SYNOPSYS®

CUSTOMER EDUCATION SERVICES

CustomSim Workshop

Student Guide

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 © 2007 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.

Right to Copy Documentation

The license agreement with Synopsys permits licensee to make copies of the documentation for its internal use only. Each copy shall include all copyrights, trademarks, service marks, and proprietary rights notices, if any. Licensee must assign sequential numbers to all copies. These copies shall contain the following legend on the cover page:
"This document is duplicated with the permission of Synopsys, Inc., for the exclusive use of _____ and its employees. This is copy number _____."

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, Cadabra, CATS, CRITIC, CSim, Design Compiler, DesignPower, DesignWare, EPIC, Formality, HSIM, HSPICE, iN-Phase, in-Sync, Leda, MAST, ModelTools, NanoSim, OpenVera, PathMill, Photolynx, Physical Compiler, PrimeTime, SiVL, SNUG, SolvNet, System Compiler, TetraMAX, VCS, Vera, and YIELDirector are registered trademarks of Synopsys, Inc.

Trademarks (™)

AFGen, Apollo, Astro, Astro-Rail, Astro-Xtalk, Aurora, AvanWaves, Columbia,Columbia-CE, Cosmos, CosmosEnterprise, CosmosLE, CosmosScope, CosmosSE, DC Expert, DC Professional, DC Ultra, Design Analyzer, Design Vision, DesignerHDL, Direct Silicon Access, Discovery, Encore, Galaxy, HANEX, HDL Compiler, Hercules, Hierarchical Optimization Technology, HSIMplus, HSPICE-Link, iN-Tandem, i-Virtual Stepper, Jupiter, Jupiter-DP, JupiterXT, JupiterXT-ASIC, Liberty, Libra-Passport,Library Compiler, Magellan, Mars, Mars-Rail, Milkyway, ModelSource, Module Compiler, Planet, Planet-PL, Polaris, Power Compiler, Raphael, Raphael-NES,Saturn, Scirocco, Scirocco-i, Star-RCXT, Star-SimXT, Taurus, TSUPREM-4, VCS Express, VCSi, VHDL Compiler, VirSim, and VMC are trademarks of Synopsys, Inc.

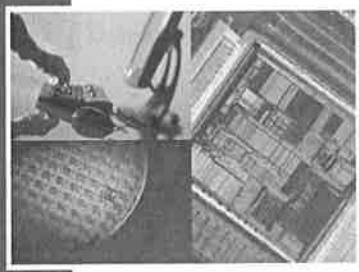
Service Marks (℠)

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: 60-I-031-BSG-008
HSPICE Essentials Student Guide

CustomSim Training



SYNOPSYS®

Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

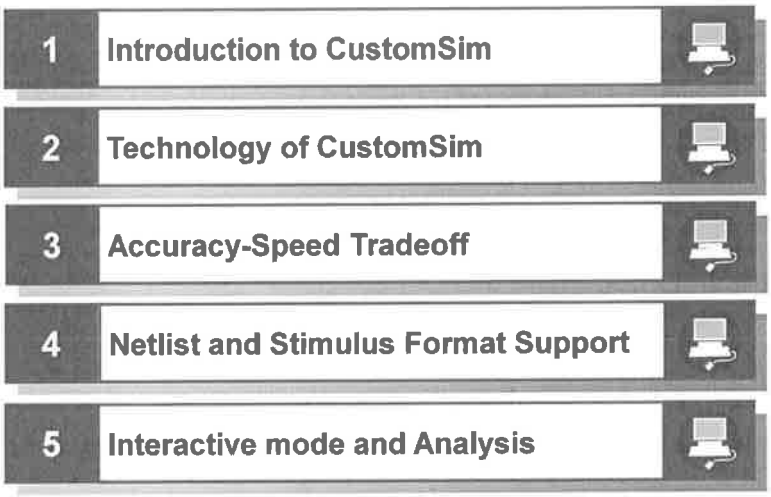
Workshop Objectives

- Understand what is Synopsys Circuit Simulation Solution
- Understand what is the technology and advantage of CustomSim
- Understand the core engines in CustomSim and use them to simulate circuit
- Understand post-layout simulation and co-simulation with CustomSim

© 2014 Synopsys, Inc. (2)

SYNOPSYS®
Predictable Success

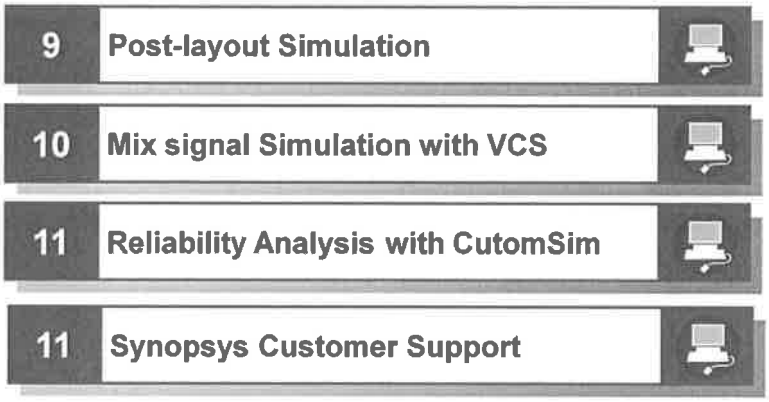
Agenda (1)



© 2014 Synopsys, Inc. (3)

SYNOPSYS®
Predictable Success

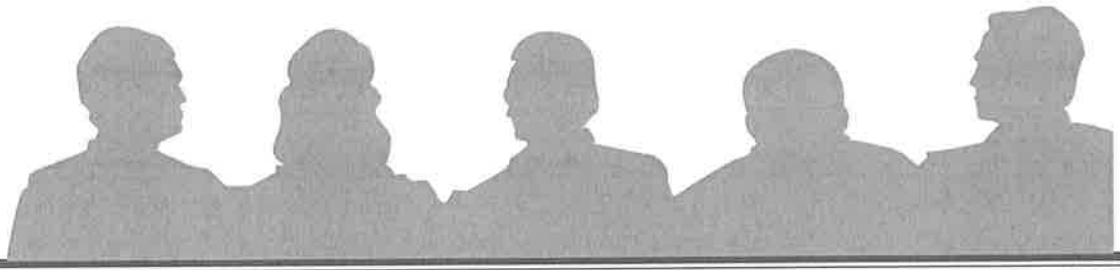
Agenda (2)



© 2014 Synopsys, Inc. (4)

SYNOPSYS®
Predictable Success

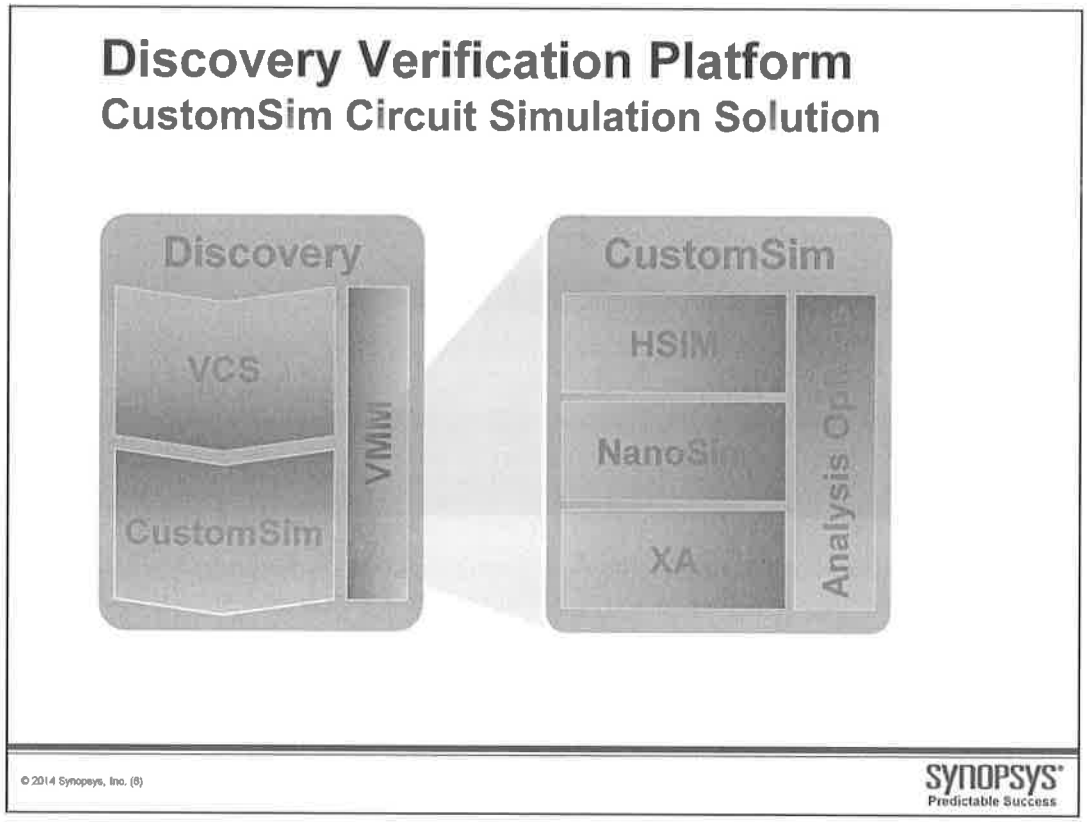
Introduction to CustomSim



© 2014 Synopsys, Inc. (5)

SYNOPSYS®
Predictable Success

All courses must have a target audience that is well defined. This sets the stage for the entire workshop. Please use the graphic provided in the slide.

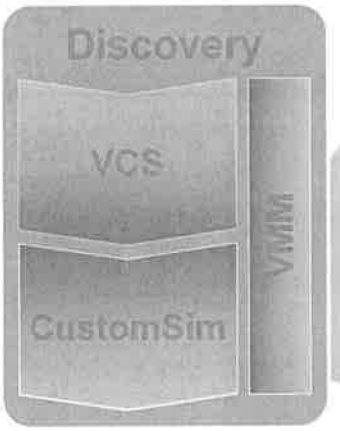


Highlight that we have a single product for them to purchase

© 2014 Synopsys, Inc. (CustomSim)

Discovery Verification Platform

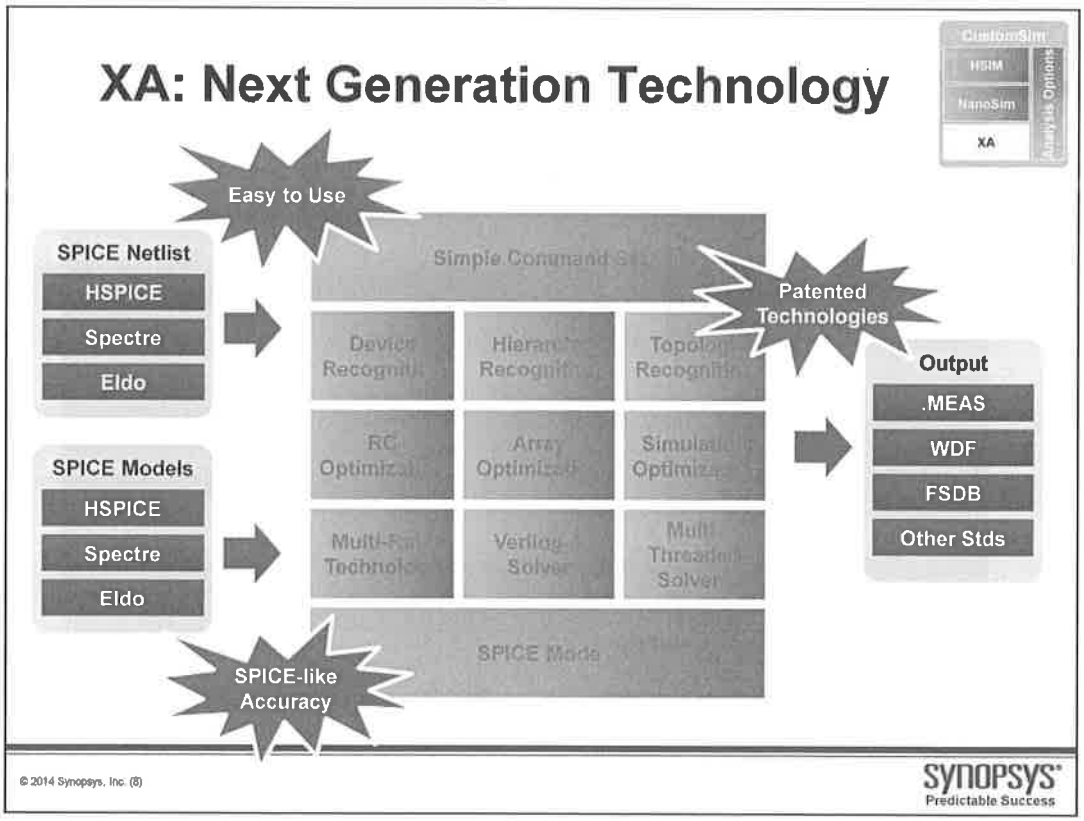
CustomSim Circuit Simulation Solution



- Unified Best-in-Class Engines with Multi-Core Technology
- Full-Chip Mixed-Signal Verification with CustomSim-VCS
- Save Time & Money with Native Circuit Checks

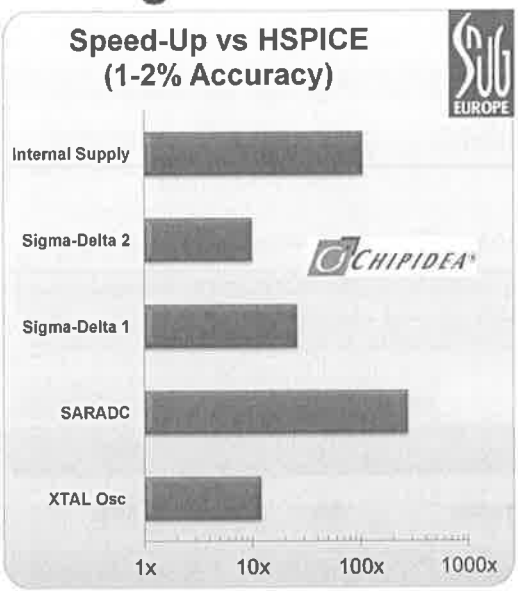
© 2014 Synopsys, Inc. (7)

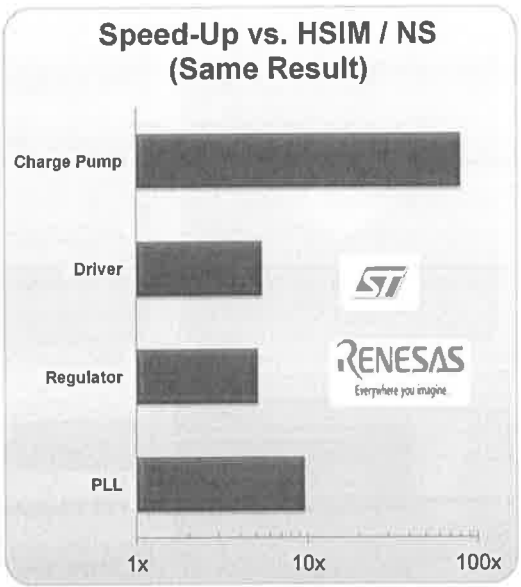
SYNOPSYS®
Predictable Success



© 2014 Synopsys, Inc. (CustomSim)

User Experience with XA Core Engine



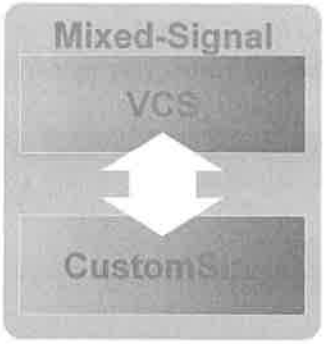


© 2014 Synopsys, Inc. (9)

SYNOPSYS®
Predictable Success

Advanced Analysis Options

Mixed-Signal Simulation



- Single Process, Single Executable
- Unified use model for all engines
  - Unified flow, inputs & outputs
    - One user manual
- C-Models, SystemVerilog, Verilog-AMS, Verilog-A, Verilog, VHDL, SPICE, DSPF, SDF

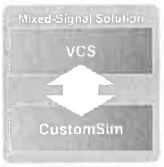
	XA-VCS	NS/HS-VCS
4-Chan SerDes	2h	24h
128 Channel Tx/Rx	5h	61h
2.5GHz Transponder	10 days	Not Possible

© 2014 Synopsys, Inc. (10)

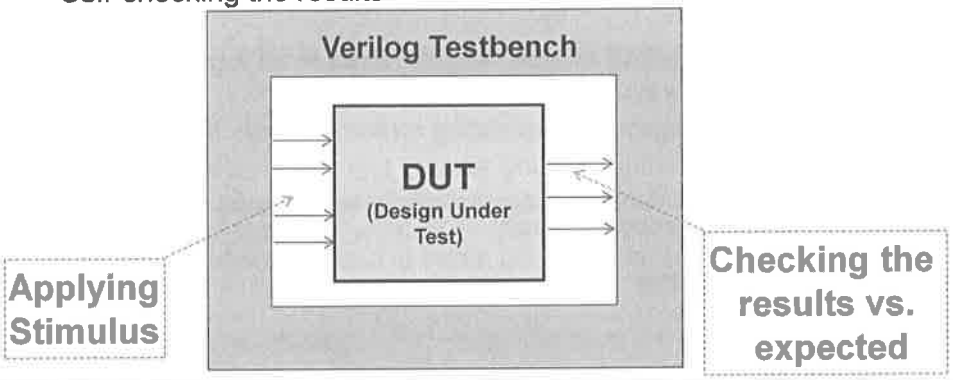
SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Mixed-Signal Verification
SystemVerilog Testbenches



- A Verilog/SystemVerilog testbench provides powerful verification capabilities:
  - Creating sophisticated test stimuli for the design
  - Self-checking the results



© 2014 Synopsys, Inc. (11)

SYNOPSYS
Predictable Success

Analog Assertions
Extending Assertions to Analog

- The objective is to extend digital verification techniques to Analog/Mixed-Signal designs for self-checking of analog output signals
- CustomSim-VCS provides two methods to support Analog Assertions:
  1. System-Verilog treating analog signals as logic values
     1. A2D and D2A conversions automatically
     2. No change in System-Verilog code is required
  2. System-Verilog treating analog signals as real values
     1. Specific Verilog system tasks/functions are used to access Analog voltages and currents
     2. New features are being added to further enhance Real Access capabilities

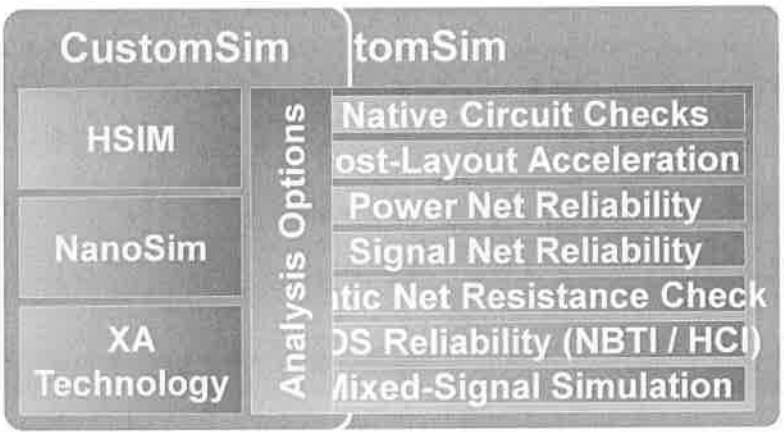
© 2014 Synopsys, Inc. (12)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

CustomSim Circuit Simulation Solution

Advanced Analysis Options



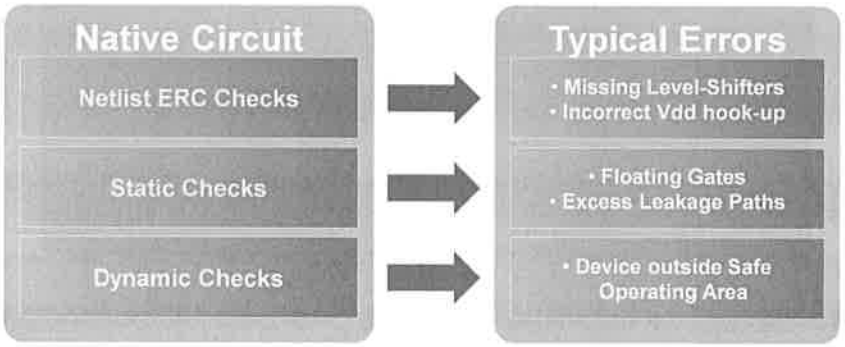
© 2014 Synopsys, Inc. (13)

SYNOPSYS® Predictable Success

Advanced Analysis Options

Native Circuit Checks

- Comprehensive Set of Static & Dynamic Checks
- Increase Confidence Finding Errors Before Tape-Out
- Increase Productivity Avoiding Wasted Simulation Time



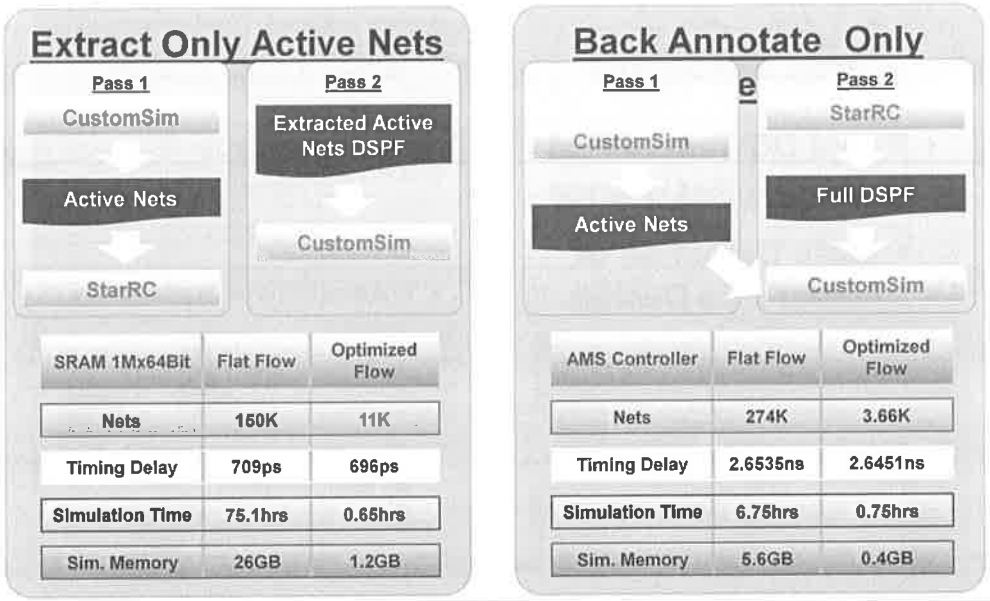
© 2014 Synopsys, Inc. (14)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Post-Layout Back-Annotation Flow

Tight Integration to StarRC



SRAM 1Mx64Bit	Flat Flow	Optimized Flow
Nets	150K	11K
Timing Delay	709ps	696ps
Simulation Time	75.1hrs	0.65hrs
Sim. Memory	26GB	1.2GB

AMS Controller	Flat Flow	Optimized Flow
Nets	274K	3.66K
Timing Delay	2.6535ns	2.6451ns
Simulation Time	6.75hrs	0.75hrs
Sim. Memory	5.6GB	0.4GB

© 2014 Synopsys, Inc. (15)

