermal models for both Siemens Flotherm and Ansys IcePak, accessed via the power design sheet or at Versal Thermal Models. See Chapter 8: Power Supply Design to enable accurate thermal simulation. It is recommended that you perform a thermal simulation. This allows you to effectively assess the conditions and use the effective ΘJA parameter from simulation to estimate accurate device junction temperature in XPE.

Platform Management Controller (PMC)

The Platform Management Controller (PMC) in Versal devices is responsible for handling the primary pre-boot tasks and management of the hardware for reliable power-up and power-down of the resources at the device level. PMC handles device management control functions such as device reset sequencing, initialization, boot, configuration, security, power management, Dynamic Function eXchange (DFX), health-monitoring, and error management. Versal® architecture supports a rich set of pre-defined power modes. The PMC option allows you to review the estimated power for different active and power down modes of the device. For more information, see *Versal ACAP Technical Reference Manual* (AM011).

Power modeling of PMC is simplified and you should enable the Core Sub-system for any design for normal operation. In certain predefined PS low power modes such as Deep Sleep and Deep Sleep-Fast Resume, it runs on a fixed frequency of 20 MHz with 0% load (more details in XPE Power Modes section). Core Sub-system configuration also includes all the interconnect configurations in it and auto-calculates power for entire sub-system. Selecting required flash interfaces, programming and debugging interfaces for the design-associated IO will estimate power for those interfaces, typically a small amount compared to total on-chip power.

Figure 7: **Platform Management Controller**



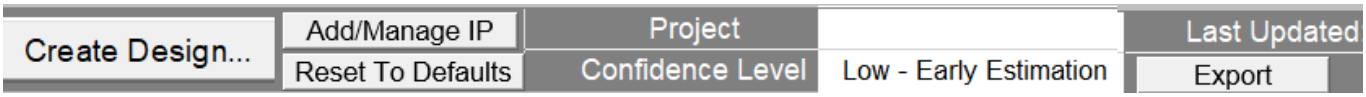
The System Monitor (SysMon) monitors the Versal device's physical environment using on-chip temperature sensors, supply sensors, external analog inputs, and an integrated analog to digital converter (ADC). In certain configuration modes, it can be operated at lower frequency during low power modes. SysMon is the only supported way to measure the device temperature and you should use it to monitor and ensure that the device temperature does not exceed its temperature range.

RECOMMENDED: *During configuration, the core load can be as high as 60% and is modeled as power-on current at maximum process in XPE. However, it would be approximately 10% during normal running state. Therefore, set the load appropriately as it is almost idle with minimal clocking and monitoring at this state. Flash and MIO interface power is calculated based on the approximate number of I/Os required for each interface and assuming 10% load across I/Os.*

Design Creation

Design entry helps you to quickly get started with Versal® device power estimation.

Figure 8: **Design Entry**



On clicking **Create Design**, the following window opens.

Figure 9: **Design Entry Window**

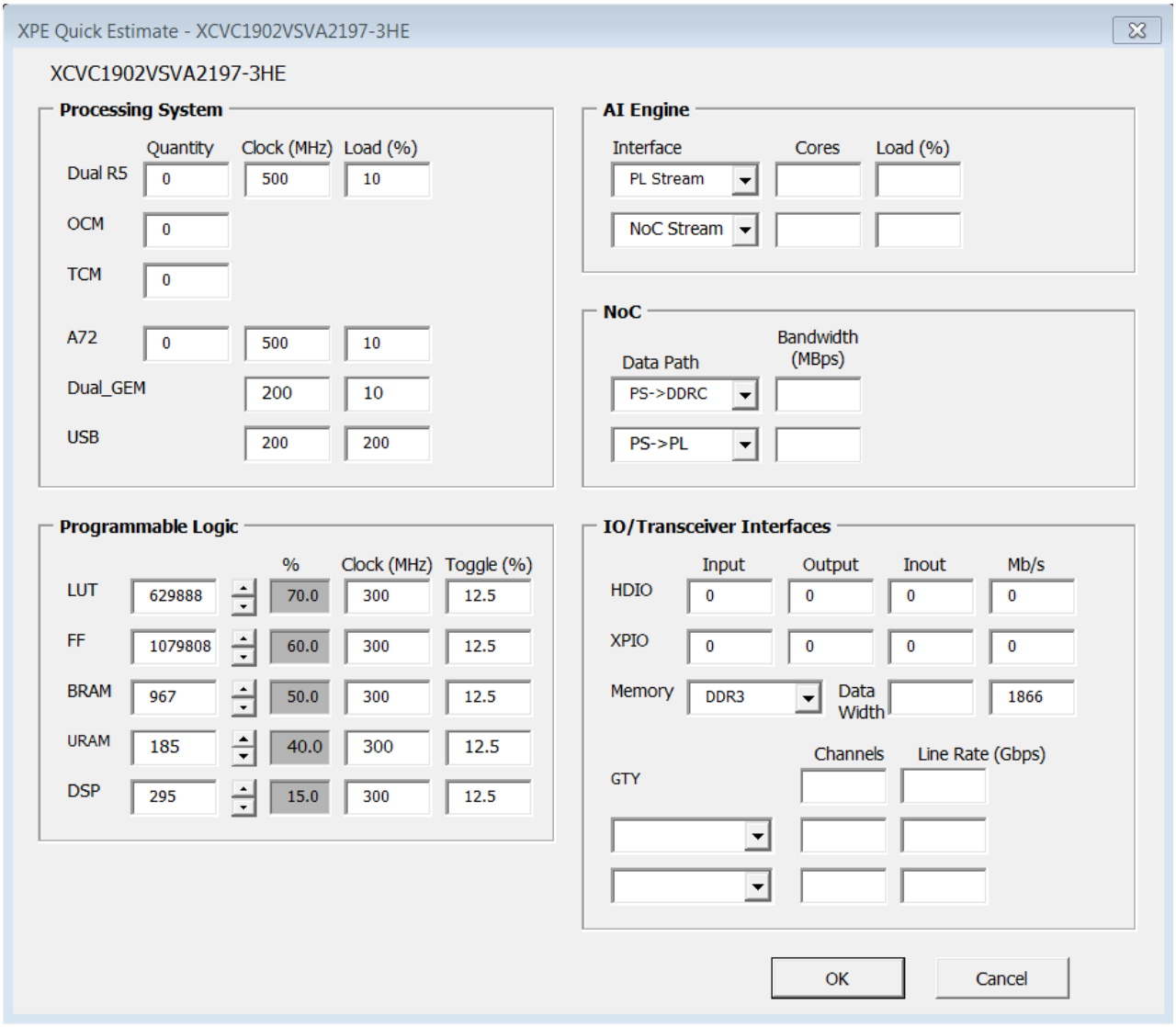
The two use models supported by Create Design are:

- Quick Estimate
- Import Flow

Quick Estimate

This wizard generates 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, and add your own entries to describe your design more explicitly. If you rerun the Quick Estimate wizard, you will replace all the spreadsheet entries from the previous run with the entries from the current run

Figure 10: **Quick Estimate Wizard**



Load or Import XPE

XPE provides multiple mechanisms to simplify data entry and manage output data depending on the design cycle. These mechanisms use the data import and data export features of XPE. For more information, see Chapter 10: Exchanging Power Information.

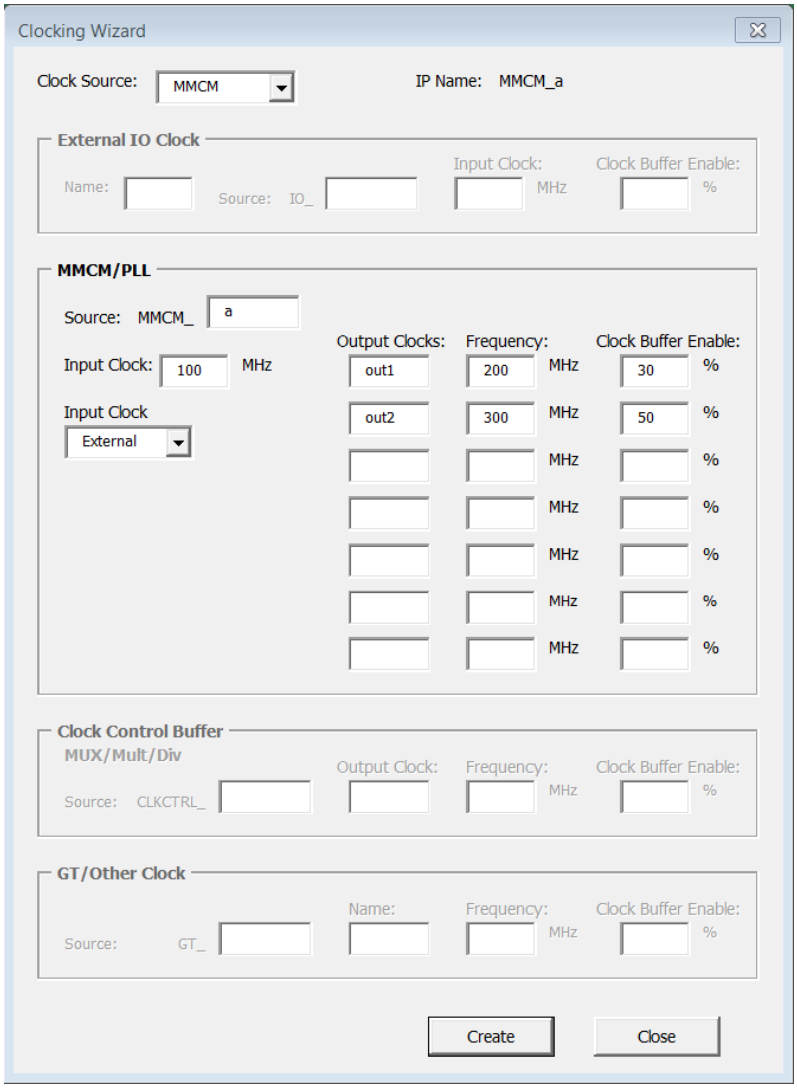
Clocking

The Clock sheet covers power estimates of clock networks and related clock generation circuits. XPE uses explicit clocks rather than arbitrary clock frequency specification. You should define each clock using the Clocking Wizard before it can be used on any other sheet. Although it requires some setup, an explicit clock has certain advantages:

- **Identification:** The unique clock name allows it to be distinguished from other clocks, particularly those with an identical frequency.
- **Ease-of-use:** Modifying a clock definition propagates changes to all sheets where the clock is used.
- **Consistency:** A clock's fanout is automatically accumulated from all sheets where it is used, resulting in a consistent estimation of clock network power on the Clock sheet.

The XPE Clocking wizard has a similar flow to that of the Clock IP wizard as shown in the following figure:

Figure 11: **XPE Clocking Wizard**



You can drive the clock using different types of sources such as:

- **External:** The clock is driven from a primary input onto a global clock resource (primitive type BUFGCE).
- **MMCM/XPLL/DPLL:** The MMCM/XPLL/DPLL generates one or more global clocks from an input clock source. In most cases the source will be either External or CLKCTRL. For ease of specification:
  - It is assumed that each generated clock is driven by a BUFGCE and it is not necessary to account for these buffers separately.
  - The MMCM/XPLL/DPLL power model requires only the input frequency and output frequency but not the internal divide and multiply counter values needed to generate each output.
- **Clock Control Buffer (CLKCTRL):** Use this option for instantiated global buffers: BUFGCTRL which are typically used as clock muxes, BUFGCE_DIV used as local clock dividers, and BUFGCE driven from logic resources.
- **GTCLK:** Use this option for global clocks sourced from GTs (type BUFG_GT).

When you define the clock, it can be selected from a drop-down menu in the Clock column of other pages.

Fanout/Site

The Fanout/Site is used to fine tune clock network power based on the number of clocked loads within a site, primarily the number of registers packed into a CLB. As total register usage rises, the packing factor increases regardless of the number of clocks in the design. This is the reason, Fanout/Site is calculated based on the total register usage for each clock in XPE and therefore different clocks with different fanouts can have same value of Fanout/Site.

You should modify the Fanout/Site value only when importing Vivado power analysis results where Fanout/Site is calculated from the actual clock load placement. For early estimation, it is recommended that you leave Fanout/Site at the default setting.

***Note*:** The default value of Fanout/Site in XPE prior to 2021.2 is different and the results from the manual population may not be consistent with import from a previous XPE version.

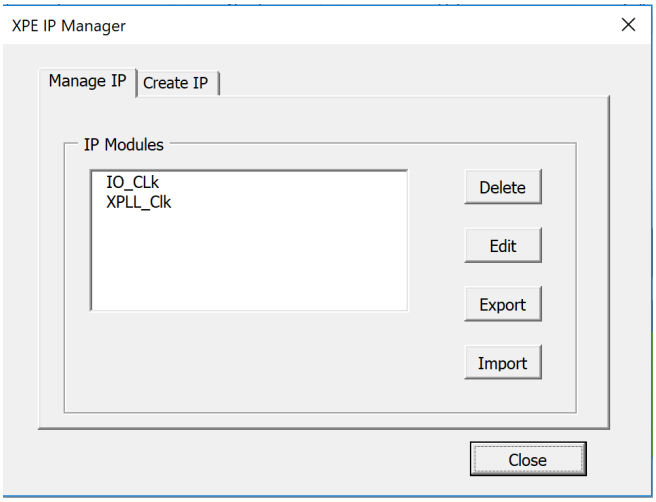
Slice Clock Enable

The Slice Clock Enable setting is used to reduce activity based on clock enables. Although the clock enable is functionally associated with logical registers, the majority of power attributed to register activity is due to the clock networks driving the registers. The clock power for registers, SRLs, and LUTRAMs are reported in clock sheet, based on the clock enable attribute. This is why the Slice Clock Enable setting appears on the Clock sheet rather than on the Logic sheet.

Edit Clocking IP

You can modify clocks created by clocking wizard using Manage IP option from IP_Manager sheet. To change clock frequency and other features, select the clock name and click **Edit**.

Figure 12: **XPE IP Manager**



You can modify the clock you created using the **Edit Clock** button in the clock sheet by selecting the clock name from the **Select Clock** drop-down list and clicking **Edit**. You can also modify clocks using the Manage IP option from the IP_Manager sheet.

Add/Manage IP

The Add/Manage IP wizard extends XPE to allow you to easily specify various types of external memory interfaces (for example, DDR4, DDR3, DDR3L, LPDDR, etc), transceiver based interfaces (for example, MRMAC, PCIe, etc), and block memory or distributed memory. The following IP wizards are available in XPE.

- Memory Generator Wizard (for Distributed Memory)
- Memory Generator Wizard (for Block Memory)
- Memory Interface Configuration Wizard (for Soft Memory)
- DDRMC Wizard
- Transceiver Configuration Wizard

Using the Memory Generator Wizard for Distributed Memory

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 **Add/Manage IP** button on the Summary sheet or the **Manage IP** button on 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 enter within XPE, see the *Versal® ACAP Configurable Logic Block User Guide*.

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. The following fields exist in the Distributed Memory tab:

- **Memory Type:** Select the type of memory your design will use.
  - Single-Port RAM
  - Simple-Dual Port RAM
  - Single-Port ROM
  - Dual-Port ROM

  ***Note*:** For a description of these memory types, see the *Versal ACAP Configurable Logic Block User Guide*.
- **Clock:** Represents the clock frequency at which the distributed memory operates. For dual-port memory types, XPE assumes the same clock frequency for both ports.

  ***Note*:** You should create clocks as a pre-step using clocking wizard and select the clock from drop-down list. Create a clock before using clocking wizard and select the clock from the drop-down list.
- **Toggle:** Represents the average toggle rate of the data signals. A toggle rate of 25% means that the data signals toggle each every fourth clock cycle.
- **Width:** Represents the bit width for each word in the memory.
- **Depth:** Represents the depth of the memory. Width × Depth is the total number of bits in the memory.
- **Registered Inputs:** Used to represent whether the memory inputs will be registered (Registered Inputs selected) or not (Registered Inputs deselected). For a description of input registering, see the *Versal Configurable Logic Block User Guide*.
- **Registered Outputs:** Represents whether the memory outputs are registered (Registered Outputs selected) or not (Registered Outputs deselected).
- **Module name:** Allows you to assign a name to the generated distributed memory configuration. This will help distinguish multiple configurations in the XPE sheets.

Once the configuration fields are filled out and created, a new row in the Logic sheet is filled in with the information you have entered in the dialog box.

Using the Memory Generator Wizard for Block Memory

The Memory Generator wizard allows you to enter block memory information in the spreadsheet. Click the **Add/Manage IP** button to access the Memory Generator Wizard on the Summary sheet or the **Manage IP** button in the IP Manager sheet. You can also click the **Add Memory** button on the block RAM sheet to access the Memory Generator Wizard. The XPE Memory Generator wizard provides a simplified method of filling in the block RAM sheet in XPE. The following fields are available in the Block Memory tab:

- **Memory Type:** The type of memory your design uses.
  - Single-Port RAM
  - Simple-Dual Port RAM
  - True-Dual Port RAM
  - Single-Port ROM
  - Dual-Port ROM
- **Clock:** The clock frequency at which the block RAM operates. For dual-port memory types, XPE assumes the same clock frequency for both Port A and Port B.
- **Toggle:** The average toggle rate of the data signals. A toggle rate of 25% means that the data signals toggle at every fourth clock cycle.
- **Port A and Port B:** If you have selected a single port Memory Type, you must enter information for Port A only. If you have selected a dual port Memory Type, enter the following information for both Port A and Port B:
- **Width:** The bit width for each word in the port.
- **Depth:** The depth of the port. `Width × Depth` is the total number of bits in the memory.
- **Enable:** The percentage of time that the port will be enabled.
- **Mode:**
  - READ_FIRST
  - WRITE_FIRST
  - NO_CHANGE
- **Module Name:** Allows you to assign a name to the generated block memory configuration. This will help you to distinguish multiple configurations in the XPE worksheets.

After it is configured and generated, a new row in the block RAM sheet and a row in the Logic sheet is filled in with the information you have entered in the dialog box. See the *Versal® ACAP Memory Resources User Guide* for more information on block RAM and memory settings.

Using the Soft Memory Interface Configuration Wizard

The Memory Interface Configuration wizard allows you to create the IPs for the I/Os involved in the interface between a Xilinx® device and external memory. It 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 is applied to the external memory. The wizard also places rows on the Clock sheet, and on the Logic sheet. Resources are added representing typical usage 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.*

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

- **Standard:** The Memory Interface Configuration wizard supports the following I/O Memory Controller Standards:
  - **Soft Memory Controllers:**
    - DDR3, DDR3L, and LPDDR3
    - DDR4
    - RLDRAM3
    - QDRII+ and QDRIV

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

- **Bank Type:** The appropriate bank type, where the choice exists between XP or HD I/O bank.
- **Mem Config:** The appropriate memory configuration.
- **Input Termination (DQ/S):** Refers to the DQ (data) and DQS (data strobe) pins. For memory interfaces using the HD or XP banks, select RTT_40, RTT_48, RTT_60 or external termination (no entry).
- **Data Rate:** The target data rate for your memory device.
- **Address Width:** The total number of address lines used in the interface, which includes Row, Column, Bank, and, if used, Rank and CS lines.
- **Data Width:** Select appropriate Data width for the selected memory interface.
- **Read/Write (%):** 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:** 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:** This option allows you to automatically generate the resources of the link layer logic for a specific memory interface. This is not applicable for Hard Memory Controllers.

