

PrimeTime PX Workshop

Lab Guide

12-I-031-SLG-003

2009.06

Synopsys Customer Education Services
700 East Middlefield Road
Mountain View, California 94043

Workshop Registration: **1-800-793-3448**

www.synopsys.com

Copyright Notice and Proprietary Information

Copyright © 2009 Synopsys, Inc. All rights reserved. This software and documentation contain confidential and proprietary information that is the property of Synopsys, Inc. The software and documentation are furnished under a license agreement and may be used or copied only in accordance with the terms of the license agreement. No part of the software and documentation may be reproduced, transmitted, or translated, in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without prior written permission of Synopsys, Inc., or as expressly provided by the license agreement.

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Registered Trademarks (®)

Synopsys, AMPS, Astro, Behavior Extracting Synthesis Technology, Cadabra, CATS, Certify, CHIPit, Design Compiler, DesignWare, Formality, HDL Analyst, HSIM, HSPICE, Identify, IN-Phase, Leda, MAST, ModelTools, NanoSim, OpenVera, PathMill, Physical Compiler, PrimeTime, SCOPE, Simply Better Results, SiVL, SNUG, SolvNet, Syndicated, Synplicity, the Synplicity logo, Synplify, Synplify Pro, Synthesis Constraints Optimization Environment, TetraMAX, UMRBus, VCS, Vera, and YIELDirector are registered trademarks of Synopsys, Inc.

Trademarks (™)

AFGen, Apollo, Astro-Rail, Astro-Xtalk, Aurora, AvanWaves, BEST, Columbia, Columbia-CE, Confirma, Cosmos, CosmosLE, CosmosScope, CRITIC, CustomExplorer, CustomSim, DC Expert, DC Professional, DC Ultra, Design Analyzer, Design Vision, DesignerHDL, DesignPower, DFTMAX, Direct Silicon Access, Discovery, Eclypse, Encore, EPIC, Galaxy, Galaxy Custom Designer, HANEX, HAPS, HapsTrak, HDL Compiler, Hercules, Hierarchical Optimization Technology, High-performance ASIC Prototyping System, HSIMplus, i-Virtual Stepper, IICE, in-Sync, in-Tandem, Jupiter, Jupiter-DP, JupiterXT, JupiterXT-ASIC, Liberty, Libra-Passport, Library Compiler, Magellan, Mars, Mars-Rail, Mars-Xtalk, Milkyway, ModelSource, Module Compiler, MultiPoint, Physical Analyst, Planet, Planet-PL, Polaris, Power Compiler, Raphael, Saturn, Scirocco, Scirocco-i, Star-RCXT, Star-SimXT, StarRC, System Compiler, System Designer, Taurus, TotalRecall, TSUPREM-4, VCS Express, VCSI, VHDL Compiler, VirSim, and VMC are trademarks of Synopsys, Inc.

Service Marks (SM)

MAP-in, SVP Café, and TAP-in are service marks of Synopsys, Inc.

SystemC is a trademark of the Open SystemC Initiative and is used under license. ARM and AMBA are registered trademarks of ARM Limited. Saber is a registered trademark of SabreMark Limited Partnership and is used under license. All other product or company names may be trademarks of their respective owners.

Document Order Number: 12-I-031-SLG-003
PrimeTime PX Lab Guide

1

Introduction to Power Analysis

Learning Objectives

After completing this lab, you should be able to:

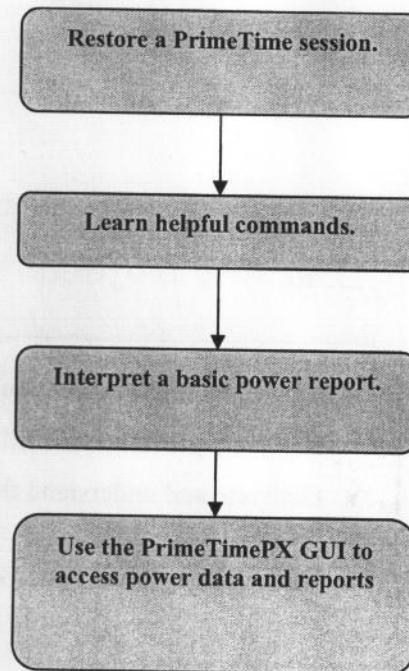
- Restore a previously saved PrimeTime session
- Generate and understand the key components of a power report.
- Navigate the PrimeTime GUI to access power data



Lab Duration:
30 minutes

Lab 1

Overview



Getting Started

If you need help...

```
pt_shell> help report*
pt_shell> printvar *power*
pt_shell> report_power -help
pt_shell> man power_analysis_mode
```

When working interactively, you do not need to type the entire command or option name. Use the <Tab> key to expand.

```
pt_shell> echo $power_en
pt_shell> set_power_analysis_opt <Tab> -help
pt_shell> check_pow -v
pt_shell> report_sw -h -lis
pt_shell> get_sw [all_inp]
```

Answers & Solutions

This lab guide contains answers and solutions to all questions.

You are encouraged to verify your results by checking the [Answers/Solutions](#) section at the end of this lab.

Objective:

In this lab, you will restore a saved PrimeTime session in order to generate power analysis reports using the shell and the GUI.

Lab 1

File Locations

Directory Structure

Reference Directory: ./ref

- ./ref/lib : Contains library files used in the labs
- ./ref/design_data : Contains netlist and SDF files
- ./ref/scripts : Contains PrimeTime scripts
- ./ref/tcl_procs : Contains Tcl procedures

Working Directories:

- ./lab1_intro
- ./lab2_average
- ./lab3_peak

Relevant Files for Lab1:

- ./lab1_intro/orca_savesession
- ./lab1_intro/.synopsys_pt.setup
- ./lab1_intro/README_lab1
- ./lab1_intro/update_session.sh
- ./lab1_intro/.solutions/run.tcl

Working Directory for Lab1:

- ./lab1_intro

Instructions

Your goal is to start mastering the power reporting commands.

Task 1. Restore a PrimeTime Session

Learn to invoke a previously saved PrimeTime session to perform STA.

1. Invoke PrimeTime from the **lab1_intro** Unix directory.

```
unix% cd lab1_intro
unix% pt_shell
```

2. Restore a previously saved PrimeTime session.

This step will read in the design netlist, libraries, and constraints. The design is now ready for analysis.

Note: The argument **orca_savesession** below is a Unix directory.

Note: Avoid typing while in **pt_shell**, type a few characters of a command or variable name and hit <Tab> to expand.

```
restore_session orca_savesession
```

Note: If you are using a different version of the tool, exit **pt_shell** and regenerate the saved session with the unix script **update_session.sh** provided in this directory.

3. The design has 6 clocks. Generate their timing reports and obtain their slacks.

```
report_clock
report_timing -path end
report_timing
```

4. Check the power setup. Execute **check_power**.

```
check_power
```

Lab 1

Question 1. What checks are performed by default? What is the setting of power_enable_analysis variable? (Hint: `printvar power*`)

-
5. Pipe the detailed listing of the failed checks into a file:
Execute `check_power -verbose > check_power.rpt`.
View the file.

```
check_power -verbose > check_power.rpt
```

Question 2. Are the out-of-range ramps over or under the library range?

.....

Question 3. What library cells have no power tables?

.....

Question 4. How many and what library cells are missing function?

Task 2. Power Analysis

1. Set the variable to enable power analysis.

```
set power_enable_analysis true
```

.....

Question 5. Does setting the variable cause a PrimeTime PX license to be checked out? (Hint: `list_licenses`)

2. Generate power reports and view the data.

```
report_power
```

.....

Question 6. What power group consumes the most power?

.....

Question 7. Why does black_box power have only net switching power?