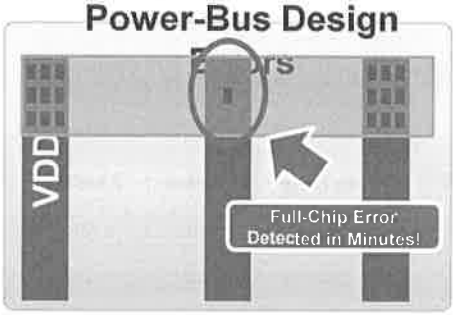
SYNOPSYS® Predictable Success

Advanced Analysis Options

Static Net Resistance Check

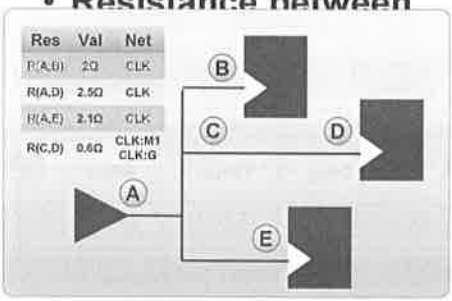
Power Bus Check

- Rapid DC Test of Pad to Device Resistance
- Early Detection of Power-Bus Design



Signal Net Check

- Resistance from Drivers to all Receivers
- Resistance between



© 2014 Synopsys, Inc. (16)

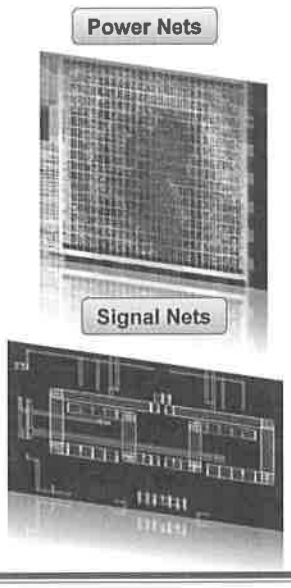
SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Advanced Analysis Options

Power & Signal Net Reliability Analysis

- Golden Solution for Transistor-Level EM/IR Drop Analysis
  - Nanometer Technology:
  - Length-Dependant EM Rules
- Fully Customizable EM Rules
- Optimized Flow with STAR-RCXT
- Visualization & "What-If" Analysis provided with Custom Designer-LE



© 2014 Synopsys, Inc. (17)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Advanced Analysis Options

MOS Reliability Analysis

- Analyze Aging Effects (NBTI & HCI)
- Common Usage Model with HSPICE
- Utilizes Built-in or Customer's Model

© 2014 Synopsys, Inc. (18)

SYNOPSYS®
Predictable Success

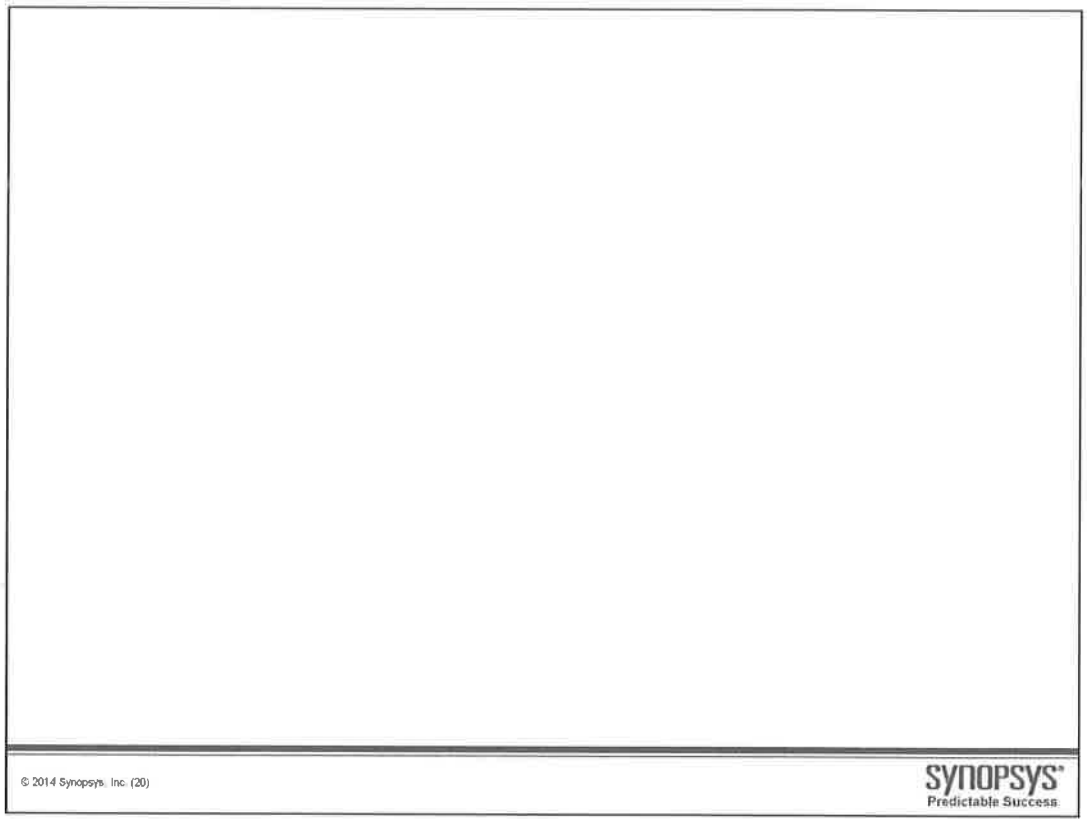
© 2014 Synopsys, Inc. (CustomSim)

CustomSim Overview Summary

- CustomSim Combines HSIM, NanoSim & XA into a Single Unified Solution
- Customers successfully deploying CustomSim and CustomSim-VCS to meet AMS verification needs
- Continue to Improve HSIM & NanoSim. Consolidated solution based on XA Engine (Roadmap in place)

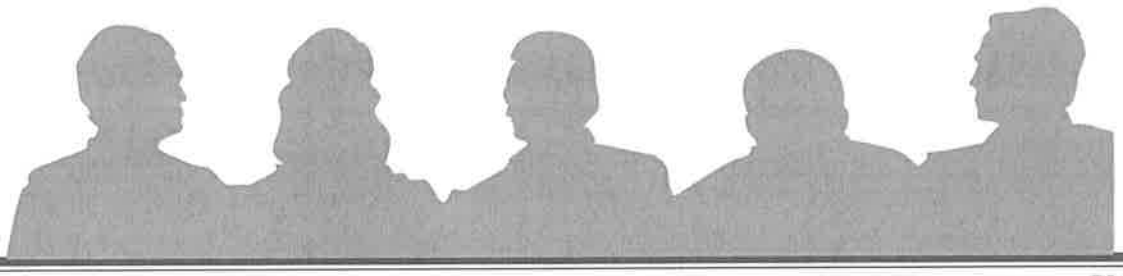
© 2014 Synopsys, Inc. (19)

SYNOPSYS®
Predictable Success



© 2014 Synopsys, Inc. (CustomSim)

Technology of CustomSim



© 2014 Synopsys, Inc. (21)

SYNOPSYS®
Predictable Success

All courses must have a target audience that is well defined. This sets the stage for the entire workshop. Please use the graphic provided in the slide.

Unit Objectives

- After completing this unit, you should know
  - The basic technologies used in CustomSim and its differences from SPICE
  - The challenges faced in CustomSim

© 2014 Synopsys, Inc. (22)

SYNOPSYS®
Predictable Success

What technologies?

- Fast-SPICE simulators are transistor-level circuit simulators that achieve faster simulation than SPICE.
- Fast-SPICE is generally used for functional verification on large designs that aren't able to simulate in SPICE.
- Fast SPICE simulators take "short-cuts" based on the knowledge of present technologies to improve the performance of the simulation.
- Common technologies used in CustomSim
  - Circuit Partitioning
  - Table Look-up Model
  - Dynamic Time Step Control

© 2014 Synopsys, Inc. (23)

SYNOPSYS®
Predictable Success

Circuit Partitioning

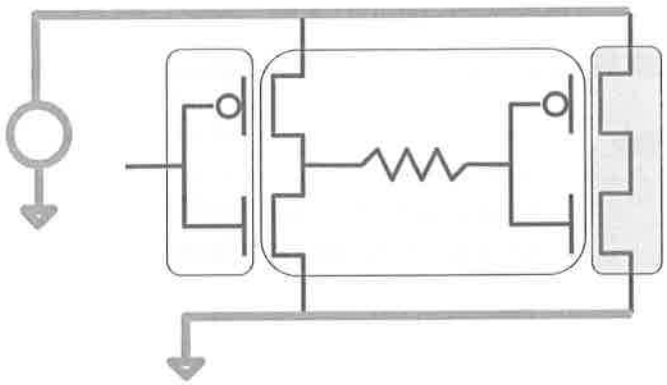
- Speed advantage over SPICE mainly comes from circuit partitioning.
- Cut a single large system into smaller "independent" groups, therefore many smaller matrices to be solved.
  - Common partitioning technology is cutting large system into many channel-connected blocks.
  - Every fast-SPICE simulator has different proprietary technologies to handle analog devices, non-ideal power supply, internal power supply, memory array, coupling capacitor, etc.

© 2014 Synopsys, Inc. (24)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

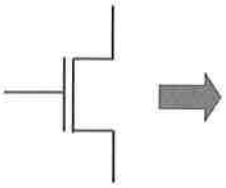
Channel-connected Blocks



© 2014 Synopsys, Inc. (25)

SYNOPSYS®
Predictable Success

Table Lookup Model



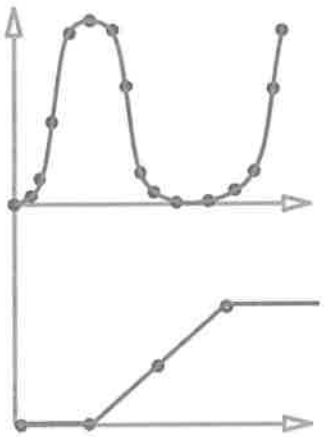
- Table lookup model is used to replace analytical model to speed up simulation.
  - It describes the characteristics of the specified MOS model based on the device size.
  - Each fast-SPICE tool has different proprietary technology to create different dimension of table model for each MOS model.

© 2014 Synopsys, Inc. (26)

SYNOPSYS®
Predictable Success

Dynamic Time Step Control

- Time step decreases during voltage and current transition, and increases during periods of lesser circuit activity.
- Individual nodes or groups of nodes have their own time step control.
- Provides an order of magnitude speedup over SPICE.



© 2014 Synopsys, Inc. (27)

SYNOPSYS®
Predictable Success

Challenges in CustomSim

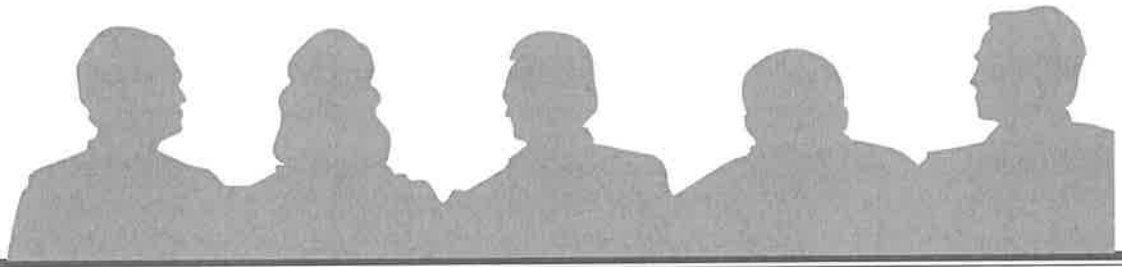
- Multiple dimension of ease-of-use commands
- Do not simulate well across different type of circuits
- DC and transient do not always converge
- Different settings for different design type and technology
- High expectation and requirements from designers
- Designers want SPICE-like accuracy at CustomSim speed

© 2014 Synopsys, Inc. (28)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to use CustomSim



© 2014 Synopsys, Inc. (29)

SYNOPSYS®
Predictable Success

All courses must have a target audience that is well defined. This sets the stage for the entire workshop. Please use the graphic provided in the slide.

Unit Objectives

- After completing this unit, you should know
  - What CustomSim is and its benefits
  - The flow of CustomSim
  - How to setup and invoke CustomSim
  - How to specify command or netlist option

© 2014 Synopsys, Inc. (30)

SYNOPSYS®
Predictable Success

What is CustomSim?

- Next generation high-speed, high-capacity, and easy-to-use transistor-level circuit simulator.
- It provides a scalable accuracy vs. speed tradeoff.
- Time to results (setup/configuration plus simulation time) is typically much better than other Fast-SPICE simulators
- Covers the application spectrum from verification to SPICE acceleration.
- Combination of the best fast-SPICE technologies from NanoSim, HSIM, and Star-SimXT.

© 2014 Synopsys, Inc. (31)

SYNOPSYS® Predictable Success

Benefits of CustomSim

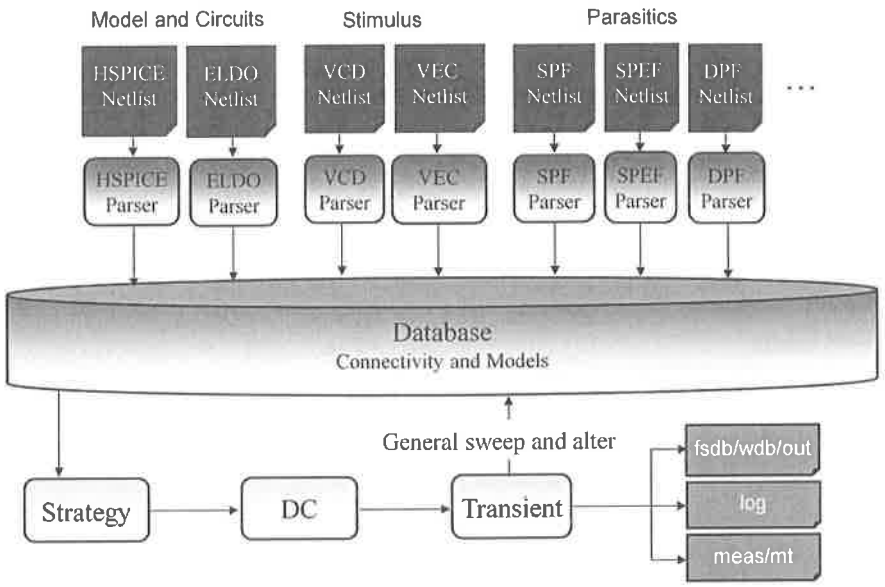
- Performance
  - New and improved partitioning and table model technology to achieve better speed-accuracy tradeoff.
- Accuracy
  - Full support of SPICE algorithm (Newton-Raphson, LTE) to achieve SPICE-like accuracy.
- Capacity
  - New and improved hierarchical data structure to achieve high capacity.
- Ease-of-use
  - New and improved auto-detection technology for identifying circuit topologies to appropriately control error tolerances and modeling complexity.
  - One dimension sliding scale ease-of-use command.

© 2014 Synopsys, Inc. (32)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

CustomSim Flow



© 2014 Synopsys, Inc. (33)

SYNOPSYS®
Predictable Success

Setting-up

- 3 environment variables
  - SNPSLMD_LICENSE_FILE or LM_LICENSE_FILE
    - Pointer to license path.
    - SNPSLMD_LICENSE_FILE takes precedence over LM_LICENSE_FILE.
    - Make sure that SNPSLMD_LICENSE_FILE is not set if LM_LICENSE_FILE is used.
  - XA_GCC
    - Pointer to QSC-standard gcc compiler.
  - PATH
    - Pointer to binary for launching.
- 64-bit binary
    - setenv SNPS_64 1 or export SNPS_64=1
- 4 Platforms
  - SUN 32/64
  - LINUX 32
  - AMD 64

© 2014 Synopsys, Inc. (34)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Executing CustomSim

xa [[-hspice|-eldo] infile] [-c scriptfile] [-out [outpath/]outfile]
[-wavefmt fmt] [-version] [-tcl] [-help] [-I[dir]]

Following options are available:

-hspice Infile	Specify the name of the Input netlist in Hspice format
-eldo Infile	Specify the name of the input netlist in Eldo format
-c scriptfile	Specify the name of the command script file
-out outfile	Set output file to outfile.log
-o	Same as -out
-wavefmt fmt	Specify the format of output waveform file
-I[dir]	Search the directory for include files
-tcl	Enable tcl programming for command script file
-version	Print the version information
-v	Same as -version
-help	Print usage information
-h	Same as -help

© 2014 Synopsys, Inc. (35)

SYNOPSYS®
Predictable Success

Command and Option Syntax

- XA-specific commands can be specified in 2 ways
  - In the command script file
    - Use –c command-line flag to include the command script file

```
set_sim_level –level 7
```

  - Embedded in the netlist using ".options" statement
    - HSPICE and Eldo

```
.opt xa_cmd="set_sim_level – level 7"
```

- Commands specified in the command script file overrides the commands embedded in the netlist.

© 2014 Synopsys, Inc. (38)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Output Files

- XA generates the following output files

file.log = Contain the log of simulation.

file.meas# = Contain result of .MEASURE statements.

file.out# = Output waveform file if output format OUT is specified. Displayed by nWave.

file.fsdb# = Output waveform file if output format FSDB is specified. Displayed by nWave and CosmosScope.

file.wdb# = Output waveform file id output format WDB is specifed. Displayed by CosmosScope.

file.rcxt# = Contain active net information if set_ba_active_file command is specified.

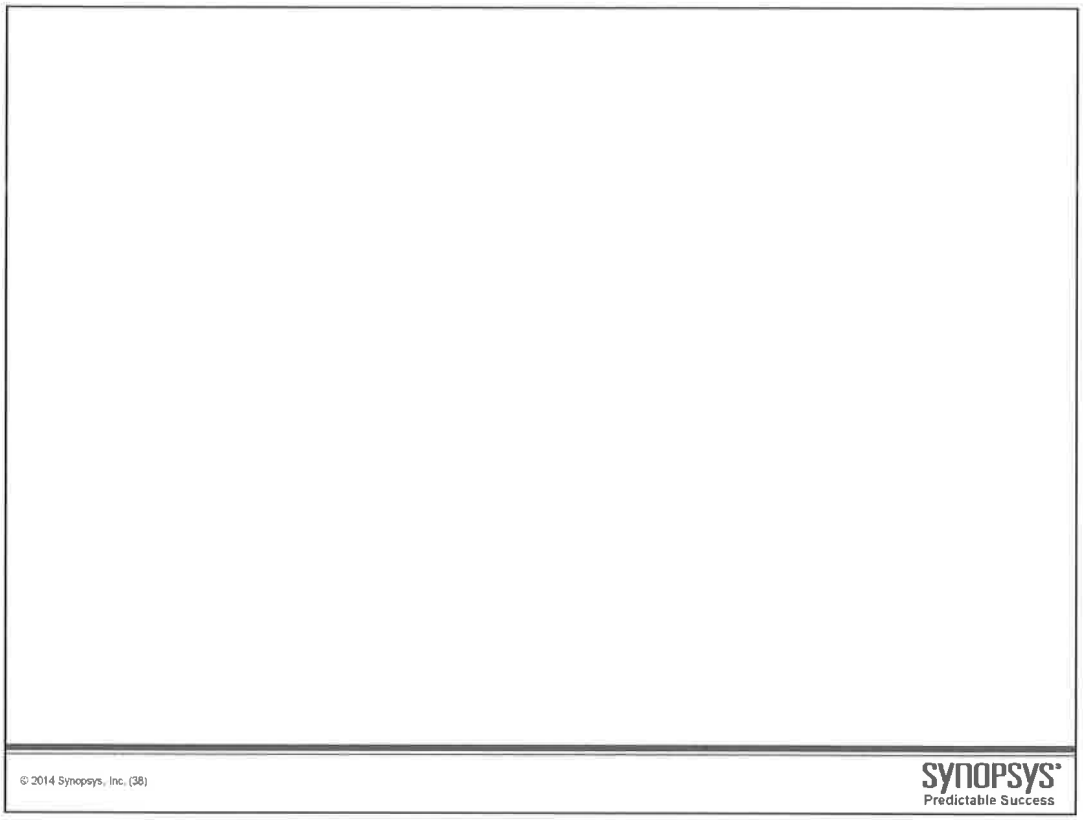
file.power# = Contain the sub-circuit power consumption report if report_power is specified.

file.*time*.ic# = Contains .op output, initial condition

= Positive Integer starting from 0. This integer is incremented for each run of a multi-run simulation (temp or alter sweep).

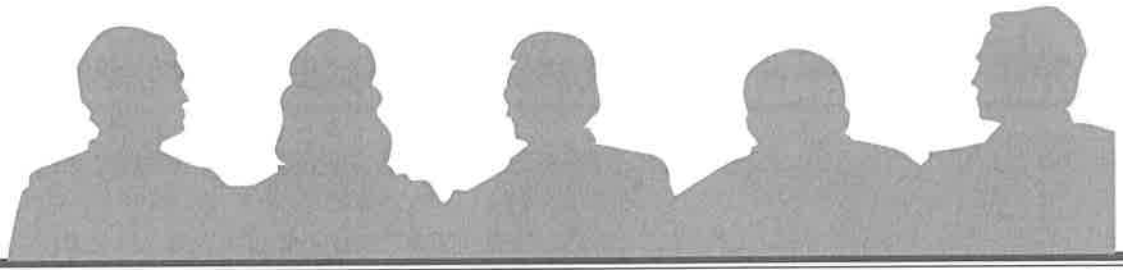
© 2014 Synopsys, Inc (37)

SYNOPSYS® Predictable Success



© 2014 Synopsys, Inc. (CustomSim)

Accuracy-Speed Tradeoff



© 2014 Synopsys, Inc. (39)

SYNOPSYS®
Predictable Success

All courses must have a target audience that is well defined. This sets the stage for the entire workshop. Please use the graphic provided in the slide.