- **Module Name:** Allows you to assign a name to the generated configuration. This helps distinguish multiple configurations on the I/O sheet.

Once configured and created, new rows in the I/O sheet are populated with the information you entered in the dialog box.

Using the DDRMC Wizard

For information on using the DDRMC Wizard, see DDRMC Wizard.

Using the Transceiver Configuration Wizard

The Transceiver Configuration wizard enables the IPs that are associated with transceiver interface in the GTY sheet. It provides a simplified method of filling in GTY sheets in the XPE spreadsheet. When you configure a transceiver interface using the wizard, rows are added to the IP Manager sheet and to the GTY sheet for the transceiver that is part of the physical transceiver interface. For some protocols, the wizard also adds rows in the Logic and Clock sheets representing typical resources used 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. You should manually add any options that are not available in a dialog box field. For cases where a quad has transceivers using both RPLL and LCPLL, you should manually enter different transmit and receive rates, or different power modes. The wizard covers many common protocols. However, you must modify the output of the wizard or enter the data manually in the GTY sheet for your specific case. For better accuracy, create an IP in Vivado and generate the resource information to manually enter in to XPE.*

To understand the capabilities of the Versal® ACAP GTY transceiver and the settings you will enter within XPE, see the *Versal ACAP GTY and GTYP Transceivers Architecture Manual* (AM002).

The following fields are available in the XPE Transceivers Configuration dialog box:

- **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 Modes, and Clock Source selections will default to values defined by the protocol. The Data Rate and number of Channels will also be reflected in the PCIe information 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 match. 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.
- **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:** Some protocols (for example, PCIe) have a specific width of the port that can be configured to be two, four, or eight bytes wide. With 8b/10b encoding, the port widths can be 16, 32 or 64 bits. With 64b/66b encoding, 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 *Versal ACAP GTY and GTYP Transceivers Architecture Manual* (AM002).
- **Clock Source:** Where the choice exists (as defined by the target device and data rate), you can choose to use the Versal device PLLs such as LCPLL and RPLL.
- **Module Name:** Allows you to assign a name to the generated configuration. This will help distinguish multiple configurations in the XPE worksheets.

Once configured and created, a new row in the GTY Sheet is populated with the information you entered in the dialog box.

Resets to Defaults

The Resets to Defaults option wipes out all user settings and resets XPE to factory defaults settings. This option is available on the Summary sheet of XPE.

Export

The Export option helps generate the power report in the `.xml` format (`.xpe` file) which can be imported back into the newer version of the Versal® ACAP XPE sheet. The button is available below the last update section on the summary sheet. For more information, see Exchanging Power Information.

Labels

The following types of labels are available for XPE:

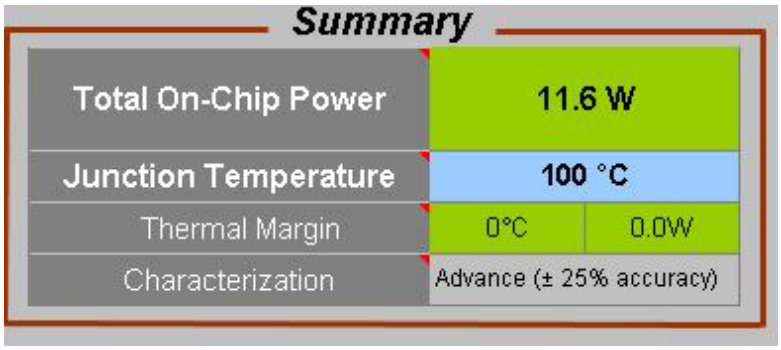
- **Project:** This is an editable field at the top of the Summary sheet that allows you to add a description of your design. For import flow from Vivado, this field lists the following information:
  - Top level design name of the Vivado project
  - Vivado version for which the design is implemented

- **Confidence Level:** This field shows the confidence about the accuracy of power report. For manual entry mode, which is an early estimation, the confidence is Low. When importing report_power results from Vivado, there are two values, Medium for the default analysis based on vectorless propagation and High for SAIF simlulation-based analysis.
- **Last Updated:** This indicates the dates on which the XPE design was created. If the design configuration is untouched, it indicates using color coding that the data is outdated. This is a unique feature to enforce refining power estimation at regular intervals to increase the confidence in power estimation. For Vivado import flow, it displays the date of report power generation in Vivado.

Power and Thermal Dashboard

The Versal® ACAP Power and Thermal dashboard is very similar to that of previous device families. A Characterization label displays the characterization status of the selected device. The following figure shows the summary of Power and Thermal reporting.

Figure 13: **Power and Thermal Reporting**

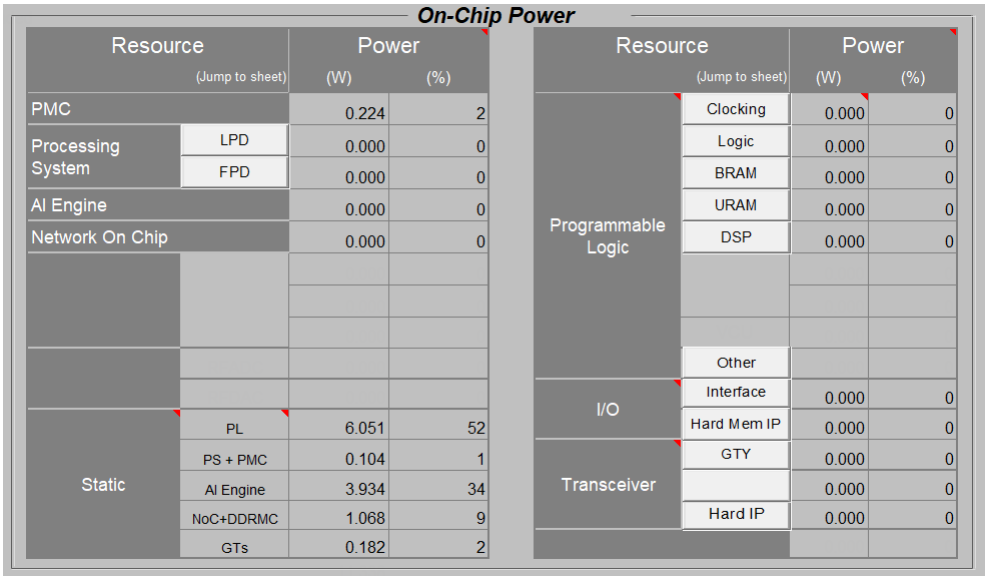


Thermal and power margins are measured against the maximum accepted ranges for this device grade. Thermal margin is negative when the estimated junction temperature exceeds the maximum specified value for the device.

On-chip Power Map by Components

The On-chip Power map by components is organized as groups. Hard blocks are categorized under I/O and Transceiver groups.

Figure 14: **On-chip Power Map**



On-Chip Power

Resource		Power (W)	Power (%)
PMC		0.224	2
Processing System	LPD	0.000	0
	FPD	0.000	0
AI Engine		0.000	0
Network On Chip		0.000	0
Static	PL	6.051	52
	PS + PMC	0.104	1
	AI Engine	3.934	34
	NoC+DDRMC	1.068	9
	GTs	0.182	2

Resource		Power (W)	Power (%)
Programmable Logic	Clocking	0.000	0
	Logic	0.000	0
	BRAM	0.000	0
	URAM	0.000	0
	DSP	0.000	0
	Other		
I/O	Interface	0.000	0
	Hard Mem IP	0.000	0
Transceiver	GTY	0.000	0
		0.000	0
	Hard IP	0.000	0

Processing System Power

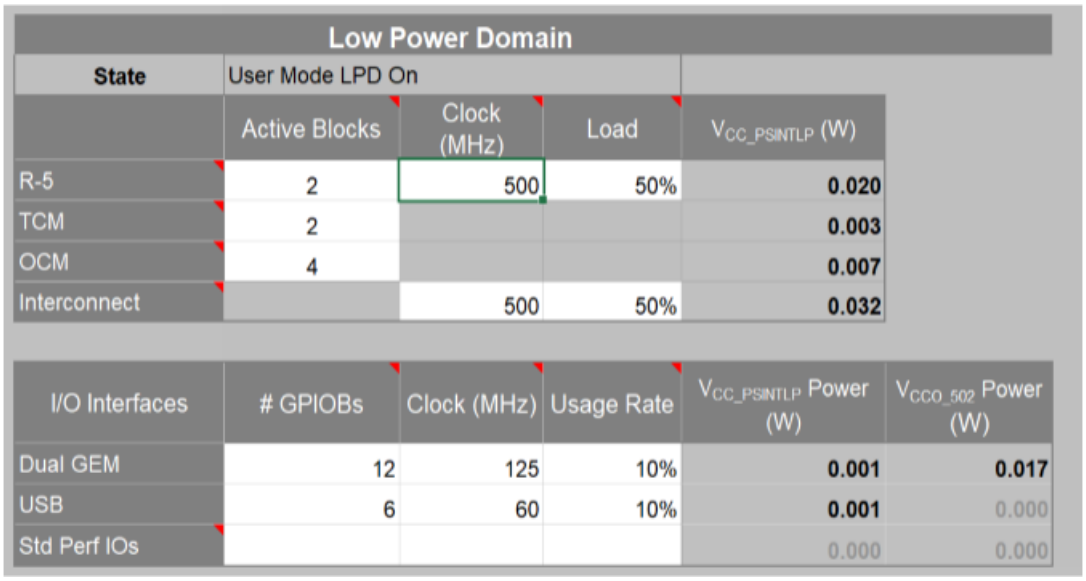
Using the Processing System Sheet

Versal® ACAP integrates a feature-rich 64-bit dual-core Arm® Cortex®-A72 and dual-core Arm Cortex®-R5F based processing system (PS), Xilinx programmable logic (PL) architecture, and AI Engine in a single device.

Low Power and Full Power Domains

The PS (Processor Subsystem) sheet is divided into two domains: Low Power Domain and Full Power Domain. These power domains can be turned on and off. The following figures show the Low Power and Full Power domains.

Figure 15: **Low Power Domain**



Low Power Domain				
State	User Mode LPD On			
	Active Blocks	Clock (MHz)	Load	$V_{CC\_PSINTLP}$ (W)
R-5	2	500	50%	**0.020**
TCM	2			**0.003**
OCM	4			**0.007**
Interconnect		500	50%	**0.032**

I/O Interfaces	# GPIOBs	Clock (MHz)	Usage Rate	$V_{CC\_PSINTLP}$ Power (W)	V_{CCO_502} Power (W)
Dual GEM	12	125	10%	**0.001**	**0.017**
USB	6	60	10%	**0.001**	0.000
Std Perf IOs				0.000	0.000

Figure 16: **Full Power Domain**

Full Power Domain				
State	Power On			
	Active Blocks	Clock (MHz)	Load	$V_{CC\_PSINTFP}$ (W)
A-72	2	1000.0	80%	**0.285**
Interconnect		550.0	1%	**0.016**
CCI		550.0	1%	**0.006**

Processor and PLLs

The PS for Versal architecture integrates a feature-rich 64-bit dual-core Arm® Cortex®-A72 (APU) for full power and dual-core Arm® Cortex-R5F (RPU) based processing system (PS) for low power domains. APU PLL is available in the full power domain and generates clocks for Arm Cortex-A72 core, L2 Cache, FPD Interconnect, and CCI. RPU PLL is available in the low power domain and generates clocks for Arm Cortex-R5F core, TCM, OCM, and LPD Interconnect.

Note: L2 cache must be enabled when using A72s, XPE does this automatically and adds power to FPD.

Memory and I/O Interfaces

The Arm® Cortex-A72 and Cortex-R5F CPU systems also include on-chip TCM, OCM memory, L2 Cache, and a rich set of peripheral connectivity interfaces.

CCI (Cache Coherent Interconnect)

The CCI refers to the block which combines part of interconnect and coherency functions into a single block. The Load field value can range from 0% -100% depending upon the application. The value for Load is the same as the Load for Interconnect. The maximum permissible frequency is the same as the range of APU frequency for a corresponding speed grade.

LPD I/O Interfaces

The three different low power domain I/O interfaces that are supported are listed here:

- **Dual GEM:** Versal device PS has 2 GEMs (Gigabit Ethernet MAC) shared between PS and PMC. The field GPIOBs represents the number of pairs of I/Os used. Each GEM needs six pairs of I/Os (12 I/Os) with a maximum clock frequency of 125 MHz. Hence, there is an option of six or twelve (pairs) for using single or both the GEMS respectively. The usage rate should be entered with expected switching rate of the interface.
- **USB:** PS and PMC has shared USB2 support. The number of I/Os required for USB data transfer is six pairs. Hence it has the option of 0 and 6. The usage rate should be entered with expected switching rate of the interface. The maximum operating frequency is 250 MHz.

- **Std Perf IOs:** The number of I/Os being used should be entered in pairs. The maximum frequency permitted is 450 MHz. The usage rate should be entered with expected switching rate of the interface.

  ***Note*:** The maximum number of I/Os per Bank is 26 (13 pairs). Each GEM interface requires 6 pairs or 12 I/O's and these I/Os are located in PMC Bank1.

XPE Power Modes

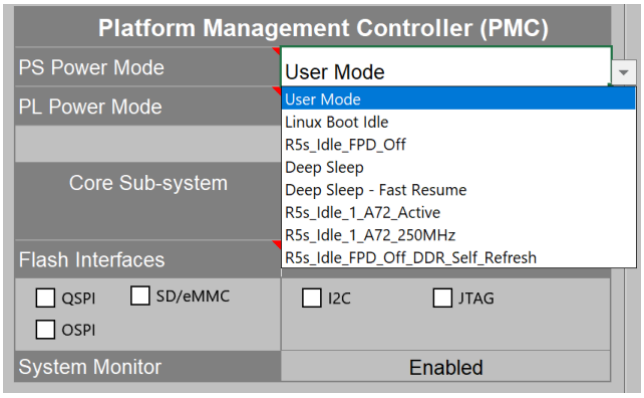
Versal® devices support different power modes of operation in XPE. The predefined independent power modes can be categorized into PS (Processing System) and PL (Programming Logic) power modes. You can select a power mode based on the power saving requirements of your application. The power modes are described in this section.

PS Power Modes

The PS power modes are described as follows.

- **Full Power Domain (FPD):** Primarily comprises of Dual A72 cores, L2 Cache, FPD Interconnect, and CCI.
- **Low Power Domain (LPD):** Primarily comprises of Dual R5 cores, TCM, OCM, LPD Interconnect I/Os.

Figure 17: **PS Power Modes**



The predefined PS power modes are described in the following table.

Table 5: **PS Power Modes**

PS Power Modes	Description	Recommended Use
User Mode	This mode has no predefined configuration. You can configure the complete PS Module based on your design requirements.	This mode is used to define the expected full processing mode of the processing subsystem. Also, it is used to estimate the worst case power for the processing subsystem.
Linux Boot Idle	• FPD powered on, APU's idle • LPD powered. RPU's idle that is load @0%	This mode emulates the linux boot time and the corresponding configuration can be used to determine power while booting.
R5s_Idle_FPD_Off	• FPD is powered off • LPD is powered on and RPU is idle where the load is 0%	This mode is suggested for PL based designs which may need LPD rarely and APU is never used. This results in total power saving of FPD and dynamic power saving of LPD during operation.
Deep Sleep	• Both LPD and FPD are powered down, both static and dynamic Power is 0 • PMC is idle at 20 MHz	This mode is suggested for power saving when the design is mainly PL based.
Deep Sleep - Fast Resume	• FPD is powered off, FPD staticPower is 0 • LPD is powered on but RPU is off/ clock gated • PMC is idle at 20 MHz	
R5s_Idle_1_A72_Active	• RPU is idle • One of the APU cores is idle • The second APU core is operational with a load of 80%	Used to save power of the RPU core and 1APU core
R5s_Idle_1_A72_250MHz	• LPD is on with RPU idle where the load is 0% • FPD is shutdown. The static power of FPD is 0	This configuration is useful if none of the blocks in the FPD are used in the design, hence fully shutting down the FPD saves static power. It is mainly used for real-time applications only
R5s_Idle_FPD_Off_DDR_Self_Refresh	• FPD is powered off • LPD is powered on and RPU is idle where the load is 0% • DDR is in self refresh mode	

***Note*:** PMC is never completely powered off in any of the above modes.

PL Power Modes

- **User Mode:** You can define this mode similar to PS User Mode. All the PL voltage rails are operational. You can manually change any setting in this mode.
- **Clock Gated Mode:** In this mode, the PL clocks are gated the same way as a zero frequency clock for Logic, BRAM, URAM and DSP. Hence dynamic power of these blocks is zero but static power is present. This mode does not affect NOC-DDRMC, AI Engine, and GT which remain fully operational. This mode is suggested for power saving when the design mainly depends on AI Engine, GT, and NOC-DDRMC.

- **Powered Off Mode:** All the PL power rails including AI Engine, NOC DDRMC, and GT are zero. Hence, no PL static power is reported in this mode. This mode is a suggested power saving mode when the design is mainly PS based.

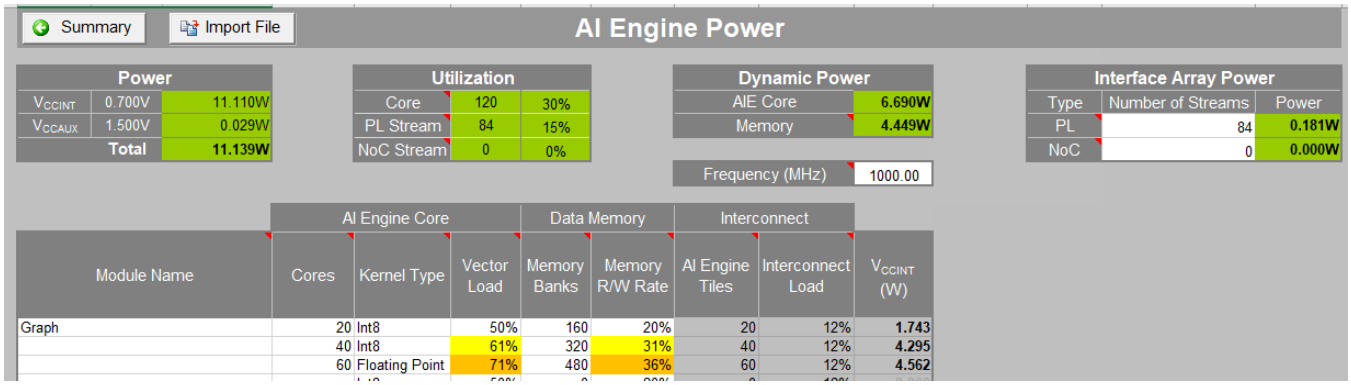
AI Engine Power

The AI Engine array introduced in the Xilinx® Versal® architecture caters to solutions for high compute or complex DSP intensive applications, like 5G Wireless or Machine Learning algorithms. AI Engine is a high performance VLIW vector (SIMD) processor with integrated memory and interconnects to help communicate with other AI Engine cores that are connected together in a two dimensional array network in the device.