3. View the power for all the instances within the io_pad power group.

```
report_power_groups  
get_power_group_objects io_pad  
report_power -group io_pad -cell
```

Question 8. How are the cells in the io_pad group sorted by default? What cell consumes the highest total power?

.....

Question 9. What option is used to sort the cells in the io_pad group by leakage power? What cells in this group consume the most leakage power? (Hint: report_power -help)

.....

4. Create a couple of your own power groups: One for all the top level cells in the design, another for the clock network objects and display their power consumption.

```
create_power_group -name all [get_cell *]  
create_power_group -name clock_tree \  
[get_clock_network_objects -type cell]  
report_power
```

Question 10. What command is used to list the available power groups? How can you tell which groups are pre-defined and which are user defined?

.....

5. View the power consumption per net. Execute **report_power -net**.

```
report_power -net
```

Question 11. Is this a multi VDD design?

.....

Question 12. How can you report the top 10 power consuming nets?

.....

Lab 1

6. Play with the different report_power options to obtain detailed analysis information.

```
report_power -hier  
report_power -cell  
report_power -net  
report_power ....
```

Task 3. Using the GUI

1. Open the PrimeTime PX GUI.

```
gui_start
```

2. Open the Power Analysis Driver from the top menu option or use the shortcut icons in the toolbar. **GUI → Power → Show Power Analysis Driver**

Question 13. What type of power is responsible for most of the total power of the design? Why?

3. Switch between Predefined Groups and User Defined Groups at the bottom left menu; note the changes. Now, set "Show by predefined groups" in this menu and select the power group with the highest internal power consumption.

Using the RMB (Right Mouse Button) or by double clicking, create a histogram of the internal power or total power distribution for this power group.

4. Select all cells that have non zero switching and internal power in the above group. Using the RMB create a toggle rate histogram of the selected cells.

5. Open menu **GUI→Power → New Power Design Map → Total Power Density**

a thermal map will appear on the screen. Area represents size, colors represent impact. Is there any "hot" cell? Using RMB **show all levels** and some red zone will appear. Positioning the pointer on the red colored cells, you'll get the total power consumption and the percentage wrt the max density value.

Question 14. Can you locate and name a cell having a very high current density?

6. Exercise different analysis types and thresholds using the GUI.

Lab 1

7. Save the session by issuing the **save_session** command in the console window and quit pt_shell.

GUI → File → Close GUI

```
save_session -replace orca_savesession  
quit
```

Congratulations. This completes lab 1 : Introduction to Power Analysis

Answers / Solutions

Question 1. What checks are performed by default? What is the setting of power_enable_analysis variable?

By default check_power performs 3 checks as revealed by the information messages and as well as :

```
pt_shell> printvar power_check_defaults
power_check_defaults =
"out_of_table_range missing_table
missing_function"
```

1. out_of_table_range:

This checks that the indeces for the transition time and output load used to access the power table energy values are within range. PrimeTime PX will extrapolate the energy value if it is beyond the range; unless the user sets the variable

power_limit_extrapolation_range to true (default false). Indeces which are far beyond the table range can result in inaccurate energy values.

2. missing_table:

This checks that power tables are available for all the cells in the design. If not available, PrimeTime PX assumes the power consumption for that cell is 0.

3. missing_function:

Checks if library cells have function statements. The function statements are used during toggle propagation and event simulation.

power_enable_analysis = "false" (check_power works regardless of this variable setting)

Question 2. Are the out-of-range ramps over or under the library range?

The out of range ramps are below the library range.

Question 3. What library cells have no internal power tables?

The library cells: CLKMUL and PLL have no internal power tables.

Question 4. How many and what library cells are missing function?

Information: Checking 'missing_function'.
library cell 'ram32x64' in library 'cb13_tsmc_memory_max' has an output pin without function.
library cell 'ram16x128' in library 'cb13_tsmc_memory_max' has an output pin without function.
library cell 'ram32x32' in library 'cb13_tsmc_memory_max' has an output pin without function.
library cell 'CLKMUL' in library 'cb13special' has an output pin without function.
library cell 'PLL' in library 'cb13special' has an output pin without function.
Warning: There are 5 library cells without function.

Question 5. Does setting the variable cause a PrimeTime PX license to be checked out?

No. The `check_power` command checks out the PrimeTime PX license. Notice that when it is checked out, the tool prints out the informative message: Checked out license PrimeTime-PX. Setting the power analysis variable enables the power calculation commands, it does not actually check out a license. Once the PrimeTime-PX license is checked out, it is not returned until the end of the session.

Question 6. What power group consumes the most power?

The `io_pad` group consumes the most power: ~.09 W; ie ~80% of the total design power.

Question 7. Why does black box power have only net switching power?

The black box cells have no power tables; so no internal power can be calculated for these cells. The net switching power is obtained via the $1/2cv^{**}2$ formula which considers the voltage, capacitance and switching of the black box outputs. Net switching power is associated with the net driver.

Question 8. How are the cells in the `io_pad` group sorted by default? What cell consumes the most total power?

The cells are by default sorted based on Total Power. The cells: `sdran_CK_ipad` and `sdran_CKn_ipad` consume the most power: ~.03W.

Lab 1

Answers / Solutions

Question 9. What option is used to sort the cells in the io_pad group by leakage power? What cells in this group consume the most leakage power?

To sort the cells in the io_pad group by leakage; issue the command: `report_power -cell -group io_pad -sort_by cell_leakage_power`. The cells: `sdram_DQ_iopad_0 - 15` consume the most leakage power: `7.215e-08`

Question 10. What command is used to list the available power groups? How can you tell which groups are pre-defined and which are user defined?

The command `report_power_group` lists the available power groups. Those which have been defined by the user have the attribute "user defined"; while the ones provided by the tool have the attribute: "default".

Question 11. Is this a multi VDD design?

Yes. (As per the Vdd column). This is simply because, the pad cells use 3v and the functional cells use 1.08v. "Multi Voltage" design will be discussed later in the Multi Voltage Power analysis unit.

Net	Vdd	Total Net Load	Static Prob.	Toggle Rate	Switching Power	Attrs
net_sdram_CK	1.08	0.141	0.500	0.2667	2.200e-05	P
net_sdram_CKn	1.08	0.141	0.500	0.2667	2.200e-05	P
buf_sys_2x_clk_G6B1I1_1ASTHIRNet792	1.08	0.061	0.500	0.5000	1.767e-05	P
buf_sys_2x_clk	1.08	0.037	0.500	0.5000	1.090e-05	P
net_sys_clk	1.08	0.043	0.500	0.2500	6.240e-06	P
net_sdram_clk	1.08	0.037	0.500	0.2667	5.774e-06	P
net_pclk	1.08	0.037	0.500	0.1333	2.987e-06	P
s_sys_2x_clk	1.08	0.009	0.500	0.5000	2.614e-06	P
n50	1.08	3.453	0.472	0.0013	2.564e-06	P
n51ASTttcNet1618	1.08	3.453	0.472	0.0013	2.564e-06	P
sd_CK	3.00	0.001	0.500	0.2667	1.758e-06	a
sd_CKn	3.00	0.001	0.500	0.2667	1.758e-06	a
n128ASTttcNet541	1.08	1.726	0.438	0.0014	1.396e-06	P

Question 12. How can you report the top 10 power consuming nets?

Issue the command: `report_power -net -nworst 10`

Question 13. What type of power is responsible for most of the total power of the design? Why?

The **internal power** is responsible for most of the total power consumption of this design. It consumes **~11W**. This is due to neither event file nor switching activity data present for power estimation. The command will propagate switching activity values for power calculation.

Question 14. Can you locate and name a cell having a very high current density?

I_CLOK_GEN/U11

Total power = 0.192185 mW
Total power density 3.49427e-05

Lab 1**Answers / Solutions**

This page is left blank intentionally.