Unit Objectives

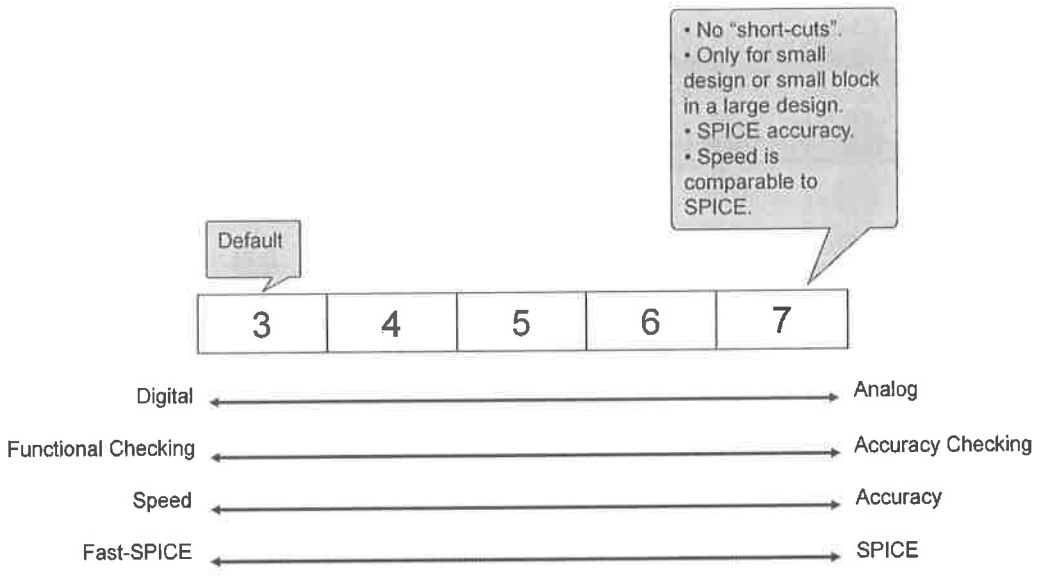
- After completing this unit, you should know
  - How to control the speed-accuracy tradeoff
  - Which setting is appropriate for a given application

© 2014 Synopsys, Inc. (40)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

1-dimension sliding scale command



© 2014 Synopsys, Inc. (41)

SYNOPSYS®
Predictable Success

Applications

Level	Digital	Low-sensitivity Analog	High-sensitivity Analog	Memory	Mixed-signal	Full-chip
3	✓	✓		✓	✓	✓
4	✓	✓	✓	✓	✓	✓
5	✓	✓	✓	✓	✓	✓
6	✓	✓	✓	✓	✓	✓
7	✓	✓	✓		✓	

© 2014 Synopsys, Inc. (42)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Sliding scale command

- Command script file syntax

  set_sim_level [-level] *accuracy_level* \

  [-inst *inst_name* {*inst_name*}] [-subckt *subckt_name* {*subckt_name*}]

- Netlist option syntax
  - HSPICE and Eldo

    .option XA_CMD="set_sim_level [-level] *accuracy_level* [-inst *inst_name* {*inst_name*}] [-subckt *subckt_name* {*subckt_name*}]"

© 2014 Synopsys, Inc. (43)

SYNOPSYS®
Predictable Success

How to set global & local command?

set_sim_level –level 5

Globally set level=5 to the whole design

set_sim_level –level 5 –subckt pll

Locally set level=5 to pll sub-circuit

set_sim_level –level 5 –inst x1 –subckt pll

Locally set level=5 to instance x1 in pll sub-circuit

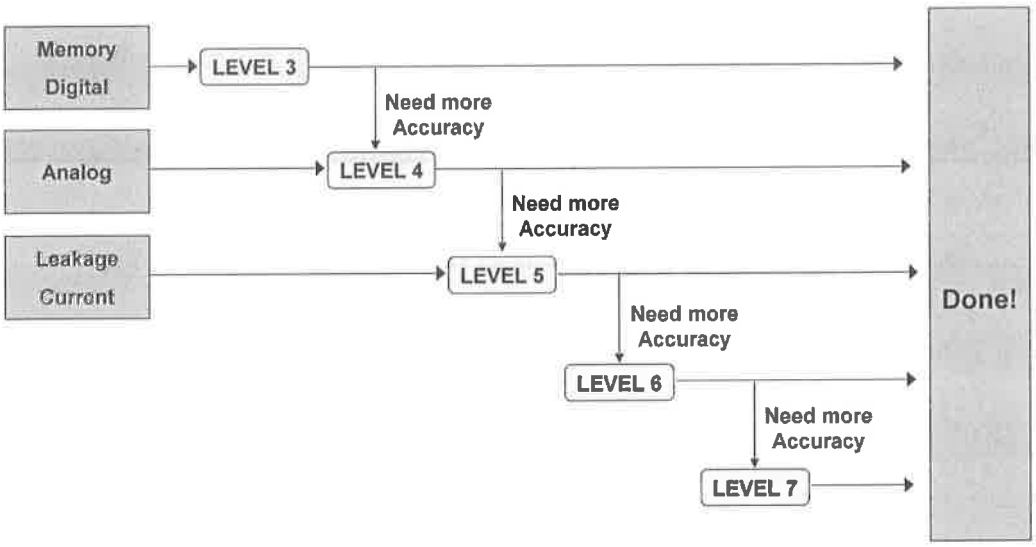
set_sim_level –level 5 –inst x1 x2 –subckt pll chgpump

Locally set level=5 to instance x1 and x2 in pll and chgpump sub-circuits

© 2014 Synopsys, Inc. (44)

SYNOPSYS® Predictable Success

General Strategy For set_sim_level



© 2014 Synopsys, Inc. (45)

SYNOPSYS®
Predictable Success

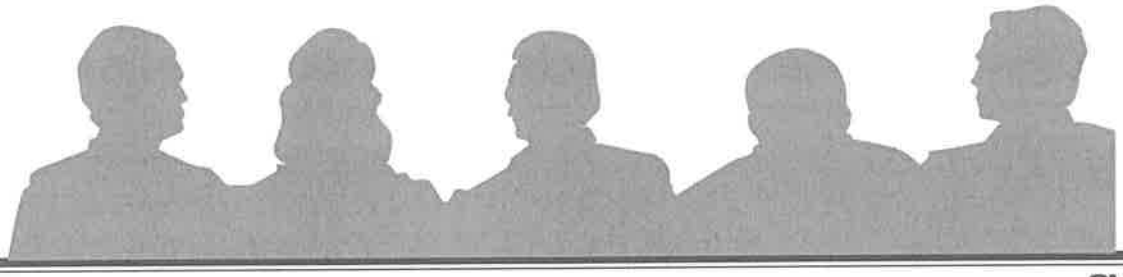
Keep In Mind

- Accuracy and Performance will not scale on every test case
- Analog often requires level 4 or higher
- Use caution when determining "golden" reference
- Pay attention to the Warning messages in the log file!
  - Floating gates
  - Non rail bulk
  - These potential netlist errors can have a much larger impact on performance in XA than in other Fast SPICE simulators, especially at levels 5, 6 and 7.

© 2014 Synopsys, Inc. (48)

SYNOPSYS® Predictable Success

Netlist and Stimulus Format Support



© 2014 Synopsys, Inc. (47)

SYNOPSYS®
Predictable Success

All courses must have a target audience that is well defined. This sets the stage for the entire workshop. Please use the graphic provided in the slide.

Unit Objectives

- After completing this unit, you should know
  - Different netlist formats supported in CustomSim
  - The incompatibility of different netlist format from the native-SPICE
  - Use VEC file as Stimulus of design

© 2014 Synopsys, Inc. (48)

SYNOPSYS®
Predictable Success

Netlist Format

- CustomSim supports the following netlist formats:
  - HSPICE
  - Eldo
  - Spectre

© 2014 Synopsys, Inc. (49)

SYNOPSYS®
Predictable Success

HSPICE Netlist Format

- HSPICE syntax in XA is predominantly identical to the newest version of HSPICE, including options and statements.
- Enhancements are implemented to avoid certain limitations.
- Unsupported syntax (options and statements) is printed in the log file.

© 2014 Synopsys, Inc. (50)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Syntax and Behavior Variations

- Curly and square bracket

  HSPICE converts {} to []; XA treats them separately.

  .param a[1]=1 a{1}=2 ➡ HSPICE treats it as .param a[1] a[1]=2

- Node names which begins with numbers

  HSPICE truncates node names to only digits XA keeps the node names as it is.

  r1 24a 0 10
  r2 24b 0 20 ➡ HSPICE truncates the nodes name from "24a" to "24", and "24b" to "b". Therefore, those 2 resistors are connected in parallel.

© 2014 Synopsys, Inc. (51)

SYNOPSYS® Predictable Success

Option Variations

- POST option

  Supported waveform formats in XA are FSDB (default), WDB, and OUT.

  .option POST=fsdb|wdb|out

- PROBE option

  HSPICE defaults the value of PROBE to 0 if it is not specified; XA defaults it to 1. To probe all waveforms in XA, you need to set PROBE=0 (This is not recommended because it prints all top-level waveforms and it slows down the simulation).

© 2014 Synopsys, Inc. (52)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Analysis Variation

- .PRINT statement

  HSPICE outputs signals specified in the .PRINT statement to an ASCII output; XA treats .PRINT the same as the .PROBE by default. Set the following XA command to generate ASCII output as HSPICE:

  enable_print_statement 1

© 2014 Synopsys, Inc. (53)

SYNOPSYS®
Predictable Success

Supported HSPICE Elements

Element	Instantiation Character
Capacitor	C
Diode	D
VCVS	E
CCCS	F
VCCS	G
CCVS	H
Current Source	I
Mutual Inductor	K
Linear Inductor	L
MOSFET	M
BJT	Q

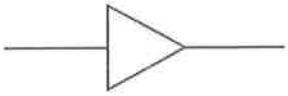
Element	Instantiation Character
Resistor	R
Transmission Line	T
Voltage Source	V
Sub-circuit Instance	X
W Element – RLGC Model	W

© 2014 Synopsys, Inc. (54)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Supported HSPICE Diode model

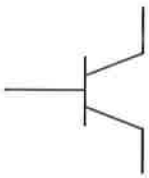


Diode	Level
Junction Diode	1
Fowler-Nordheim Diode	2
Geometric Junction Diode Processing	3
Philips JUNCAP Diode	4
JUNCAP2 Diode	6

© 2014 Synopsys, Inc. (55)

SYNOPSYS®
Predictable Success

Supported HSPICE BJT Model



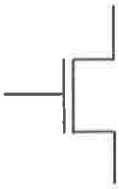
BJT	Level
Gummel-Poon Model	1
Quasi-saturation Model	2
Philips Bipolar Model	6
HiCUM Model	8
HiCUM0 Model	13

© 2014 Synopsys, Inc. (56)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Supported HSPICE MOSFET Model



MOSFET	Level
BSIM Model	13
HSPICE-enhanced UC Berkeley BSIM3v3 Model	49
Philips MOS9 Model	50
UC Berkeley BSIM3v3 Model	53
UC Berkeley BSIM4 Model	54
Philips MOS11 Model	63
PSP Model	69

© 2014 Synopsys, Inc. (57)

SYNOPSYS®
Predictable Success

Supported Analysis Statements

- The following analysis statements from HSPICE are supported in XA
  - .PROBE
  - .PRINT
  - .MEASURE
  - .DATA
  - .ALTER
  - .TEMP

© 2014 Synopsys, Inc. (58)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Eldo Netlist Support

- Eldo syntax in XA follows the description in Eldo User Guide.
- If there is a behavior difference between the description of Eldo User Guide and the simulator, XA follows the behavior described in the Eldo User Guide.
- Unsupported syntax (options and statements) is printed in the log file.

© 2014 Synopsys, Inc. (59)

SYNOPSYS®
Predictable Success

Supported Eldo Elements

Element	Instantiation Character
Capacitor	C
Diode	D
VCVS	E
CCCS	F
VCCS	G
CCVS	H
Current Source	I
Coupled Inductor	K
Linear Inductor	L
MOSFET	M
BJT	Q

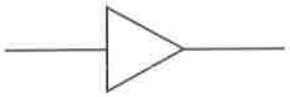
Element	Instantiation Character
Resistor	R
Transmission Line	T
Voltage Source	V
Sub-circuit Instance	X

© 2014 Synopsys, Inc. (60)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Supported Eldo Diode model

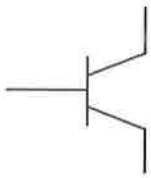


Diode	Level
Berkeley Level 1 Model	1
Modified Berkeley Level 1 Model	2
Fowler-Nordheim Model	3
STMicroelectronics Level 1 Model	4
STMicroelectronics Level 2 Model	5
STMicroelectronics Level 3 Model	6
JUNCAP Model	8

© 2014 Synopsys, Inc. (61)

SYNOPSYS®
Predictable Success

Supported Eldo BJT Model



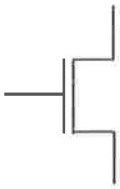
BJT	Eldo Level
Eldo BJT	1
ST BJT level 1	2
Philips Mextram 503.2 Model	4
Improved Berkeley Model	5
HiCUM Model	9
Philips Mextram 504 Model	22
HiCUM0 Model	24

© 2014 Synopsys, Inc. (62)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Supported Eldo MOSFET Model



MOSFET	Level
STMicroelectronics Level 3 Model	19
Berkeley BSIM3v3 Model	53
Philips MOS9 Model	59
Berkeley BSIM4 Model	60
Philips MOS 1102 Model	69
PSP Model	70

© 2014 Synopsys, Inc. (63)

SYNOPSYS®
Predictable Success

Eldo-specific Syntax in XA (1)

- Duplicate port definitions in a .SUBCKT statement

  Both Eldo and XA allow duplicate port definitions in the .SUBCKT statement. If different signals are connected to the duplicate ports, XA shorts the signals.

```
.subckt dupe n1 n2 n2
...
.ends
x1 a b c dupe
```

XA shorts node "b" and node "c".

© 2014 Synopsys, Inc. (64)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Eldo-specific Syntax in XA (2)

- Pre-processor directives

  Pre-processor can only be detected by XA when invoking with the –D or –U command-line flag.

  -D*x* = Define string x and run pre-processor
  -D*x*=*y* = Define string x to be string y and run pre-processor
  -U*x* = Un-define string x and run precessor

- ST-Eldo

  ST version of Eldo can be enabled by adding –stver command-line flag or by adding .opt stver to the netlist.

© 2014 Synopsys, Inc. (65)
SYNOPSYS® Predictable Success

Spectre neltist support

- Spectre Command-line Syntax
- Supported Spectre Elements
- Supported Spectre Device Models
- Supported Spectre Statements
- Supported Spectre Option and Tran Parameters

© 2014 Synopsys, Inc. (68)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Spectre Command-line Syntax

- A Spectre netlist format can be identified by the –nspectre or –spectre command-line flag
  - xa –nspectre input.scs –c cmd –o xa/result

    or
  - xa –spectre input.scs –c cmd –o xa/result

© 2014 Synopsys, Inc. (67)

SYNOPSYS®
Predictable Success

Supported Spectre Elements

- bsource behavioral source
- capacitor two-terminal capacitor
- cccs linear CCCS
- ccvs linear CCVS
- Inductor two-terminal inductor
- iprobe current probe
- mutual_ inductor mutual inductor
- Paramtest parameter value tester
- Pcccs polynomial CCCS
- pccvs polynomial CCVS
- phy_res physical resistor
- pvccs polynomial VCCS
- pvcvs polynomial VCVS
- rdiff diffusion resistor model
- resistor two-terminal resistor
- Switch ideal switch
- Transformer linear two winding transformer
- vccs linear VCCS
- vcvs linear VCVS
- vsource independent voltage source

© 2014 Synopsys, Inc. (68)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Supported Spectre Device Models

- bsim3v3 BSIM3v3 Level-11 Model
- bsim4 BSIM4 Level-14 Model
- mos1 MOS Level-1 Model
- mos903 Compact MOS-transistor Model
- mos11020 MOS model 11, Level 1102
- mos11021 MOS Model 11, Level 1102
- psp PSP Model
- diode Diode Level-1 Model
- diode Diode Level-2 Model
- bjt BJT Model
- bht HiCUM Model
- bht0 HiCUM Level-0 Model
- bjt503 Vertical BJT
- bjt504 Compact BJT
- vbic, version 1.15 VBIC v1.1.5
- vbic, version 1.2 VBIC v1.2
- dio500 Diode Level-500 Model
- jfet JFET Model

© 2014 Synopsys, Inc. (69)

SYNOPSYS®
Predictable Success

Supported Spectre Statements

- global global nodes
- ic initial conditions
- include include file
- library library - sectional include
- parameters netlist parameters
- subckt subcircuit definition
- veriloga Verilog-A include
- simulator SPICE and Spectre mode
- if structural if-statement
- nodeset node sets
- save output selection
- analogmodel using Analogmodel for model passing
- options immediate set option
- tran transient analysis

© 2014 Synopsys, Inc. (70)

SYNOPSYS®
Predictable Success

Supported Spectre Option and Tran Parameters

Option parameter

- currents
- gmin
- nestlvl
- redundant_currents
- rforce
- save
- scale
- scalem
- subcktprobelvl
- temp
- title
- tnom
- useterms

Tran parameter

- autostop
- cmin
- ic
- nestlvl
- oppoint
- readic
- readns
- save
- skipdc
- stop
- write
- writefinal

© 2014 Synopsys, Inc. (71)

SYNOPSYS® Predictable Success

VEC Stimulus Format

- XA supports the following stimulus formats:
  - HSPICE Vector Format (HVEC)
  - Value Change Dump Format (VCD)

© 2014 Synopsys, Inc. (72)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

HSPICE Vector Format (HVEC)

- Identical to HSPICE behavior.
- Is a tabular digital description of nodal behavior.
- Can be used as
  - Drivers to the circuits
  - Expected output comparison to the output nodes

© 2014 Synopsys, Inc. (73)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Content of VEC

- Consist of 3 sections
  - Vector Pattern Definition
    - Contain the size, name, direction, sequence and order of the signals.
    - Radix command must appear 1st in the non-command line to specify the number of bit for each vector.
  - Waveform Characteristics
    - Contain the definition of various attributes of each vector.
  - Tabular Digital Data
    - Define the signal logic state for a specified time.

```
; Vector Pattern Definition
Radix 4 4 1
Vname a[[3:0]] b[[3:0]] ci
Io i i i

; Waveform Characteristics
Vih 3.0
Vil 0.0

; Tabular Digital Data
Tunit ns
0    0 0 0
10   1 1 0
20   2 2 1
30   4 4 0
40   E E 0
50   1 F 1
```

© 2014 Synopsys, Inc. (74)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Commands (1)

Command	Description	Example
Radix	Specify the number of bit associated with each vector. Must be the 1st non-comment line in the VEC file.	Radix 11 4 2
Vname	Specify the name of node driven by each vector.	Vname a b c[3:0] d[1:0]
Io	Define the direction of each vector. Can be I (input), o (output), or b (bi-direction).	Io ii o b
Tunit	Define time and unit for time-related statements and absolute time in the VEC file.	Tunit 2ns
Vref	Specify the reference voltage name for the input.	Vref vss 11 0 1
Period	Define the time interval for the tabular data section so each individual time does not need to be specified	Period 10
Tskip	Specify the absolute time field in the data tabular to be ignored.	Tskip

© 2014 Synopsys, Inc. (75)

SYNOPSYS® Predictable Success

Commands (2)

Command	Description	Example
Enable	Specify the control signal(s) for a bidirectional vector to which the bit-mask applies. The control signal(s) control the direction of bidirectional vector.	Enable va 0010
CBC	Specify the direction of bidirectional vector based on the characters used in the tabular data section. 0,1, and Z are treated as input; L,H,U, andX are treated as output.	Option cbc
Vih	Specify the logic-high voltage for input vector to which the bit-mask applies.	Vih 1.0 11111000
Vil	Specify the logic-low voltage for input vector to which the bit-mask applies.	Vil 0.0 11111100
Vth	Specify the voltage threshold of logic-high and logic-low for output vector to which the bit-mask applies.	Vth 0.5 00000011

© 2014 Synopsys, Inc. (76)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Commands (3)

Command	Description	Example
Voh	Specify the logic-high voltage for output vector to which the mask-bit applies.	Voh 0.8 00000011
Vol	Specify the logic-low voltage for output vector to which the mask-bit applies.	Vol 0.2 00000011
Out, Outz	Specify the resistance of the voltage source when input vectors drive high impedance state to which the bit-mask applies.	Out 100 11000000
Triz	Specifies the output impedance when in XA for input vector to which the bit mask aplies.	triz 10 1000
Tdelay	Specify the delay time for input and output vectors to which the bit-mask applies.	Tdelay 1 1100
Idelay	Specify the delay time only for input vector to which the bit-mask applies.	Idelay 10 1100
Odelay	Specify the delay time only to output vector to which the bit-mask applies.	Odelay 15 1010

© 2014 Synopsys, Inc. (77)

SYNOPSYS®
Predictable Success

Commands (4)

Command	Description	Example
Slope	Define the rise and fall time for input vector to which the bit-mask applies.	Slope 0.5 1011
Trise	Specify the rise time for input vector from Vil to Vih to which the bit-mask applies.	Trise 0.1 1101
Tfall	Specify the fall time for input vector from Vih to Vil to which the bit-mask applies.	Tfall 0.1 1101

© 2014 Synopsys, Inc. (78)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Additional Commands

- These commands are enhancements to HSPICE behavior.

Command	Description	Example
check_window	Define time window for output comparison to which bit mask applies.	check_window 1.5 2.0
vchk_ignore	Define time window to disable output comparison.	vchk_ignore 1 1.2
stop_at_error	Stop simulation whenever there is mismatch for output comparison.	stop_at_error

© 2014 Synopsys, Inc. (79)

SYNOPSYS®
Predictable Success

How to create VEC?

Netlist
Input ports: a[3]-a[0] b[3]-b[0] ci

Requirement
Input logic-high = 3.0V
Input logic-low = 0.0V

1. Create input vectors.
2. Define attribute of the input vectors.
3. Create logic state for each vector at specified time and unit.

```
Radix 4 4 1
Vname a[[3:0]] b[[3:0]] ci
Io i i i

Vih 3.0
Vil 0.0

Tunit ns

0    0 0 0
10   1 1 0
20   2 2 1
30   4 4 0
40   E E 0
50   1 F 1
```

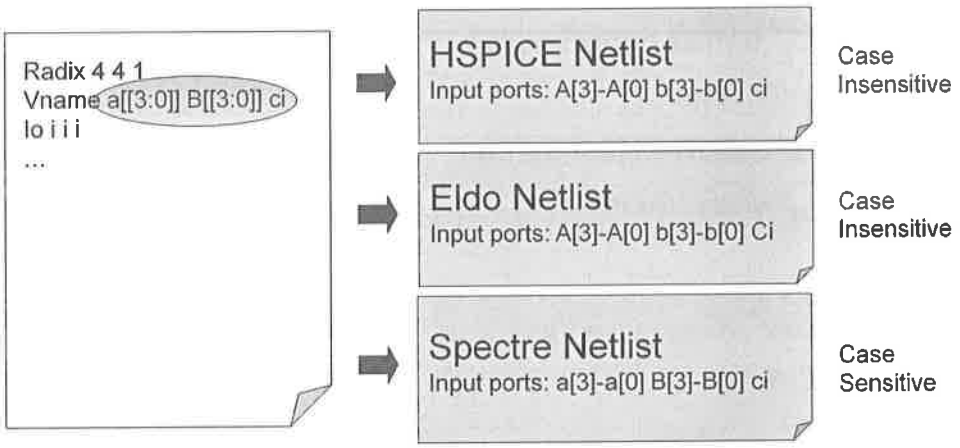
© 2014 Synopsys, Inc. (80)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Case Sensitivity

- Node names are case insensitive in ELDO and HSPICE, and case sensitive in Spectre.



© 2014 Synopsys, Inc. (61)

SYNOPSYS® Predictable Success

How to include VEC file into netlist?

- Command file syntax

  load_vec_file [-file] *filename* [-format vec]

- HSPICE netlist option syntax

  .vec *filename*

- Spectre option netlist syntax

  vec_include "*filename*"

© 2014 Synopsys, Inc. (82)

SYNOPSYS®
Predictable Success

How to create output comparison?

Netlist
Input ports: a[3]-a[0] b[3]-b[0] ci
Output ports: s[3]-s[0]

Requirement
Input logic-high = 3.0V
Input logic-low = 0.0V
Output logic-high = 2.7V
Output logic-low = 0.3V

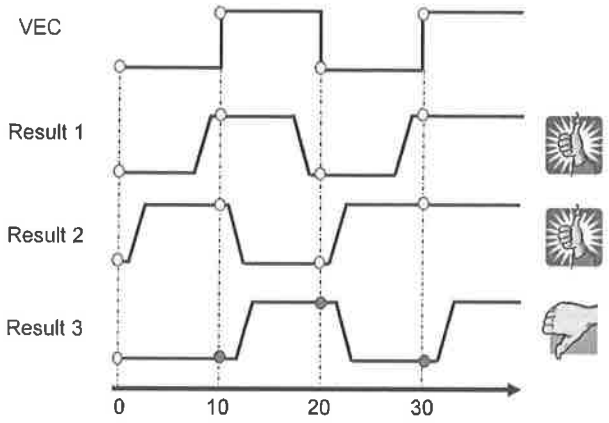
1. Create input and output vectors.
2. Define attribute of the input and output vectors.
3. Create logic state for each vector at specified time and unit. Output logic state specified is the expected output and will be compared against simulation result.

```
Radix 4 4 1 4
Vname a[[3:0]] b[[3:0]] ci s[[3:0]]
lo i i i o

Vih 3.0
Vil 0.0
Voh 2.7
Vol 0.3

Tunit ns

0    0 0 0 0
10   1 1 0 2
20   2 2 1 5
30   4 4 0 8
40   E E 0 D
50   1 F 1 1
```

© 2014 Synopsys, Inc. (83)
SYNOPSYS® Predictable Success

Output Comparison in VEC

- Output comparison is performed only on output vector.
- VEC performs discrete checking




```
...
Vname out ...
Io o ...
...
0    0 ...
10   1 ...
20   0 ...
30   1 ...
...
```

© 2014 Synopsys, Inc. (84)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to create bidirectional port?