Power Estimation by XPE

The AI Engine sheet in XPE for Versal ACAP is available for the AI Core Series family. XPE helps you estimate the power consumption of AI Engine blocks for a particular configuration. The following figure shows the AI Engine Power interface:

Figure 18: **AI Engine Power Sheet**



For an early power estimation, you should provide the configuration details of the AI Engine array such as clock frequency, number of cores, kernel type, and the Vector Load average percentage for the cores. The supported kernel types are Int8, Int16, and Floating Point.

TIP: *When considering the Vector Load percentage, the average loading percentage should be used. The kernel may be using 100% of the available core run-time, however overhead from pre-Fetch, memory accesses, NOPs, stream, and lock stalls should be considered. The recommended range is 30% to 70%.*

Data Memory and Interconnect Load fields are auto-populated based on the number of AI Engine cores used and can be overridden based on the application requirement. There are eight memory banks in an AI Engine Tile (each bank is 4 KB in size totaling 32 KB per tile), by default, XPE uses all of them, this can be overridden if the application requires fewer bank accesses. Memory R/W rate is average Read/Write memory access for each bank.

TIP: *The Memory R/W rate is an average value. XPE uses 20% by default. Recommended value range is 10% to 30%.*

The AI Engine array interface allows access to rest of the Versal® ACAP, there are interface tiles for both the Programmable Logic (PL) and Network On Chip (NoC), these interfaces tiles are represented as streams. You can override the PL/NoC streams based on your design and application. The interconnect fields are read-only and calculated based on your input. PL streams show the available streams in the first row of AIE tiles and allows you to specify the number of 64b PL streams that are utilized. It is recommended that PL streams are set at default 14 streams per 20 AIE tiles used. However, PL streams can be changed, you can see a DRC (cell turns orange) when the PL streams exceed the available streams within the total AIE array. Interconnect load is averaged to a fixed value of 12% and has minimum impact to power and can be overridden by import flow described in the next section. The maximum range for clock speed depends on the speed grade of a device with 1300 MHz for -3H grade. For more information, see the *Versal ACAP AI Engine Architecture Manual* (AM009).

TIP: *There are multi level DRCs for Vector Load and Memory R/W Rate. Yellow indicates values are in the higher range of typical application and orange indicates the values are higher than typical expected applications.*

Import Flow

Compiler Flow

Vitis™ software platform generates an `.xpe` file that can be imported to provide an accurate starting point for AI Engine power estimation. Once imported, all the configuration is generated and power can be estimated more accurately compared to the manual entry mode. The `.xpe` file generated by the Vitis™ software platform (in `Work/reports` directory), when imported, averages out Vector Load and Memory R/W rate for a particular kernel type. For example, all the cores of kernel type INT8 vector load and the R/W rate is averaged and populated in a single row of XPE. The Interconnect Load is not a default of 12% in the import flow. Instead, it is computed by the tool based on the stream usage for each AI Engine tile.

***Note*:** You can import the `.xpe` file to the AI Engine sheet using the Import button at the top.

AI Engine Simulation Based Power Estimation

You can use the simulation flow to get more accurate power analysis of the AI Engine. You can run the AI Engine simulator after the AI Engine compiler flow. Perform the following steps for simulation based power estimation of the AI Engine:

1. Generate a `vcd` file while running AI Engine simulator.
2. Use the vcdanalyze tool to create more accurate `.xpe` file using the following command:

   ```
   vcdanalyze --vcd <vcdfile> --xpe
   ```

   a. <vcdfile> is the generated `.vcd` file from the AI Engine simulator. The above command generates a new `.xpe` file in `aiesim_xpe` folder.
3. Import the newly generated `.xpe` file to XPE tool for more accurate AI Engine power estimation.

   a. This `.xpe` file has more accurate vector load and memory R/W rate.

You need to consider the following before performing simulation based power estimation of the AI Engine:

- Presently only vector instructions such as VMAC and VMUL are supported in this flow. Therefore, simulation flow should only be used if your design has vector instructions. If the AI Engine has scalar instructions (not the expected use of AI Engines), it is recommended to go with the compiler import flow.
- While using the vcdanalyze utility on a `vcd` file, navigate to the location of the `.vcd` file as the work directory is also required to generate `.xpe` output.
- To create a `.xpe` file in a specific directory, use the following command:

  ```
  vcdanalyze --vcd <vcdfile> --xpe --xpe-dir <dir_name>
  ```

  where <dir_name> is the directory name where generated .xpe file is expected.

For more information on how to generate the `.vcd` file and vcdanalyze tool, see *Versal ACAP AI Engine Programming Environment User Guide* (UG1076).

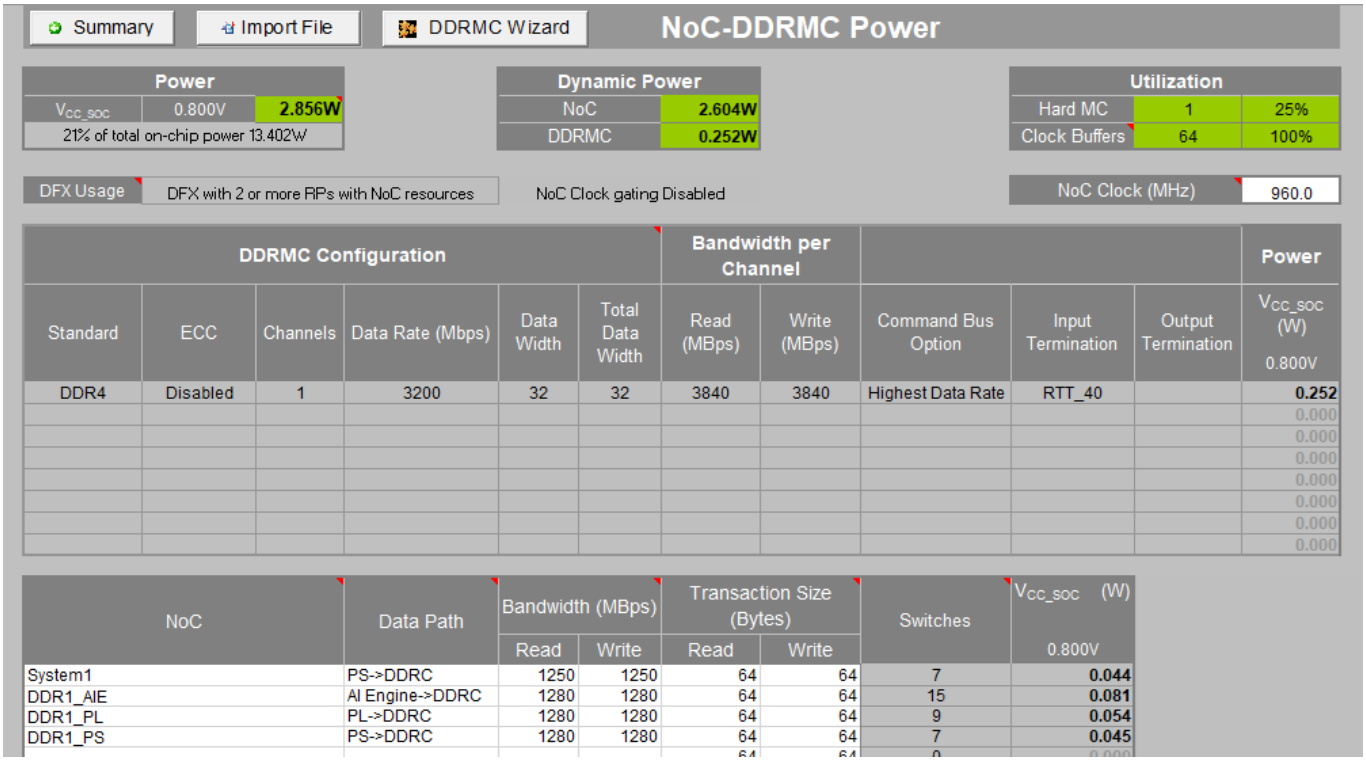
Network-On-Chip and DDRMC Power

NoC is a new connectivity hard block of the Versal® architecture that replaces some aspects of the fabric and fabric logic with a narrower high speed network. It provides connectivity between PMC, DDRMC, CPM, PL, AI Engine, and PS. It also provides a configuration function to the blocks that it interfaces with.

Power Estimation in XPE

The Versal ACAP XPE has a sheet for NoC and DDRMC power estimation. A dedicated power rail VCC_SOC is used for both NoC and DDRMC hard blocks. The hardened DDRMC can only be accessed using NoC irrespective of the master type. The following figure shows the NoC-DDRMC sheet as part of Versal ACAP XPE:

Figure 19: **Network-On-Chip Power Sheet**



The NoC block power depends on various factors of your design. The following section describes the properties that you should specify into the NoC_DDRMC sheet to get the estimated power.

Table 6: **NoC Property Description**

Property	Description
NoC Clock	Clock frequency of NoC operation in MHz, you can edit it based on the allowed range, speed grade, and voltage selected. Fmax of the NoC clock is 1080MHz in a -3H device.
Data Path	Select the data path between Master and Slave for which NoC power is being estimated. This entry has the drop-down list of around 17 valid data paths between different masters and slaves available in Versal devices like PS, PL, AI Engine, PMC, CPM, and DDRMC.
Bandwidth	This field is to specify the read and write bandwidth requirement for that particular data path. The unit is MB/s and the maximum bandwidth supported is 19200 MB/s.
Transaction Size	Specify the transaction size of the traffic data for read and write interface being transferred through NoC. This size is in Bytes and the maximum size supported for write and read transaction is 64 Bytes.
Switches	This field is auto populated based on your input and this represent the average number of NoC programmable switches required in the given data path.
Clock Buffers	This field is auto populated based on your data path input. It determines the average number of Clock Buffers used for a path. It adds up per path and maximizes to total available for the device. NoC Clock Power is directly proportional to the number of Clock Buffers used.

DFX Usage and NoC Clock Gating

The state of NoC clock gating enabled/disabled i.e. number of used NoC clock buffers are dependent of the DFX usage of the design. The mode of DFX used can be selected from DFX Usage section of the Summary Tab.

NoC clock gating can only be enabled if DFX is unused or only 1 reconfigurable partition (RP) is used in the design. This setting is with respect to the NoC containing elements in the RP, if the NoC is static for all RPs then select DFX is unused. If 2 or more RPs are used that contain NoC elements then NoC clock gating is disabled, this means all NoC clock buffers are turned on, XPE then accounts for the additional NoC clock power in the VCC_SOC rail. Following is the detailed description of each mode:

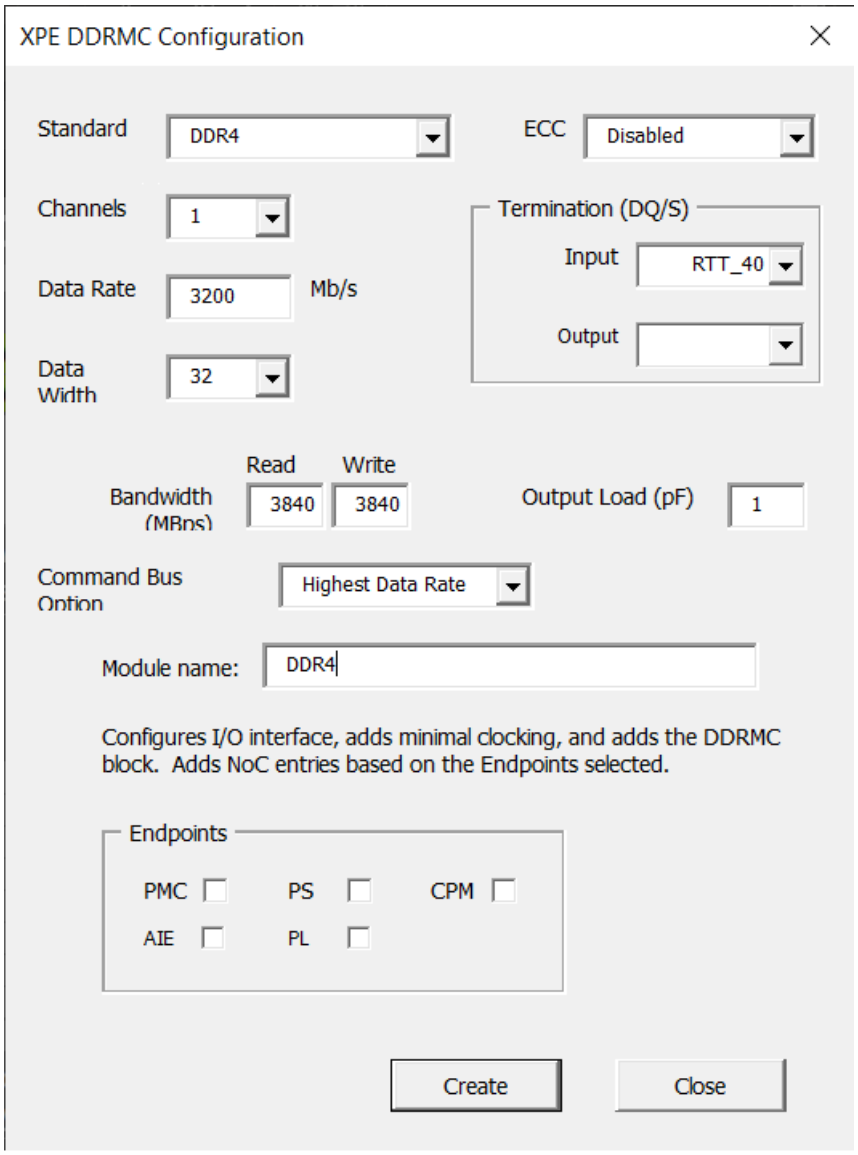
- **No DFX, or no RPs include NoC resources (Default):** Select this option if DFX is not used in your design or if DFX is being used but none of the RPs contain NoC resources.
- **Classic SoC Boot mode:** This mode allows users to configure the PS and System Domains and boot Linux, before the PL. To enable this mode a single DFX RP is used for the PL. You must connect an external memory via the SYSTEM domain. NoC clock gating is still possible, but you must specify the DDRMC and NoC paths you plan to use in the NoC_DDRMC tab for XPE to correctly estimate the SYSTEM domain power.

- **DFX with 1 RP that includes NoC resources:** Select this setting when using DFX and if there is 1 RP that contains NoC resources
- **DFX with 2 or more RPs with NoC resources:** When 2 or more RPs contain NoC resources, NoC clock gating is not possible, select this option to allow XPE to correctly account for the additional VCC_SOC power.

DDRMC Wizard

The dedicated, hardened DDRMC can be configured using the DDRMC Wizard in the NOC-DDRMC sheet. Creating a DDRMC using the wizard creates the required DDR I/Os in the I/O sheet. The following figure shows the DDRMC Configuration wizard:

Figure 20: **XPE DDRMC Configuration**



The parameters in the DDRMC wizard are:

- **Standard:** Hardened DDR memory controllers support only DDR4 and LPDDR4 standards. (For DDR3 and other standards, use Soft DDR).
- **Channels:** The number of channels. Values range between 1 and 2.
- **Data Rate:** The maximum permitted data rates for standards.
  - DDR4 - 3200 Mb/s
  - LPDDR4 - 4266 Mb/s
- **Data Width:** The possible values of Data Width are 16, 32, and 64.

- **ECC:** This field represents the 8 bit error correction code. This field can either be set to Enabled or Disabled.
- **Bandwidth (MBps):** Read and Write bandwidth in MB/s (Mega Byte per second). The sum of read and write bandwidth {Data Rate (Mb/s)*Data Width} cannot exceed total bandwidth.
- **Command Bus Option:** The address width is fixed in the hard memory controller but there are different command/address busses to trade off between Fmax and pin count. In the current release of XPE, only the Highest Data Rate option is supported. The Minimize Pin Count option will be supported in later releases.
- **Endpoints:** The DDRMC wizard allows you to quickly add NoC connectivity with the DDMC, you can select endpoints such as, PMC, PS, CPM, AIE, and PL. When selecting applicable endpoints, the bandwidth entered in the DDRMC wizard is divided equally among them, this can be altered in the NoC page if needed.

***Note*:** To create NoC path to different DDRMCs, different module names should be given. Else, it generates path to the same DDR controller.

NoC Power Estimation Flow

For the current release of the XPE, following are the two ways to estimate NoC power from Vivado:

- **Manual Entry:** For early power estimation when a Vivado design is not yet available or ready.
  1. Manually specify the configuration of NoC data path as described above to get the estimated power.
  2. Create a DDR interface using the DDRMC Wizard if you are using a hard DDRMC in your design.

  ***Note*:** For fewer paths, number of clock buffers may be under reported. It is suggested to import clock buffers from Vivado whenever possible.
- **Importing from Vivado:** If there is an IP integrator design available in Vivado with NoC present as an IP, use this flow for NoC power estimation. In this flow, Vivado generates the `.xpe` file with all the information required for NoC power estimation.
  1. In the Vivado IP integrator design, when validate_bd_design is run, the `NOC_Power.xpe` file is generated with all the NoC configurations.
  2. Once the `.xpe` file for NoC design is generated, import this `.xpe` file into the Versal® ACAP XPE NoC_DDRMC sheet.
  3. The design taken from the design flow has NoC configuration embedded in the exported .xpe file which can be imported to XPE from summary sheet.

The power estimated by this flow is more accurate than the manual entry mode. In this flow, the NoC configuration is being populated from the actual NoC design and the number of switches are very close to what has been used in the design.

Programmable Logic Power

The Programmable Logic consists of the following components:

- Logic
- Block RAM
- UltraRAM
- DSP

You should define the clocks first so that they can be associated with the resource entries on the other sheets. Organization of data entries is important for accurate power estimation. This allows you to avoid underestimation resulting from parts of the design being overlooked and to avoid overestimation by duplication (entering the same resources more than once). Use the following common strategy for estimating the power of large designs:

- Enter rows of resources corresponding to each major hierarchical design block, using the block name or instance name as a base name for the row.
- Enter at least one row of resources for each clock within each hierarchical block.
- For logic on clock domain crossings, group the logic with its associated clock.
- Within each clock domain, use multiple rows for resources with different Toggle Rates.