2

Average Power Analysis

Learning Objectives

After completing this lab, you should be able to:

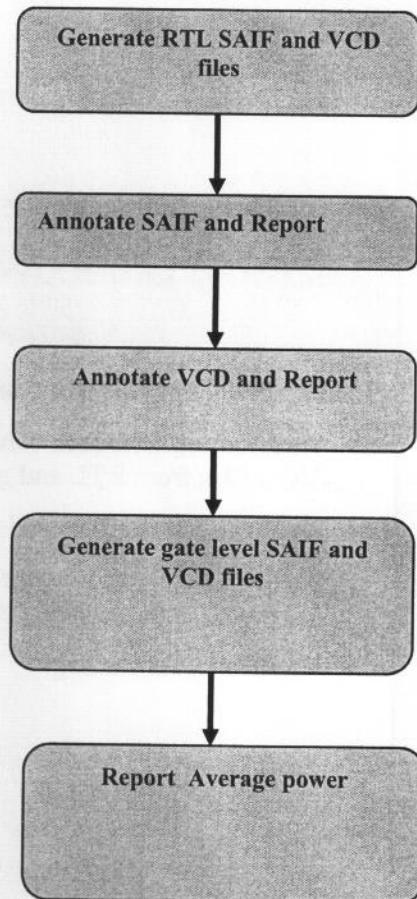
- Add Verilog testbench commands to generate SAIF and VCD files from RTL and gate-level Verilog simulations.
- Read the activity files and report the switching activity.
- Perform statistical average power analysis using SAIF based vs. VCD based annotations
- Compare your findings



Lab Duration:
30 minutes

Lab 2

Overview



Getting Started

If you need help...

```
pt_shell> help report*
pt_shell> printvar *power*
pt_shell> read_saif -help
pt_shell> man read_vcd
```

When working interactively, you do not need to type the entire command or option name. Use the <Tab> key to expand.

```
pt_shell> echo $power_en
pt_shell> report_sw -h -lis
pt_shell> set_power_analysis_opt -help
pt_shell> get_sw [all_inp]
```

Answers & Solutions

This lab guide contains answers and solutions to all questions.

You are encouraged to verify your results by checking the [Answers/Solutions](#) section at the end of this lab.

Objective:

In this lab, you will:

- Run VCS simulation to generate the switching activity files (SAIF and VCD)
- Annotate switching activity from SAIF file and report average power
- Change design activity information by specifying switching activity from VCD file
- Perform average power analysis and compare your findings with the previous report

Lab 2

File Locations

Directory Structure

Reference Directory: ./ref

./ref/lib	:	Contains library files used in the labs
./ref/design_data	:	Contains netlist and SDF files
./ref/scripts	:	Contains PrimeTime scripts
./ref/tcl_procs	:	Contains Tcl procedures

Working Directories:

```
./lab1_intro  
./lab2_average ←  
./lab3_peak
```

Relevant Files for Lab2:

```
./lab2_average/ptpx/lab2.tcl  
./lab2_average/sim/runsim_rtl  
./lab2_average/sim/runsim_gate  
./lab2_average /src/hdl/tb/tb.v  
./lab2_average /src/hdl/rtl/mac.v  
./lab2_average/src/lib/verilog/core.v  
./lab2_average/src/lib/snps/core_typ.db  
./lab2_average/src/hdl/gate/mac.vg  
./lab2_average/src/hdl/gate/mac.sdc  
./lab2_average/src/annotate/mac.spf
```

Working Directory for Lab2:

```
./lab2_average
```

Instructions

Task 1. Generate RTL SAIF & VCD files from RTL Simulation and Compare the Power

1. Go to the lab2_average lab directory.

```
unix% cd lab2_average
```

2. Using your favorite text editor, edit the testbench. Your goal is to ensure that RTL points will be saved when generating RTL SAIF.

```
unix> <Editor> src/hdl/tb/tb.v
```

Scroll down to the RTL_SAIF section and verify that the option to \$set_gate_level_monitoring is correct.

Question 1. What option to the \$set_gate_level_monitoring command is specified to write out an RTL SAIF file?

.....

3. Execute the simulation runscript: sim/runsim_rtl to generate the RTL SAIF and VCD files.

```
unix> cd sim
```

```
unix> ./runsim_rtl (OR) source ./runsim_rtl
```

Question 2. Where are the RTL SAIF and RTL VCD files placed? View the testbench/src/hdl/tb/tb.v to determine the name and location of the activity files.

.....

4. Load the design data, verify the timing, and save the timing data into "mac_timing" using pre-written script.

```
unix> cd ../../ptpx
```

```
unix> pt_shell -f lab2.tcl | tee -i lab2.log
```

Lab 2

5. Read in the RTL SAIF file generated from simulation by supplying the "strip_path" information.

```
read_saif -strip_path ????.src/annotate/rtl.saif
```

- Question 3.** What is the strip path? View the SAIF file to determine the path to the top level instance starting with the test bench out-of-range ramps over or under the library range?
-

6. Report the percentage annotation and note that some of the ports/pins are listed as not annotated.

```
report_switching_activity -include_only rtl
```

7. Generate a list of the unannotated pins/ports.

```
report_switching_activity -include_only rtl \
                         -list_not_annotated
```

8. View the gate-level netlist,/src/hdl/gate/mac.vg and notice how the RTL signal names "a_lt" and "b_lt" have been renamed.

- Question 4.** What difference does the -include_only rtl option make to the report_switching_activity command?
-

9. Ensure that the analysis mode is “averaged” and Report power consumption.

```
printvar power_analysis_mode
report_power
```

10. Save the switching annotation and power report into a file and store the session

```
report_switching -include_only rtl > rtl_saif.pwr
report_power >> rtl_saif.pwr
save_session -replace mac_avg
```

Lab 2

11. Reset the switching activity and annotate activity from the RTL VCD file. Verify the annotation before reset and read_vcd.

```
reset_switching_activity
report_switching
read_vcd -strip_path tb/macinst \
          ../../src/annotate/rtl.vcd
```

12. View the annotation and power consumption and save it into a file. Exit pt_shell.

```
report_power
report_switching -include_only rtl > rtl_vcd.pwr
report_power >> rtl_vcd.pwr
```

13. Compare the annotation and total power between the SAIF and VCD flows. Record your findings in the following table

Annotation Data	% Nets	% Sequential cells	Total Power
SAIF			
VCD			

- Question 5.** Why does the RTL VCD file annotate more sequential cells? What nets are in the RTL VCD file which are not in the RTL SAIF file?
-

Lab 2

14. Use the `set_switching_activity` command to change design activity. Example follows.

```
set_switching_activity -static_probability 0.015 \
                       -toggle_count 0.25 \
                       -glitch_count 0.05 \
                       -type inputs

report_switching

set_switching_activity -clocks clk \
                       -period 12 -toggle 1 \
                       -static .5 -type nets

report_switching
```

Note: report_switching should have different number of objects passing from the Activity File column into SSA column. Exercise more options of `set_switching_activity` command and `report_power` to see differences.

```
exit
```

Task 2. Generate Gate level SAIF/VCD and Compare the Average Power values reported

1. Execute the runscript to generate VCD/SAIF from gate-level simulation.

```
% cd ../sim
% ./runsim_gate (OR) source ./runsim_gate
```

2. Restore mac_timing session, annotate activity and compare average power.

```
% cd ../ptpx
% pt_shell
restore_session mac_timing
read_saif -strip_path tb/macinst
../src/annotate/gate.saif
report_power
reset_switching_activity
read_vcd -strip_path tb/macinst
../src/annotate/gate.vcd
report_power
```

3. Record the number of annotations and power consumption in the following table

Annotation Data	% annotated nets	% annotated leaf cells	Total Power
SAIF			
VCD			

Question 6. How close are the results? Why?