Netlist
Input port: ctl_a ctl_c
Bidirectional ports: a c

Requirement
Input logic-high = 3.0V
Input logic-low = 0.0V
Output logic-high = 2.7V
Output logic-low = 0.3V
a is input if ctl_a is logic-high
c is inout if ctl_c is logic-low

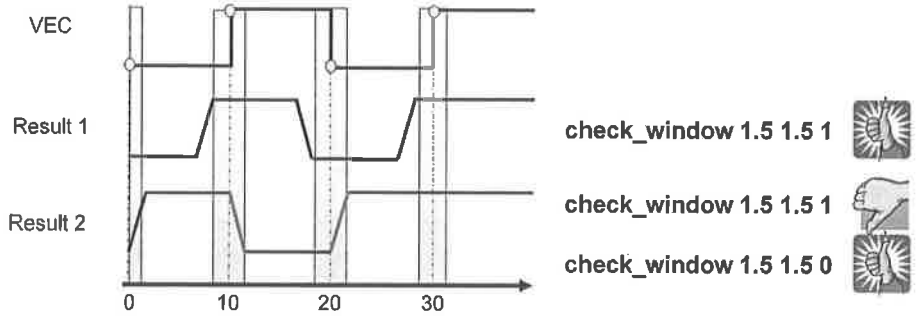
1. Create bidirectional vectors.
2. Create control signal for each bidirectional vector. Control signal can come from input vector or any signal from the netlist.
3. Define attribute of the vectors.
4. Create logic state for each vector at specified time and unit.

```
Radix 1 1 1 1
Vname ctl_a ctl_c a c
Io i i b b

Enable ctl_a 0 0 1 0
Enable ~ctl_c 0 0 0 1

Vih 3.0
Vil 0.0
Voh 2.7
Vol 0.3

Tunit ns

0    1 0 1 1 ; a=in c=in
10   1 1 0 1 ; a=in c=out
20   0 0 1 0 ; a=out c=in
30   0 1 1 1 ; a=out c=out
```

© 2014 Synopsys, Inc. (85)
SYNOPSYS® Predictable Success

How check_window work?

- check_window defines a time frame around the strobe time for output comparison.



© 2014 Synopsys, Inc. (88)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Value Change Dump (VCD)

- Is an open industry standard ASCII format file.
- Is generated by a test pattern generator, or a RTL simulator.
- Contains the digital data of value changes for variables at time increment. Only the variables which changes during a time increment are listed.
- Is case-sensitive for variable names.
- Is hierarchical.

© 2014 Synopsys, Inc. (67)

SYNOPSYS®
Predictable Success

Content of VCD

- Consists of 3 sections
  - Header Information Section
    - Describe the date, simulator, the version of simulator, and time-scale used.
  - Variable Definition Section
    - Contains the scope of the hierarchy and type of variables.
    - Each variable is represented by a unique character.
  - Value Change Section
    - Contains the logic-state of a variable which changes during a time-increment.

© 2014 Synopsys, Inc. (88)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

VCD File

```
$date
Sun Oct 10 18:20:30
$end
$version
Synopsys VCS Version 7.0
$end
$timescale
1ns
$end
$scope module adder $end
$var reg 1 ! CIN $end
$var reg 4 " A [3:0] $end
$var reg 4 # B [3:0] $end
$var wire 1 $ COUT $end
$var wire 4% S [3:0] $end
$upscope $end
$enddefinitions $end
$dumpvars
0!
b0000 "
b0000 #
0$
b0000 %
$end
```

```
#10
b0000 #
b0000 "
0!
#12
b0000 %
0$
#20
b0001 #
#22
b0001 %
#30
b0010 #
#32
b0010 %
#40
b0011 #
#42
b0011 %
#50
b0100 #
```

© 2014 Synopsys, Inc. (89)

SYNOPSYS®
Predictable Success

How to use VCD file?

- Need an intermediate file (signal information file or VCD control file) to map the variables in the VCD file to the nodes in the netlist.
- To avoid modifying the VCD file which could be extremely large in some cases.
- Direction of the vector and various waveform attributes can be defined in the VCD control file.

© 2014 Synopsys, Inc. (90)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Commands (1)

Command	Description	Example
#format	Specify the bus format used in the netlist.	#format %[#]
#input	Specify the input variable name.	#input a b c
#output	Specify the output variable name.	#output out
#bidirectional	Specify the bidirectional variable name and its control signal(s).	#bidirectional (in en)
#alias	Map variable name to node name in netlist.	#alias a ain
#vih	Specify the logic-high voltage for input vectors.	#vih 1.0
#vil	Specify the logic-low voltage for input vectors.	#vil 0.0
#voh	Specify the threshold for logic-high voltage for output vectors.	#voh 0.9
#vol	Specify the threshold for logic-low voltage for output vectors.	#vol 0.1
#outz	Specify the resistance of the voltage source for input vectors.	#outz 10000

© 2014 Synopsys, Inc. (91)

SYNOPSYS®
Predictable Success

Commands (2)

Command	Description	Example
#idelay	Specify the delay time only for the input vectors.	#idelay 1ns
#odelay	Specify the delay time only for the output vectors.	#odelay 10ns
#tfall	Specify the fall time for input vectors from #vih to #vil.	#tfall 0.5us
#trise	Specify the rise time for input vectors from #vil to #vih.	#trise 0.5us
#iw	Define the "don't care" window to which output comparison is ignored.	#iw 0.1u 0.2u
#scope	Specify the hierarchical scope name in the VCD file, from which the variables are used.	#scope adder

© 2014 Synopsys, Inc. (92)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to create VCD control file?

Netlist
Input ports: a[3]-a[0] b[3]-b[0] ci
Output ports: s[3]-s[0]

Requirement
Input logic-high = 3.0V
Input logic-low = 0.0V
Output logic-high = 2.7V
Output logic-low = 0.3V

1. Define bus syntax used in netlist.
Only one #format per VCD control file.
2. Define name and direction of vectors.
Many ways to define the name.
3. Define attributes for vectors.

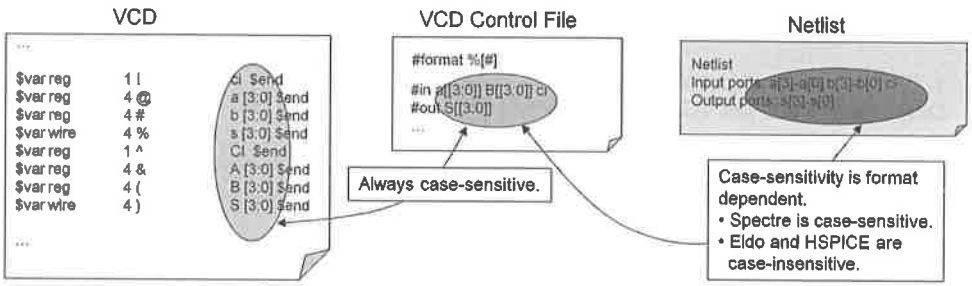
```
$date
   Sep 28, 1993  16:20:30
$end
$version
   VERILOG-XL 1.6b
$end
$timescale
   1us
$end

$var reg     1 !   ci  $end
$var reg     4 "   a [3:0] $end
$var reg     4 #   b [3:0] $end
$var wire    4 %   s [3:0] $end

$enddefinitions $end

$dumpvars
0!
b0000 "
b0000 #
b0000 %
$end
#10
b0000 #
b0000 "
0!
#12
b0000 %
#20
b0001 #
```

```
#format %[#]

#in a[[3:0]] b[[3:0]] ci
#out s[[3:0]]

#vih 3.0
#vil 0.0
#voh 2.7
#vol 0.3
```

© 2014 Synopsys, Inc. (93)
SYNOPSYS® Predictable Success

Case-sensitivity in VCD

- Mapping of variable names from VCD control file to VCD file is always case-sensitive.
- Case-sensitivity of mapping of node names from VCD control file to netlist is format-dependent.



© 2014 Synopsys, Inc. (94)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to create VCD control file for Spectre?

Netlist
Input ports: a[3]-a[0] B[3]-B[0] ci
Output ports: s[3]-S[0]

Requirement
Input logic-high = 3.0V
Input logic-low = 0.0V
Output logic-high = 2.7V
Output logic-low = 0.3V

1. Define bus syntax used in netlist.
Only one #format per VCD control file.
2. Define name and direction of vectors.
Many ways to define the name.
3. Map variable names to node names.
4. Define attributes for vectors.

```
$date
   Sep 28, 1993  18:20:30
$end
$version
   VERILOG-XL 1.6b
$end
$timescale
   1us
$end

$var reg     1 !   ci  $end
$var reg     4 "   a [3:0] $end
$var reg     4 #   b [3:0] $end
$var wire    4 %   s [3:0] $end

$enddefinitions $end

$dumpvars
0!
b0000 "
b0000 #
b0000 %
$end
#10
b0000 #
b0000 "
0!
#12
b0000 %
#20
b0001 #
```

```
#format %[#]
#in a[[3:0]] b[[3:0]] ci
#out s[[3:0]]
#alias b% B%
#alias s% S%
#vih 3.0
#vil 0.0
#voh 2.7
#vol 0.3
```

© 2014 Synopsys, Inc. (95)
SYNOPSYS® Predictable Success

How to include VCD file into netlist?

- Command file syntax

  load_vec_file [–file] *filename* –format vcd –ctl *ctlname*

- Spectre netlist option syntax

  vcd_include "*filename*" "*ctlname*"

© 2014 Synopsys, Inc. (98)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to map to a different name?

Netlist
Input ports: aa[3]-aa[0] b[3]-b[0] ci
Output ports: ss[3]-ss[0]

Requirement
Input logic-high = 3.0V
Input logic-low = 0.0V
Output logic-high = 2.7V
Output logic-low = 0.3V

1. Define bus syntax used in netlist. Only one #format per VCD control file.
2. Define name and direction of vectors. Many ways to define the name.
3. Map variable names to netlist node names.
4. Define attributes for vectors.

```
$date
   Sep 28, 1993  16:20:30
$end
$version
   VERILOG-XL 1.6b
$end
$timescale
   1us
$end

$var reg     1 !   ci  $end
$var reg     4 "   a [3:0] $end
$var reg     4 #   b [3:0] $end
$var wire    4 %   s [3:0] $end

$enddefinitions $end

$dumpvars
0!
b0000 "
b0000 #
b0000 %
$end
#10
b0000 #
b0000 "
0!
#12
b0000 %
#20
b0001 #
```

```
#format %[#]
#in a[[3:0]] b[[3:0]] ci
#out s[[3:0]]

#alias a% aa%
#alias s% ss%

#vih 3.0
#vil 0.0
#voh 2.7
#vol 0.3
```

© 2014 Synopsys, Inc. (97)

SYNOPSYS® Predictable Success

How to use hierarchical VCD?

Netlist
Input ports: aa[3]-aa[0] b[3]-b[0] ci
Output ports: ss[3]-ss[0]

Requirement
Input logic-high = 3.0V
Input logic-low = 0.0V
Output logic-high = 2.7V
Output logic-low = 0.3V

1. Specify the hierarchical scope name. Only one #scope per VCD control file.
2. Define bus syntax used in netlist. Only one #format per VCD control file.
3. Define name and direction of vectors. Many ways to define the name.
4. Map variable names to netlist node names.
5. Define attributes for vectors.

```
$date
   Sep 26, 1993  18:20:30
$end
$version
   VERILOG-XL 1.6b
$end
$timescale
   1us
$end

$scope module adder $end
$var reg     1 !   ci  $end
$var reg     4 "   a [3:0] $end
$var reg     4 #   b [3:0] $end
$var wire    4 %   s [3:0] $end
Supscope $end

$enddefinitions $end
$dumpvars
0!
b0000 "
b0000 #
b0000 %
$end
#10
b0000 #
b0000 "
0!
#12
b0000 %
#20
b0001 #
```

```
#scope adder
#format %[#]
#in a[[3:0]] b[[3:0]] ci
#out s[[3:0]]
#alias a% aa%
#alias s% ss%
#vih 3.0
#vil 0.0
#voh 2.7
#vol 0.3
```

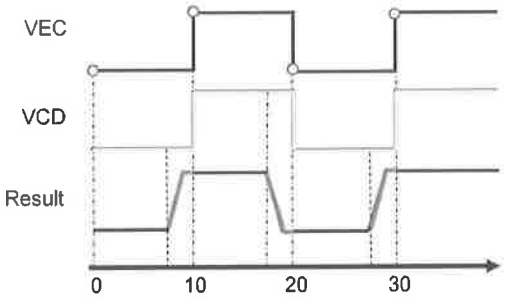
© 2014 Synopsys, Inc. (98)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Output Comparison in VCD

- Output comparison is performed only on output vector.
- Output comparison in VCD is continuous
  - Different from VEC which is discrete



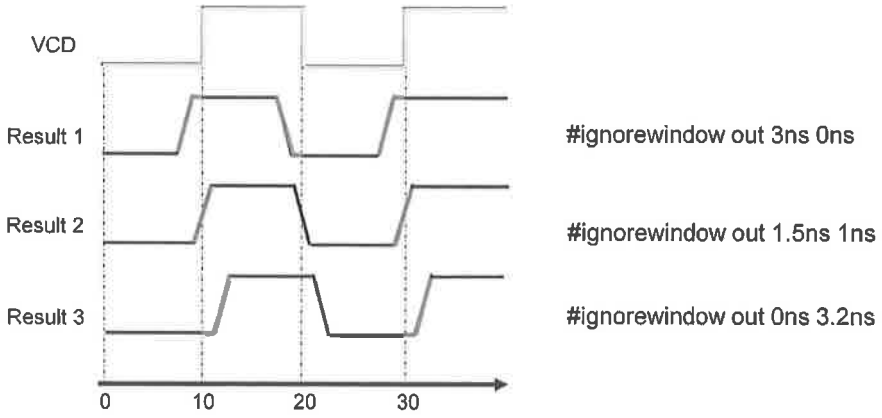

```
...
Vname out ...
lo o ...
...
0    0 ...
10   1 ...
20   0 ...
30   1 ...
...
```

```
...
$var reg 1 l out  $end
...
$dumpvars
0!
$end
#10
1!
#20
0!
#30
1!
...
```

© 2014 Synopsys, Inc. (99)

SYNOPSYS®
Predictable Success

How to define "Don't Care" window?



© 2014 Synopsys, Inc. (100)

SYNOPSYS®
Predictable Success

Interactive mode and analysis

© 2014 Synopsys, Inc. (101)

SYNOPSYS®
Predictable Success

All courses must have a target audience that is well defined. This sets the stage for the entire workshop. Please use the graphic provided in the slide.

Unit Objectives

- After completing this unit, you should know
  - How to enable interactive mode debug in CustomSim

© 2014 Synopsys, Inc. (102)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to get into interactive mode

- Invoke the interactive mode at specified time on command line option -inter
  - xa design.sp –inter 20n
- Add breakpoint in interactive mode:
  - XA>iset_break_point at time
- Ctrl-C can be used to re-enter interactive mode only if -intr command-line flag is specified. Otherwise, XA terminates the simulation.

© 2014 Synopsys, Inc. (103)

SYNOPSYS®
Predictable Success

Get help in interactive mode

- Print all of interactive command
  - XA>help
- Print specific interactive command usage
  - XA>help command

© 2014 Synopsys, Inc. (104)

SYNOPSYS®
Predictable Success

Interactive Commands

Command	Description
iset_break_point –at	Pause the simulation at the specified time.
ilist_break_point	List all the break point.
idelete_break_point	Delete the break point.
icontinue_sim	Continue the simulation.
iquit_sim	Quit the simulation.
iprint_node_info	Print node voltage and current simulation time.
iprint_elem_info	Print detail element information.
iprint_connectivity	Print detail node connectivity.
iopen_log	Start recording the interactive process in a file.
iclose_log	Stop recording the interactive process in a file.
iprint_help	Display description of commands.
source	Read and execute command from a file.
alias	Create alias for command.
history	Display history of last 20 interactive commands

© 2014 Synopsys, Inc. (105)

SYNOPSYS®
Predictable Success

probe_waveform_voltage

- Syntax

  probe_waveform_voltage [[-v] *node* {*node*}] \

  [-v*n* *instance* {*instance*}] [-vall *instance* {*instance*}]] [-limit *level*]

- Probe the voltage of a node or a pin of an instance.
- Example

  probe_waveform_voltage a b c

  This command probes the voltage of top-level nodes which names a, b and c.

© 2014 Synopsys, Inc. (106)

SYNOPSYS®
Predictable Success

probe_waveform_current

- Syntax

  probe_waveform_current [[-i|-i1] *instance* {*instance*}] \

  [-i*n* *instance* {*instance*}] [-iall *instance* {*instance*}] [-limit *level*]

- Probe the current through an instance pin.
- Example

  probe_waveform_current -i vvdd vvss -iall x1.M1

  This command probes the first branch current of vvdd and vvss and all branch current of x1.M1.

© 2014 Synopsys, Inc. (107)

SYNOPSYS®
Predictable Success

probe_waveform_logic

- Syntax

  probe_waveform_logic [-node] *node* {*node*} [-vol *VLth*] [-voh *VHth*] \
  [-limit *level*]

- Probe logic waveform of a node. Need to specify either *VLth* or *VHth*.
- Example

  probe_waveform_logic a3 -vol 3

  This command probes the logic waveform of node a3. If the voltage value is equal of less than 3, the logic state is LOW; Otherwise, it is HIGH.

© 2014 Synopsys, Inc. (108)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

report_power

- Syntax

  report_power -port *port_name* {*port_name*} [-label *label_name*] \

  [-from *start_time*] [-to *stop_time*] [-limit *level*] [-subckt *subckt*] \

  [-avg *enable_value*] [-rms *enable_value*] \

  [-max *enable_value*] [-min *enable_value*]

- Analyze sub-circuit power consumption of a port and generate an ASCII report (*file*.power0).
- Example

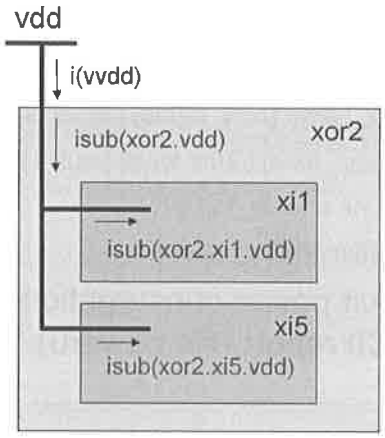
  report_power vdd -label power

  This command reports the power consumption of port vdd down to the 3rd hierarchy.

© 2014 Synopsys, Inc. (109)

SYNOPSYS® Predictable Success

How Does report_power work?



i(vvdd) = -isub(xor2.vdd)
isub(xor2.vdd) = -isub(xor2.xi1.vdd) - isub(xor2.xi5.vdd)

© 2014 Synopsys, Inc. (110)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Example of *file*.power0

LABEL=POWER FROM=0 TO=1.85E-07

Port Name: XOR2.VDD (vdd) isub(xor2.vdd)
Sub-circuit Definition:
Max(A)= 2.7565584e-03 Min(A)= -4.3443018e-04 Avg(A)= 1.0163091e-05 Rms(A)= 1.2646468e-04
Max(W)= 1.3782792e-02 Min(W)= -2.1721509e-03 Avg(W)= 5.0815455e-05 Rms(W)= 6.3232338e-04

Port Name: XOR2.XI1.VDD (vdd) isub(xor2.xi1.vdd)
Sub-circuit Definition:
Max(A)= 2.5055710e-03 Min(A)= -2.9511190e-04 Avg(A)= 6.1408835e-06 Rms(A)= 1.0377710e-04
Max(W)= 1.2527855e-02 Min(W)= -1.4755595e-03 Avg(W)= 3.0704417e-05 Rms(W)= 5.1886550e-04

Port Name: XOR2.XI5.VDD (vdd) isub(xor2.xi5.vdd)
Sub-circuit Definition:
Max(A)= 1.4685299e-03 Min(A)= -4.3469633e-04 Avg(A)= 4.0219188e-06 Rms(A)= 5.9391395e-05
Max(W)= 7.3426495e-03 Min(W)= -2.1734816e-03 Avg(W)= 2.0109594e-05 Rms(W)= 2.9695697e-04

© 2014 Synopsys, Inc. (111)
SYNOPSYS® Predictable Success

Wildcard

- Supported wildcard patterns

Pattern	Description
?	Matches any single character, except the period (.) and white space.
[list]	Matches any single character in the list. It can be alphabetical characters or numerical characters. A range of characters can be specified by using a dash (-).
[~list] or (!list] or [^list]	Matches any single character that is not white space, period (.) or in the lost.
*	Matches any contiguous group of characters, except for white space across all levels of hierarchy. Use set_wildcard_rule to change of the rule.

© 2014 Synopsys, Inc. (112)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

set_wildcard_rule

- Syntax

  set_wildcard_rule [-match*] all|one

- Select the wildcard rule for * character. If one is specified, XA matches * character to only one level of hierarchy. Otherwise, it matches to all levels of hierarchy. Keep in mind that this rule does not apply to the netlist statements such as .probe and .print.
- Example

  set_wildcard_rule –match* one

© 2014 Synopsys, Inc. (113)

SYNOPSYS® Predictable Success

How wildcard work?

- Example 1

  probe_waveform_voltage *.* -limit 4

  XA probes voltage of all nodes from the 1st hierarchy down to the 4th hierarchy.

- Example 2

  probe_waveform_voltage *.* -limit 0

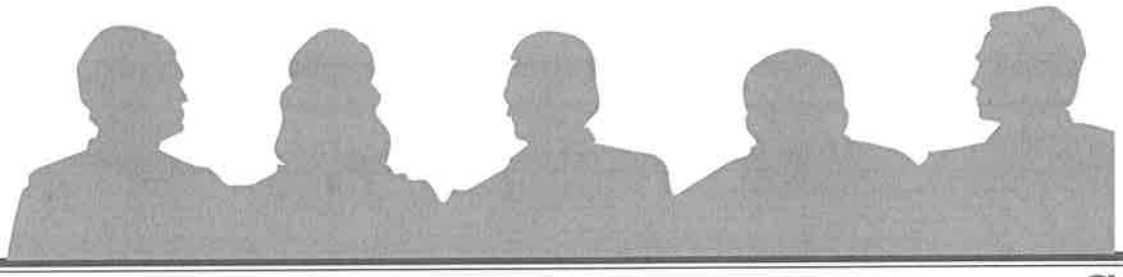
  XA does not probe any node. *.* indicates any nodes at the 1st hierarchy and below and -limit 0 limits only to the top hierarchy. The arguments contradict each others.

© 2014 Synopsys, Inc. (114)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Post-layout Simulation



© 2014 Synopsys, Inc. (115)

SYNOPSYS®
Predictable Success

All courses must have a target audience that is well defined. This sets the stage for the entire workshop. Please use the graphic provided in the slide.