Logic

The Logic sheet covers power estimates of CLB logic: LUTs and Registers as shown in the following figure. Each row represents a group of logic that is associated with:

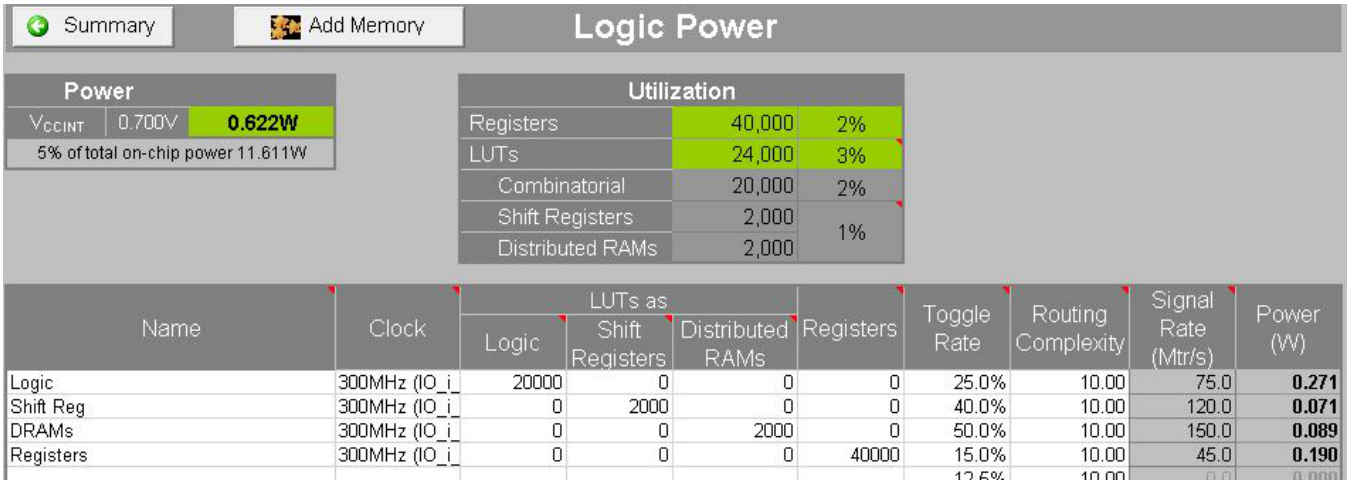
- A particular Clock whose frequency is used to calculate dynamic power.
- A Toggle Rate which represents an average over the inputs and outputs of all logic.

LUTs fall into the following three categories while Registers are primarily the CLB registers.

- **LUT as Logic:** For simplified entry, XPE assumes an average sized LUT of about five inputs and also assumes a small percentage of LUTs use two outputs.
- **LUT as Shift Registers:** SRL primitives.

- **LUT as Distributed RAMs:** LUTRAM primitives.

Figure 21: **XPE Logic**



Both Shift Registers and Distributed RAMs use the M-type CLB LUTs that you can configure as memory. It may be difficult to estimate the total LUTs needed for Distributed RAM-based memories. Instead use the Add Memory button to launch the XPE Memory Configuration wizard. Specify the memory array size, clock and options and XPE calculates the expected number of LUTs and registers and enters them into a row. Toggle Rate is defined as the percentage of clock cycles where a transition occurs. The default value of 12.5% means one transition every eight cycles.

Routing Complexity is an abstract model of the interconnect power. The number represents the average number of routing resources per logical net. A design with higher complexity requires more routing resources per net which increases power. Routing Complexity is typically only modified when importing Vivado power analysis results where Routing Complexity is calculated from the actual routing resources used to route the design. For early estimation, Xilinx recommends that you leave Routing Complexity at the default setting.

Block RAM

The block RAM sheet covers power estimates of the dedicated 36 kb block RAMs which are used to implement much larger memory arrays than the memories that are feasible with LUTRAMs. Block RAMs typically comprise an array of a particular depth, width, and cascade height. The Memory Configuration wizard available from the Add Memory button is recommended for entering large arrays because it closely matches the synthesized results. Often synthesis chooses certain architectural trade-offs to balance performance, area, and power which may be slightly different than expected. Following are descriptions of the columns used for block RAM size configurations.

- **Cascade Group Size:** block RAMs have dedicated cascade circuitry for efficiently building deeper memories without incurring extra logic cost for addressing, decoding, and multiplexing. Most importantly cascading block RAMs results in significant dynamic power savings as only one block RAM of a cascade group is active at a time. There is a trade-off between power and performance which is reflected in the XPE default cascade group size of 4. Synthesis creates cascades as deep as 8 (least power) and as shallow as 1 (best performance).
- **Mode:** This represents the configuration of the block RAM size and error correction mode if used. The different sizes modeled by XPE include:
  - RAMB18: 18 kbit capacity with width up to 18 bits
  - RAMB36: 36 kbit capacity with width up to 36 bits
  - RAMB18SDP: A Simple Dual-Port mode with fixed width of 36 bits
  - RAMB36SDP: A Simple Dual-Port mode with fixed width of 72 bits

  The SDP modes can be configured to enable Error Correction Coding circuitry. Also note that the SDP RAM modes are not always used for simple dual-port RAMs. It depends on whether or not the port Bit Width fits efficiently into the fixed widths of the SDP mode configurations: 36 bits wide for RAMB18 and 72 bits wide for RAMB36. Intermediate widths are more likely to be mapped to RAMB18 and RAMB36 modes.
- **Bit Width:** This is the data width of each block RAM port with choices limited to width values that are supported by the RAM block configuration.
  - For intermediate widths, select the next higher width.
  - For SDP mode configurations you must select a width of 36 for RAMB18SDP and 72 for RAMB36SDP.
  - Asymmetric widths are supported.

Once the structural configurations are determined, additional settings configure the activity rates affecting dynamic power.

- **Clock:** Each port can be clocked by the same clock or independent clocks. As soon as a clock is assigned, the respective port consumes power, even when disabled. This is because the clock network is still active in parts of the block RAM circuit.
- **Toggle Rate:** This is the percentage of clock cycles where block RAM data is toggling. This is the average value of all input and output data pins over all block RAMs in the row entry.
- **Write Mode:** This mode determines what happens when there is a simultaneous write and read to the same address. The default value of NO_CHANGE results in the lowest power as there is no transition seen at the block RAM outputs. The other modes of WRITE_FIRST and READ_FIRST result in higher dynamic power because output data undergoes transitions to the Write data or Read data respectively.
- **Enable Rate:** This is the percentage of time a port is active versus on standby. When a port is enabled, it consumes more power.

- **Write Enable:** This is the percentage of time a port is being written, independent of the Enable Rate. So, make sure Write Enable <= Enable Rate.

To minimize power for RAMs, the enable logic should be designed such that ports are only enabled when necessary. A port that is constantly enabled (100% Enable Rate) wastes power. Vivado block RAM power optimizations enforce this recommendation with logical optimizations. For example a write-only port with a Write Enable active for 25% of cycles and Enable tied high could be estimated using a 100% Enable Rate for that port. However, Vivado swaps the Write and Enable control signals in the logical netlist such that the Write Enable is tied active (Write Enable 100%) and the Enable is driven by the Write Enable logic (Enable Rate 25%). Review entries carefully to ensure the Enable Rate is minimized.

UltraRAM

The UltraRAM sheet covers power estimates of the dedicated 288 Kb UltraRAM block. The UltraRAM in Versal® ACAP has more flexibility than UltraScale+™. The Add Memory IP wizard does not yet support UltraRAM. So, the resources must be estimated or calculated manually. The settings that define UltraRAM structure include the following:

- **Cascade Group Size:** UltraRAM blocks support cascading to create larger memory arrays while reducing the overall power by enabling only one UltraRAM of a cascade at a time. Example: 20 UltraRAM blocks with a Cascade Group Size of 4 represents 20/4 = 5 sets of cascaded UltraRAM of 4 blocks each. If there is no cascading, use 1 as the value of cascade group size. The Cascade Group Size applies to vertical cascading, increasing array depth.
- **Latency:** The optional UltraRAM pipeline registers are IREG_PRE (input) or REG_CAS (cascade). The default value is Cascade Group Size divided by 3. If there is no UltraRAM cascading, only IREG_PRE can be used which corresponds to a Latency of 1.
- **Mode:** Chooses between `URAM288 (no ECC)` and `URAM288_with_ECC`.

The following settings define the UltraRAM activity:

- **Sleep Rate:** The percentage of time the UltraRAM 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 UltraRAM or UltraRAM module.

Following are the values specified for the UltraRAM 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 UltraRAM 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 UltraRAM RDB_WR_A and RDB_WR_B pins.

***Note*:** 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%.

DSP

The DSP sheet covers estimation of DSP58 block resources. Similar to previous generations, a Versal® ACAP DSP block can implement a wide variety of arithmetic and logical functions ranging from addition, subtraction, multiplication, and also common DSP functions such as multiply-accumulate. Like previous generation DSP blocks can implement wide logic functions such as XOR and can be cascaded to form digital filters. The Versal ACAP DSP block has a wider 27x24 complex multiplier that can also be configured as three 9x8 multipliers and also has a wider 58-bit accumulator. Versal ACAP DSP also supports floating point addition and multiplication.

The following settings descriptions are used to configure Versal DSP blocks for power estimation:

- **Configurations:** Versal ACAP XPE allows you to use mode-specific configurations for the DSP block. You can select the suitable configurations from the drop-down list based on the DSP operation that you are performing. The Versal DSP58 power model supports much more fine-grained accuracy compared to previous generations. You can choose from various sizes of integer mulitpliers, MACs, dot-products, complex multipliers, and floating point operations.
- **DSP58 Slices:** This is the number of DSP58 blocks. In the Versal architecture, one DSP58 block can implement a 27x24 fixed-point multiplier while two DSP58 blocks can be paired with common logic to implement an 18-bit complex multiplier.
- **Clock:** Choose the DSP58 Slice clock from the drop-down menu.
- **Block Toggle Rate:** This is the average toggle rate of all DSP block signals. Manually adjust the toggle rate when necessary.
  - If the DSP block is enabled for only a fraction of cycles, scale the Block Toggle Rate by the enable rate. For example if the DSP is enabled for half the cycles, multiply the Block Toggle Rate by 0.5 to get the new Block Toggle Rate.
  - If the DSP block does not use all the multiplier outputs, scale the Block Toggle Rate by the fraction of output bits used. If only 48 bits are used then multiply the Block Toggle Rate by (48 / 58) to reflect the proportion of actively switching signals in the DSP block.
- **DSP Mode:** This indicates the operational mode of the DSP blocks. It is auto populated and is read only for the configuration that is specified.

  - **INT24:** This mode is compatible with the DSP48 from previous generations. INT24 indicates the DSP block is configured as a 27x24 signed, fixed point multiplier. If using a smaller sized multiplier, scale the Block Toggle Rate by the proportion of used output bits.
  - **INT8:** DSP58 uses the Vector Fixed Point ALU mode in this configuration. This mode is used for computing three-element 9x8 vector dot-products with accumulate or post add options.
  - **CINT18:** This mode indicates that two adjacent DSP58 blocks are configured to implement an 18-bit complex multiplier. Ensure that the DSP Slice total takes into account two DSP Slices per complex multiplier.
  - **FP32:** DSP58E5 uses floating point multiplier and adder in this configuration. This mode is used for FP32 single precision or FP16 half precision with accumulate or post add options.
- **MULT Used?:** This indicates whether or not the DSP58 multiplier is used. The default value is Yes because the multiplier is expected to be used for the majority of cases. Set the value to No for non-multiplication use cases. It is auto populated and is a read only for a given configuration.
- **Multiplier Pipeline Used?:** When MULT Used is Yes, this indicates whether or not the multiplier is pipelined. The multiplier is typically pipelined due to its relatively large propagation delay so the default value is Yes. The value should be set to No only for very low clock speeds. This field is auto populated and you override this setting.
- **Pre-Add Used?:** The DSP58 contains a 27-bit signed adder that can drive one or both inputs of the multiplier. Choose Yes if you are implementing an arithmetic function that requires the pre-add, for example (B + D) * A. The default value is No. This field is auto populated and is a read only for a given configuration.
- **AD Reg Used?:** This indicates that the Pre-Adder output is pipelined before feeding to multiplier input. The default value is No. This field is auto populated and you can override this setting.

Chapter 7

I/O and Transceiver Power

The Versal® ACAP IO Sheet is very similar to prior device families. Versal architecture introduces Xtreme Performance IOs and are found under the XP I/O Type. The GTY sheet includes Hard IP/PCS modes and CPM Blocks. The XPE for Versal® devices consolidates all IP power estimation into the new IP Manager sheet, which includes the Memory Interface IP and Transceiver Interface IP.

IP Manager

You can generate and manage commonly used IP modules in XPE for power estimation using the Manage IP wizard. XPE IP generation is generally aligned with Vivado® IP generation but only provides a subset of configuration detail that is relevant to power estimation. Unlike Vivado IP, no output products are generated, only the resources required for each IP instances are entered into XPE sheets upon creation. Each IP is stored as an instance that can be viewed and deleted, and exported and imported using XPE Exchange format.

The Manage IP wizard is accessible from the IP_Manager sheet which keeps track of each created IP:

- **IP Power:** The Memory Interface power is displayed here.
- **Usage:** All hard IP (embedded) blocks usage is tracked in Memory Controllers, PCIe, MRMAC, and ILKN.

Additionally, the IP Manager sheet displays tables of created IPs with configuration and power details. The tables are divided into three major categories:

- Memory Interface IP
- Transceiver Interface IP
- IP Modules which includes other IPs such as Clocking Wizard, block RAM, and Distributed RAM.

Creating IP

You can create an IP using the Manage IP wizard Create IP tab. For proper identification and management of IP, assign unique names for each instance created. The Clocking Wizard is typically the first type of IP to be created because all other XPE sheets, Block Memory IP, and Distributed Memory IP require a clock to be defined. From the Standard drop-down menu the Memory Interface IP provides several types of interfaces based on the hard memory controller or soft memory controller. When choosing a hard memory controller standard:

- There is an option to select between one and two channels with two channels resulting in slightly higher power.
- There is a Command Bus Option to choose between Minimum Pin Count and Highest Data Rate with the latter resulting in higher power.
- The hard memory controller power is shown in the Hard IP column of the Memory Interface IP table.

When choosing a soft memory controller standard, you can select the check box option to add typical link layer logic. This adds logic and block RAM resources used by the soft IP. The Transceiver IP configuration includes presets for the following protocols:

- PCIe Gen1
- PCIe Gen2
- PCIe Gen3
- PCIe Gen4
- MRMAC
- MRMAC with RS-FEC

In the GTY sheet, Hard IP block is selected by the wizard based on the protocol.

Managing IP

Once you create an IP, you can manage it using the Manage IP tab of the Manage IP Wizard. You cannot modify the IP using the Manage IP wizard and its table entry cannot be modified either. It is possible to delete and recreate IPs if changes are necessary. The IPs can be exported and imported to XPE Exchange format files.

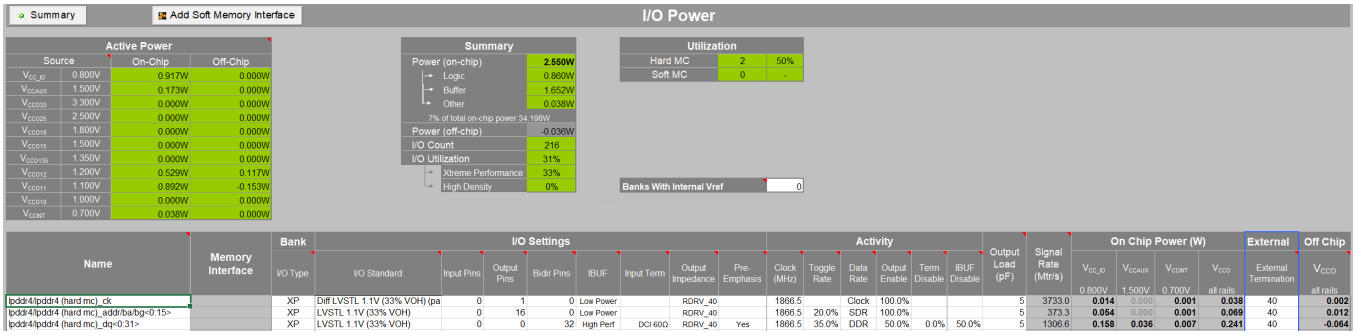
If a Clocking Wizard IP is deleted, its clocks are also deleted from the available selection of clocks. Sheets that refer to a deleted clock will continue to use the clock frequency of the deleted clock to estimate power, but the clock cell will be highlighted in yellow to indicate that it is no longer valid. The clock can be recreated using the Clocking Wizard IP and restored by selecting it from the list of available clocks.

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 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, use the Using the Soft Memory Interface Configuration Wizard.

The following figure illustrates the three main types of information entered on the I/O sheet: I/O Settings, Activity, and External Termination:

Figure 22: **I/O Sheet**



The following section provides more information on how to fill in each of these columns.

- **I/O Setting :**
  - 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 VCCO level (for example, 3.3V) that is closest to the nominal listed in the data sheet for that I/O standard.

    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).*

  - 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, it is recommended that 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.*

  - On-Chip Termination: The termination values are same as that of UltraScale+ devices.
- **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 if 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: *Setting Output Enable to 100% is a common mistake which degrades the XPE 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.

Transceivers and Hard Blocks

GTY, GTYP, and GTM are serial transceivers used for serial data transmission in Versal® ACAP.

In Versal devices, hard IP block selection is dissociated from GT sheet to Hard_Blocks sheet unlike previous device families. Hard IPs should be configured on Hard_Blocks sheet in Versal.

GTY

The Versal ACAP GTY supports continuous data rate from 1.2 Gb/s to 32.75 Gb/s. Versal GTY sheet has few differences when compared to previous device families.

- The following Versal device Clock Sources are different:
  - **LCPLL:** LC-tank based VCO for low jitter, supports maximum data rates
  - **RPLL:** Ring-oscillator based, slightly lower power, lower data rates

RECOMMENDED: *It is recommended to use Add GTY Interface or Manage IP to enter a GTY configuration for a desired protocol so that all columns are correctly specified.*

Versal devices include the following changes to the GTY configuration:

Table 7: **Transceiver Power Estimation (GTY)**

Source	UltraScale+ Device	Versal Device
LC PLL	QPLL (shared to 4 channels)	LCPLL (shared to 2 channels)
Ring PLL	CPLL	RPLL
Ethernet MAC	CMAC	MRMAC
Debug	Eyescan	Not currently supported

GTYP

GTYP is a superset of GTY in every way, including IP and pin compatibility. It is an upgraded version of GTY which supports PCIe® Gen5. For more information on GTYP, see *Versal ACAP GTY and GTYP Transceivers Architecture Manual* (AM002).

GTM

GTM is a high performance transceiver that supports PAM4 and NRZ modulations. Versal® ACAP GTM supports non-continuous data rates between 9.5 Gb/s to 116 Gb/s.

Hard IP Interface

In Versal® ACAP, various hard IP transceiver interfaces and their configuration is specified on Hard_Blocks sheet. Few of the hard IP blocks such as PCIe and MRMAC are configured using Add Hard IP wizard.