.....

Question 7. What type of analysis (average vs time_based) was performed?

.....

Lab 2

4. Do a specific report of the annotated activity using the `get_switching` command. You may try more examples.

```
get_switching_activity \
    -include_only {sequential combinational} \
    -average mult_21
get_switching_activity -only_related_clock clk \
    [all_inputs]
```

Congratulations. This completes lab 2 : Average Power Analysis

Answers / Solutions

Question 1. What option is specified to the **\$set_gate_level_monitoring** command to write out an RTL SAIF file from simulation?

The option "**rtl_on**" should be applied. This causes VCS to store the ports and registers in the design.

Question 2. Where are the RTL SAIF and RTL VCD files placed? View the test bench `../src/hdl/tb/tb.v` to determine the name and location of the activity files

The SAIF and VCD files are located under:
src/annotate/rtl.saif and
src/annotate/rtl.vcd respectively.

Question 3. What is the strip path?

The strip path is: **tb/macinst**

Question 4. What difference does the -rtl option make to the **report_switching_activity** command? What if it is not applied to an RTL saif?

The -rtl option tells the tool to report switching activity on the primary ports and sequential element outputs. If the user does NOT apply the -rtl option, then the **report_switching_activity** command checks ALL the nodes in the design and issues a report. If only RTL activity has been provided, it makes sense to use the -rtl option, since the combinational logic is not included.

Question 5. Why does the RTL VCD file annotate more sequential cells? What nets are in the RTL VC file that is not in the RTL SAIF file?

When generating RTL SAIF from simulation, only the activity of nets declared as registers are saved in the RTL SAIF file. When generating VCD, both signals declared as regs and wires are saved in the VCD.

When mapping RTL regs to gate-level nets, PrimeTime PX assumes the gate-level register Q pin name matches that of the driving register. In this design, the registers are connected to the **_concat_out** nets which are included in the RTL VCD file; but not the RTL SAIF file because they were declared as wires.

Question 6. How close are the results? Why?

The results match exactly: both report $\sim 3.9e-03 \text{ W}$. This is because internally, PrimeTime PX converts the VCD data to SAIF. Both sets of data contain the toggle information for all the nets in the design; the calculation of power is the same for both.

Question 7. What type of analysis (average vs / time_based) was performed when this command was issued?

Information: Running averaged power analysis... (PWR-601)

Report : Averaged Power

Design : mac

Version: C-2009.06-SP2

Date : Mon Oct 12 13:19:47 2009

3

Peak Power Analysis

Learning Objectives

After completing this lab, you should be able to:

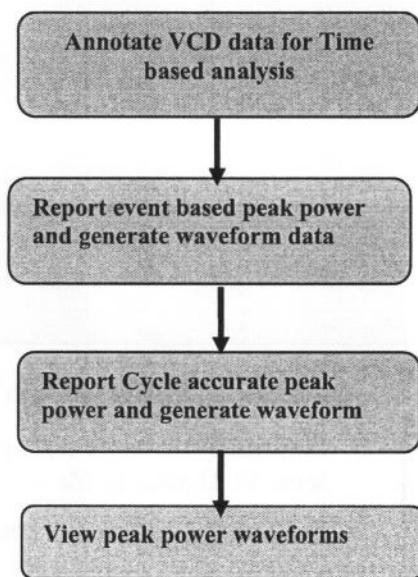
- Perform event based peak power analysis using gate level VCD activity file
- Perform Cycle accurate peak power analysis using RTL and gate level VCD activity files
- Generate and view peak power waveforms
- Explain the reported power results: peak power and peak time



Lab Duration:
30 minutes

Lab 3

Overview



Getting Started

If you need help...

```
pt_shell> help report*
pt_shell> printvar *power*
pt_shell> man power_analysis_mode
pt_shell> read_vcd -help
```

When working interactively, you do not need to type the entire command or option name. Use the <Tab> key to expand.

```
pt_shell> echo $power_en
pt_shell> report_sw -h -lis
pt_shell> set_power_analysis_options -help
pt_shell> get_sw [all_inp]
```

Answers & Solutions

This lab guide contains answers and solutions to all questions.

You are encouraged to verify your results by checking the [Answers/Solutions](#) section at the end of this lab.

Objective:

In this lab, you will:

- Using the VCD switching activity file, perform a time based analysis to report event based peak power and generate waveform data
- Perform Cycle accurate peak power analysis and generate waveform data
- View the peak power waveforms using nWave waveform displayer

Lab 3

File Locations

Directory Structure

Reference Directory: ./ref

./ref/lib	:	Contains library files used in the labs
./ref/design_data	:	Contains netlist and SDF files
./ref/scripts	:	Contains PrimeTime scripts
./ref/tcl_procs	:	Contains Tcl procedures

Working Directories:

./lab1_intro
./lab2_average
./lab3_peak

Relevant Files for Lab3:

./lab2_average/ptpx/mac_timing
./lab3_peak/sim/runsim_rtl
./lab3_peak/src/annotate/rtl.vcd
./lab3_peak/src/annotate/gate.vcd

Working Directory for Lab3:

./lab3_peak

Instructions

Task 1. Perform Event Based Peak Power Analysis Using Gate Level VCD Activity

1. Invoke pt_shell from the lab directory containing a previously saved session

```
unix% cd lab3_peak/ptpx  
unix% ln -s ../../lab2_average/ptpx/mac_timing/  
unix% pt_shell
```

Note: If you do not have a previously saved session, use the following UNIX commands to create one, prior to this step.

```
unix% cd lab2_average/ptpx  
unix% pt_shell -f lab2.tcl
```

2. Restore the previously saved design mac_timing

```
pt_shell> restore_session mac_timing
```

Question 1. What power analysis mode is the design currently set to?

.....

Question 2. Does the design have any activity annotation?

.....

3. Specify the power analysis mode for event based peak power analysis and annotate the gate level VCD activity file

```
pt_shell> set power_analysis_mode time_based  
pt_shell> read_vcd -strip_path tb/macinst  
.../src/annotate/gate.vcd
```

Question 3. How well is the design annotated now?

.....

Lab 3

4. Specify waveform output file name to be “**peak_gate.fsdb**”

```
pt_shell> set_power_analysis_options -help  
pt_shell> set_power_analysis_options \  
        -waveform_output peak_gate
```

5. Perform peak power analysis and generate report.

```
pt_shell> update_power  
pt_shell> report_power
```

6. Record your findings from the above power report

- a. Total Power: _____
- b. Peak Power: _____
- c. Peak Time: _____

Question 4. What confirms that this to be a peak power report?

.....

Task 2. Perform Cycle Accurate Peak Power Analysis Using RTL and Gate Level VCD Activity Files

1. Specify the power analysis mode for cycle accurate peak power (CAPP) analysis and annotate the RTL VCD activity file

```
pt_shell> set power_analysis_mode time_based  
pt_shell> reset_switching_activity; # time_based?  
pt_shell> read_vcd -rtl -strip_path tb/macinst  
./src/annotate/rtl.vcd
```

Question 5. What is the percentage of annotation?

.....

2. Specify waveform output file name to be “**capp_rtl.fsdb**”

```
pt_shell> set_power_analysis_options -help  
pt_shell> set_power_analysis_options \  
-waveform_output capp_rtl
```

3. Perform cycle accurate peak power analysis and generate report.

```
pt_shell> update_power; report_power
```

4. Record your findings from the above power report

- a. Total Power: _____
- b. Peak Power: _____
- c. Peak Time: _____

Question 6. What confirms this to be a cycle accurate peak power report?

.....

Question 7. What caused the analysis to be CAPP?

.....

