



XAPP152 (v1.1) February 18, 2000

Virtex Power Estimator User Guide

Summary

This application note is offered as complementary text to the Virtex power estimator worksheet. A completed Virtex design and a successful functional simulation should be performed before using this worksheet. The power estimator worksheet URL is:

<http://www.xilinx.com/support/techsup/powerest/index.htm>

Introduction

The Virtex power estimator worksheet estimates power consumption for a Virtex design before it is downloaded. It considers the design resource usage, toggle rates, I/O power, and many other factors in the estimation. The formulas used for calculations in the program are based on test design measurements.

Xilinx provides two versions of the power estimator, an Excel 97 version that works with Microsoft Office 97 software, and a CGI version for use with web browsers. They are identical in terms of estimations and data entries.

This application note explains how to use the Power Estimator Worksheet to calculate estimated power consumption for Virtex designs. Since this is an estimation tool, results may not match precisely with what is measured on the board.

The power estimator consists of five categories: CLB (configurable logic block) logic power, block SelectRAM power, CLKDLL (clock delay-locked loop power), Input/Output power, and the results. To estimate power with the worksheet, a designer must determine how to group portions of the design into modules, what resources each module contains, the respective clock frequencies, and average toggle rates.

CLB Logic Power

Table 1 shows the data entries required for the CLB Logic Power section in the Power Estimator. This section estimates the power consumption of the CLBs for a Virtex design. In this section, users need to partition designs into modules, specify area utilization, and toggle rates.

Table 1: CLB Logic Power

Module	Frequency (MHz)	CLB Slices	Flip-Flops/Latches	LUTs		Average Toggle Rate (%)	Routing Amount
				Shift Register	SelectRAM		
User Module 1	0	0	0	0	0	0%	Medium
User Module 2	0	0	0	0	0	0%	Medium
User Module 3	0	0	0	0	0	0%	Medium
User Module 4	0	0	0	0	0	0%	Medium
User Module 5	0	0	0	0	0	0%	Medium
User Module 6	0	0	0	0	0	0%	Medium
User Module 7	0	0	0	0	0	0%	Medium
User Module 8	0	0	0	0	0	0%	Medium

Modules

Modules are portions of a design. A designer could treat the entire design as one module and calculate its toggle rate. However, estimating power this way is not as accurate as when the design is divided into multiple modules. Generally, with more modules the estimate is better.

This Virtex power estimator allows designs to be partitioned into a maximum of eight modules. Determining how to partition the design into modules depends on user preference. Three partitioning approaches are presented below as guidelines.

Grouping by hierarchy

If a design contains hierarchical components at the top level, these components may be separated or grouped together to represent modules.

Grouping by clocks

If a design has several different clocks, the logic associated with each clock should be treated as a module. For accuracy, it is recommended that each module contains only one clock.

Grouping by functionality

For a design with sub-components that perform different functions, each sub-component can be considered as a module. For example, a microprocessor can be thought of as three main modules: an ALU, a Register File, and a Control System.

Frequency (MHz)

Frequency is the clock speed for the module. Again, it is strongly recommended that each module contains only one clock.

CLB Slices

This involves the total CLB usage of a module. This number is available from the synthesis report in a specific synthesis tool. For a more accurate result, MAP only this module in Xilinx Foundation software, and take the numbers from the map.mrp file. The map.mrp file is the output resource usage file produced by running the MAP program in the Xilinx Foundation software.

For schematic-based designs, obtaining this number is slightly more difficult. Designers can either estimate CLB usage based on the design structure or MAP the module and read the numbers from the map.mrp file.

Flip Flops or Latches

The total number of flip-flop and latch elements used for each module can be obtained from the synthesis report, the map.mrp file, or by adding up the registers from the schematics.

Shift Register LUTs

This is the total number of SRL16 elements used in each module.