CPM

The Hard_Blocks sheet supports an independent subsystem called the CPM. The CPM contains a Type-A PCIe (Gen4 x16) controller and also the necessary hardened components to allow a fabric accelerator to act as a cache coherent interconnect for accelerator (CCIX). The CPM subsystem power is estimated based on the number of controllers used. XPE allows selection of PCIe configuration such as link speed and width.

The controller supports Gen1, Gen2, Gen3, Gen4 PCIe modes, up to x16 lanes. It also supports a CCIX only ESM mode (20 or 25 Gb/s). XPE uses PCIe Core A0 by default for CPM and XPE supports four different modes for CPM based on CPM use models. The four modes are as follows.

- CPM_CCIX – CPM cores used in CCIX mode. PCIe carries CCIX traffic.
- CPM_Stream – PL accessing CPM. AXI-PL Interconnect enabled.
- CPM_PS – PS accessing CPM. AXI-PS enabled.
- CPM_Stream_wDMA – PCIe controller acting in PCIe-only mode (as PCIe bridge). In this mode only core A0 is used and core A1 remains unused.

DCMAC

The DCMAC block provides 100G, 200G, and 400G standard ethernet capability in combinations up to 600G to clients in fixed configuration. It also provides a 600G client frontend that performs standard ethernet processing for up to 60 channels in a flexible dynamic configuration that works with an external FlexE shim to sequence the data out and in from the physical transceivers. XPE supports the following modes based on use models:

- 3x200GE with RS-FEC
- Transcoder_Bypass with RS-FEC
- Power_gated without RS-FEC
- 1x400G with RS-FEC
- 1 x200G with RS-FEC
- 1x100G with RS-FEC
- 1x100G without RS-FEC
- 6x100GE with RS-FEC
- 6x100GE without RS-FEC

ILKN

The Versal ACAP Interlaken block consists of one Interlaken core with aggregate bandwidth of up to 600 Gb/s that supports certain combinations of ports and lane rates with different user interface widths.

The Interlaken block can interface to 12.5G, 25.78125G, and 53.125G transceivers. Up to 24 lanes are supported for 12.5G and 25.78125G transceivers. With 53.125G transceivers, the maximum number of lanes supported are 12. Overclocking modes (28.21G and 56.42G) are also supported in certain configurations for higher speed grades. Error correcting logic (FEC) is required to support 36G+ transceiver lanes (GTMs). The RS-FEC extension of the Interlaken protocol leverages some of Ethernet Clause 91 FEC functions for Interlaken and requires a 100G capable RS-FEC block to be used for two adjacent 36G+ transceiver lanes. Therefore, the following are all combinations with 36G+ transceivers:

- 6x100G with RS-FEC
- 12x53G with RS-FEC
- 24x25G without RS-FEC
- Power_gated without RS-FEC
- 6x53G with RS-FEC
- 12x25G without RS-FEC

- 6x25G without RS-FEC
- Transcoder_bypass without RS-FEC

HSC

The high speed cryptography (HSC) Hard IP block provides high performance encryption/decryption of single or multiple concurrent data streams using a range of ciphers. HSC is an independent subsystem that exists in Versal devices, including those targeted for deployment in networking and data center applications. As such, the off-chip sources and destinations of the data streams serviced by HSC are typically connected via industry standard networking and data center interfaces, for example, OTN/FLexE, Ethernet, and PCI Express. The primary intended usage of HSC is for cryptography services to enable secure transmission of information over the data center interconnect (DCI). The HSC block also optionally gathers statistics whilst performing those cryptography function.

CPM5

CPM5 contains Type-A Gen-5 PCIe controllers and two DMA controllers. CPM5 also contains the necessary hardened components to allow a fabric accelerator to act as a CCIX accelerator over the PCIe transport. Like CPM, CPM5 subsystem power is estimated based on the number of controllers used and respective PCIe configuration. The controller supports Gen1, Gen2, Gen3, Gen4, and Gen5 PCIe modes, up to x16 lanes. It also supports a CCIX only ESM mode (20 or 25 Gb/s). In XPE, both PCIe Core A0 and A1 can be configured separately. There are three different modes supported for CPM based on its usage models. The modes are as follows:

- **PCIESTREAM:** PCIe controller configured to connect to AXI4-Stream interface to access PL
- **CCIX:** PCIeA carries CCIX traffic
- **DMA:** Dedicated DMA controllers used to carry traffic to/from PCIeA controllers

XILINX

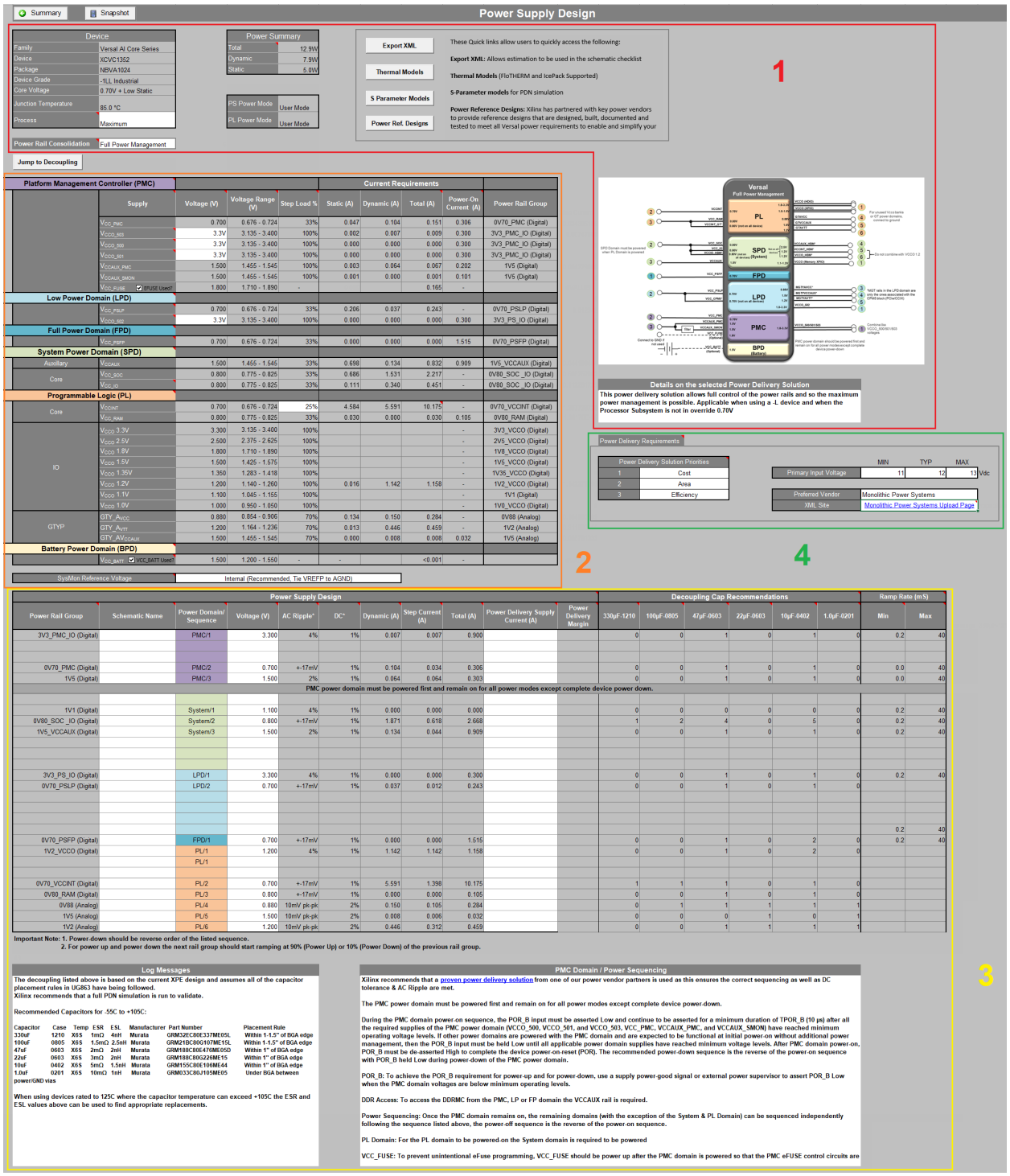
Chapter 8

Power Supply Design

The Power Design sheet in XPE for Versal® ACAP is redesigned to allow you to design a complete, validated power delivery solution using correct sequencing and tolerances. Following are the four main sections in this sheet:

1. Device Estimation Overview
2. Power Rails Table
3. Power Supply Design Table
4. Power Delivery Requirements

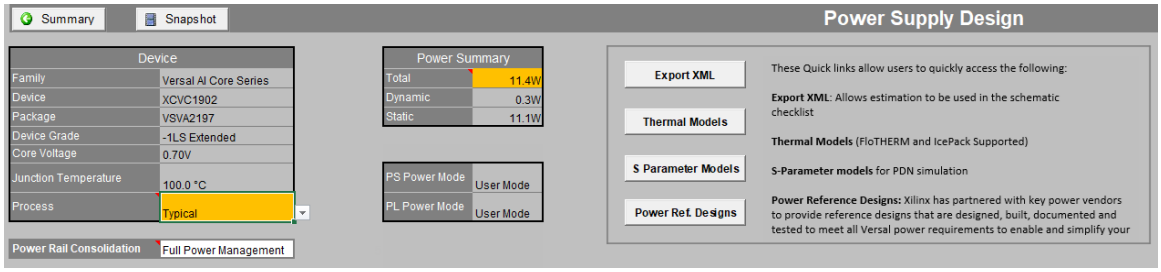
Figure 23: **Power Supply Design Sheet**



Device Estimation Overview

The Device Estimation Overview section of the Power Design tab shows you the current selected device including the Voltage, Static Screen, Temperature grade as well as the total power broken down to static and dynamic. There are design rule checks (DRCs) on the Process, PS and PL Power Mode selection. The power delivery should be designed for the worst-case power estimation, meaning the maximum Process with PS and PL domains fully active. If any of these are not met, then the cell is highlighted in orange to indicate that the setting should be changed. The following figure shows you an example where the Process is set to Typical, the DRC highlights this as it should be set to maximum.

Figure 24: **Device Estimation Overview**



- **Process:** You can toggle between Typical and Maximum process.

  RECOMMENDED: *The power supply should be specified for the worst-case scenario. This means that the Maximum process must be set and the Junction temperature should be set based on the thermal simulation result. If that is not available, Xilinx recommends to set the Junction temperature to the maximum for the device temperature range.*

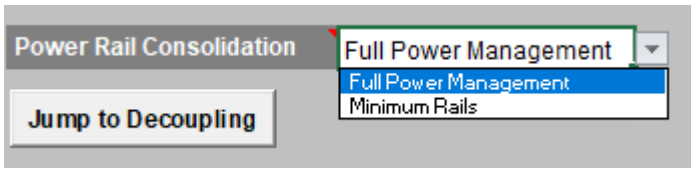
  TIP: *A DRC highlights if the worst-case settings are not used for Process, PS and PL power modes*

- **Power Delivery Solution:** You can define the desired power delivery solution. Following are the two options.

  - **Full Power Management:** This is the recommended solution. It allows for full power management, which means that an individual rail can be sequenced on and off as required to reduce power.
  - **Minimum Power Rails:** This mode allows to consolidate power supply while maintaining the required sequence and voltage specification for every rail.

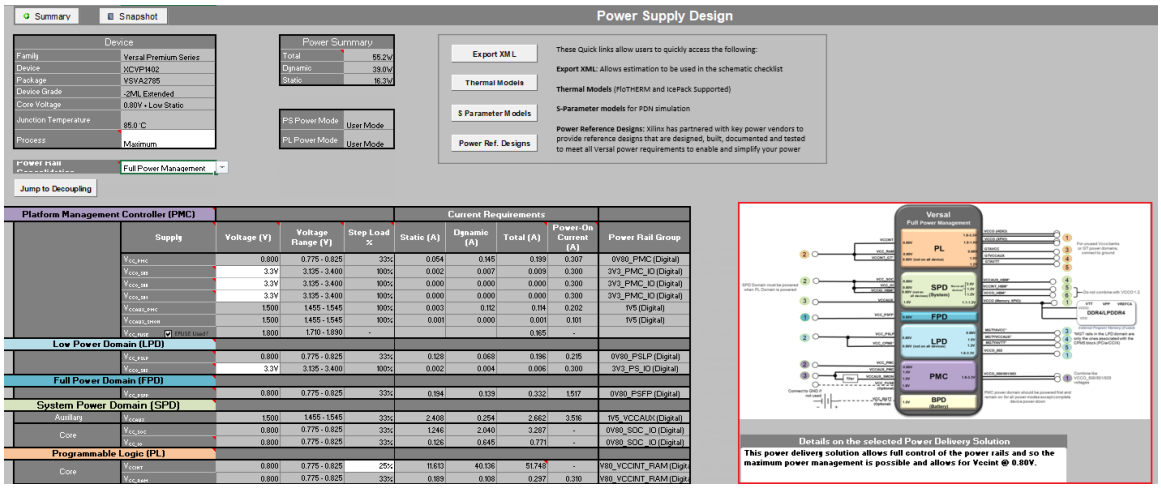
    TIP: *All proven power delivery solutions can be seen here.*

Figure 25: **Power Rail Consolidation**



Based on the power rail consolidation selection, the following image changes to show the connectivity based on the targeted device. Additional information on the power rail consolidation is provided in the note following the image:

Figure 26: **Power Supply Design Based on Rail Consolidation (as highlighted in red)**



TIP: *There are two options for every device, Full Power Management and Minimum Rails. These options change automatically based on the VCCINT voltage of the targeted device or when the PS Overdrive is used in a Low Voltage device.*

Quick links are provided here for you to find additional reference material.

- **Export XML:** Generates `.XML` file that can be used with the Schematic Checklist or the Supported Power vendors.
- **Thermal Models:** Provides access to Versal Thermal Models. Xilinx provides models for both Siemens Flotherm and Ansys Icepak.
- **S Parameter Models:** Provides a link to the lounge to access the S Parameter models.
- **Power Reference Designs:** Provides a link to the proven power solution from our power supply partners.
- **Jump to Decoupling:** Moves the sheet to the bottom to display the decoupling table.

Power Rails Table

All the required power rails for a given device are displayed within the power rail table. The rails are grouped based on their respective power domains (PMC, LPD, FPD, System, PL, and Battery).

TIP: *The color shown for each domain matches that of the power delivery solution as shown in the following figure and it is common across Xilinx documentation.*

Figure 27: **Power Rails**

Platform Management Controller (PMC)					Current Requirements				
	Supply	Voltage (V)	Voltage Range (V)	Step Load %	Static (A)	Dynamic (A)	Total (A)	Power-On Current (A)	Power Rail Group
	V_{CC_PMC}	0.700	0.676 - 0.724	33%	0.076	0.105	0.181	0.311	0V70_PMC (Digital)
	V_{CCO_503}	3.3V	3.135 - 3.400	100%	0.000	0.007	0.008	0.300	3V3_PMC_IO (Digital)
	V_{CCO_500}	3.3V	3.135 - 3.400	100%	0.000	0.000	0.000	0.300	3V3_PMC_IO (Digital)
	V_{CCO_501}	3.3V	3.135 - 3.400	100%	0.000	0.000	0.000	0.300	3V3_PMC_IO (Digital)
	V_{CCAUX_PMC}	1.500	1.455 - 1.545	100%	0.016	0.080	0.097	0.216	1V5 (Digital)
	V_{CCAUX_SMON}	1.500	1.455 - 1.545	100%	0.014	0.000	0.014	0.108	1V5 (Digital)
	V_{CC_FUSE} ☑ EFUSE Used?	1.800	1.710 - 1.890	-			0.165	-	
Low Power Domain (LPD)									
	V_{CC_PSLP}	0.700	0.676 - 0.724	33%	0.168	0.045	0.213	0.214	0V70_PSLP (Digital)
	V_{CCO_502}	3.3V	3.135 - 3.400	100%	0.000	0.000	0.000	0.300	3V3_PS_IO (Digital)
Full Power Domain (FPD)									
	V_{CC_PSFP}	0.700	0.676 - 0.724	33%	0.000	0.000	0.000	1.617	0V70_PSFP (Digital)
System Power Domain (SPD)									
Auxillary	V_{CCAUX}	1.500	1.455 - 1.545	33%	3.267	0.014	3.281	4.442	1V5_VCCAUX (Digital)
Core	V_{CC_SOC}	0.800	0.775 - 0.825	33%	3.363	0.000	3.363	4.634	0V80_SOC _IO (Digital)
	V_{CC_IO}	0.800	0.775 - 0.825	33%	0.499		0.499	-	0V80_SOC _IO (Digital)
Programmable Logic (PL)									
Core	V_{CCINT}	0.700	0.676 - 0.724	25%	25.428	0.009	25.437	-	0V70_VCCINT (Digital)
	V_{CC_RAM}	0.800	0.775 - 0.825	33%	0.186		0.186	0.280	0V80_RAM (Digital)
IO	V_{CCO} 3.3V	3.300	3.135 - 3.400	100%				-	3V3_VCCO (Digital)
	V_{CCO} 2.5V	2.500	2.375 - 2.625	100%				-	2V5_VCCO (Digital)
	V_{CCO} 1.8V	1.800	1.710 - 1.890	100%				-	1V8_VCCO (Digital)
	V_{CCO} 1.5V	1.500	1.425 - 1.575	100%				-	1V5_VCCO (Digital)
	V_{CCO} 1.35V	1.350	1.283 - 1.418	100%				-	1V35_VCCO (Digital)
	V_{CCO} 1.2V	1.200	1.140 - 1.260	100%				-	1V2_VCCO (Digital)
	V_{CCO} 1.1V	1.100	1.045 - 1.155	100%				-	1V1_VCCO (Digital)
	Vcco 1.0V	1.000	0.950 - 1.050	100%				-	1V0_VCCO (Digital)
GTY	GTY_A_{VCC}	0.880	0.854 - 0.906	70%				-	0V88 (Analog)
	GTY_A_{VTT}	1.200	1.164 - 1.236	70%				-	1V2 (Analog)
	GTY_AV_{CCAUX}	1.500	1.455 - 1.545	70%				-	1V5 (Analog)
Battery Power Domain (BPD)									
	V_{CC_BATT} ☑ VCC_BATT Used?	1.500	1.200 - 1.550	-	-		<0.001	-	
SysMon Reference Voltage		Internal (Recommended, Tie VREFP to AGND)							

The Power Rails table is divided into the following columns:

- **Voltage:** With the exception of VCCO_500, VCCO_501, VCCO_502, and VCCO_503 this column is for reference only. The voltage can only be changed by adjusting the regulator voltage in the Power Supply Design table. For 50x banks, the desired voltage can be selected here.
- **Voltage Range (Read Only):** This column shows the allowed voltage range for each power rail.

> **TIP:** *It is recommended to keep the voltage at the TYP value, because this means that there is a balanced positive and negative range for the power supply design to cover AC ripple and DC tolerance.*

- **Step Load:** Step load is the maximum percentage change of the dynamic current on a given rail.