Question 8. Would you have performed event based peak instead of CAPP in this case? Why (not) ?

.....

Lab 3

5. Reset existing switching activity and annotate gate level VCD activity file for CAPP analysis

```
pt_shell> reset_switching_activity
pt_shell> printvar power_analysis_mode; #time_based?
pt_shell> read_vcd -zero_delay \
           -strip_path tb/macinst ../src/annotate/gate.vcd
```

Question 9. What is the percentage of annotation?

.....

6. Specify waveform output file name to be “**capp_gate.fsdb**”. Specify sampling interval to be same as your clock period (HINT: **report_clock**)

```
pt_shell> set_power_analysis_options -help
pt_shell> set PER [get_attr [get_clocks clk] period]
pt_shell> set_power_analysis_options \
           -waveform_output capp_gate \
           -waveform_interval $PER
```

7. Perform cycle accurate peak power analysis and generate report.

```
pt_shell> update_power; report_power
```

Question 10. What caused the analysis to be CAPP?

.....

8. Record your findings from the power report

- a. Total Power: _____
- b. Peak Power: _____
- c. Peak Time: _____

9. Verify that the waveform files have been created

```
pt_shell> ls -al *.fsdb
```

Task 3. View the Peak Power Waveforms

1. View gate-level peak power waveforms (event based and CAPP) within the PT GUI that invokes nWave. Notice that the waveforms cover the entire simulation period.

```

pt_shell> start_gui
PT GUI: Power → View Waveforms...
Waveform File to Open: peak_gate.fsdb
<Wait for the nWave window to open up)
nWave GUI → Signal → Get All Signals (OK)
<View the waveform>

nWave GUI → Tools → New Waveform
nWave GUI → File → Open → capp_gate.fsdb
nWave GUI → Signal → Get All Signals
<View the waveform>

nWave GUI → File → exit
PT GUI: File → Close GUI

```

Note: You may need to "Maximize" and/or "Zoom" the waveform window (**Z = zoom in** and **z = zoom out**)

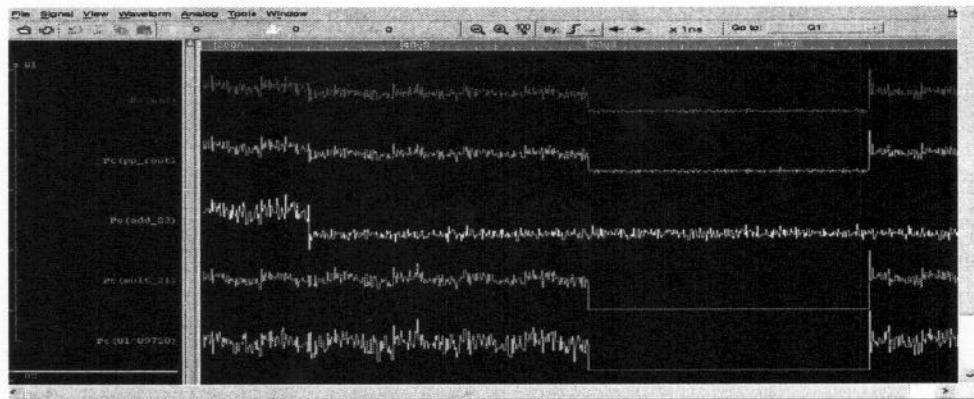


Fig-1: Instantaneous Peak power waveform

Lab 3

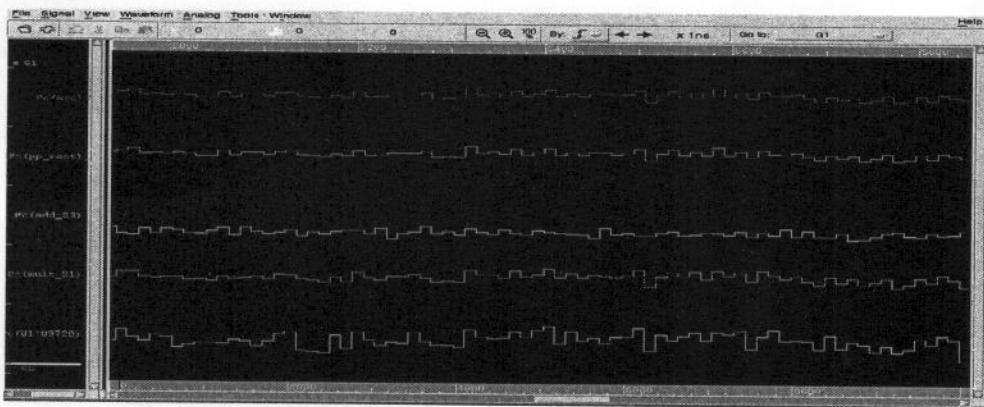


Fig-2: Cycle Accurate peak power waveform

2. Exit pt_shell

```
pt_shell> exit
```

Task 4. Explain reported peak power and peak time

- From the previous tasks, you should have seen the following reported:

Analysis performed	Activity annotated	Annotation percentage	Peak power (mW)	Peak time (ns)
Event based Peak	Gate level VCD (full timing SDF based)	100%	62.7mW	1596ns
CAPP (“rtl”, waveform interval = 12ns)	RTL VCD	27%	5.476 mW	9000ns
CAPP (“zero_delay”, waveform interval = 12ns)	Gate Level VCD (full timing SDF annotated)	100%	6.79mW	9000ns

Table-1: Peak Power (Time based) analysis results

Note: You are encouraged to consult the Answers section for the questions in this task

Question 11. Can you explain the gate level event based peak power compared to the zero delay peak power (CAPP)?

.....

Question 12. Can you explain the gate level event based peak time compared to the zero delay peak time (CAPP)?

.....

Question 13. Can you explain the RTL CAPP compared to gate level zero delay peak power (CAPP)?

.....

Congratulations. This completes lab 3: Peak Power Analysis

Answers / Solutions

Question 1. What power analysis mode is the design currently set to?

```
pt_shell> printvar power_analysis_mode  
power_analysis_mode = "averaged"
```

Question 2. Does the design have any activity annotation?

There is only “Default” annotation as per
report_switching_activity

Question 3. How well is the design annotated now?

```
pt_shell> read_vcd -strip_path  
tb/macinst ../src/annotate/gate.vcd
```

Summary:

Total number of nets = 1629

Number of annotated nets = 1629 (100.00%)

Total number of leaf cells = 1339

Number of fully annotated leaf cells = 1339 (100.00%)

Question 4. What confirms that this to be a peak power report?

```
pt_shell> update_power
```

Information: Running time_based power analysis... (PWR-601)

```
pt_shell> report_power
```

```
*****
```

Report : Time Based Power

Design : mac

Version: C-2009.06-SP2

Date : Wed Oct 14 06:57:21 2009

```
*****
```

Peak Power = 0.0627

Peak Time = 1596.00

Question 5. What is the percentage of annotation?

```
pt_shell> read_vcd -rtl -strip_path  
tb/macinst ../src/annotate/rtl.vcd
```

Summary:

Total number of nets = 1629

Number of annotated nets = 453 (27.81%)

Total number of leaf cells = 1339

Number of fully annotated leaf cells = 130 (9.71%)

Question 6. What confirms this to be a cycle accurate peak power report?

```
pt_shell> update_power; report_power
```

Information: Clock clk is selected as the reference clock for cycle accurate peak power analysis of the current design.
(PWR-280)

```
*****
```

Report : Time Based Power (Cycle Accurate)

Design : mac

Version: C-2009.06-SP2

Date : Wed Oct 14 07:03:57 2009

```
*****
```

Question 7. What caused the analysis to be CAPP?

```
set power_analysis_mode time_based
```

AND

```
read_vcd -rtl -strip_path tb/macinst  
../src/annotate/rtl.vcd
```

(The “-rtl” switch)

Question 8. Would you have performed event based peak instead of CAPP in this case? Why (not) ?