SelectRAM LUTs

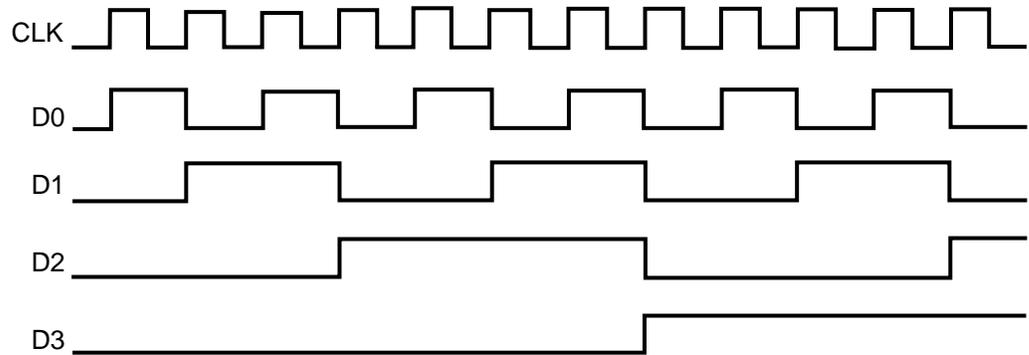
This is the total number of LUTs used as Distributed Select RAM components. For Virtex devices, one 16 x 1 synchronous RAM is equivalent to one LUT, and one 16 x 1 dual-port RAM is equivalent to two LUTs.

Average Toggle Rate (%)

The toggle rate describes how often the output changes with respect to the input clock, usually between 6% and 12% for a typical module. Functional simulation is required to accurately calculate the toggle rate. Designers need to simulate all the flip-flop outputs in each module

with regard to the clock, and calculate how often the flip-flop outputs change in relation to the clock.

Measuring the toggle rate becomes a more complex and a time-consuming process as module size increases. A toggle flip-flop has a 100% toggle rate, an 8-bit counter has 28%, and 16-bit counter has 14%. **Figure 1** is an example of how to calculate the toggle rate for a 4-bit counter.



x152_01_122999

Figure 1: Output Waveform of a 4-bit Counter

Figure 1 shows the simulation wave form of a 4-bit counter. D0 stands for the LSB of the count, and D3 stands for the MSB. The toggle rate of D0 is 100% because D0 changes after every clock cycle. The toggle rate of D1 is 50% because D1 changes after every two clock cycles. The toggle rate of D2 is 25% because D2 changes after every four clock cycles. The toggle rate of D3 is 12.5% because D3 changes after every eight clock cycles. The average toggle rate of a 4-bit counter is derived as follows:

$$(100\% + 50\% + 25\% + 12.5\%) / 4 = 46.875\%$$

Routing Amount

There are three levels concerning the amount of routing to be used: low, medium, and high. The routing level is determined by the primary logic type of the module. Typical data path logic typically requires a low routing usage, random logic calls for a medium level, and control logic needs a high level.

Typical data path logic is structured combinatorial logic, such as multiplexers, adders, AND gates, OR gates, or any other signals that have one to two fanouts between structures.

Random logic is logic such as decoders, encoders or any logic that has three to five fanouts.

Control logic is typically logic with high fanout signals (excluding clocks) such as clock enables or reset signals. Control logic used in state machines also belongs to this category.

Each designer needs to determine the routing that is most appropriate for each module.

Routing, which is determined by the type of logic in the module, is divided into three levels: low, medium, and high. Each designer needs to determine the routing that is most appropriate for each module.

1. Typical data path logic, which uses combinatorial logic such as multiplexers, adders, AND gates, and OR gates, usually requires a low routing usage. This also applies to any other signals that have one or two fanouts between structures.
2. Random logic, such as decoders, encoders, or any logic that has three to five fanouts, calls for a medium level of routing usage.
3. Control logic is typically logic with high fanout signals (excluding clocks) such as clock enables or reset signals. Control logic used in state machines also belongs to this category.

Block SelectRAM Power

Table 2 shows the data entries required for the Block SelectRAM Power section. This section is used to specify how many block RAMs are used and to determine their estimated power consumption. Before doing the calculation, designers can either treat all the RAMB4 cells as one module or break them down into smaller modules. RAMB4 is the base name for the Virtex Block SelectRAM component.

RAMB4 Cells

This is total number of Block Select RAMs (RAMB4 cells) used in each module.

Port A Frequency (MHz)

This is the frequency on the CLKA pin.

Port A Width

This is data width of DIA and DOA busses.

Port A Enable Rate (%)

This specifies how often ENA is enabled with respect to the clock. For a typical design, the rate may be 100% because the enable could be enabled all the time. For a FIFO design, the rate could be approximately 50% due to bursting of data into and out of the RAM.

Port B Frequency (MHz)

This is the frequency on the CLKB pin.

Port B Width

This is the data width of DIB and DOB busses.

Port B Enable Rate (%)

This specifies how often ENB is enabled with respect to the clock.