Unit Objectives

- After completing this unit, you should know
  - XA-supported post-layout flows
  - XA-supported post-layout netlist formats
  - How to setup Star-RCXT and XA for different flows
  - How to write analysis statements that are compatible for pre-layout and post-layout flows
  - The limitation of current post-layout flows

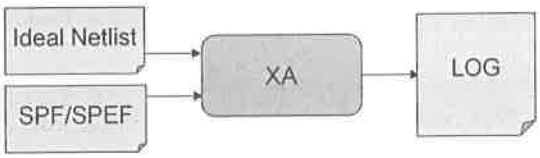
© 2014 Synopsys, Inc. (116)

SYNOPSYS®
Predictable Success

Post-layout Simulation

- XA supports
  - Flat post-layout simulation

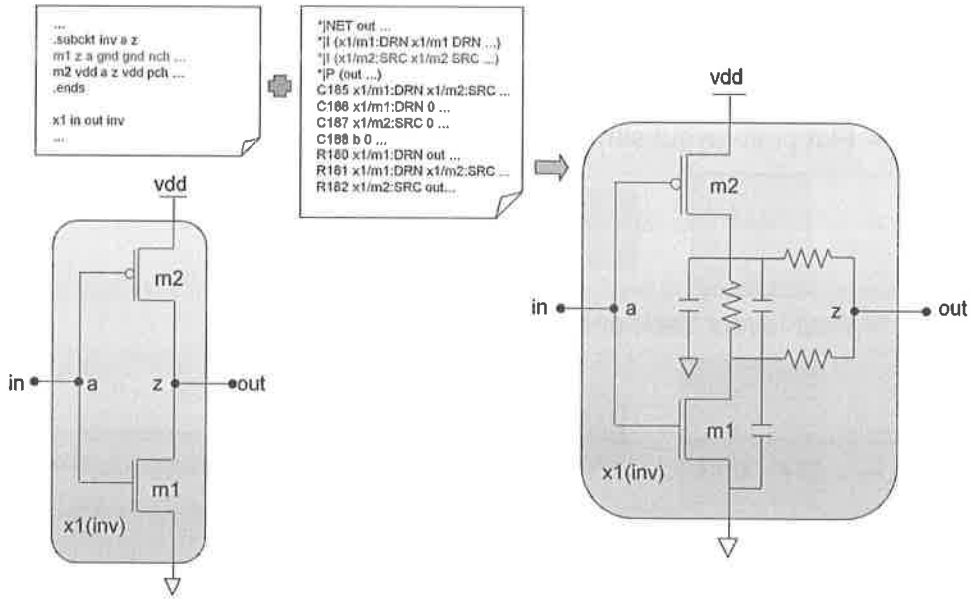
  - Post-layout back-annotation simulation



© 2014 Synopsys, Inc. (117)

SYNOPSYS®
Predictable Success

What is back-annotation?



© 2014 Synopsys, Inc. (116)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

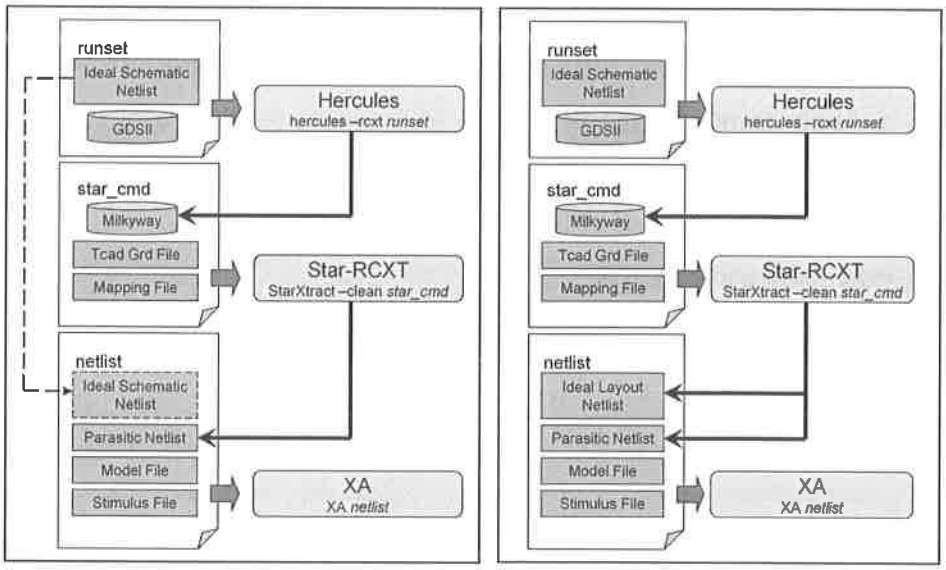
Advantage of Back-annotation Flow

- Full control on the hierarchical schematic netlist
  - Apply command hierarchically
  - Re-use analysis statements from pre-layout simulation
- Faster simulation runtime
- Less memory usage
- Support both DPF, SPEF and SPF formats

© 2014 Synopsys, Inc. (119)

SYNOPSYS®
Predictable Success

Back-annotation Flow



© 2014 Synopsys, Inc. (120)

SYNOPSYS
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to set up Star-RCXT?

- Commands used by Star-RC XT to complete back-annotation flow in XA.
  - XREF: NO|NETS|YES|COMPLETE
    - Determines which set of names to report for Star-RCXT netlisting and analysis flows and which devices and nets to retain.
    - XA supports only YES and COMPLETE.
    - XREF:YES is layout-based; every layout device and net is reported. The parasitic netlist will use the schematic names for the matched nets and devices.
    - XREF: COMPLETE is schematic-based; parasitics are netlisted only for nets that were successfully cross-referenced to schematic nets.

© 2014 Synopsys, Inc. (121)

SYNOPSYS® Predictable Success

How to set up Star-RCXT? (cont)

- NETLIST_FORMAT: SPF|STAR|SPEF|SBPF|MW|CONLY|NETNAME|NONE
  - Defines the structure and format of the output parasitic netlist.
  - XA supports only SPF and SPEF.
- NETLIST_FILE: *file_name*
  - Specifies the name of the file to which the output parasitic netlist is written.

© 2014 Synopsys, Inc. (122)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to set up Star-RCXT? (cont)

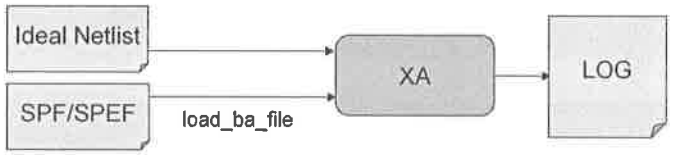
- NETLIST_IDEAL_SPICE_FILE: *file_name*
  - Specifies the name of an ideal SPICE-format netlist for use with simulation.
- NETLIST_IDEAL_SPICE_HIER: YES|NO|SCHEMATIC
  - Specifies whether or not to preserve the original hierarchy when you are generating the NETLIST_IDEAL_SPICE_FILE.
  - YES is recommended for best performance.

© 2014 Synopsys, Inc. (123)

SYNOPSYS®
Predictable Success

How to set up XA?

- One command to include parasitic file for back-annotation simulation
  - load_ba_file [-file] *file_name*
  - Automatically detects the parasitic file format, either DPF, SPF or SPEF format

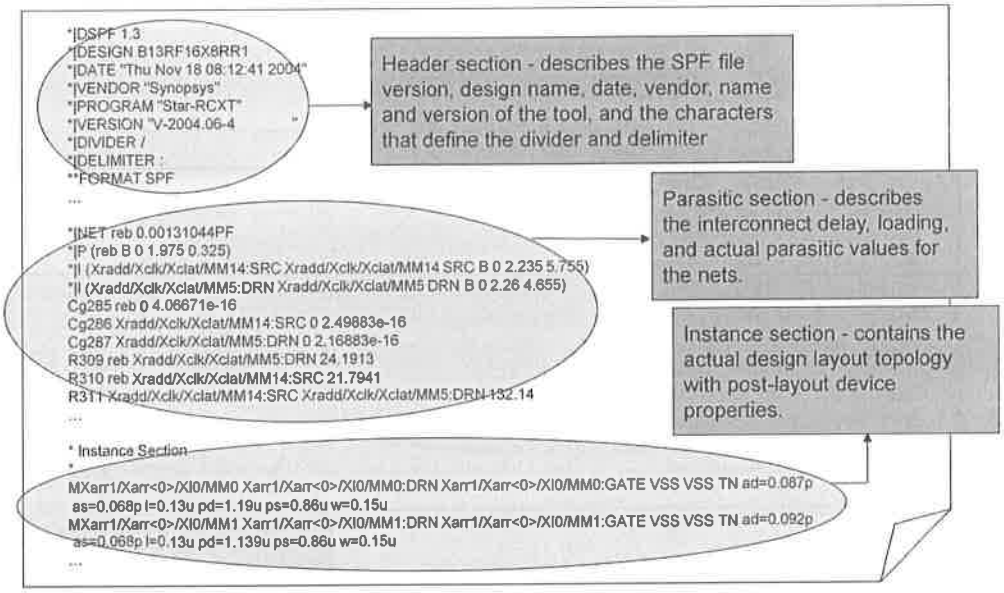


© 2014 Synopsys, Inc. (124)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

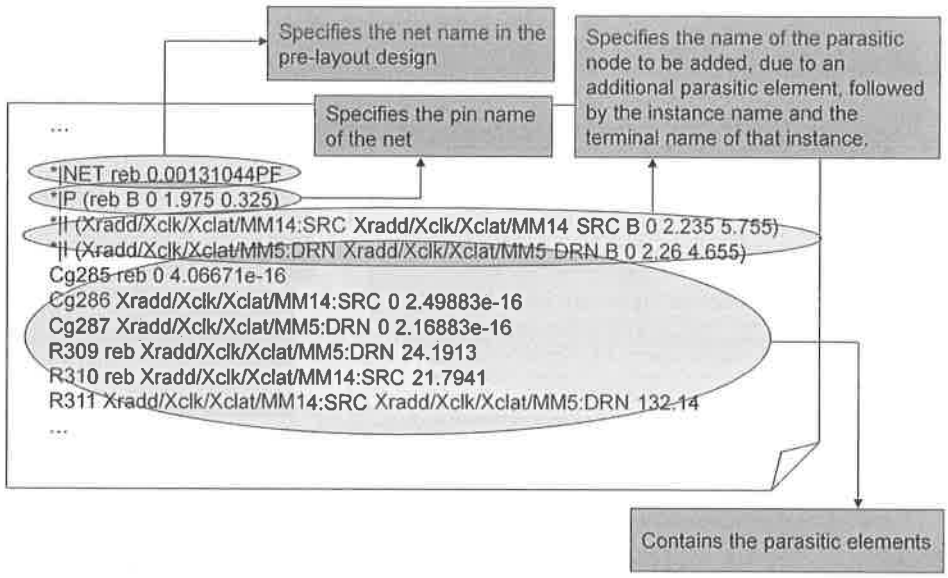
Structure of SPF File



© 2014 Synopsys, Inc. (125)

SYNOPSYS®
Predictable Success

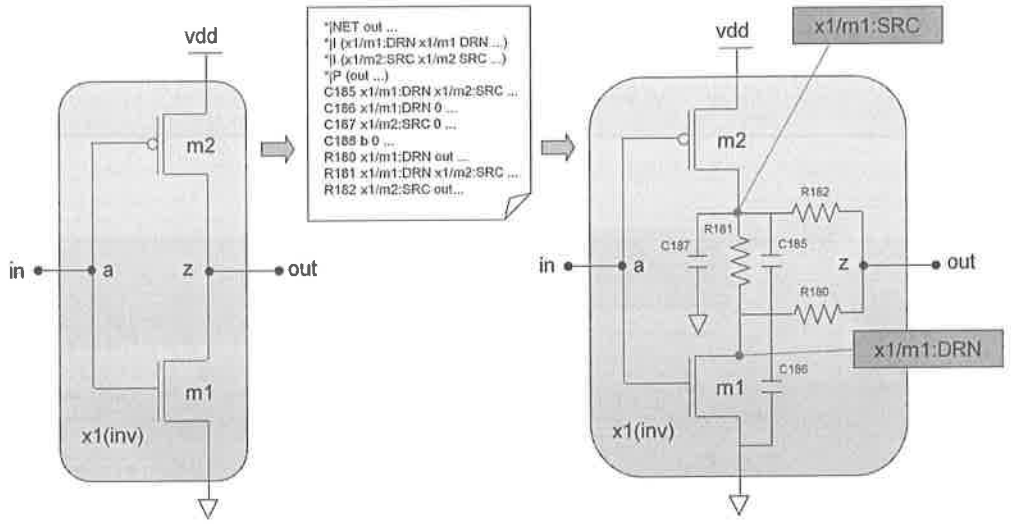
Structure of Parasitic Section



© 2014 Synopsys, Inc. (126)

SYNOPSYS® Predictable Success

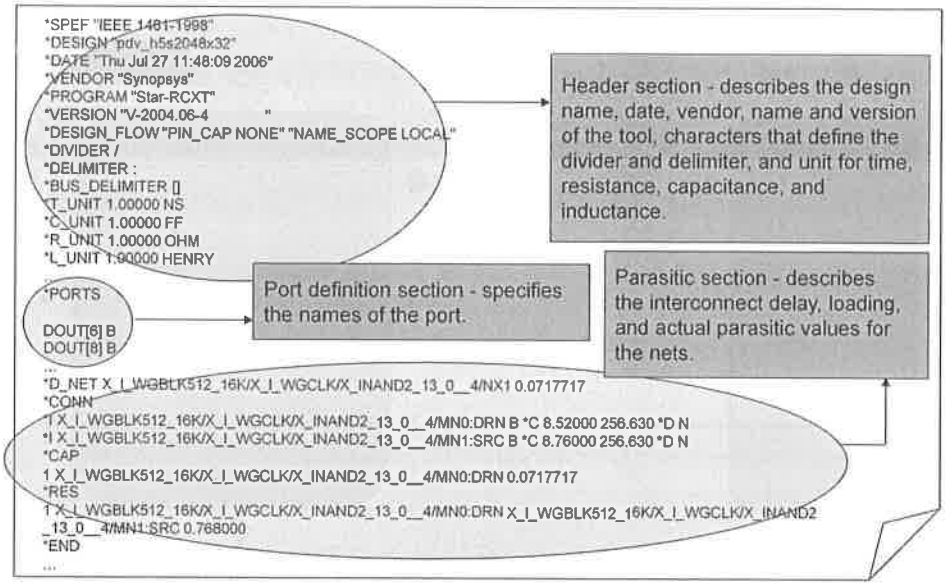
How does back-annotation work?



© 2014 Synopsys, Inc. (127)

SYNOPSYS
Predictable Success

Structure of SPEF



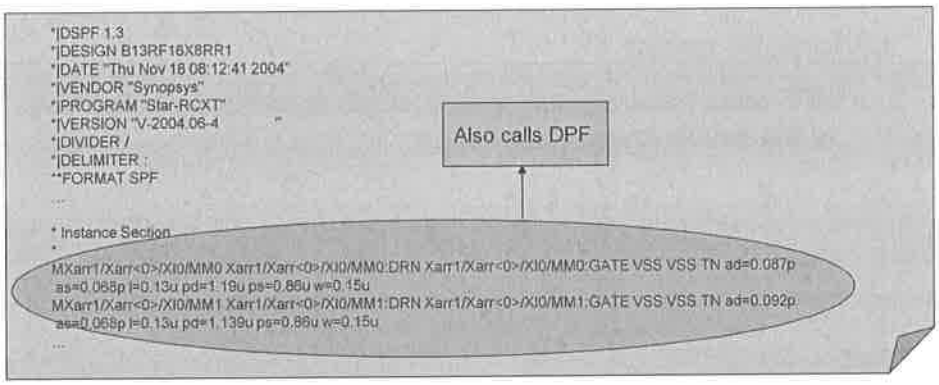
© 2014 Synopsys, Inc. (128)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

What is DPF?

- Stands for Device Parasitic File.
  - Contains the properties of the device layout parameters.
  - Support only flat DPF.
- Is the instance section of a SPF file.



© 2014 Synopsys, Inc. (129)

SYNOPSYS®
Predictable Success

When to use DPF?

- When ideal schematic netlist is used.
  - Ideal layout netlist contains device layout parameter so DPF is not needed.
- When SPEF is used for back-annotation into an ideal schematic netlist.
  - SPF often has instance section which contains the properties of the device layout parameters.

© 2014 Synopsys, Inc. (130)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to generate a DPF file?

- NETLIST_IDEAL_SPICE_FILE: *file_name*
  - Specifies the name of an ideal SPICE-format netlist for use with simulation.
- NETLIST_IDEAL_SPICE_HIER: NO
  - NO creates a flattened ideal SPICE-format netlist which is always a DPF.
  - YES creates a ideal layout-hierarchy-base SPICE-format netlist.

© 2014 Synopsys, Inc. (131)

SYNOPSYS® Predictable Success

How to include DPF?

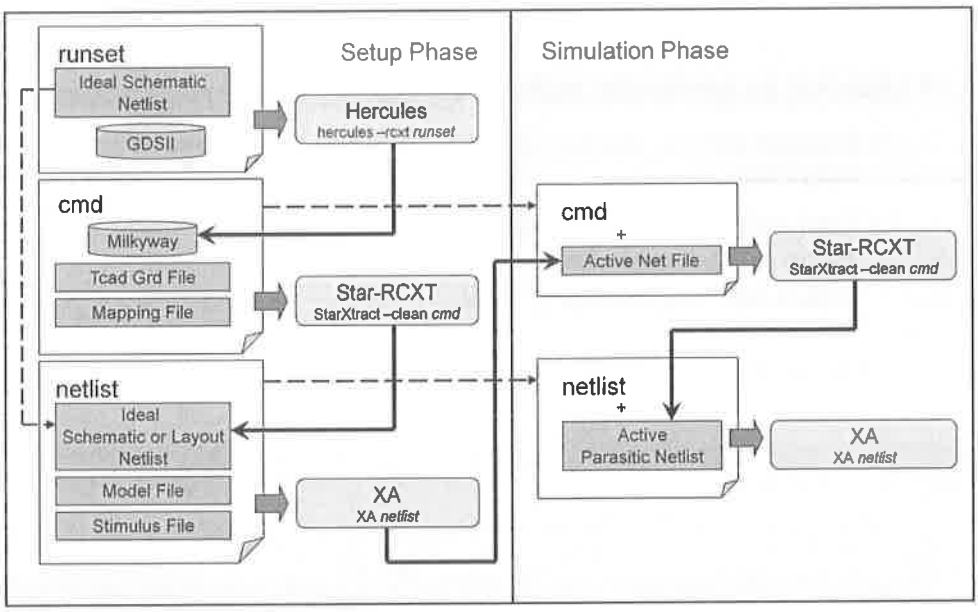
- If SPF has instance section, XA automatically back-annotates the device layout parameters.
- To skip back-annotation of instance section of SPF, use
  - skip_ba_dpf [-switch] *enable_value*
  - *enable_value* can be 1, ON, TRUE, YES to turn on or 0, OFF, FLASE, NO to turn off
- To include DPF file, use
  - load_ba_file [-file] *file_name*
  - Automatically detects the parasitic file format, either DPF, SPF or SPEF format

© 2014 Synopsys, Inc. (132)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Selective-net Extraction-Simulation



© 2014 Synopsys, Inc. (133)

SYNOPSYS®
Predictable Success

How to generate active net file?

- Use XA to generate active net file.
  - Include set_ba_active_file [-file] *filename* in the command script file.
  - Can also change the voltage tolerance with -vtol *voltage*. Node with voltage value exceeding *voltage* is considered as active net. The default is 0.1V.

© 2014 Synopsys, Inc. (134)

SYNOPSYS®
Predictable Success

How to set up Star-RCXT?

- You must modify Star-RCXT command file from setup phase to only extract the selected nets.
  - Comment out NETS: * if it is specified. This command instructs Star-RCXT to extract all nets.
  - Include NET_FILES: *XA-generated-active-file* to only extract the nets specified in the file. The nets which are specified in the file are the nets XA considers as active.

© 2014 Synopsys, Inc. (135)

SYNOPSYS®
Predictable Success

How to set up XA?

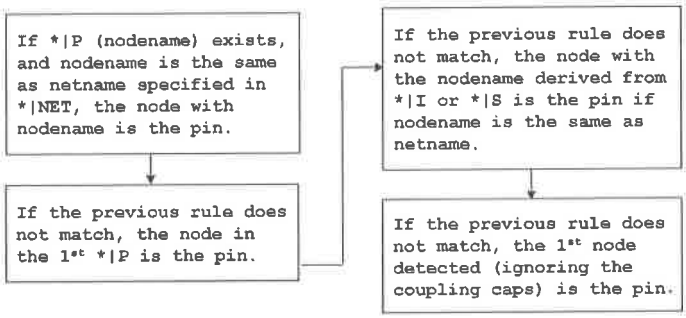
- You must include the Star-RCXT generated extraction file to run back-annotation flow.
  - Comment out set_ba_active_file command in the command script file.
  - Add load_ba_file –file *rcxt-generated-file* in the command script file.

© 2014 Synopsys, Inc. (136)

SYNOPSYS®
Predictable Success

What is pin?

- XA assigns a pin to each net.
- The exact location of a net after back-annotation
  - Connects the missing instance(s) (*|I) to the pin of each net

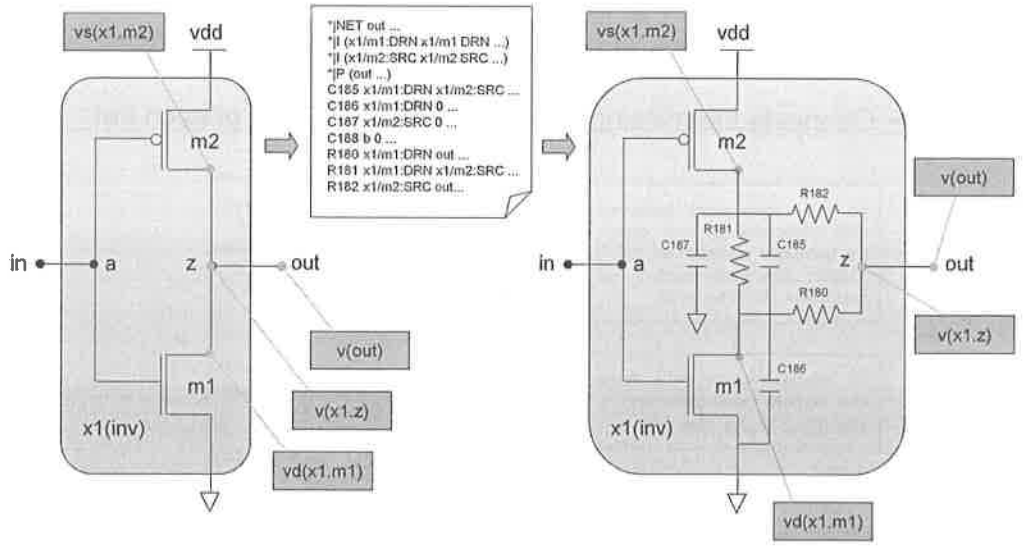


© 2014 Synopsys, Inc. (137)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How to create analysis statements?

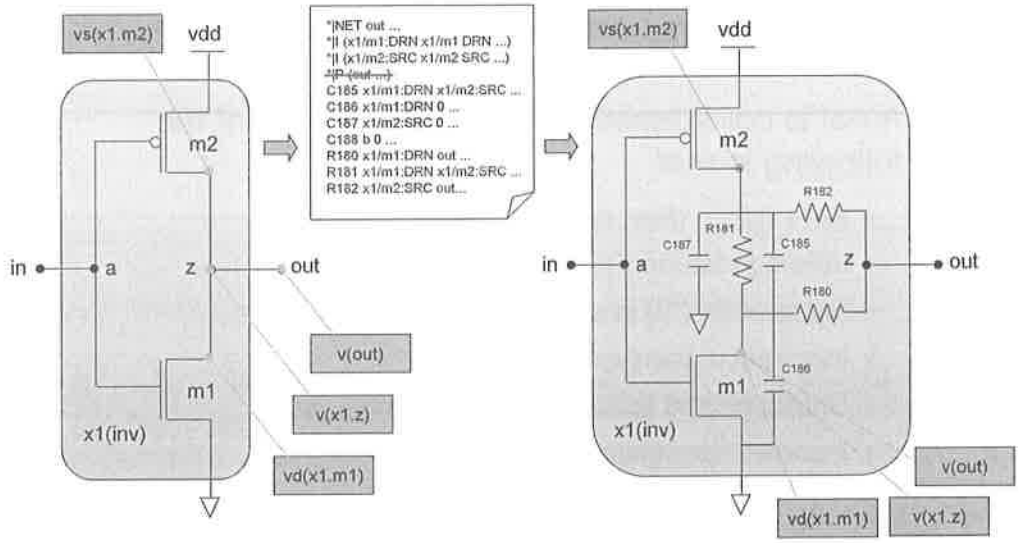


© 2014 Synopsys, Inc. (138)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

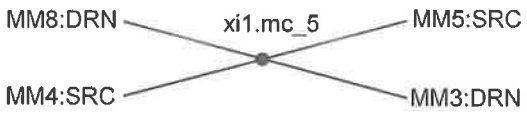
What if *|P does not exist?