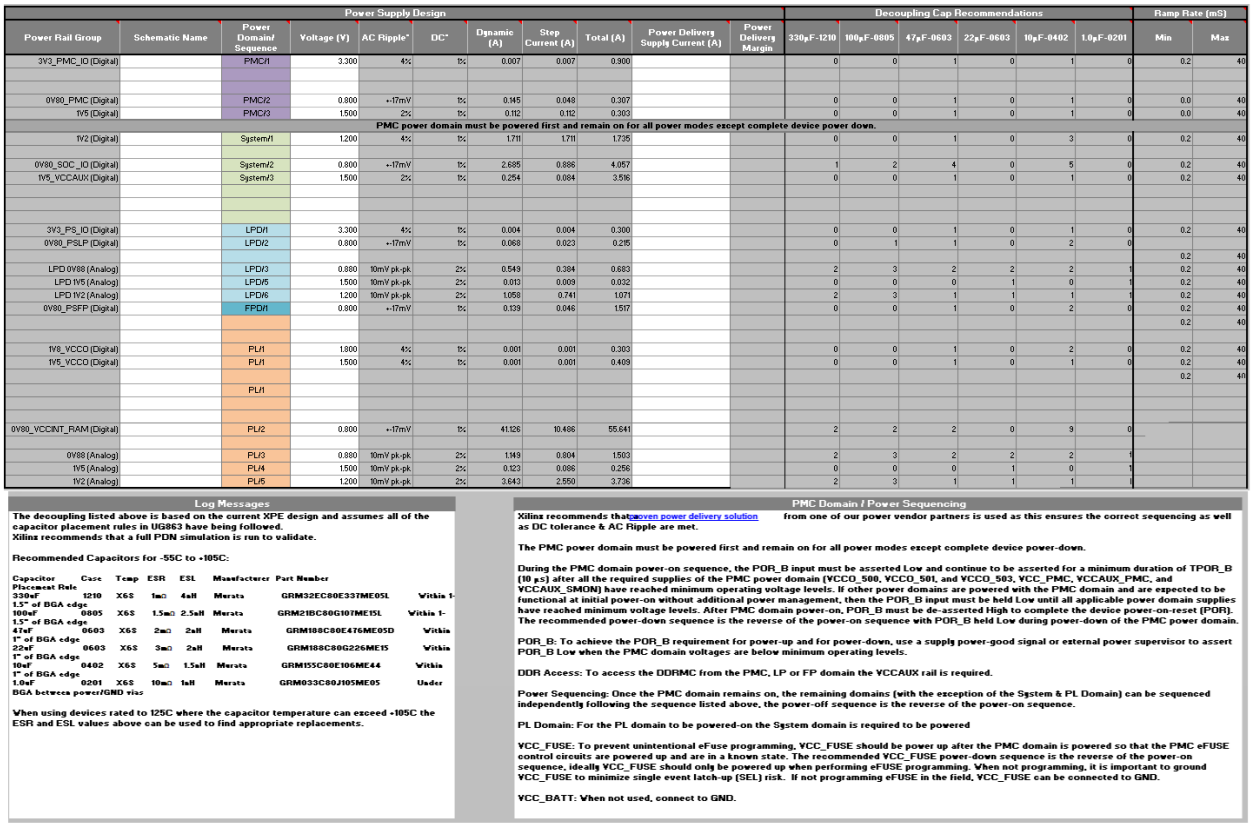
> **TIP:** *Only the VCCINT step load can be altered as this impacts the required numbers of decoupling capacitors, the other rails step loads are fixed.*

- **Current Requirements (Ready Only):** This is the current requirement for each rail based on the current estimation. It is categorized into Static, Dynamic, Total and Power on Current (A). Power on current is only displayed when Maximum process is set.
- **Power Rail Group (Read Only):** This is the power rail group of the power rail, it is based on the selected power delivery solution. All power rails that are part of the same power rail group are supplied by the same group in the Power Supply Design Table.
- **SysMon Reference Voltage:** You can select the SysMon Reference Voltage connection, it is recommended to use the Internal Vref, if the external Vref is used then the required reference voltage needs to be supplied on the PCB.

Power Supply Design Table

The power supply design table displays all of the required information to correctly design the power supply and power decoupling network based on the current estimation.

Figure 28: **Power Supply Design and Decoupling Recommendations**



Power Supply Design

Power Rail Group	Schematic Name	Power Domain/ Sequence	Voltage (V)	AC Ripple*	DC*	Dynamic (A)	Static Current (A)	Total (A)	Power Delivery Supply Current (A)	Power Delivery Margin
3V3_PMC_IO (Digital)		PMC/1	3.300	4%	1%	0.007	0.007	0.900		
0V80_PMC (Digital)		PMC/2	0.800	+-17mV	1%	0.145	0.048	0.307		
1V5 (Digital)		PMC/3	1.500	2%	1%	0.112	0.112	0.303		
PMC power domain must be powered first and remain on for all power modes except complete device power down.										
1V2 (Digital)		System/1	1.200	4%	1%	1.711	1.711	1.735		
0V80_SOC_IO (Digital)		System/2	0.800	+-17mV	1%	2.685	0.886	4.057		
1V5_VCCAUX (Digital)		System/3	1.500	2%	1%	0.254	0.084	3.516		
3V3_PS_IO (Digital)		LPD/1	3.300	4%	1%	0.004	0.004	0.300		
0V80_PSLP (Digital)		LPD/2	0.800	+-17mV	1%	0.068	0.023	0.215		
LPD 0V88 (Analog)		LPD/3	0.880	10mV pk-pk	2%	0.549	0.384	0.683		
LPD 1V5 (Analog)		LPD/5	1.500	10mV pk-pk	2%	0.013	0.009	0.032		
LPD 1V2 (Analog)		LPD/6	1.200	10mV pk-pk	2%	1.058	0.741	1.071		
0V80_PSFP (Digital)		FPD/1	0.800	+-17mV	1%	0.139	0.046	1.517		
1V8_VCCO (Digital)		PL/1	1.800	4%	1%	0.001	0.001	0.303		
1V5_VCCO (Digital)		PL/1	1.500	4%	1%	0.001	0.001	0.409		
		PL/1								
0V80_VCCINT_RAM (Digital)		PL/2	0.800	+-17mV	1%	41.126	10.486	55.641		
0V88 (Analog)		PL/3	0.880	10mV pk-pk	2%	1.149	0.804	1.503		
1V5 (Analog)		PL/4	1.500	10mV pk-pk	2%	0.123	0.086	0.256		
1V2 (Analog)		PL/5	1.200	10mV pk-pk	2%	3.643	2.550	3.736		

Decoupling Cap Recommendations						Ramp Rate (mS)	
330µF-1210	100µF-0805	47µF-0603	22µF-0603	10µF-0402	1.0µF-0201	Min	Max
0	0	1	0	1	0	0.2	40
0	0	1	0	1	0	0.0	40
0	0	1	0	1	0	0.0	40
0	0	1	0	3	0	0.2	40
1	2	4	0	5	0	0.2	40
0	0	1	0	1	0	0.2	40
0	0	1	0	1	0	0.2	40
0	1	1	0	2	0		
						0.2	40
2	3	2	2	2	1	0.2	40
0	0	0	1	0	1	0.2	40
2	3	1	1	1	1	0.2	40
0	0	1	0	2	0	0.2	40
						0.2	40
0	0	1	0	2	0	0.2	40
0	0	1	0	1	0	0.2	40
						0.2	40
2	2	2	0	9	0		
2	3	2	2	2	1		
0	0	0	1	0	1		
2	3	1	1	1	1		

Log Messages

The decoupling listed above is based on the current XPE design and assumes all of the capacitor placement rules in UG863 have being followed.
Xilinx recommends that a full PDN simulation is run to validate.

Recommended Capacitors for -55C to +105C:

Capacitor Placement Rule	Case	Temp	ESR	ESL	Manufacturer	Part Number	
330nF 1.5" of BGA edge	1210	X6S	1mΩ	4nH	Murata	GRM32EC80E337ME05L	Within 1-
100nF 1.5" of BGA edge	0805	X6S	1.5mΩ	2.5nH	Murata	GRM21BC80G107ME15L	Within 1-
47nF 1" of BGA edge	0603	X6S	2mΩ	2nH	Murata	GRM188C80E476ME05D	Within
22nF 1" of BGA edge	0603	X6S	3mΩ	2nH	Murata	GRM188C80G226ME15	Within
10nF 1" of BGA edge	0402	X6S	5mΩ	1.5nH	Murata	GRM155C80E106ME44	Within
1.0nF BGA between power/GND vias	0201	X6S	10mΩ	1nH	Murata	GRM033C80J105ME05	Under

When using devices rated to 125C where the capacitor temperature can exceed +105C the ESR and ESL values above can be used to find appropriate replacements.

PMC Domain / Power Sequencing

Xilinx recommends that a [proven power delivery solution] from one of our power vendor partners is used as this ensures the correct sequencing as well as DC tolerance & AC Ripple are met.

The PMC power domain must be powered first and remain on for all power modes except complete device power-down.

During the PMC domain power-on sequence, the POR_B input must be asserted Low and continue to be asserted for a minimum duration of TPOR_B (10 µs) after all the required supplies of the PMC power domain (VCCO_500, VCCO_501, and VCCO_503, VCC_PMC, VCCAUX_PMC, and VCCAUX_SMON) have reached minimum operating voltage levels. If other power domains are powered with the PMC domain and are expected to be functional at initial power-on without additional power management, then the POR_B input must be held Low until all applicable power domain supplies have reached minimum voltage levels. After PMC domain power-on, POR_B must be de-asserted High to complete the device power-on-reset (POR). The recommended power-down sequence is the reverse of the power-on sequence with POR_B held Low during power-down of the PMC power domain.

POR_B: To achieve the POR_B requirement for power-up and for power-down, use a supply power-good signal or external power supervisor to assert POR_B Low when the PMC domain voltages are below minimum operating levels.

DDR Access: To access the DDRMC from the PMC, LP or FP domain the VCCAUX rail is required.

Power Sequencing: Once the PMC domain remains on, the remaining domains (with the exception of the System & PL Domain) can be sequenced independently following the sequence listed above, the power-off sequence is the reverse of the power-on sequence.

PL Domain: For the PL domain to be powered-on the System domain is required to be powered

VCC_FUSE: To prevent unintentional eFuse programming, VCC_FUSE should be power up after the PMC domain is powered so that the PMC eFUSE control circuits are powered up and are in a known state. The recommended VCC_FUSE power-down sequence is the reverse of the power-on sequence, ideally VCC_FUSE should only be powered up when performing eFUSE programming. When not programming, it is important to ground VCC_FUSE to minimize single event latch-up (SEL) risk. If not programming eFUSE in the field, VCC_FUSE can be connected to GND.

VCC_BATT: When not used, connect to GND.

- **Power Rail Group (Read Only):** Displays the required power rail groups based on the device selection and the power rail consolidation.
- **Schematic Name:** Allows you to add your own schematic name reference for each of the power rail groups. This is added to the exported XML file when using with the power vendors or the schematic checklist.
- **Power Domain/Sequence (Read Only):** Displays the power domain that the regulator is part of and its position in the power on and off sequence.

> **TIP:** *The power off sequence is the reverse of the power on sequence. XPE provides all sequencing information for the PMC and other domains based on the power rail consolidation options selected.*

- **Voltage:** Allows you to alter the voltage of the power rail. Any changes here impact all the power supplies connected to it in the power supply table.

> **TIP:** *It is recommended to keep the voltage at the TYP value, because this means that there is a balanced positive and negative range for the power supply design to cover AC ripple and DC tolerance. The AC ripple and DC tolerance values only apply when the voltage is set to TYP, if you adjust the voltage then the AC ripple and DC tolerance requirements need to be re-calculated to ensure the power rail stays within the allowed range.*

- **AC Ripple:** Shows the allowed AC ripple on each of the Power Rail groupings.
- **DC Tolerance:** Shows the allowed DC tolerance of the regulator output.
- **Dynamic (A):** Sum of the dynamic current of the Versal ACAP rails supplied by the power rail based on the current estimation.
- **Step Current (A):** Sum of the step loads of the Versal ACAP rails supplied by the power rail based on the current estimation.
- **Total (A):** Total current requirement for the power rail, sum of the Versal ACAP rails supplied by the power rail. If the Power ON Current is greater than the operating current, then the PoC requirement is shown.
- **Power Delivery Supply Current (A):** Allows you to enter the power delivery current specification for each rail for validation. When you enter the current for each rail, it is validated against the current power estimation. For most accurate results, ensure that the estimation is up to date. If possible, an import from the Report_Power is used and validated.
- **Power Delivery Margin:** Allows you to validate your selected power delivery when entering the current for each power rail group. XPE validates and indicates the power delivery solution margin. Characterization level is used to ensure margin is added to the dynamic requirements. The calculation is, Static Requirement + (Dynamic Requirement + Char Level).
  - **Ramp Rate Min:** Fastest ramp rate from GND to 95% is allowed.
  - **Ramp Rate Max:** Slowest ramp rate from GND to 95% is allowed.

Decoupling Capacitors Table

The decoupling capacitor table indicates the required decoupling based on the current estimation and the step load entered for VCCINT. The table lists six types of capacitors. The following is the size and self resonant frequency of each capacitor:

- **330µF-1210:** Typical self resonant frequency is 0.3 MHz.
- **100µF-0805:** Typical self resonant frequency is 0.5 MHz.
- **47µF-0603:** Typical self resonant frequency is 0.8 MHz.
- **22µF-0603:** Required for the transceivers. Typical self resonant frequency is 1.0 MHz.
- **10µF-0402:** Typical self resonant frequency is 2 MHz.
- **1.0µF-0201:** Typical self resonant frequency is 10 MHz.

RECOMMENDED: *Xilinx recommends a PDN simulation is done to validate the decoupling capacitors quantity and placement is based on the completed PCB design.*

TIP: *More details such as ESR / ESL, manufacturer part number, and temperature range can be seen in the message on the power supply tab of the XPE.*

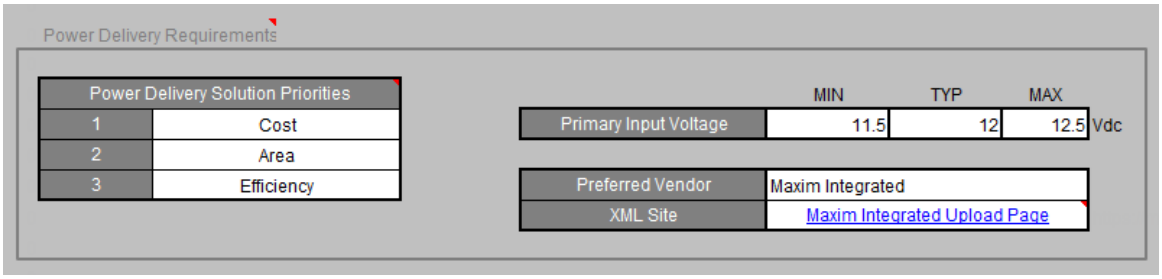
Power sequencing requirements are given in the PMC / Power Sequencing dialog box.

Power Delivery Requirements

Finding the right power delivery solution for your Xilinx application is now easier than ever. Xilinx has collaborated extensively with the industry's leading power delivery vendors to provide optimized solutions to meet the needs of any Xilinx application. Hardware verified reference designs ensure that all Xilinx power specifications are optimally met and followed Xilinx supported power up/down sequencing. Visit Xilinx Power Delivery Solutions to explore variety of solutions available.

The final section of the power delivery requirements page. It doesn't impact the estimation but it helps to ensure that the power delivery is designed correctly.

Figure 29: **Power Delivery Requirements**



The power delivery requirements section allows you to select the following:

- **Power Delivery Solution Priorities:** You can select the top three preferences like Cost, Area, and Efficiency.
- **Primary Input Voltage:** You can input the board input voltage that allows the power vendors to specify full power delivery from input to the Versal ACAP.
- **Preferred Vendors:** Xilinx has partnered with the following vendors. Select your preferred partner and the application. XML portal is displayed for you to upload your `.xml` file from XPE.

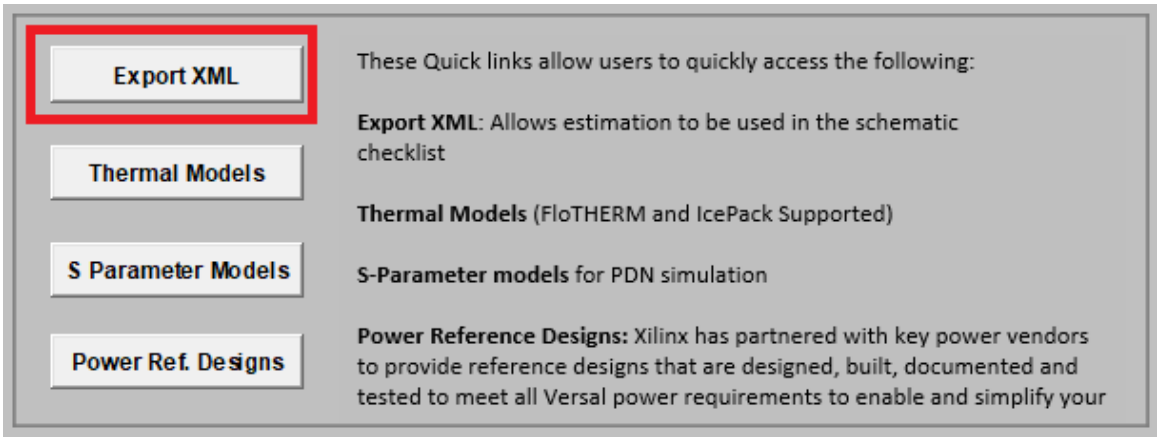
Table 8: **Preferred Vendors**

Vendor	XML Portal
Infineon	https://www.infineon.com/cms/en/about-infineon/company/contacts/product-support-form/
Maxim Integrated	https://maximsupport.microsoftcrmportals.com/en-US/xilinx
MPS	https://www.monolithicpower.com/en/contact/xilinx-file-upload.html
Renesas	https://powercompass.renesas.com/us/en/support/design-tools/powercompass/editor.html#/?_k=veafc7