No, since the VCD coverage is too small as revealed by the PWR-007 message as below

```
pt_shell> read_vcd -strip_path  
tb/macinst ../src/annotate/rtl.vcd
```

Summary:

Total number of nets = 1629

Number of annotated nets = 453 (27.81%)

Total number of leaf cells = 1339

Number of fully annotated leaf cells = 130 (9.71%)

Warning: VCD coverage on the design is too small. Power may be underestimated. (PWR-007)

Question 9. What is the percentage of annotation?

```
pt_shell> read_vcd -zero_delay -  
strip_path tb/macinst  
../src/annotate/gate.vcd
```

Summary:

Total number of nets = 1629

Number of annotated nets = 1629 (100.00%)

Total number of leaf cells = 1339

Number of fully annotated leaf cells = 1339 (100.00%)

Question 10. What caused the analysis to be CAPP?

```
set power_analysis_mode time_based
AND
read_vcd -zero_delay -strip_path
tb/macinst ../src/annotate/gate.vcd
(The "-zero_delay" switch)
```

Note: The following *issue* has been fixed for the D-2009.12 version of PrimeTime PX

STAR 9000347027 FIXED B1|

*report_power header is incorrect for CAPP
using gate level zero delay*

VCD|PrimeTime PX

Question 11. Can you explain the gate level event based peak power compared to the zero delay peak power (CAPP)?

Gate level event based peak power	= 62.7mW
Gate level zero-delay peak power (CAPP)	= 6.79mW

Although, there is ~10X difference in the peak power reported, the difference between the CAPP peak power and the event-based peak power is expected and is due to the difference in the waveform interval applied to the analysis.

- In event-based analysis, PTPX calculates the energy for every event and averages the energy over the path delay (or input transition time if the energy is associated with an input transition). The resolution of the resulting waveform corresponds to the resolution of the vcd file (ie .01ns in this case).
- During CAPP analysis, PTPX propagates the VCD activity per clock cycle, sums up the energy for all the transitions in that cycle, and then averages the energy over the clock cycle (ie 12ns in this case).

If you run the gate-level event based analysis with a 12 ns interval (ie `set_power_analysis_options -waveform_interval 12`); you will see that the peak power reported is 6.79mW which matches that of the zero-delay analysis. Fundamentally, any time you change the interval, the peak will be reduced because rather than averaging the energy over the time when the event occurred, it is being averaged over a larger interval.

Question 12. Can you explain the gate level event based peak time compared to the zero delay peak time (CAPP)?

Gate level event based peak time = 1596ns
Gate level zero-delay peak time (CAPP) = 9000ns
Again, because of the difference in the waveform interval, the gate-level event-based peak power is not expected to correspond to the CAPP "peak" power. If you examine the waveform file produced by the event-based analysis, to measure the average power over 12ns for:

- Time 1596 (where event based peak occurred)
- Time 9000 (where CAPP peak occurred)

You will see that, when averaged over 12ns; the power for:

- $1596 - 1608 = 5.36\text{mW}$
- $9000 - 9012 = 6.80\text{mW}$

Yes – at time 1596 there is a big spike, but immediately following it – the power drops. As a result, when averaged over 12ns, the "average" over the clock cycle is less at time 1596 than it is at time 9000.

Note:

To measure the average power over an interval in nWave, use the cursors (left mouse for one, and right mouse for the other) to mark the desired interval, then select the analysis signal name; go to the menu option: Analog => Average/Min/Max and it pops open a window displaying these values for the desired interval.

Question 13. Can you explain the RTL CAPP compared to gate level zero delay peak power (CAPP)?

RTL CAPP reported = 5.476mW
Gate level zero-delay peak power (CAPP) = 6.79mW
For this design, the total number of events in the RTL CAPP analysis is less than the total number of events in the zero-delay CAPP analysis using gate-level VCD. The reason for this is:

- RTL CAPP analysis uses zero-delay simulation to instantly propagate the RTL events.
- The gate.vcd file is not actually a VCD file produced by zero-delay simulation, and includes all the “glitch” events which occur during sdf based simulation. All the logic settling which occurs through the multiplier are in fact contained in the gate level VCD file.

The zero-delay option is intended for zero-delay VCD files. In this mode, PTPX sums up the energy for the events per cycle and averages it over the cycle. The gate-level VCD has more events than a zero-delay VCD which does not contain all the intermittent logic values. The results will be comparable if using a VCD file produced by zero-delay verilog simulation for the CAPP zero delay analysis.

Lab 3

Answers / Solutions

This page is left blank intentionally.

4

Clock Network Power Analysis

Learning Objectives

After completing this lab, you should be able to:

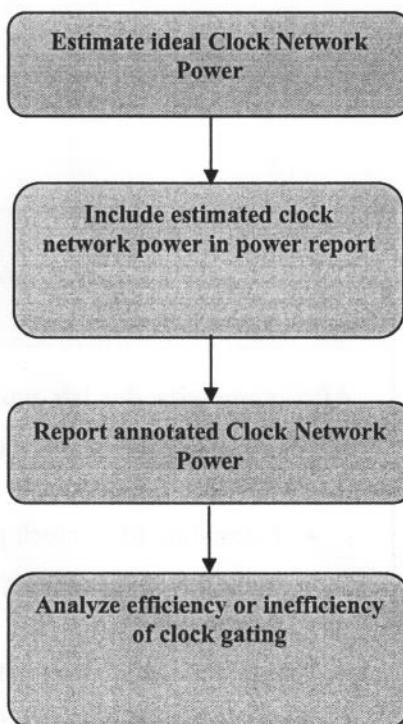
- Analyze and estimate pre-CTS clock tree power that includes:
 - Estimating ideal clock network power
 - Including the estimated clock network power in the design report
- Report annotated clock network power
- Analyze the efficiency or inefficiency of clock gating in the design



Lab Duration:
30 minutes

Lab 4

Overview



Getting Started

If you need help...

```
pt_shell> help estimate*power
pt_shell> printvar *power*
pt_shell> report_clock_gate_savings -help
pt_shell> man estimate_clock_network_power
```

When working interactively, you do not need to type the entire command or option name. Use the <Tab> key to expand.

```
pt_shell> echo \
$power_clock_network_include_register_clock_pin_power
pt_shell> report_pow -gr
pt_shell> set_ideal_network [get_ports clk2]
pt_shell> report_power -include_estimated
```

Answers & Solutions

This lab guide contains answers and solutions to all questions.

You are encouraged to verify your results by checking the [Answers/Solutions](#) section at the end of this lab.

Objective:

In this lab, you will:

- Estimate and report pre-layout ideal clock network power
- Report annotated clock network power
- Analyze the efficiency or inefficiency of clock gating in the design

Lab 4

File Locations

Directory Structure

Reference Directory: ./ref

- | | |
|-------------------|---|
| ./ref/lib | : Contains library files used in the labs |
| ./ref/design_data | : Contains netlist and SDF files |
| ./ref/scripts | : Contains PrimeTime scripts |
| ./ref/tcl_procs | : Contains Tcl procedures |

Working Directories:

- ./lab1_intro
- ./lab2_average
- ./lab3_peak
- ./lab4_clock_tree**

Relevant Files for Lab4:

- ./lab4_clock_tree/ptpx/lab3.tcl
- ./lab4_clock_tree/src/lib/snps/core_*.db
- ./lab4_clock_tree/src/hdl/gate/mac.vg
- ./lab4_clock_tree/src/hdl/gate/mac.sdc
- ./lab4_clock_tree/src/annotate/mac.spf

Working Directory for Lab4:

- ./lab4_clock_tree**

Instructions

Task 1. Analyze and Estimate Clock Network Power

1. Invoke PrimeTime from the **lab4_clock_tree/ptpx** Unix directory and source the lab4.tcl script to load the design data

```
unix% cd lab4_clock_tree/ptpx
unix% pt_shell -f lab4.tcl | tee -i lab4.log
```

2. View clock network power.

```
report_power -group clock_network -cell
report_power -group register -cell
```

Question 1. Why are registers listed in both groups? When is the register activity considered as part of the clock network power?