© 2014 Synopsys, Inc. (139)

SYNOPSYS®
Predictable Success

What is error net?



- A net is considered as an error net if one of the following is met
  - Net (*|NET) does not exist
  - Missing instance(s) (*|I) in a net
  - Instance(s) (*|I) in a net does not exist
  - Incorrect instance(s) connection
  - Unrecognized terminal name(s)
  - Incorrect terminal(s) connection

© 2014 Synopsys, Inc. (140)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How XA handle with error nets?

- Net (*|NET) does not exist
  - Ignore of the back-annotation of the net
- Missing instance(s) (*|I) in a net
  - Connect the missing instance(s) to the pin of each net
- Instance(s) (*|I) in a net does not exist
  - Ignore the instance(s)
- Incorrect instance(s) connection
  - Ignore the back-annotation of the net unless enable_ba_error_net is set
- Unrecognized terminal name(s)
  - Ignore the back-annotation of the net unless map_ba_terminal is set
- Incorrect terminal(s) connection
  - Automatically swap SRC and DRN
  - Ignore the back-annotation of the net unless enable_ba_error_net is set

© 2014 Synopsys, Inc. (141)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

enable_ba_error_net

- Syntax

  enable_ba_error_net [-switch] setting_value

  setting_value ::= 0|off|no|false = ignore back-annotation of the net
  1|on|yes|true = ignore incorrect elements
  lump = back-annotates the lumped capacitance

- Control the back-annotation behavior for error nets.
- Example

  enable_ba_error_net -switch lump

  This command enable back-annotation of the lumped capacitance to the error nets.

© 2014 Synopsys, Inc. (142)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Recognizable Terminal Names

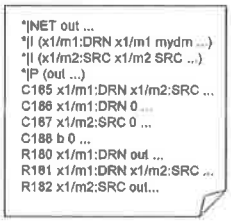
Terminal Index	M	Q	R, C, D
1	D[R][A][I][N]	C[O][L][L][E][C][T][O][R]	A[N][O][D][E], P[L][U][S], P[O][S][I][T][I][V][E]
2	G[A][T][E]	B[A][S][E]	B, C[A][T][H][O][D][E], M[I][N][U][S], N[E][G][A][T][I][V][E]
3	S[O][U][R][C][E]	E[M][I][T][T][E][R]	S[U][B][S][T][R][A][T][E]
4	B[U][L][K]	S[U][B][S][T][R][A][T][E]	n/a

© 2014 Synopsys, Inc. (143)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

map_ba_terminal



- Syntax

  map_ba_terminal [-name] ba_file_terminal_name] \

  [-alias] valid_terminal_name

- Map the terminal name in the back-annotation file to the terminal name recognized by XA.
- Example

  map_terminal_name mydrn drn

  This command maps the unrecognized terminal name "mydrn" to the drain terminal".

© 2014 Synopsys, Inc. (144)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Summary of Back-annotation

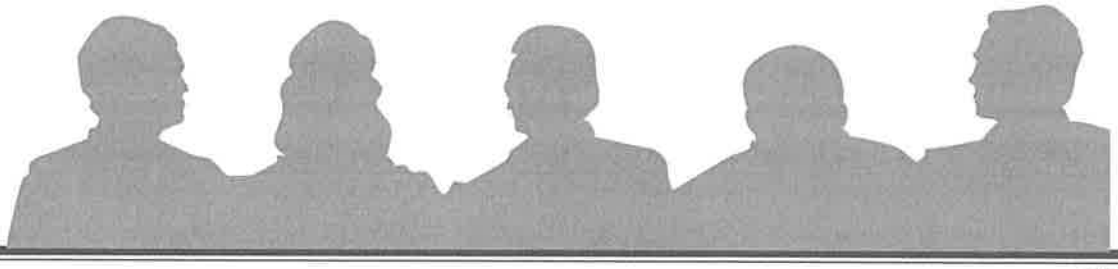
Summary of parsing DSPF file "d16384x32_complete.spf"

module name	PDV_H5D16384X32
Lines parsed	15012244
Nets parsed	18926
Nets back-annotated without error	18926 (100%)
Nets back-annotated with enable_ba_error_net	0 (0%)
Nets not back-annotated	0 (0%)
Empty nets	0 (0%)
Parasitic nodes added	763690
Resistors back-annotated	1754669
Grounded capacitors back-annotated	473026
Coupling capacitors back-annotated	273443
Elements parameters updated (DPF)	12262555

© 2014 Synopsys, Inc. (145)

SYNOPSYS
Predictable Success

CustomSim Co-simulation with VCS



© 2014 Synopsys, Inc. (146)

SYNOPSYS®
Predictable Success

All courses must have a target audience that is well defined. This sets the stage for the entire workshop. Please use the graphic provided in the slide.

Unit Objectives

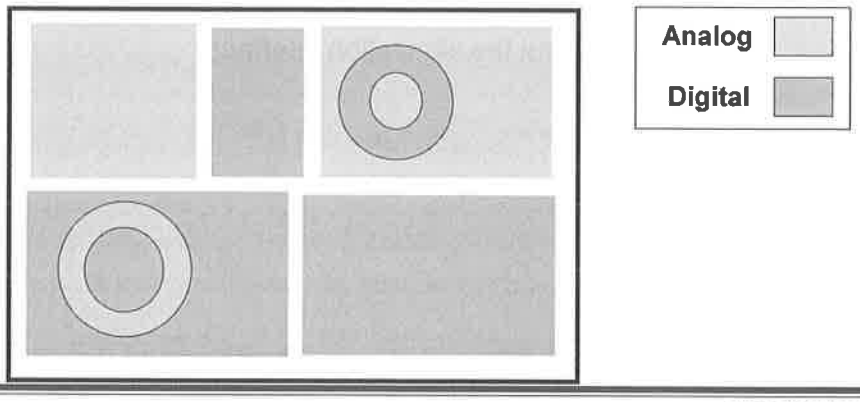
- After completing this unit, you should know
  - Discovery-AMS solution
  - Mixed-signal simulation flow with CustomSim-VCS
  - How to setup Co-SIM environment and run Co-SIM
  - How to control the simulation interface

© 2014 Synopsys, Inc. (147)

SYNOPSYS®
Predictable Success

Mixed-Signal Hierarchy

- Analog/Digital donut configuration
  - **Discovery-AMS** allows arbitrary layering of Analog and Digital blocks
  - A Verilog **parent cell** can instantiate a SPICE **child cell** or vice-versa
  - These layerings are called "donuts"



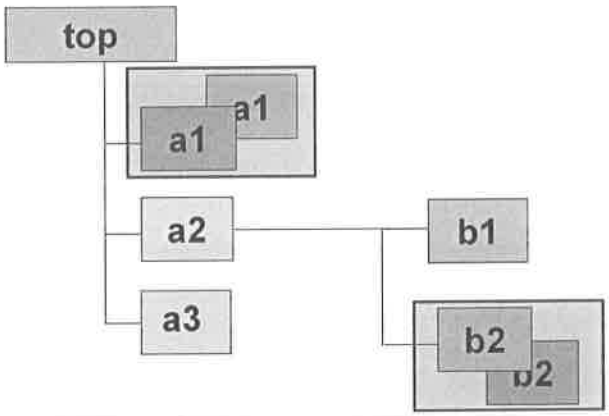
© 2014 Synopsys, Inc. (145)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Mixed-Signal Hierarchy

- **Discovery-AMS** allows cells to have multiple views
  - For example, a cell can have both a SPICE and Verilog view
  - By default, the same view as the parent cell is selected for the child cell
  - User can override this default behavior



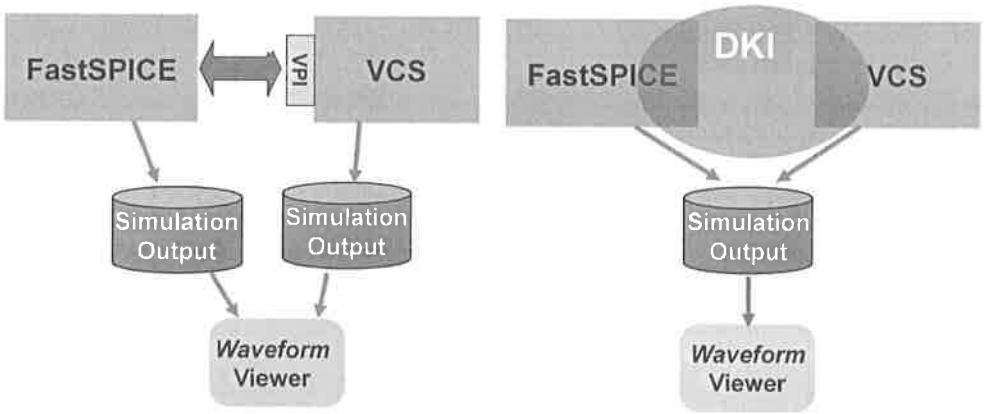
© 2014 Synopsys, Inc. (149)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

VPI and DKI: What are They?

- VPI (Verilog Procedural Interface)
- DKI (Direct Kernel Interface)



© 2014 Synopsys, Inc. (150)

SYNOPSYS
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

What is so special about DKI?

- Better performance over VPI
- Better flexibility than VPI, for example
  - XMR
  - SPICE-top, VHDL Top
  - advanced donutings etc. (any combination of Verilog/VHDL/SPICE layers in the hierarchy)
- Much easier to implement new features
  - Single output file
  - Verilog-AMS and VHDL support
  - Any future sophisticated feature (e.g. UPF power management in mixed-signal)

© 2014 Synopsys, Inc. (151)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

VPI vs. DKI features

Some prominent mixed-signal features that are supported in DKI and not in VPI:

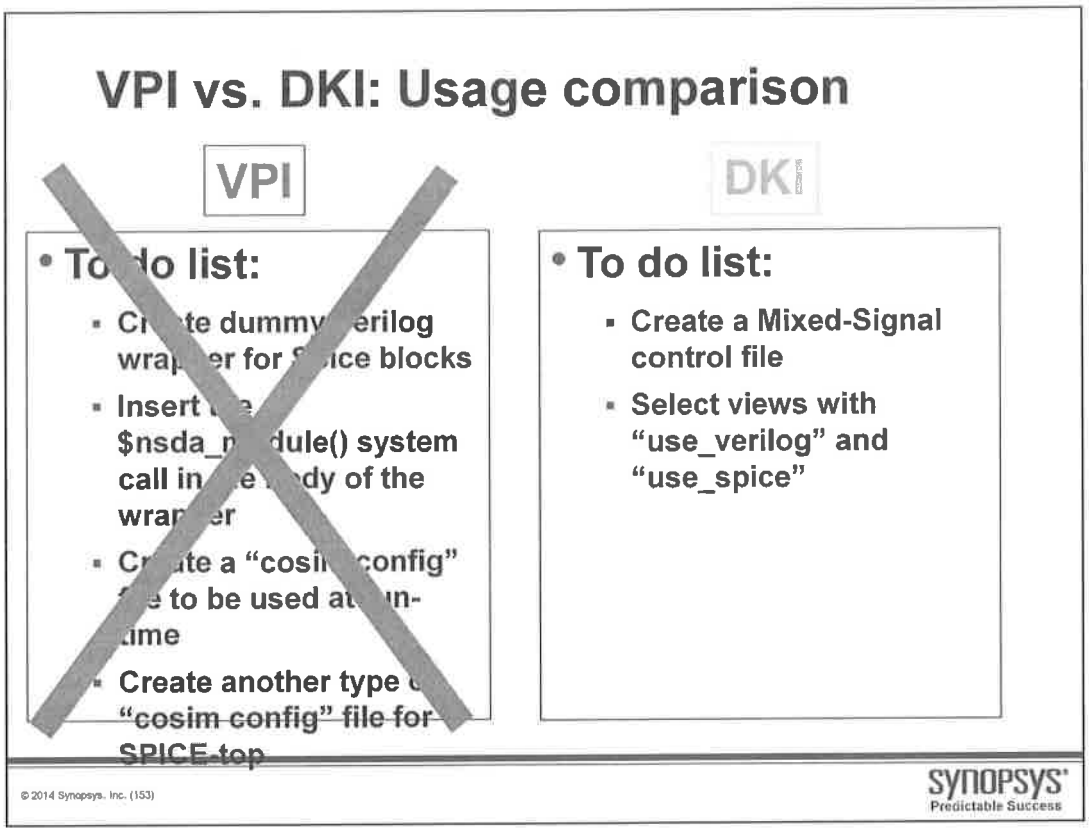
- Advanced A/D interfaces
  - ✓ Real Numbers
  - ✓ Analog <-> Digital power strength conversions
  - ✓ Dynamic supply: a2d/d2a levels following changes in VDD
  - ✓ 3D-IC (multiple SPICE design libraries) support in mixed-signal
  - ✓ Analog <-> digital port map and port mismatch resolution
  - ✓ Verilog-SPICE parameter passing

© 2014 Synopsys, Inc. (152)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

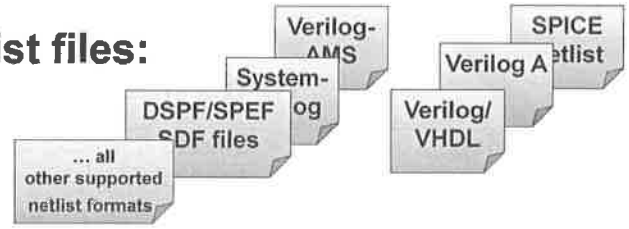
VPI vs. DKI: Usage comparison



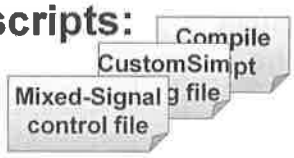
© 2014 Synopsys, Inc. (CustomSim)

How Discovery-AMS mixed-signal simulation works

Netlist files:



Setup files and scripts:



Tools and licenses:

VCS

CustomSim

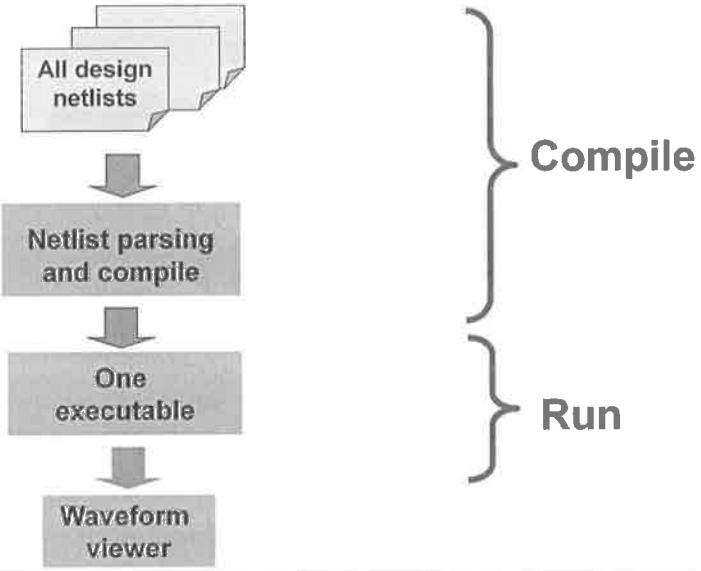
Licenses

© 2014 Synopsys, Inc. (154)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How Discovery-AMS mixed-signal simulation works

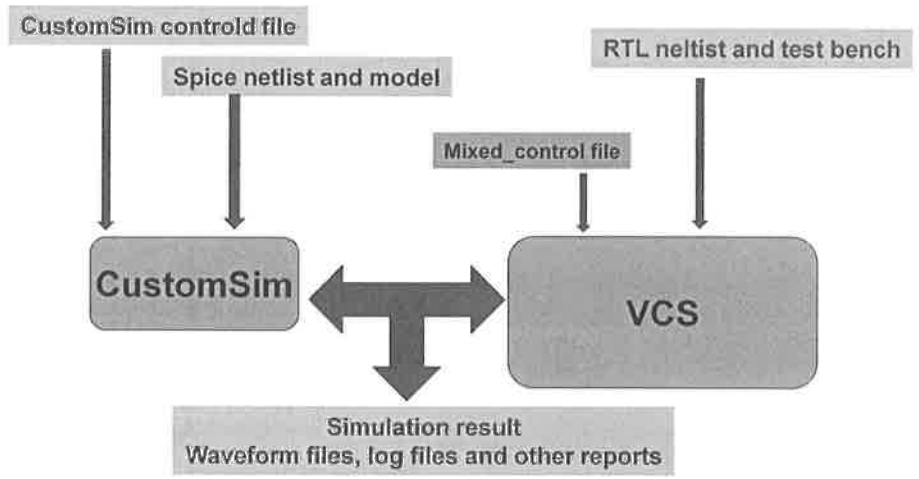


© 2014 Synopsys, Inc. (155)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

CustomSim-VCS Simulation Flow



© 2014 Synopsys, Inc. (156)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Compatible CustomSim and VCS version

CustomSim/FineSim	VCS	Platform		
J-2014.09-SP1/SP2	J-2014.12	The platforms supported in this release must be compliant with the Synopsys QSC J-Foundation which is described in the link below: http://www.synopsys.com/Support/LI/SupportPlatform/ReleaseSupport/Pages/support-foundation-j.aspx		
J-2014.09	I-2014.03-SP1	The platforms supported in this release must be compliant with the Synopsys QSC J-Foundation which is described in the link below: http://www.synopsys.com/Support/LI/SupportPlatform/ReleaseSupport/Pages/support-foundation-j.aspx		
I-2013.12-SP1/SP2	I-2014.03	**linux/amd64:** RHEL v5.7 (IA32/x86/x86_64)	RHEL v5.9, v6.2, v6.4	gcc 4.7.2
		suse32/suse64: SLES 10-SP3 (IA32/x86/x86_64)	SLES 10-SP4, 11-SP1, 11-SP2	gcc 4.7.2
		sparcOS5/sparc64: Solaris 10 (Sun Sparc)	Solaris 10	SUN Studio 12
CustomSim	**VCS**	**Platform**	**Binary Comp**	**Compiler**
I-2013.12	I-2014.03-Beta1	**linux/amd64:** RHEL v5.7 (IA32/x86/x86_64)	RHEL v5.9, v6.2, v6.4	gcc 4.7.2
		suse32/suse64: SLES 10-SP3 (IA32/x86/x86_64)	SLES 10-SP4, 11-SP1, 11-SP2	gcc 4.7.2
		sparcOS5/sparc64: Solaris 10 (Sun Sparc)	Solaris 10	SUN Studio 12
CustomSim	**VCS**	**Platform**	**Binary Comp**	**Compiler**
I-2013.12	H-2013.06-SP1	**linux/amd64:** RHEL v4.8 (IA32/x86/x86_64)	RHEL v5.5, v5.7, v6.1, v6.2	gcc 4.5.2
		suse32/suse64: SLES 10-SP3 (IA32/x86/x86_64)	SLES 10-SP4, 11-SP1, 11-SP2	gcc 4.5.2
		sparcOS5/sparc64: Solaris 10 (Sun Sparc)	Solaris 10	SUN Studio 12
CustomSim	**VCS**	**Platform**	**Binary Comp**	**Compiler**
H-2013.03-SP1/SP2	H-2013.06	**linux/amd64:** RHEL v4.8 (IA32/x86/x86_64)	RHEL v5.5, v5.7, v6.1, v6.2	gcc 4.5.2
		suse32/suse64: SLES 10-SP3 (IA32/x86/x86_64)	SLES 10-SP4, 11-SP1, 11-SP2	gcc 4.5.2
		sparcOS5/sparc64: Solaris 10 (Sun Sparc)	Solaris 10	SUN Studio 12
CustomSim	**VCS**	**Platform**	**Binary Comp**	**Compiler**
H-2013.03	G-2012.09-SP1	**linux/amd64:** RHEL v4.8 (IA32/x86/x86_64)	RHEL v5.0, v5.3, v5.5, v5.7	gcc 4.5.2
		suse32/suse64: SLES 10-SP2 (IA32/x86/x86_64)	SLES 10-SP3, 10-SP4, 11-SP1	gcc 4.5.2

© 2014 Synopsys, Inc. (157)

SYNOPSYS®
Predictable Success

Compile Script

- The compilation is done by calling "vcs"


```
vcs   file_1.v  file_2.v ...   -l vcs.log        -ad=vcdAD.init
```

  - **If "-ad" alone is used (no "=file_name"), by default the name "vcsAD.init" is assumed as the file name**
  - **The output of compilation is an executable**
    - **Called "simv" by default and can be renamed by VCS "-o output_file" switch**

© 2014 Synopsys, Inc. (158)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

How Discovery-AMS mixed-signal simulation works

- Compile script
  - Pointing to the **Mixed-Signal control** file

```
vcs   file_1.v  file_2.v ...   -l vcs.log        -ad=vcsAD.init *
```

```
choose xa|finesim  <options>;
    use_spice -cell addr4;
use_verilog –module inv;
   set bus_format _%d;
```

*Note: +ad switch is still supported but phasing out

© 2014 Synopsys, Inc. (159)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Mixed-Signal Control File

- Contains co-simulation configuration commands such as:

```
choose xa/finesim
use_verilog
use_spice
a2d
d2a
spice_top
```

- Commands end with ";"
- Comments after "//" characters

© 2014 Synopsys, Inc. (160)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

View Selection for multi-view cells

- **use_verilog:**
  - Mixed-Signal command
  - Allows picking the Verilog view of a multi-view cell
- Cell based view selection

  use_verilog –module *module_name;*

  Example:
  use_verilog –module nand21;

- Instance based view selection

  use_verilog *hierarchical_path_to_instance*

  Example:
  use_verilog top.i1.x2;

© 2014 Synopsys, Inc. (161)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

View Selection for multi-view cells

- **use_spice :**
  - Mixed-Signal command
  - Allows picking the SPICE view of a multi-view cell
- Cell based view selection

  use_spice –cell *subckt_name;*

  Example:
  use_spice –cell nand21;

- Instance based view selection

  use_spice *hierarchical_path_to_instance*

  Example:
  use_spice top.i1.x2;

© 2014 Synopsys, Inc. (182)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Using SPICE as top-level

- If top-level design (and stimulus) is in SPICE format
  - add '`spice_top`' command in vcsAD.init setup file

- VCS compile and simv as usual:

```
cmdose xa -n top.spi -c xa.cfg;
          spice_top;
use_verilog -module digital_core;
```

```
> vcs digital_core.v -ad
> simv
```

© 2014 Synopsys, Inc. (163)

SYNOPSYS® Predictable Success

A2D conversion control

- **a2d :**
  - Allows overriding the default a2d settings (e.g. low and high thresholds)
  - **a2d** loth=<lo_thrsh>[V | %] hith=<hi_thrsh>[V | %] node=<node_name>
  - Example:
    Setting the A2D high and low thresholds for node **top.i1.n1** to 0.3V and 1.3V

```
a2d loth=0.3v hith=1.3v node=top.i1.n1;
```

© 2014 Synopsys, Inc. (164)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

D2A conversion control

- **d2a :**
  - Allows overriding the default d2a settings (e.g. low and high voltages)
  - **d2a** hiv=<*high_voltage*> *lov*=<*low_voltage*> node=<node_name> rf_time=<slope_time> rise_time=<rise_time> fall_time=<fall_time> delay=<delay_time> powernet
  - Example:
    Setting the D2A high and low voltages for node **top.i1.n1** to 0.1 V and 1.7 V

```
d2a  hiv=1.7  lov=0.1  node=top.i1.n1;
```

© 2014 Synopsys, Inc. (165)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Run CustomSim-VCS simulation

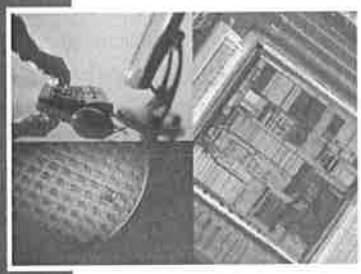
- Phase 1
  - **vcs *verilog_design_file(s) -ad [=mixed-signal_control_file]*** *[vcs options]*
  - vcs -f my_list.txt -ad=my_setup.init -o my_simv
- Phase 2
  - Run simv
  - my_simv

© 2014 Synopsys, Inc. (166)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

CustomSim Reliability Analysis



SYNOPSYS®

Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Overview of CustomSim Reliability Analysis

- Three components in CustomSim Reliability Analysis (RA):
  - Static Power net RESistance calculation (**SPRES**): Calculate effective resistance from pads to device pins
  - Power Net RA (**PWRA**): Perform IR drop and electromigration on power nets (internal power nets supported)
  - Signal Net RA (**SIGRA**): Perform electromigration on signal nets

© 2014 Synopsys, Inc. (168)

SYNOPSYS® Predictable Success

Advances in process technology and changing design styles are increasing the impact of electromigration (EM) and IR-drop effects on the performance and reliability of analog, mixed-signal, memory and custom digital IP blocks at 28nm and below.