Table 2: Block SelectRAM Power

Module	RAMB4 Cells	Port A			Port B		
		Frequency (Mhz)	Width	Enable Rate (%)	Frequency (MHz)	Width	Enable Rate (%)
User Module 1	0	0	0	0	0%	0	0%
User Module 2	0	0	0	0	0%	0	0%
User Module 3	0	0	0	0	0%	0	0%
User Module 4	0	0	0	0	0%	0	0%
User Module 5	0	0	0	0	0%	0	0%
User Module 6	0	0	0	0	0%	0	0%
User Module 7	0	0	0	0	0%	0	0%
User Module 8	0	0	0	0	0%	0	0%

Clock Delay Locked Loop Power

Table 3 shows the data entries required for the CLKDLL Power section and is used to estimate how much power CLKDLLs consume. Only the clock input frequencies to the CLKIN pin needs to be entered.

Table 3: Clock Delay Locked Loop Power

Module	Clock Input Frequency (MHz)
User DLL 1	0
User DLL 2	0
User DLL 3	0
User DLL 4	0

Input/Output Power

Table 4 shows the data entries for the Input/Output Power section used to estimate the power dissipation of the Inputs and Outputs. Users should try to group the I/Os into modules based on their I/O standard type. If the entire design has only one I/O standard type, all the I/Os can be treated as one module. However, separating the I/Os into smaller modules makes it easier to obtain better accuracy.

Frequency (MHz)

This is the frequency of the module.

I/O Standard Type

This is the type of I/Os used in the module. Each module may have only one I/O standard type. I/O power is strongly influenced by the I/O standard used.

Inputs

This is the total number of the input buffers in each module.

Outputs

This is the total number of the output buffers in each module.

Average Output Toggle Rate (%)

This number can be obtained in the same way as obtaining the Average Toggle rate in the CLB Logic Power section.

Average Output Load (pF)

This specifies the average capacitive load on the outputs.

Table 4: Input/Output Power

Module	Frequency (MHz)	I/O Standard Type	Inputs	Outputs	Average Output Toggle Rate (%)	Average Output Load (pF)
User Module 1	0	LVTTTL_12	0	0	0%	0
User Module 2	0	LVTTTL_12	0	0	0%	0
User Module 3	0	LVTTTL_12	0	0	0%	0
User Module 4	0	LVTTTL_12	0	0	0%	0
User Module 5	0	LVTTTL_12	0	0	0%	0
User Module 6	0	LVTTTL_12	0	0	0%	0
User Module 7	0	LVTTTL_12	0	0	0%	0
User Module 8	0	LVTTTL_12	0	0	0%	0

Results

The results section of the power estimator are shown in Table 5. The four sections of the power estimator program independently estimate power consumption, and the results are displayed at the end of each section.

The total design power consumption is the summation of those, and is displayed at the very top of the program.

Table 5: Results

Target		Estimated Design Power Values (mW)					
Device	Package	Total Power	V _{CCINT} 2.5 V	V _{CCO} 3.3 V	V _{CCO} 2.5 V	V _{CCO} 1.5 V	Output Sink Power
XCV300	BG432	0	0	0	0	0	0

Target Device

This refers to the target Virtex device size.

Note: No checking is done to verify that the module entries fit into the amount of resources available in the selected devices.

Target Package

This refers to the package of the device.

Note: No checking is done to verify that the selected device-package combination is valid.

Estimated Total Power

This section displays the total power consumption of the design. It is the summation of the CLB Logic power, the Block Select RAM power, the Clock DLL power, and the Input/Output power.

Estimated V_{CCINT} 2.5V Power

This section displays the total power consumption from the core supply voltage (V_{CCINT}). It does not include the power consumption from the output source voltage (V_{CCO}).

Estimated V_{CCO} 3.3V Power

This section displays the V_{CCO} power consumption for 3.3 V applications. The I/O standards that use 3.3V V_{CCO} are LVTTTL, PCI, SSTL3 Class I and II, CTT, and AGP.

Estimated V_{CCO} 2.5V Power

This section displays the V_{CCO} power consumption for 2.5 V applications. The supported I/O standards are LVCMOS and SSTL2 Class I and II.

Estimated V_{CCO} 1.5V Power

This section displays the V_{CCO} power consumption for 1.5 V applications. The supported I/O standards are HSTL Class I, III and IV.

Estimated Output Sink Power

This section displays the power consumption when sinking current to ground. The supported I/O standards are GTL and GTL+.

Revision History

The following table shows the revision history for this document.

Date	Version #	Revision
4/28/99	1.0	Initial release.
2/18/00	1.1	Reformatted text.