3. Report clock network and register power groups disabling the variable: `power_clock_network_include_register_clock_pin_power`

```
set power_clock_network_include_register_clock_pin_power
false
report_power -group clock_network -cell
report_power -group register -cell
```

Question 2. Explain the results. Why are registers no longer part of the clock network power?

4. Estimate the clock network power and determine the power consumption of the clock tree.

```
estimate_clock_network_power ssc_core_typ/buflal
```

Question 3. What is the estimated average transition time on the clock tree output?

Lab 4

5. Generate power reports with and without including the estimated clock network power in the power report

```
report_power -include_estimated_clock_network  
report_power
```

6. Set the clock network as ideal, and specify transition times based on estimation.

```
set_ideal_network [get_ports clk2]  
set_ideal_transition .26 [get_ports clk2]  
report_power -group clock_network
```

7. Remove the ideal network and compare the power consumption of the clock network.

```
remove_ideal_network [get_ports clk2]  
report_power -group clock_network
```

Question 4. Why is the power consumption on the clock network greater when clk2 is NOT ideal?

.....

Task 2. Annotated Clock Network Power

1. Generate a report to show power consumed by the `clock_network` power group

```
restore_session mac_timing
report_power -group clock_network
```

Question 5. How much is the power reported for `clock_network` group?

2. Replace this calculated value using the following values to be annotated

- a. Internal Power: `1e-5 w`
- b. Switching Power: `2e-5 w`
- c. Leakage power: `3e-5 w`

```
set_annotated_clock_network_power \
    -internal_power 1e-5 \
    -switching_power 2e-5 \
    -leakage_power 3e-5 \
    -clock [get_clocks clk2]
report_power -group clock_network
```

Question 6. How much is the power reported for `clock_network` group? How are the annotated values shown?

3. Dis regard the power consumption due to transitions on the clock pins and regenerate the above power report

```
set power_clock_network_include_register_clock_pin_power
false
report_power -group clock_network
```

Question 7. How has the report changed?

Lab 4

4. Remove the annotated power values and enable the “include clock pin power” variable. You should now be able to generate power report that compares to that in Step-1

```
remove_annotated_clock_network_power  
set power_clock_network_include_register_clock_pin_power  
true  
report_power -group clock_network
```

Question 8. Have you restored the power values as in Step-1?

.....

5. You may exit from the tool

```
pt_shell> exit
```

Task 3. Clock Gate Savings Report

1. You are provided with a set of reports in the “reports” directory. Your task is look into the relevant report to answer questions that follow.

Note: Refer to the lecture material and/or man pages for the hints. Several questions don’t have a single answer while comparing your answer with the Answers/Solutions section.

```
unix% cd ../reports  
unix% ls -al
```

Question 9. What can you comment on the quality of clock gating on this design? What report(s) did you use?

.....

Question 10. What is one register where the value of Q “never” changes? What report did you use?

.....

Question 11. What is one block having no clock gating? What report did you use?

.....

Congratulations. This completes lab 4 : Clock Network Power Analysis

Answers / Solutions

Question 1. Why are registers listed in both groups? When is the register activity considered as part of the clock network power?

The power due to toggles on the Q output are considered to be register power. The register power due to transitions on the clock pin when the Q output does not toggle, are considered to be part of the clock network power.

Question 2. Explain the results. Why are registers no longer part of the clock network power?

When the variable is set to false, the power consumption due to transitions on the clock pin (when the Q does not toggle) is considered to be part of register power.

Question 3. What is the average estimated transition time on the clock tree output?

Clock Tree Output Transition	min	ave	max
<hr/>			
rise	3.39	3.5	3.52
fall	0.994	1.03	1.03

Question 4. Why is the power consumption on the clock network greater when clk2 is NOT ideal?

When the net is ideal, the capacitive load is 0; and the transition time is the value specified. These are less than the current values.

Question 5. How much is the power reported for `clock_network` group?

Power Group	Internal Power	Switching Power	Leakage Power	Total Power	(%)	Attrs
clock_network	8.757e-04	3.072e-04	1.017e-10	1.183e-03	(100.00%)	i
Net Switching Power	= 3.072e-04	(25.97%)				
Cell Internal Power	= 8.757e-04	(74.03%)				
Cell Leakage Power	= 1.017e-10	(0.00%)				
Total Power	= 1.183e-03	(100.00%)				

Answers/ Solutions

Lab 4

Question 6. How much is the power reported for `clock_network` group? How are the annotated values shown?

Power Group	Internal Power	Switching Power	Leakage Power	Total Power	(%)	Attrs
clock_network	8.712e-04	0.0000	0.0000	8.712e-04	(93.56%)	ei
Clock	Internal Power	Switching Power	Leakage Power	Total Power	(%)	Attrs
clk2	1.000e-05	2.000e-05	3.000e-05	6.000e-05		
All Annotated Clocks	1.000e-05	2.000e-05	3.000e-05	6.000e-05	(6.44%)	
Net Switching Power	= 2.000e-05	(2.15%)				
Cell Internal Power	= 8.812e-04	(94.63%)				
Cell Leakage Power	= 3.000e-05	(3.22%)				
Total Power	= 9.312e-04	(100.00%)				

Question 7. How has the report changed?

The `clock_network` power group's Internal power is now zero!

Note: The "i" attribute is now missing.

Power Group	Internal Power	Switching Power	Leakage Power	Total Power	(%)	Attrs
clock_network	0.0000	0.0000	0.0000	0.0000	(0.00%)	e
Clock	Internal Power	Switching Power	Leakage Power	Total Power	(%)	Attrs
clk2	1.000e-05	2.000e-05	3.000e-05	6.000e-05		
All Annotated Clocks	1.000e-05	2.000e-05	3.000e-05	6.000e-05	(100.00%)	
Net Switching Power	= 2.000e-05	(33.33%)				
Cell Internal Power	= 1.000e-05	(16.67%)				
Cell Leakage Power	= 3.000e-05	(50.00%)				
Total Power	= 6.000e-05	(100.00%)				

Question 8. Have you restored the power values as in Step-1?

Yes, the report compares exactly to the one in Step-1

Power Group	Internal Power	Switching Power	Leakage Power	Total Power	(%)	Attrs
clock_network	8.757e-04	3.072e-04	1.017e-10	1.183e-03	(100.00%)	i
Net Switching Power	= 3.072e-04	(25.97%)				
Cell Internal Power	= 8.757e-04	(74.03%)				
Cell Leakage Power	= 1.017e-10	(0.00%)				
Total Power	= 1.183e-03	(100.00%)				

Lab 4

Answers / Solutions

Question 9. What can you comment on the quality of clock gating on this design? What report(s) did you use?

It looks good ☺

From the "report_clock_gate_savings" report:

Clock: clock

- + Clock Toggle Rate: 0.04
- + Number of Registers: 2553
- + Number of Clock Gates: 43
- + Average Clock Toggle Rate at Registers: 0.0144628
- + Average Toggle Savings at Registers: 63.8%

From "report_clock_gate_savings.bycg" report:

The toggle savings range anywhere between 58.2% up to 98.2%

Question 10. What is one register where the value of Q "never" changes? What report did you use?

Using "report_clock_gate_savings.seq.bycg":

> memory_1_hier/memory_1 0 0.0%

Question 11. What is one block having no clock gating? What report did you use?