Table 8: **Preferred Vendors** *(cont'd)*

Vendor	XML Portal
Texas Instruments	http://www.ti.com/fpga-xilinx-XML

Figure 30: **Export XML**



XILINX

Chapter 9

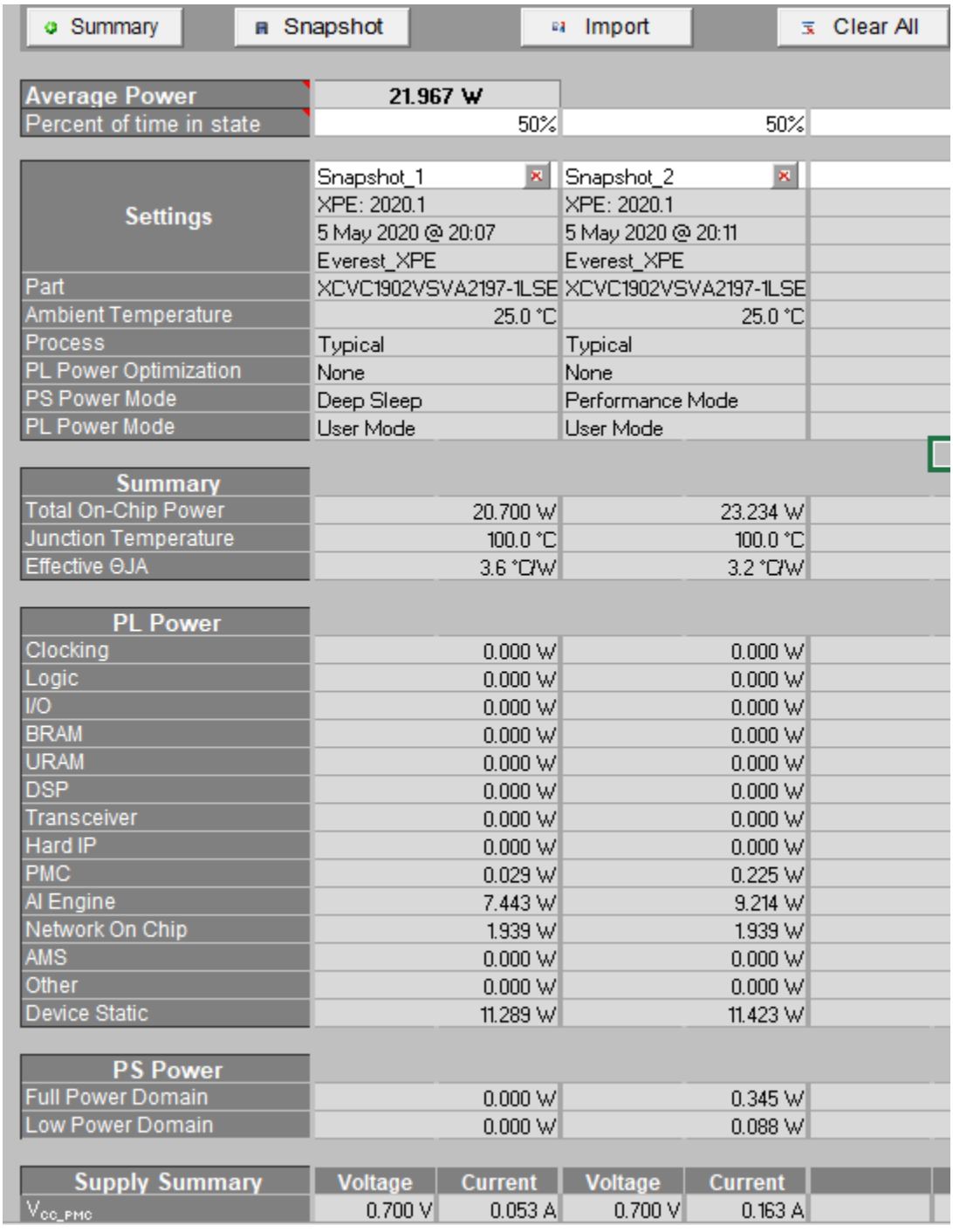
Snapshot, Device Duty Cycle and Graph Sheet

Snapshot and Device Duty Cycle

Snapshots allow you to capture the summary of a design or a design state capturing total power and individual block powers for a particular configuration. The snapshot feature also displays the power supply summary for each power rail. This feature can be used to compare the power difference between two different configurations and helps in the design level if-else analysis.

Duty cycle feature is introduced for Versal® devices. This feature helps you to determine the design power accurately based on the amount of time the design remains in a particular state. In real time designs, all blocks are not enabled for complete operation time. For example, some application use PS in high performance mode only for 50% of the design operational time, rest of the time it operates in Deep-Sleep mode. Similarly, it can be used for PL dynamic clock scaling, and for other blocks. The expected percentage of time a design is in a particular state can be provided to XPE. XPE then determines the weighted average of all the states based on the amount of time in each state. The weighted average is used to determine thermal solutions.

Figure 31: **Snapshot Sheet**



Summary | Snapshot | Import | Clear All

Average Power	**21.967 W**			
Percent of time in state	50%	50%		
Settings	Snapshot_1	Snapshot_2		
	XPE: 2020.1	XPE: 2020.1		
	5 May 2020 @ 20:07	5 May 2020 @ 20:11		
	Everest_XPE	Everest_XPE		
Part	XCVC1902VSVA2197-1LSE	XCVC1902VSVA2197-1LSE		
Ambient Temperature	25.0 °C	25.0 °C		
Process	Typical	Typical		
PL Power Optimization	None	None		
PS Power Mode	Deep Sleep	Performance Mode		
PL Power Mode	User Mode	User Mode		
Summary				
Total On-Chip Power	20.700 W	23.234 W		
Junction Temperature	100.0 °C	100.0 °C		
Effective ΘJA	3.6 °C/W	3.2 °C/W		
PL Power				
Clocking	0.000 W	0.000 W		
Logic	0.000 W	0.000 W		
I/O	0.000 W	0.000 W		
BRAM	0.000 W	0.000 W		
URAM	0.000 W	0.000 W		
DSP	0.000 W	0.000 W		
Transceiver	0.000 W	0.000 W		
Hard IP	0.000 W	0.000 W		
PMC	0.029 W	0.225 W		
AI Engine	7.443 W	9.214 W		
Network On Chip	1.939 W	1.939 W		
AMS	0.000 W	0.000 W		
Other	0.000 W	0.000 W		
Device Static	11.289 W	11.423 W		
PS Power				
Full Power Domain	0.000 W	0.345 W		
Low Power Domain	0.000 W	0.088 W		
Supply Summary	**Voltage**	**Current**	**Voltage**	**Current**
V_{CC_PMC}	0.700 V	0.053 A	0.700 V	0.163 A

Note:

- Average power is displayed only when the total percent of time in state = 100%
- Reset to Defaults option in the Summary Sheet cannot clear the snapshot state. To clear the snapshot state you need to select **Clear All** in the Snapshot sheet.

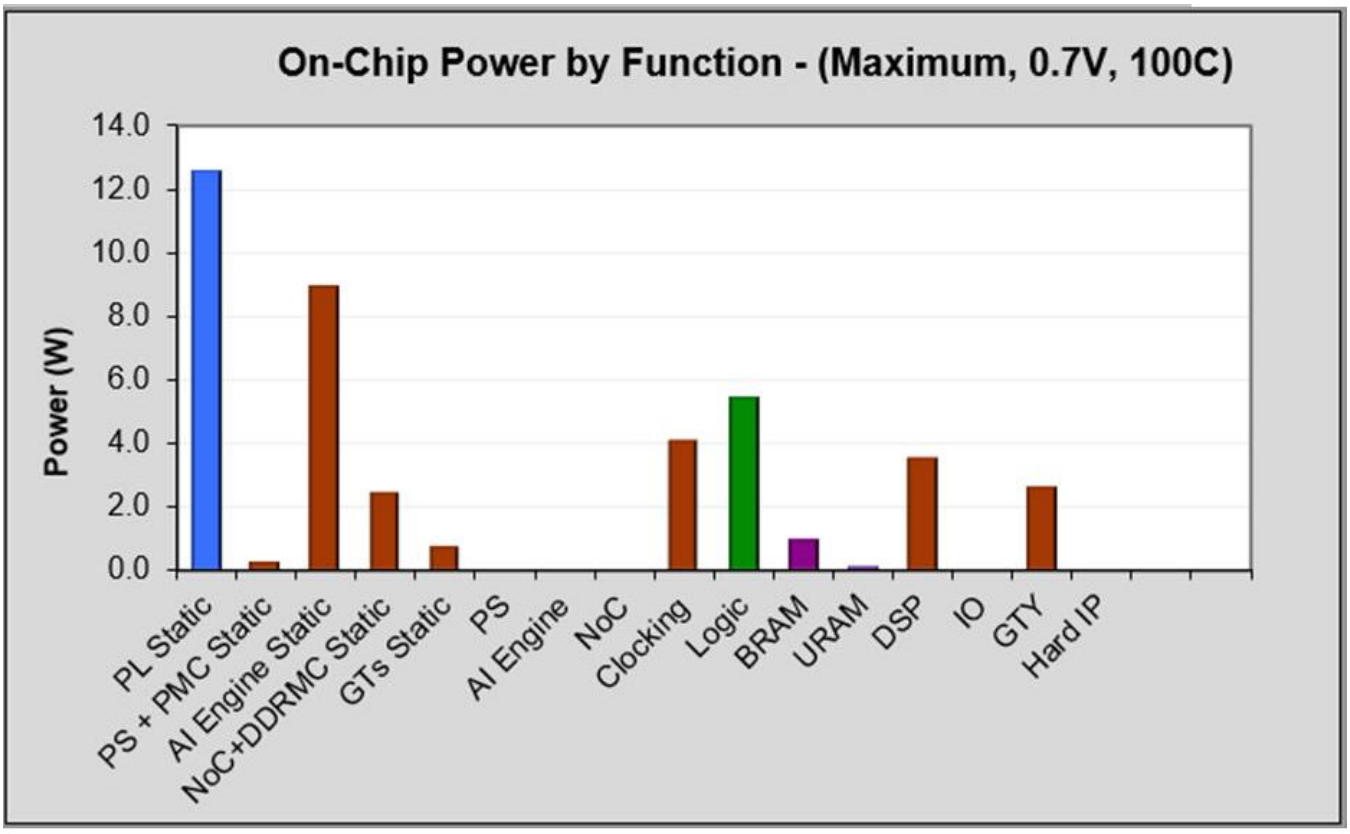
Using Graph Sheet

The following power graphs show the graphical representation of your power estimates.

This graph displays the variation of power for each functional block.

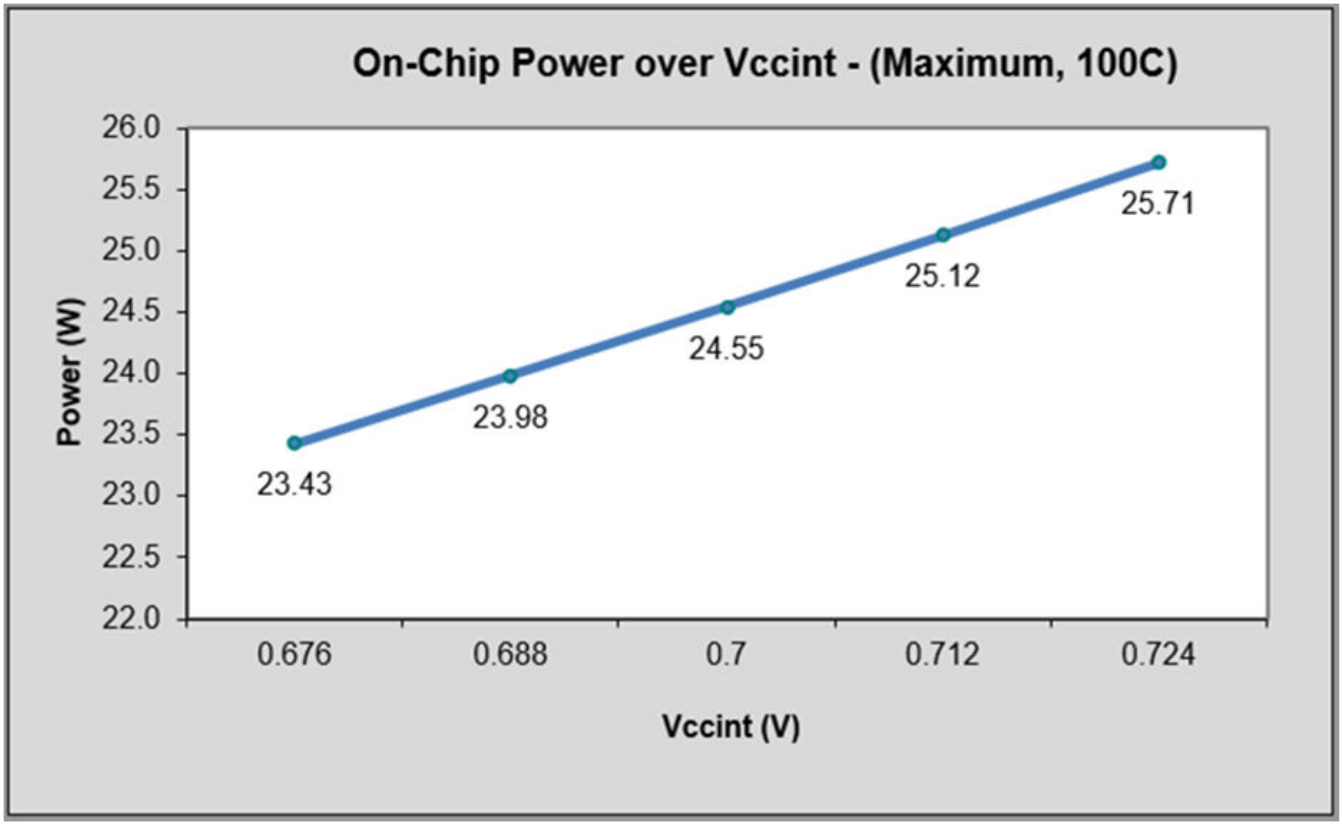
***Note*:** Graph sheet computation may take several minutes to complete. When the graph sheet is selected, a pop-up appears with the option to select computation or decline.

Figure 32: **On-Chip Power by Function**



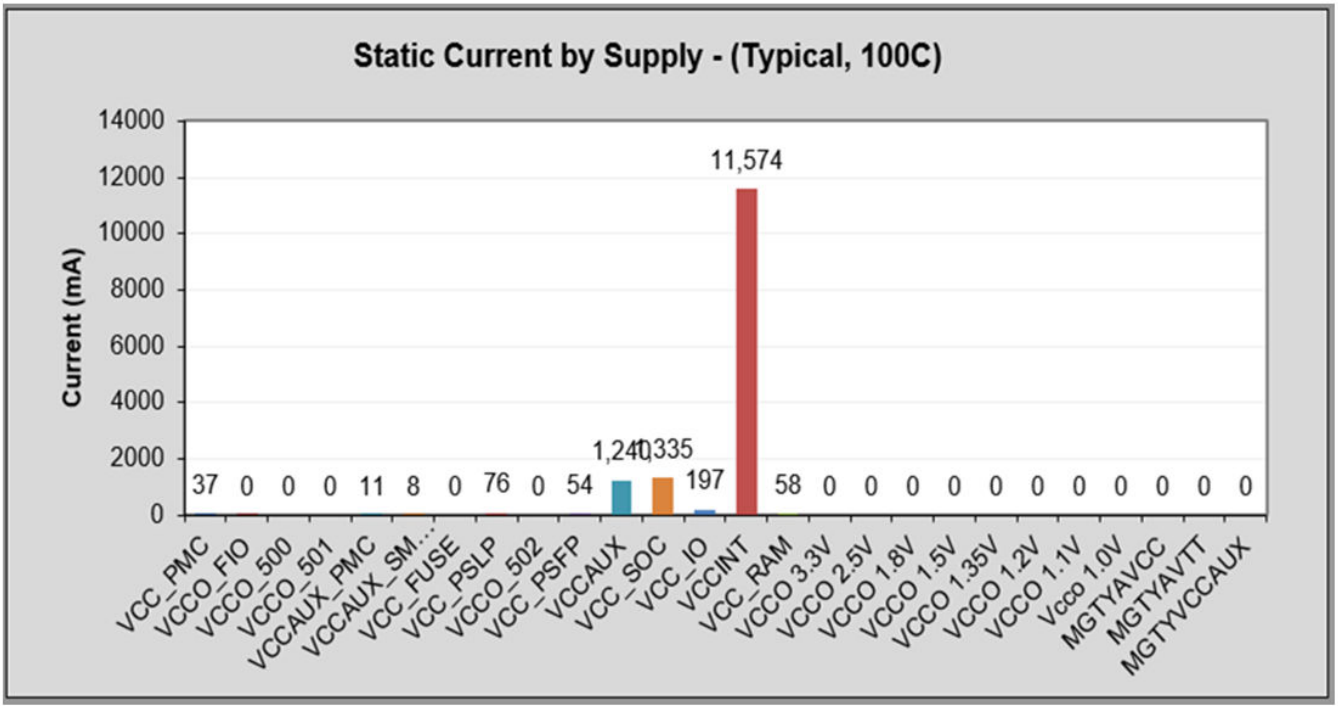
This graph displays the variation of power with respect to Vccint (core voltage).

Figure 33: **On-Chip Power over Vccint**



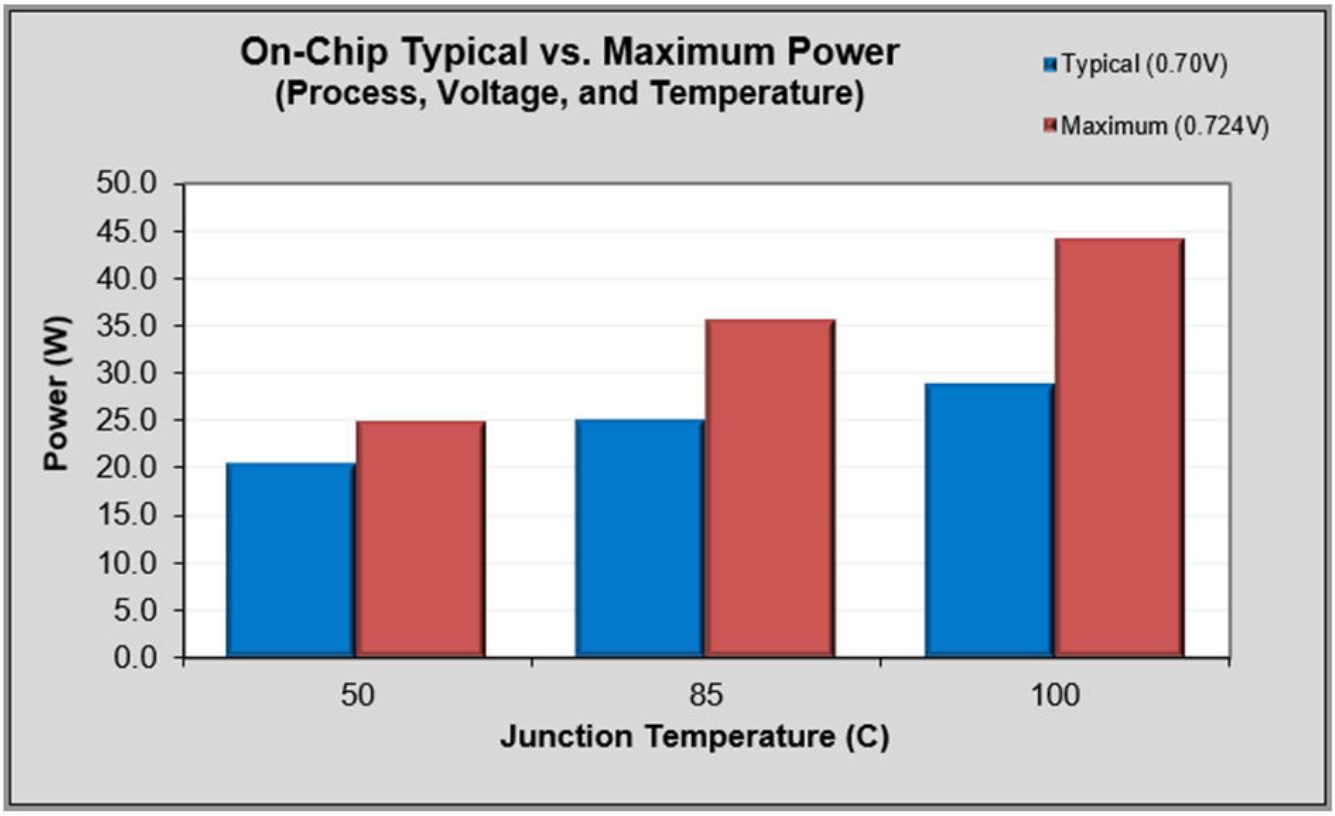
This graph displays the power for each supply rails.