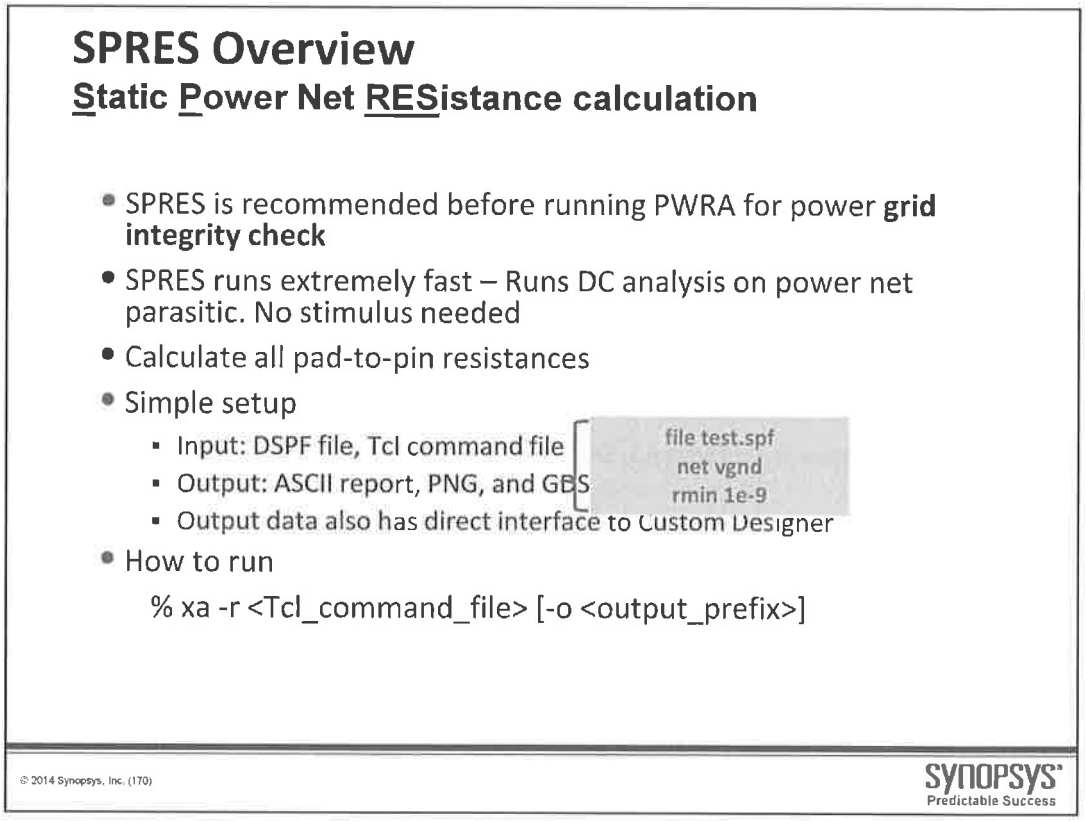
This presentation will introduce the various trends exacerbating EM and Irdrop effects as well as design and reliability analysis (RA) techniques to avoid them, and introduce Synopsys' transistor-level analysis solution, which includes Static SPRES and Dynamic PWRA/SIGRA capabilities.

Overview of CustomSim Reliability Analysis

- Three components in CustomSim Reliability Analysis (RA):
  - Static Power net RESistance calculation (**SPRES**): Calculate effective resistance from pads to device pins
  - Power Net RA (**PWRA**): Perform IR drop and electromigration on power nets (internal power nets supported)
  - Signal Net RA (**SIGRA**): Perform electromigration on signal nets

© 2014 Synopsys, Inc. (169)

SYNOPSYS® Predictable Success



SPRES stands Static Power Net RESistance.

It provides information on performing a DC analysis of power net resistance.

During the layout design of complex ICs, it can be valuable to quickly assess the parasitic effects of power net wiring, without performing a complete simulation.

To provide this capability, the SPRES option performs the complex calculation of all pad-to-pin and/or pad-to-internal instance node resistances.

More specifically, for each instance pin (*|I) in the DSPF file, an effective resistance is calculated to all pads (*|P) connected together.

To perform such calculations, no pre-layout netlist is necessary, only the net description in DSPF format is needed.

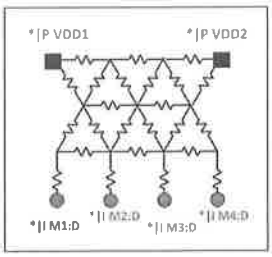
The result of the calculation is output as an ordered list of instance pins and their resistances (in the order from largest to smallest values).

In addition to a text output, a graphical display of the resistance distribution on the layout can be generated.

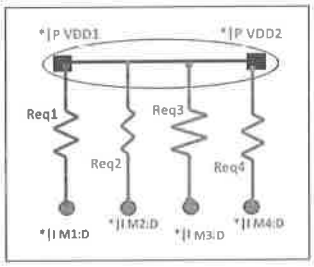
SPRES Static Power Net Resistance Calculation

SPRES Computes Pad-to-Pin Equivalent Resistance

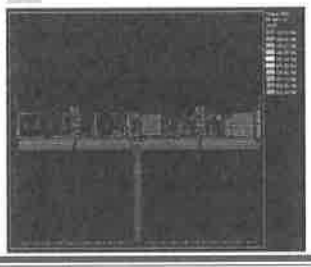
1 Reads in power-net DSPF



2 Computes pad-to-pin equivalent resistance



3 Generates visual and text reports

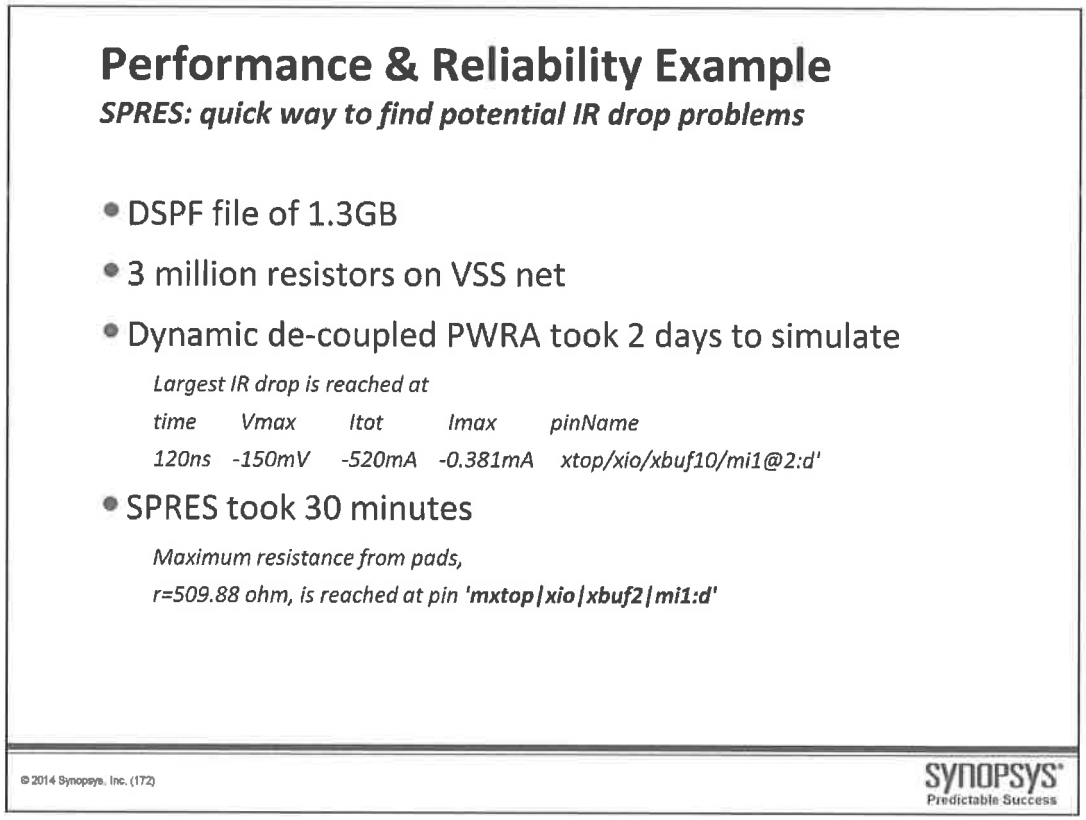


```
;;net: vgnd
;;sortby=R
;;xmin=0 ymin=2.085 xmax=121.76 ymax=301.025
;;rmin=0.01

Count   R      W/L   (x:y)           name

#1      205.9  0     (59.165:33.91)  X1/M1:D
#2      162.0  0     (58.705:33.91)  X1/M2:D
#3      107.5  0     (60.13:36.31)   X1/M3:D
#4      105.2  2.8   (74.5:17.9)     X1/M4:D
```

© 2014 Synopsys, Inc. (171)

SYNOPSYS® Predictable Success

SPRES doesn't require any vectors, or even netlist. All it needs is a DSPF with extracted power net. After reading DSPF net, SPRES computes equivalent pad-to-pin resistance for every instance pin and reports this information in various formats like ASCII, PNG, and GDSII.



Here is an example of how customer found a problem in his power net in just 30 minutes using SPRES. He could've discovered the same problem using CustomSim PWRA but it would've taken him at least 2 days.

SPRES Multi-thread

- Used to speed up simulation. It can be invoked by Tcl command or in command line option:

Tcl file:

```
file test.spf
net vgnd
rmin 1e-9
mt 2
```

Command line:

```
% xa –r file.tcl –mt2 –o
output
```

© 2014 Synopsys, Inc. (173)

SYNOPSYS® Predictable Success

Node to Node Resistance Calculation

- User specifies two nodes from a DSPF net, SPRES calculates the effective resistance between these two nodes
- A node can be:
  - Netpin (*|P)
  - Instance pin (*|I)
  - Sub-node (*|S)
- Example:

  node2node vdd vdd:1 x1/x2/m0:s

  The resistance between the subnode vdd:1 and instance pin xa/x2/m0:s will be reported in the .rlog file

© 2014 Synopsys, Inc. (174)

SYNOPSYS® Predictable Success

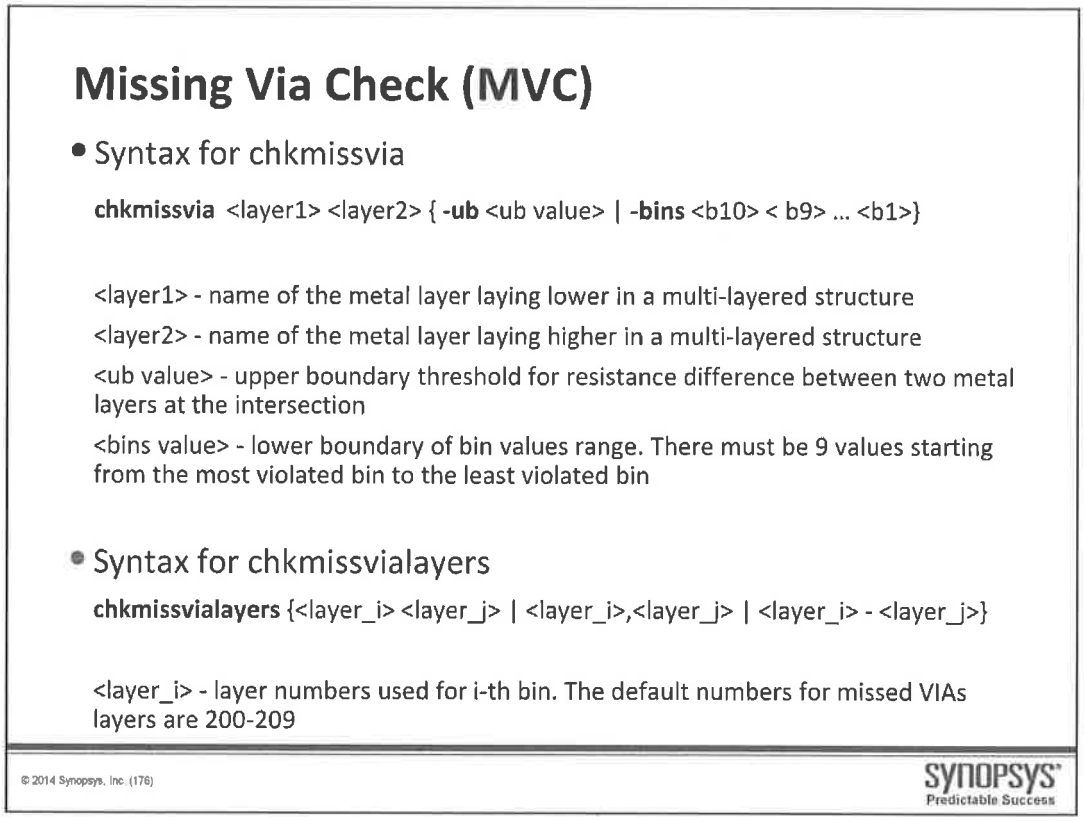
© 2014 Synopsys, Inc. (CustomSim)

Missing Via Check (MVC)

- MVC makes use of the resistance values to identify possible missing vias
  - Calculate resistance from net pin to all nodes (including sub-nodes)
  - Find all intersections between resistors belonging to pairs of metal layers specified by the user;
  - For all intersections, calculate resistance difference
  - Create shapes for the output of violation in ASCII/GDSII format if resistance difference is larger than threshold specified by the user

© 2014 Synopsys, Inc. (175)

SYNOPSYS® Predictable Success



chkmissvia

Flags missing vias between two metal layers.

Any intersection of geometries of two specified layers with a resistance value difference within the threshold ranges you specify is flagged as a violation.

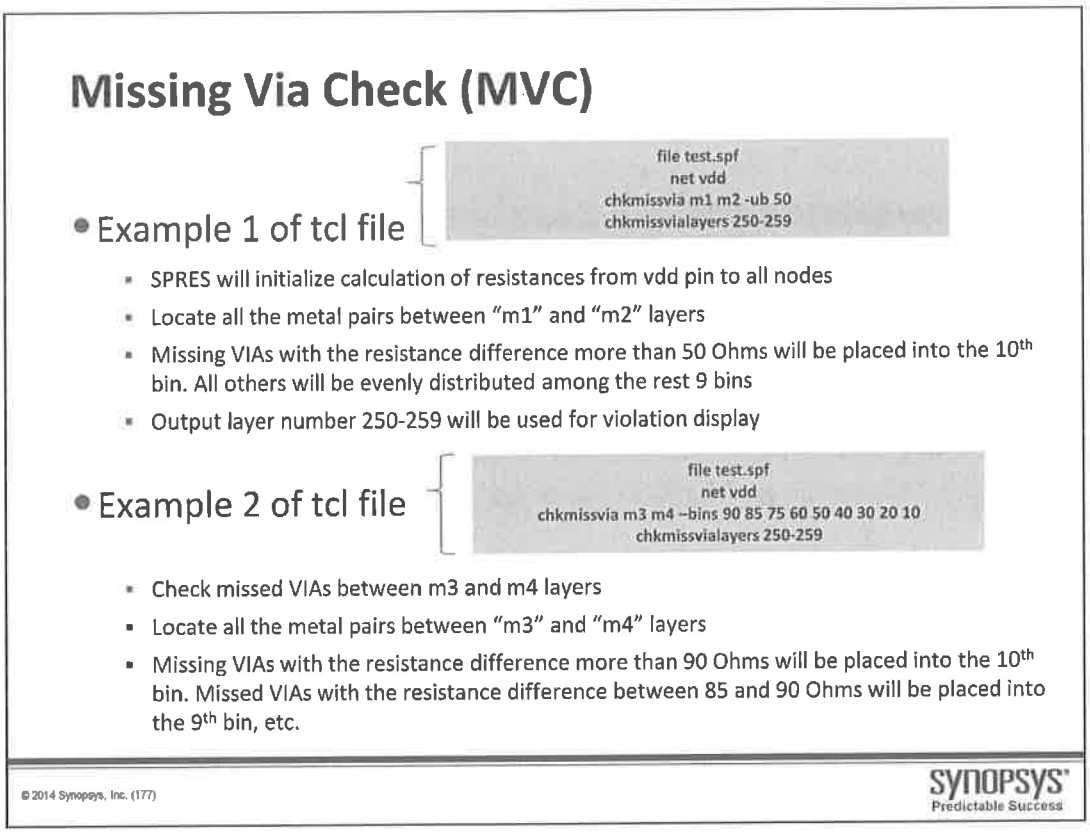
If the resistance difference is higher than the upper boundary, the violation is in blinking red.

chkmissvialayers

Specifies the via layers to be check with the chkmissvia command.

chkmissvialayers {<layer_i> <layer_j> | <layer_i>,<layer_j>

© 2014 Synopsys, Inc. (CustomSim)



The following example shows how you can specify 10 bin layers:

chkmissvialayers 200-209 (this could be used as default value if chkmissvialayers command is not specified).

chkmissvialayers 250 251 252 253 254 255 256 257 258 259

For the Tcl file example 1:

SPRES only analyzes the vdd net. SPRES initializes the calculation of resistances from the vdd pin to all nodes and finds missed vias between the m1 and m2 layers.

Missed vias with a resistance difference more than 50 Ohms are placed into the 10th bin. All others are evenly distributed among the rest of the nine bins.

Advantage of Using SPRES

- Very easy to set up
- Doesn't require any vectors
- Very fast
- Complements PWRA dynamic feature

© 2014 Synopsys, Inc. (178)

SYNOPSYS®
Predictable Success

Overview of CustomSim Reliability Analysis

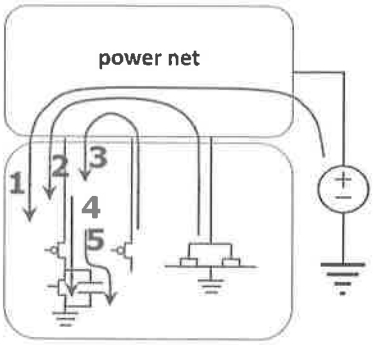
- Three components in CustomSim Reliability Analysis (RA):
  - Static Power net RESistance calculation (**SPRES**): Calculate effective resistance from pads to device pins
  - Power Net RA (**PWRA**): Perform IR drop and electromigration on power nets (internal power nets supported)
  - Signal Net RA (**SIGRA**): Perform electromigration on signal nets

© 2014 Synopsys, Inc. (179)

SYNOPSYS® Predictable Success

"I" stands for "Current"

- Analysis accuracy depends on accuracy of the current (V=I*R)
- How many types of currents are there?
  - Supply current
  - Current from on-chip decoupling capacitors
  - Diffusion charge stored in neighboring transistors
  - Crowbar and leakage currents
  - Load current



© 2014 Synopsys, Inc. (180)

SYNOPSYS® Predictable Success

PWRA option gives information on performing IR drop and electro-migration analysis of power nets

Power nodes in simulated circuits are considered to have constant voltage. In reality, resistive networks between the power node pad and the devices always exist.

When devices draw current from the power source, the voltage or IR drop appears on the device terminals connected to power nodes. If significant, this IR drop may affect the performance of the circuit; such as the delay value. The IR drop value depends on the value of the resistive path from the power pad, to the device connection. The current flowing through each power net RC network resistor may cause electro-migration (EM) problems in the circuit if the current density is greater than a specified threshold.

Challenges of dynamic IR drop verification

- Capacity and performance
  - Ability to handle millions of transistors and parasitic resistors
  - Full chip transistor level simulation without reduction is impossible
  - Power-net reduction impacts the accuracy of the IR drop
  - Traditional partitioning techniques are impossible because power-nets are global coupling nets

© 2014 Synopsys, Inc (181)

SYNOPSYS® Predictable Success

Overview of PWRA and SIGRA

- True dynamic simulation and analysis
  - Not static or statistical
- Take advantage of performance, capacity, and accuracy benefits of CustomSim transient simulation
- Apply instantaneous transistor currents for the best accuracy
  - No characterization, estimation or averaging
  - Handle temperature and process variations
- Analyze influence of IR drop on design performance
- Identify power net IR drop hot spots
- Locate electromigration issues

© 2014 Synopsys, Inc. (182)

SYNOPSYS® Predictable Success

The CustomSim tool provides power net reliability analysis (PWRA) to address these problems.

■ Uses a DSPF file representation of the power net to calculate all the pad-to-pin resistances. The number of resistors in the power nets of modern circuits makes it impossible to simulate the circuit with back-annotated power nets as is, so power net RC reduction must be applied.

■ Accepts power net RC networks by back-annotating the resistors and capacitors from a DSPF file.

■ Simulates the circuit with the reduced power net RC network added to the circuit devices, providing dynamic IR drop on circuit behavior.

■ Calculates the maximum IR drop at each connection between the power net and circuit devices.

■ Calculates the maximum, average, or RMS current density through each power net resistor and compares it with the threshold for the purpose of electro-migration analysis.

■ Disp

Signal net reliability analysis (SIGRA) is similar to the power net reliability analysis Reliability Analysis (PWRA).

The analysis consists of two phases:

■ Phase I: The CustomSim tool simulates a circuit with back-annotated signal

nets and saves currents flowing through the net ports.

- Phase II: The saved currents are injected into the non-reduced signal nets and the current through each individual resistor of the net is computed.

Only electro-migration (EM) analysis is performed for signal nets so that the current density for each resistor is calculated, which then may be displayed in the form of a violation map in a GDSII file. The spots where the current density exceeds the specified threshold value are highlighted. If you use the set_ra_net_type command, only the nets or net pattern specified in this command will be considered for RA.lays the violation map highlighting locations where IR drop or current density exceeds the threshold.

© 2014 Synopsys, Inc. (CustomSim)

CustomSim RA Flow

Take two-phase approach

- First phase: Run Transient Simulation → to Collect Device Current of power and signal nets
- Second phase: Simulate the Parasitic Network by Injecting Device Current Collected from phase one
- Advantage of two-phase approach:
  - Allow RC reduction for better first phase simulation throughput.
  - Limit amount of parasitic in transient simulation → handle designs with large amount of RC's
  - Allow efficient What-If analysis
  - ...