Using

"report_clock_gate_savings.hier.bycg":

Blocks with no clock-gating are reported "NA"

memory_0_hier	0	NA
feed_between	0	NA
memory_1_hier	0	NA
DFT_system_controller_U_compressor_ScanCompression_mode	0	NA
DFT_system_controller_U_decompressor_ScanCompression_mode	0	NA
column_processor	0	NA
wrapper0	0	NA
wrapper5	0	NA
wrapper6	0	NA
wrapper3	0	NA
wrapper4	0	NA
wrapper1	0	NA
wrapper2	0	NA
wrapper7	0	NA
status_ctrl	0	NA

5

Multi Voltage Power Analysis

Learning Objectives

After completing this lab, you should be able to:

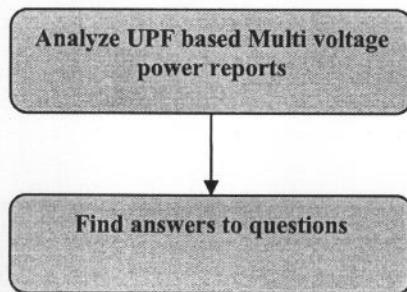
- Answer questions related to UPF based multi voltage power analysis in PTPX. The questions will test your understand for:
 - Power domain and Supply net based power reporting
 - Method used to link the design
 - Library scaling
 - Power saving due to turning off a switch



Lab Duration:
20 minutes

Lab 5

Overview



Getting Started

If you need help...

```
pt_shell> help *voltage*
pt_shell> set_voltage -help
pt_shell> printvar link*instance
pt_shell> man define_scaling_lib_group
```

When working interactively, you do not need to type the entire command or option name. Use the <Tab> key to expand.

```
pt_shell> echo $link_path_per_instance
pt_shell> load_upf <file>
pt_shell> set_current_power_domain <PD>
pt_shell> get_sw [all_inp]
```

Answers & Solutions

This lab guide contains answers and solutions to all questions.

You are encouraged to verify your results by checking the [Answers/Solutions](#) section at the end of this lab.

Objective:

In this lab, you will answer questions related to UPF based multi voltage power analysis. You'll be provided with the necessary analysis data.

Lab 5

File Locations

Directory Structure

Reference Directory: ./ref

./ref/lib	: Contains library files used in the labs
./ref/design_data	: Contains netlist and SDF files
./ref/scripts	: Contains PrimeTime scripts
./ref/tcl_procs	: Contains Tcl procedures

Working Directories:

./lab1_intro
./lab2_average
./lab3_peak
./lab4_clock_tree
./lab5_multi_voltage

Relevant Files for Lab5:

./lab5_multi_voltage/ptpx/play_gates.upf
./lab5_multi_voltage/ptpx/run_ptpx.tcl
./lab5_multi_voltage/ptpx/play_gates.link
./lab5_multi_voltage/ptpx/run.log
./lab5_multi_voltage/ptpx/pt_shell_command.log
./lab5_multi_voltage/ptpx/reports/*

Working Directory for Lab5:

./lab5_multi_voltage

Instructions

Task 1. Answer Questions on UPF Based Multi Voltage Power Analysis

After performing a multi voltage power analysis on a sample design, our power guru has generated all the needed files for you. Your task is to find information out of these files to answer the following questions.

Note: Assume that the library nominal voltages are 1.08v and 0.864v for the multi voltage libraries used in the design

1. The working directory for this lab is lab5_multi_voltage/pt_px

```
unix% cd lab5_multi_voltage/pt_px
```

Question 1. Is this analysis UPF based?

.....

Question 2. How many power domains are contained in the design? How did you determine?

.....

Question 3. What are some indications that this is a multi voltage design?

.....

Question 4. Which method of linking is used to link the design? i.e.,
a) link_path_per_instance or
b) define_scaling_lib_group?

.....

Question 5. How much is average power consumed by this design during normal operation using default annotation?

.....

Question 6. How much power is drawn from the supply net VDD1_ADD of power domain PD_ADD? How was this report generated?

.....

Lab 5

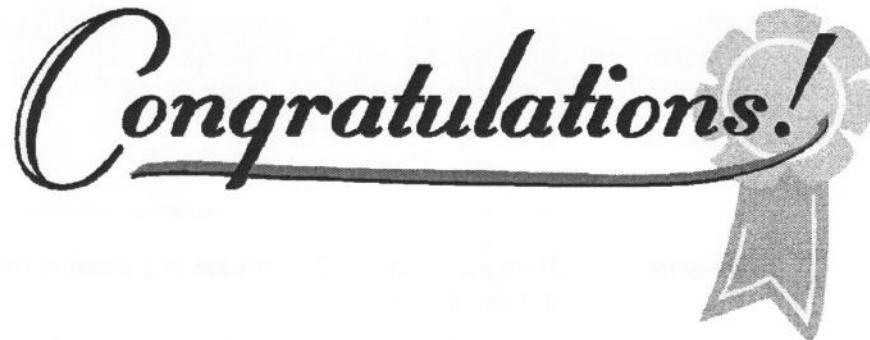
Question 7. How much is the power saving when the switch is turned on vs. off?

.....

Question 8. Is voltage scaling performed during analysis?

.....

This lab concludes the PrimeTime-PX workshop!



Answers / Solutions

Question 1. Is this analysis UPF based?

The run_ptpx.tcl script contains

```
load_upf play_gates.upf
```

Also, per the run.log file, the UPF file has been loaded successfully

```
load_upf play_gates.upf  
1
```

Question 2. How many power domains are contained in the design?
How did you determine?

From the UPF file used, we see there are 2 power domains specified

```
create_power_domain PD_TOP .....  
create_power_domain PD_ADD .....
```

Question 3. What are some indications that this is a multi voltage design?

- In the UPF file, there are 2 level shifting strategies specified (ls_in and ls_out). Level shifters are not needed in a single voltage design.
- From the run script, we see the set_voltage commands that are required to analyze a multi voltage design.

Question 4. Which method of linking is used to link the design? i.e.,
a) link_path_per_instance or
b) define_scaling_lib_group?

The run script sources the file containing the link path per instance setting:

```
source -e -v play_gates.link
```

Question 5. How much is average power consumed by this design during normal operation using default annotation?

From the report file: reports/full_power.rpt

Total Power = 6.623e-05 (100.00%)

The run.log file indicates how this file was generated.

Question 6. How much power is drawn from the supply net VDD1_ADD of power domain PD_ADD? How was this report generated?

From the report file: reports/ADD_power.rpt

Total Power = 2.043e-05 (100.00%)

The report was generated by specifying the power domain and supply net prior to the analysis. From the run script:

```
set_current_power_net VDD1_ADD  
set_current_power_domain PD_ADD  
update_power
```

This is confirmed by the above report file:

```
Current Power Domains : PD_ADD  
Current Power Nets : VDD1_ADD
```

Question 7. How much is the power saving when the switch is turned on vs. off?

From the report files: ON_power.rpt and OFF_power.rpt

Power savings = 2.672e-05 (100.00%)

Question 8. Is voltage scaling performed during analysis?

No voltage scaling is performed (not needed). The run script contains the following set_voltage commands:

```
set_voltage 1.08 -object_list VDD  
set_voltage 0.864 -object_list VDD1  
set_voltage 0.864 -object_list VDD1_ADD  
set_voltage 0.00 -object_list GND
```

Since the library nominal voltages are the same as that of set_voltage specifications (See Note at the beginning of this task), there is no voltage scaling needed. (An example requiring voltage scaling would be when IR drop effects are reflected in the set_voltage commands.)

Also, for the tool to perform voltage scaling, library grouping must be defined using the define_scaling_lib_group command