Figure 34: **Static Current by Supply**



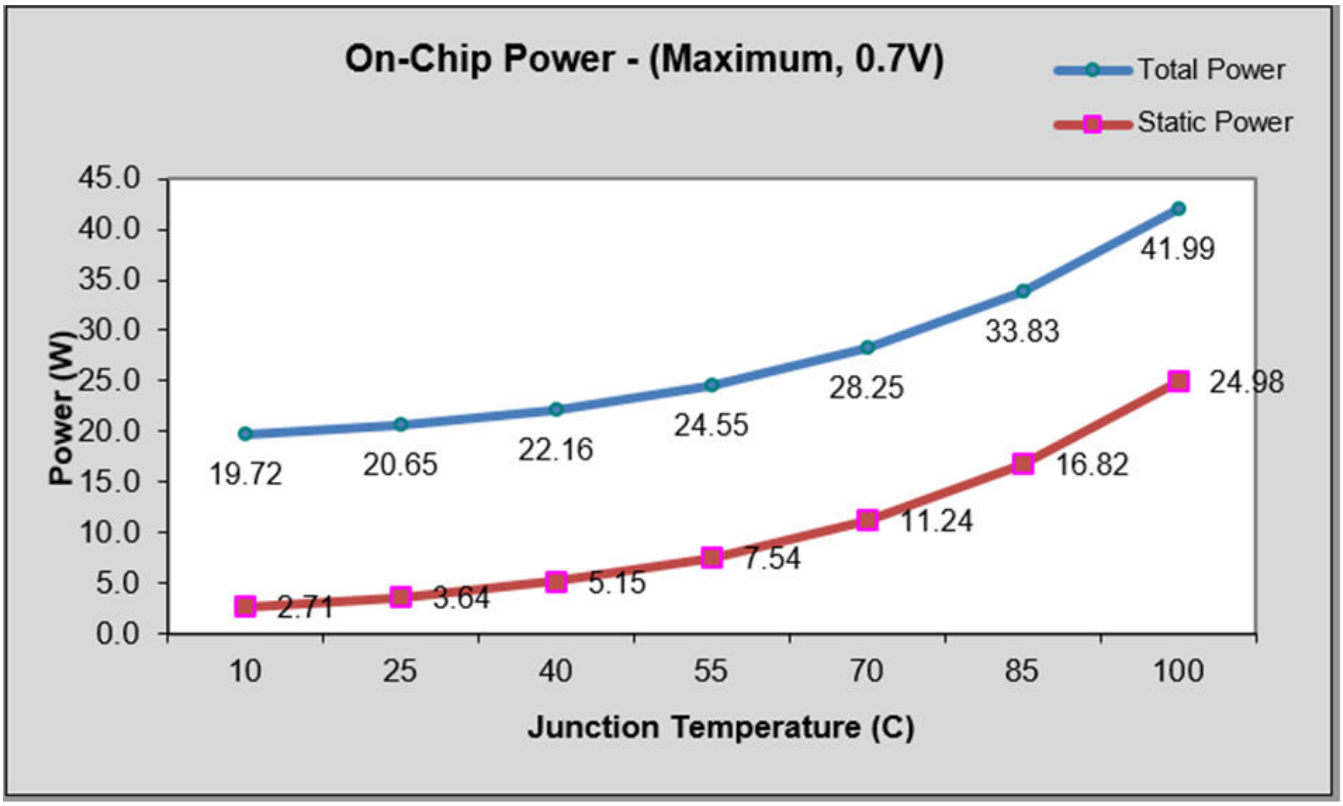
This graph displays the power with respect to process, voltage, and temperature (PVT) changes.

Figure 35: **On-Chip Typical vs Maximum Power**



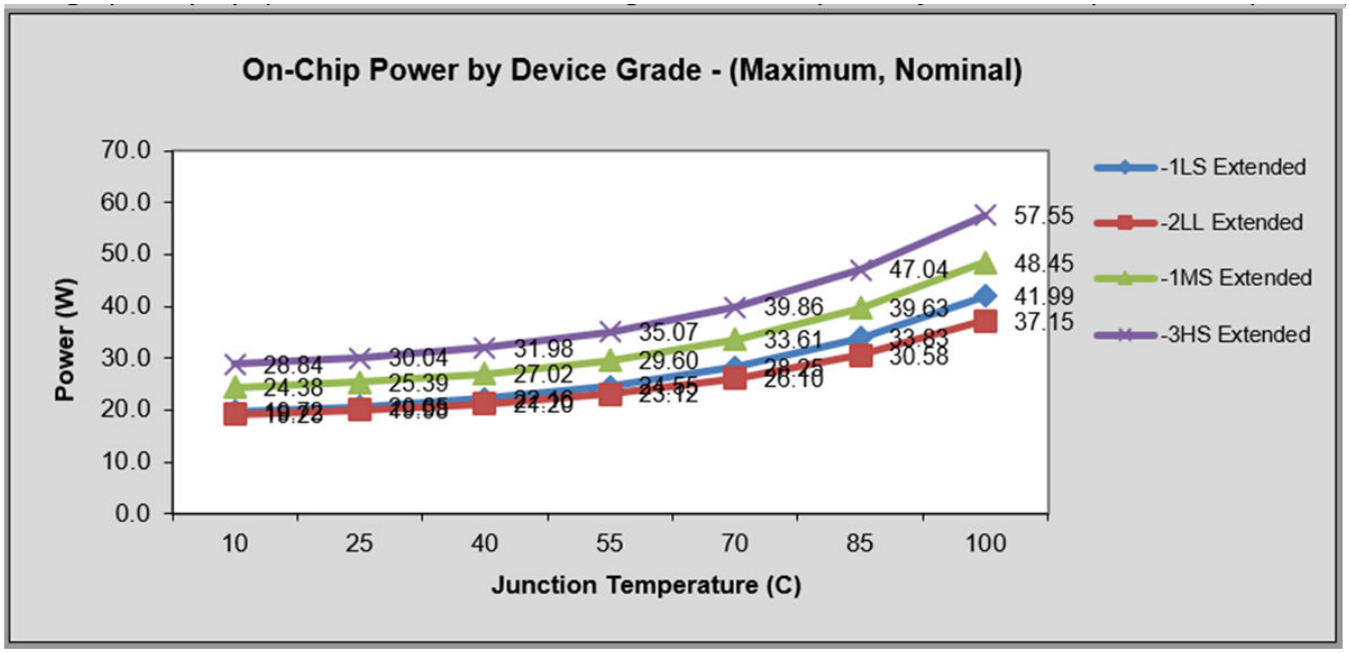
This graph displays the very fine variation of power (static and total) at maximum process with respect to junction temperatures.

Figure 36: **On-Chip Power**



This graph displays power for different device grade with respect to junction temperature depending on the selected device grade.

Figure 37: **On-Chip Power by Device Grade**



Exchanging Power Information

The following table summarizes the current level of support for importing data into XPE and exporting data from XPE.

Table 9: **File format support for Export/Import**

File Format	File Extension	Import into XPE	Export from XPE
Text Power Report	.pwr	NA	Not Supported
Xilinx Design Constraints	.xdc	NA	Planned
XPE Exchange	.xpe	Yes	Yes
Export XML	.xml	No	Yes

Notes:

1. Not all data is supported for XPE Exchange when importing XPE data from earlier device families to a Versal® device.
2. XPE does not yet support import of Vivado Report Power analysis as it is not yet available.
3. XPE Exchange enables import and export of XPE data. Typical use cases include:
   - Transfer of data between XPE spreadsheets.
   - Transfer of IP estimation between spreadsheets.
   - Import of power estimation data from high-level compilers such as the NoC Compiler.
   - Import of results from Vivado® Report Power to analyze in XPE (not yet supported for Versal® devices).
4. XML export allows you to export .xml file that provides estimation to be used in the schematic checklist for the preferred power vendors.

Exporting XPE Data

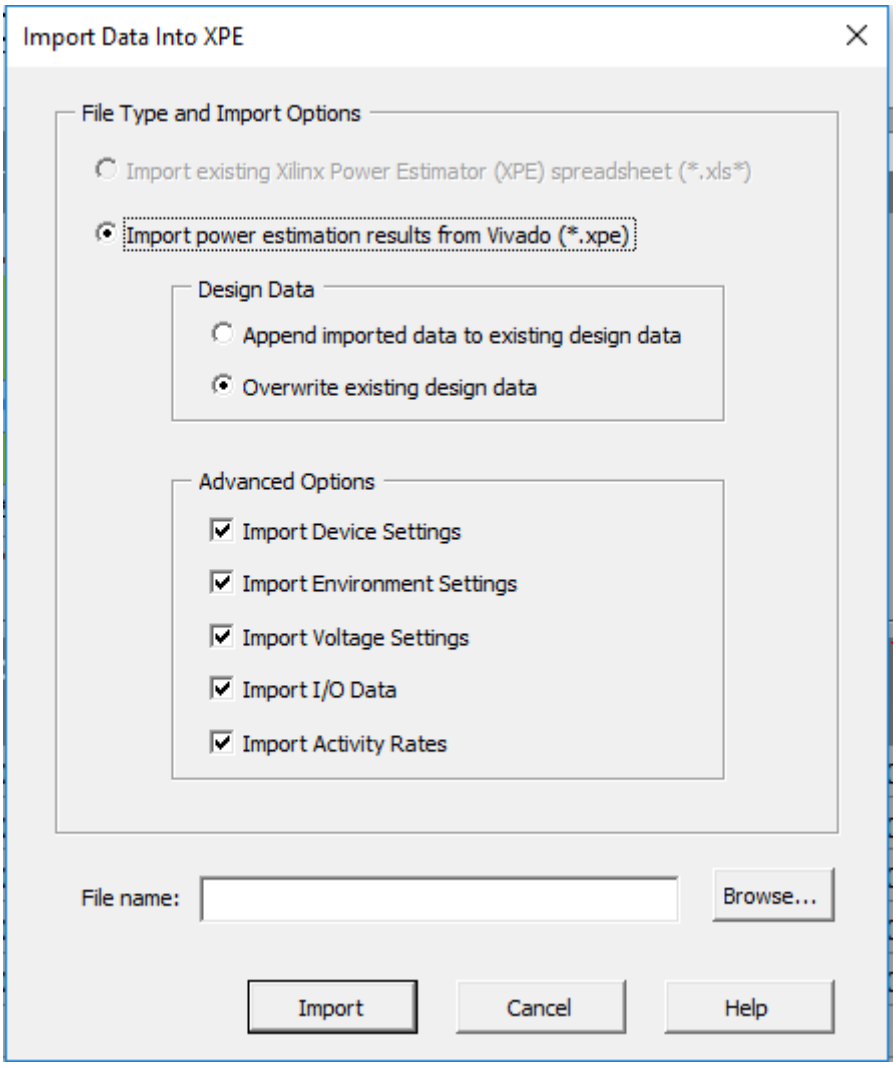
As XPE is an Excel macro-enabled workbook, you can save and open it as an MS Excel (`.xlsm`) file with all the data exactly restored. You can also export XPE data into a file (`.xpe`). This only stores XPE sheet configurations but not Snapshots or User sheet entries.

To export data from XPE, use the **Export** button on the Summary sheet at the right top corner.

Importing XPE Data

XPE can import data from a file (`.xpe`) that is generated by exporting the `.xpe` file from Versal® ACAP XPE or from Vivado. With XPE Exchange import, you can choose to either overwrite the existing data or append incoming data to the data that is already in XPE. There are Advanced Options to selectively import different data types as shown in the following figure:

Figure 38: **Importing XPE Data**



Importing XPE Data from a Previous Device Family

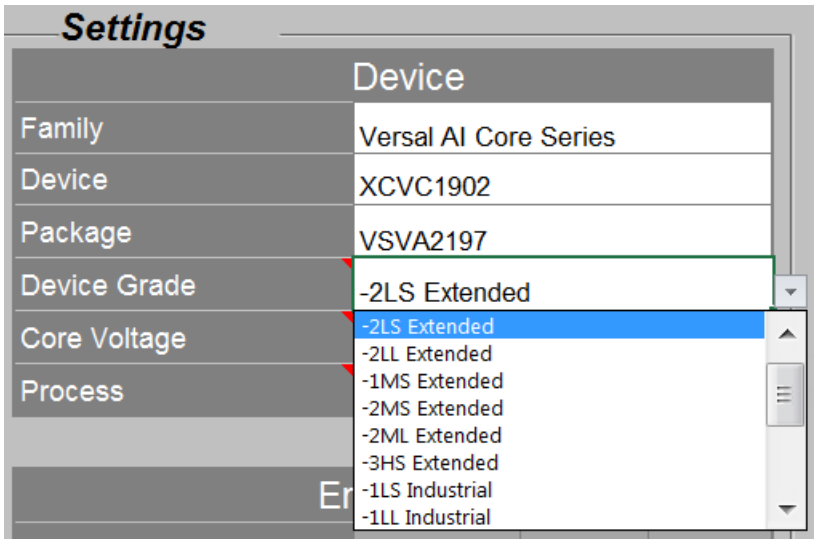
Some device resources are compatible with the Versal® architecture, making it possible to import XPE data from prior architectures. Versal ACAP XPE will apply the data to match Versal architecture as closely as possible. In the Versal ACAP XPE, use the **Load/Import .xpe** button under Create Design Wizard from the Summary sheet to import the data from the previously generated .xpe file.

***Note*:** Only data from UltraScale+™ families are supported for import.

After importing, review each page for incompatibility.

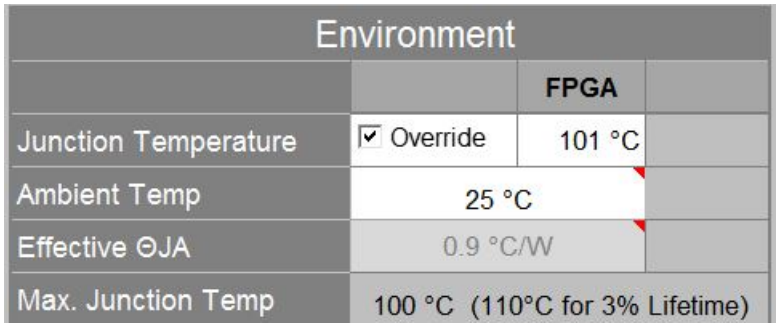
On the Summary sheet, review Device Settings and choose the target Versal device. For Versal ACAP XPE, Speed Grade and Temperature Grade have been combined into a single selection called Device Grade.

Figure 39: **Importing XPE Data Settings**



Next review Environment Settings. For Versal devices, Environment settings have been simplified to provide only Junction Temperature override, Ambient Temp, and Effective ΘJA. Either Junction Temperature must be overridden, or Effective ΘJA overridden with Ambient Temp specified.

Figure 40: **Importing XPE Data Environment**



Review the Power Design Sheet to ensure all voltage supply values are configured properly.

Incompatible Sheets

The following sheets are incompatible with prior generation devices and no data can be imported:

- **NoC_DDRMC:** Network on Chip and hardened DDR controller are new features that are not available for earlier devices.
- **AI Engine:** AI Engine is a new feature that is not available for earlier devices.

Compatible and Partially-Compatible Sheets

Data can be imported from prior families for certain sheets with further actions required. Incompatibilities such as invalid IO Standards are highlighted in yellow and must be corrected. Following sheets are mostly compatible.

- Logic
- IO
- Block RAM
- UltraRAM

***Note*:** The Logic, block RAM, and UltraRAM sheets are identical to those of the previous families. For BRAM, FIFO18 / FIFO36 modes from UltraScale+ is imported as RAMB18SDP / RAMB36SDP modes. The IO Sheet is also similar to the previous families. However, the higher performance I/O are called XPIO and are located under the XP I/O Type.

These sheets are partially-compatible and require careful review after import.

- **Clock:** Clocks are mapped one-to-one with the following rules:
  - Each clock in UltraScale+ is created as an externally driven clock in Versal ACAP XPE.
  - Clock fanout is ignored as it will be calculated based on the other sheet resources.

  - After import, it is important to review and modify all the clock definition in the Versal ACAP XPE.
- **PS:** PS sheet is mapped with the following rules:
  - Only user configuration are imported into Versal® XPE and power-down and deep-sleep are ignored.
  - Number of cores for A53 will be divided by 2 to covert to A72 in Versal ACAP XPE.
  - Retention mode for L2 cache and battery domain are ignored during import.
  - After import, all the settings must be reviewed and modified.
- **DSP:** DSPs are mapped one-to-one with the following rules:
  - Versal ACAP DSP Mode defaults to INT24.
  - Toggle Rate translates to Block Toggle Rate.
  - MREG translates to Multiplier Pipeline Used.
  - Versal Input Pipeline Used defaults to No.
- **GTH:** This will be imported as GTY with all other parameter imported as it is like data rate, mode etc.
- **GTY:** The following GTY values can be applied to Versal devices:
  - Name
  - GTY Channels
  - Operational Mode (Power down not currently supported)
  - Power Mode
  - Data Rate
  - Data Path
  - Data Mode

Other values such as Hard IP Block usage must be manually reviewed and entered. The PCS/ Hard-IP Mode must be chosen manually after importing.

***Note*:** Some incompatible values may not be highlighted in yellow.

Appendix A

Additional Resources and Legal Notices

References

These documents provide supplemental material useful with this guide:

1. *Versal ACAP AI Engine Programming Environment User Guide* (UG1076)
2. *Versal ACAP PCB Design User Guide* (UG863)
3. *Versal ACAP GTY and GTYP Transceivers Architecture Manual* (AM002)
4. *Versal ACAP AI Engine Architecture Manual* (AM009)
5. *Versal ACAP Technical Reference Manual* (AM011)
6. *Extending the Thermal Solution by Utilizing Excursion Temperatures* (WP517)

Please Read: Important Legal Notices

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.

Copyright