© 2014 Synopsys, Inc. (183)

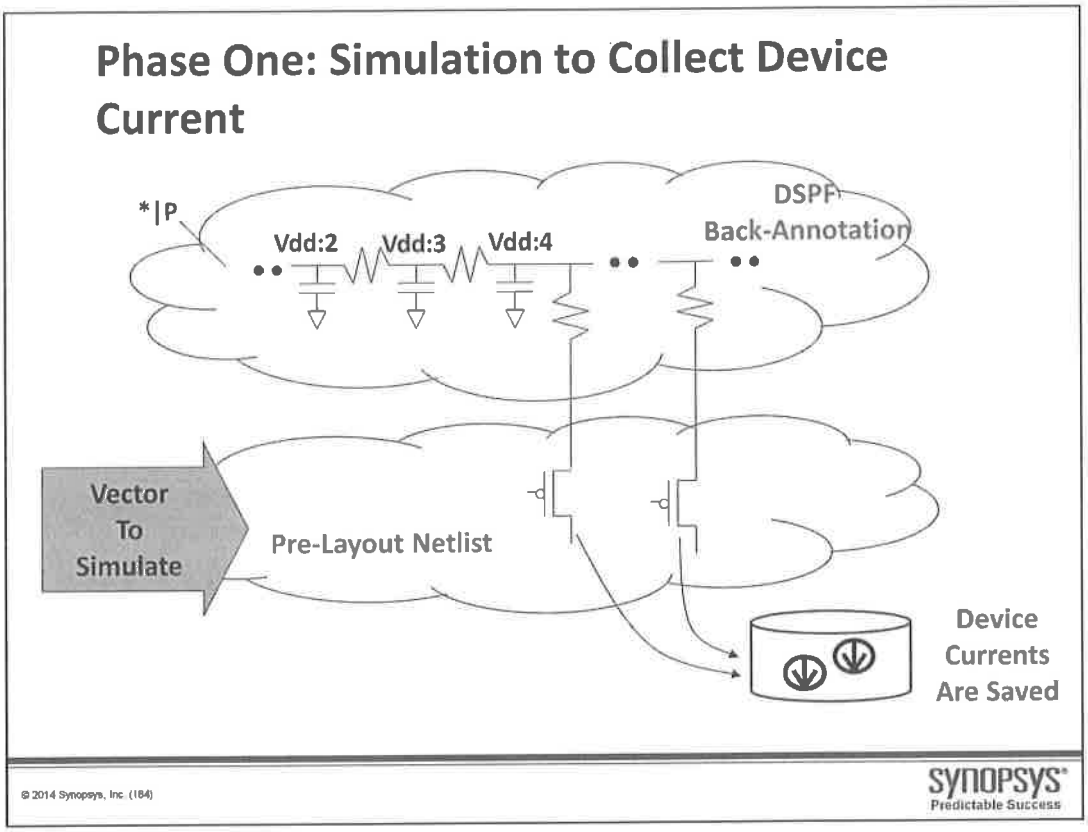
SYNOPSYS® Predictable Success

The CustomSim tool performs PWRA in two phases:

■ Phase I: A reduced power net is back-annotated to the circuit and transient simulation is performed. This simulation determines the impact of dynamic voltage drop on circuit behavior and captures degradation of delay or characteristics induced by IR drop.

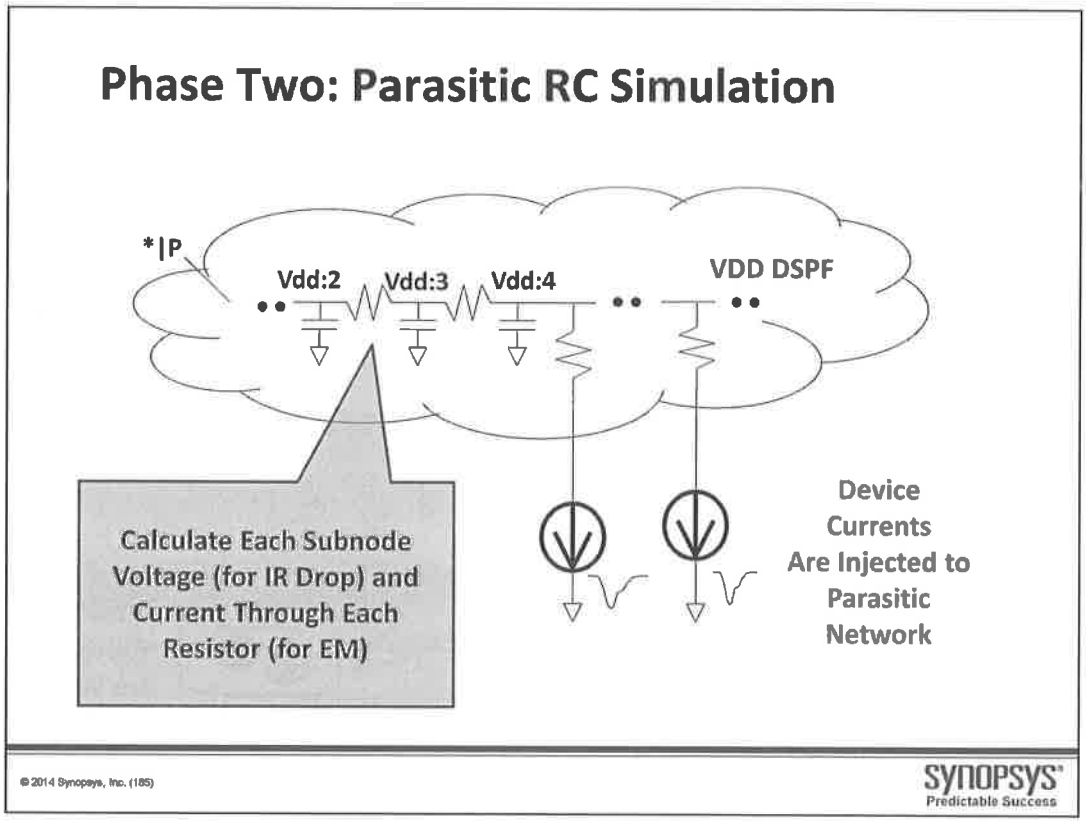
■ Phase II: Node voltage and branch current density values are computed for the raw, unreduced power net and graphical visualization data is created.

© 2014 Synopsys, Inc. (CustomSim)



The figure shows an overview of the IR/EM flow, which consists of:

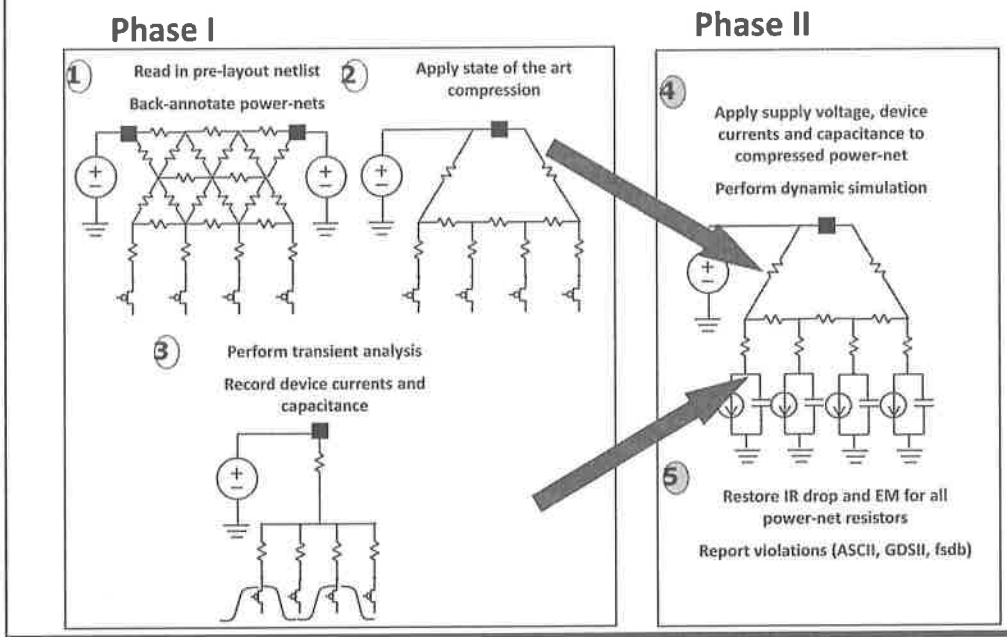
■ Phase I, in which a simulation is performed to capture the currents of devices connected to the power nets. The current information is stored in the .rasim file for phase II use.



Phase II, in which the power nets (which may include internal supply net as well if you define the internal supply net as power net) are simulated and the stored currents are injected into the power net and the IR drop and resistor current values are calculated.

© 2014 Synopsys, Inc. (CustomSim)

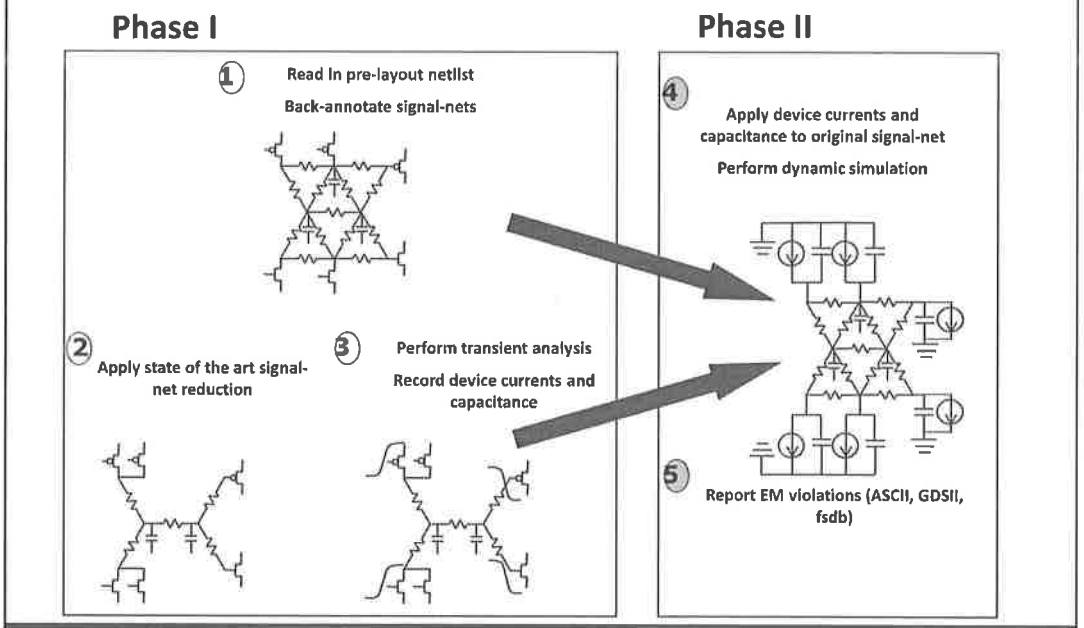
Power-net Reliability Analysis



© 2014 Synopsys, Inc. (186)

SYNOPSYS®
Predictable Success

Signal-net Reliability Analysis



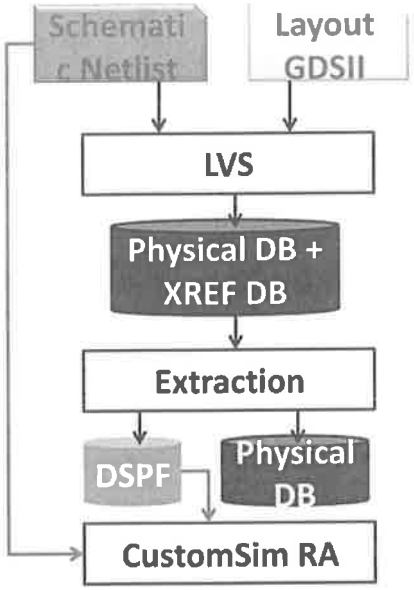
© 2014 Synopsys, Inc. (187)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

PWRA and SIGRA Require Back-Annotation Flow

- Both a pre-layout netlist and DSPF are required
- Names (instance/device/node name) need to match between Pre-layout and DSPF netlists →
  - Recommend to use LVS netlist as the pre-layout netlist
  - Or, an IDEAL netlist from extractor can be used
  - Option to use XBA (Extended BA) if flow does not allow back-annotation(BA)

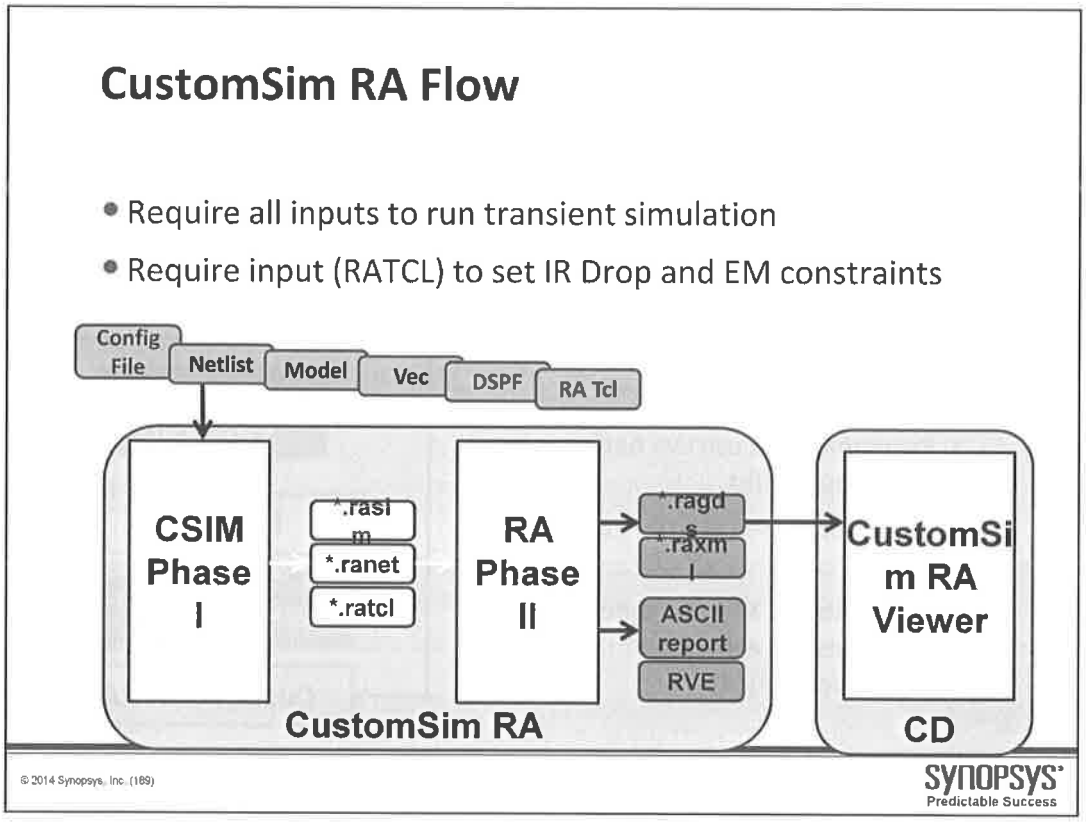


© 2014 Synopsys, Inc. (188)

SYNOPSYS® Predictable Success

In the CustomSim-RA flow, back-annotation (BA) is a prerequisite setup.

However, the RA analysis is independent of the BA setup. This lets you define the RA nets to maximize the performance/accuracy tradeoff.



Setting Up the IR/EM Flow

To set up the IR/EM flow, do the following steps:

1. Prepare your simulation data (vector/model file/BA flow) to enable a signal net simulation.

2. Insert or create the RA global command in your existing CustomSim command file. For example:

set_ra_option -sigra 1 -ratcl ratcl_file

sct_ra_net_type -signet *signalnet_name*

• You can specify more than one ratcl file with the -ratcl command argument.

For example: set_ra_option -sigra 1 **-ratcl *ratcl_file1 ratacl_file2***

• The CustomSim tool has the flexibility to allow you to specify multiple set_ra_net_type commands for signal nets, or provide one or more signal nets in a single set_ra_net_type command.

3. Launch the CustomSim RA simulation with the configuration file that contains the RA commands.

For example: xa top.sp -c cmd -o output/xa_ra

4. Review the RA analysis results.

xa -ra <output file name>-<power net name1> .. <output file name>-<power net nameN>

© 2014 Synopsys, Inc. (CustomSim)

To start the Phase II for the VDD net only, use the following syntax:

xa -ra test-vdd

PWRA reads the .ratcl file for the correspondent power net and performs the specified types of analysis in Phase II. The following analysis types may be performed during the Phase II:

- 1|vmax: Maximum node IR drop.
- 2|imax: Peak resistor current.
- 3|irms: RMS resistor current.
- 4|iabs: Average absolute magnitude of resistor current.
- 5|iavg: Average resistor current.

During the Phase II, the currents stored from Phase I are injected into the entire power net. For each time interval equal to the RATAU, the following are recalculated and updated:

- Voltages at every power net node
- Currents through each power net resistor

If vmax analysis is specified, CustomSim PWRA generates the following files from Phase II:

- *.ralog: VDD log file
- *.radb: VDD net results
- *_vmax.ragds: vmax analysis violation map of the VDD net in GDSII format.

Ease-of-Use: CustomSim RA

4 groups of RA-specific commands

- ***set_ra_option***
  - Control global options for RA simulation
- ***set_ra_net***
  - Select nets to be simulated by 2nd phase simulation (filtering/selective)
- ***set_ra_net_type,***
  - Define net type, either a power net or signal net
- ***set_ra_pwnet_option***
  - Define internal power net, source node, source node voltage reference

© 2014 Synopsys, Inc. (190)

SYNOPSYS® Predictable Success

There are two ways to select RA nets for analysis.

You use the set_ra_net command to select the nets that go into RA analysis. By default the CustomSim tool selects all nets for RA analysis.

When selecting RA nets, you can specify both explicit name or use a wildcard character. The net name is based on BA net name, not the schematic name.

You can also apply multiple set_ra_net commands. If the net names are matched by both -selectnet/-skipnet arguments, -skipnet has a higher

priority. The RA flow always honors your setup when you specify an explicit net name.

set_ra_option Command

Option Syntax	Description
[*-pwra* 1/0]	Set to "1" to enable PWRA. Default: 0
[***-sigra*** 1/0]	Set to "1" to enable SIGRA. Default: 0
[*-ratcl* RA_Tcl_filename]	Define RA Tcl file name(s). Default: none
[***-ratau*** RA_timestep]	Define RA simulation timestep. Default: 10ps
[*-twindow* RA_sim_windows]	Define RA simulation window(s). Default: Transient window
[***-rap2auto*** 1/0]	Default is to auto start 2nd phase. Default: 1
[*-waveform_split* file_size]	Control maximum *.rasim file size. Default: 2GB

Example
set_ra_option –pwra 1 –sigra 1 –ratcl ra.tcl –twindow 1n 2n 5n 6n
Enables PWRA and SIGRA. Set "ra.tcl" as RA Tcl file. Set 2nd phase simulation windows to [1n 2n] [5n 6n]

© 2014 Synopsys, Inc. (191)

Predictable Success

set_ra_option

Specifies options for RA analysis.

In the CustomSim RA flow, phase II is run automatically after the completion of the phase I simulation. If you want to stop the simulation after the completion of the phase I, you can use the following RA command.

set_ra_option -pwra 1 -ratcl ra.tcl **-rap2auto 0**

set_ra_net Command

Option Syntax	Description
[*-selectnet* list_of_nets]	Enter net names explicitly for selective net RA simulation. Support wildcard character (*). Default: "*"
[***-skipnet*** list_of_nets]	Enter net names to be skipped for RA simulation. Default: none

- Net name specification takes the form of SPF net names! Use SPF hierarchy divider character!
- Multiple **set_ra_net** commands are allowed
- **-skipnet** option has higher precedence when there is a net name conflict with **-selectnet**
- Explicit net name specification take precedence over wildcard specification

Example
set_ra_net –selectnet * -skipnet vdd x1/x1/net*
Select all but net names match "vdd" and "x1/x1/net*" for RA simulation. Note: "/" is the hierarchical divider used in SPF

© 2014 Synopsys, Inc. (192)

SYNOPSYS® Predictable Success

set_ra_net

Specifies the nets to consider for RA.

© 2014 Synopsys, Inc. (CustomSim)

set_ra_net_type Command

Option Syntax	Description
[*-pwnet* list_of_nets]	Define a list of power nets. Default: Auto detection
[***-signet*** list_of_nets]	Define a list of signal nets for SIGRA analysis.

- Net name specification takes the form of SPF net names. Use SPF hierarchy divider character.
- Multiple **set_ra_net_type** commands are allowed
- **-pwnet** option has higher precedence when there is net name conflict with **-signet**
- Explicit net name specification take precedence over wildcard specification

Example

```
set_ra_option -pwra 1 -sigra 1 -ratcl ra.tcl
set_ra_net_type -pwnet vdd vss vddhd
```

"vdd", "vss", "vddhd" are chosen for PWRA analysis. All other nets are considered signal nets for signal net analysis

© 2014 Synopsys, Inc. (193)

SYNOPSYS® Predictable Success

The set_ra_net_type command selects the type of net that goes into the RA flow. There are only two types of nets that can go into RA: power nets or signal nets.

If a net is selected as both power and signal, then the power selection takes precedence.

When you do not specify a set_ra_net_type command, the CustomSim tool uses internal ID methods to identify net types. When you specify one or more set_ra_net_type commands, the automatic identification is disabled, allowing you to control net types manually.

You can specify multiple set_ra_net_type commands. Only nets that are selected by -pwnet or -signet are selected for RA (set_ra_option -pwra 1 triggers PWRA and set_ra_option -sigra 1 triggers SIGRA).

set_ra_pwnet_option Command

Option Syntax	Description
[*-net* net_name]	Specify internal power net name or net pattern (net name with wildcard)
[*-src* source_net_name]	Specify netlist name of the corresponding source net (for power switch design). Required.
[*-vref* vref_value]	Specify the reference voltage value of the corresponding external power net. This is the reference voltage for internal power net IR drop. If not specified, tool will auto detect. Recommended.

- Apply this command for internal power nets PWRA analysis
- Net name specification takes the form of SPF net names. Use SPF hierarchy divider character.
- Multiple **set_ra_pwnet_option** commands are allowed
- One command for net name/pattern
- Command syntax is effective after 2013.03-ENG2 release (Apr, 2013)

© 2014 Synopsys, Inc. (194)

SYNOPSYS® Predictable Success

set_ra_pwnet_option

Lets you define the reference voltage and source external power supply node for the internal power net.

By default the CustomSim tool automatically identifies internal power nodes and selects appropriate reference voltages and source nodes for them. An output file .pw_driver is created that contains information about internal power nets and their drivers. You can override the automatic identification of vref and source_pwnet_node with the set_ra_pwnet_option command. If you specify source_pwnet_node, the automatic driver ID selects only those drivers that connect to the specified node.

Coupled and De-coupled Simulation

- Coupled and De-coupled simulation setting is control by back-annotation command
- De-coupled simulation on VDD, use:

```
load_ba_file –file dspf.file  –cnet vdd
```

- For coupled simulation, use:

```
load_ba_file –file dspf.file
```

© 2014 Synopsys, Inc. (195)

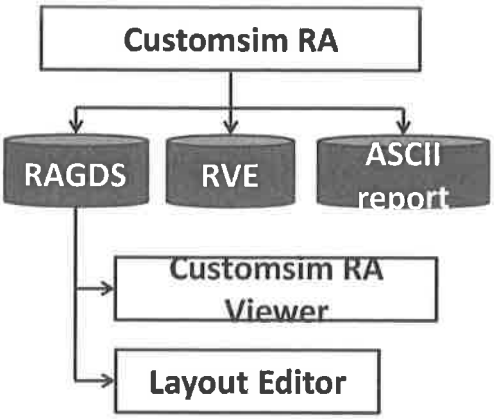
SYNOPSYS® Predictable Success

Load_ba_file: specifies a post layout back-annotation file

-cnet option: specifies for the nets to use lumped capacitance back-annotation.

The lumped capacitance value uses the net capacitance value in the *|NET line. All the nets not specified by this argument use full RC back-annotation (assuming the net itself contains RCs).

View and Debug RA Violation



- Graphical Viewing and Debug
  - Step through RA violations using CustomSim RA Viewer
  - CustomSim RA Viewer is based on Custom Designer
  - Or, RAGDS can be stream into any layout editor
  - Or, viewing violation using RVE
- ASCII Report Debug
  - Configurable ASCII report
  - Identify violations based on XY locations

© 2014 Synopsys, Inc. (196)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

RA Requirement to Set Up Viewer

- Set cdesigner as the target viewer in hsimratcl file

```
raviewer cdesigner
```

- To overlay violations to the original design (optional)

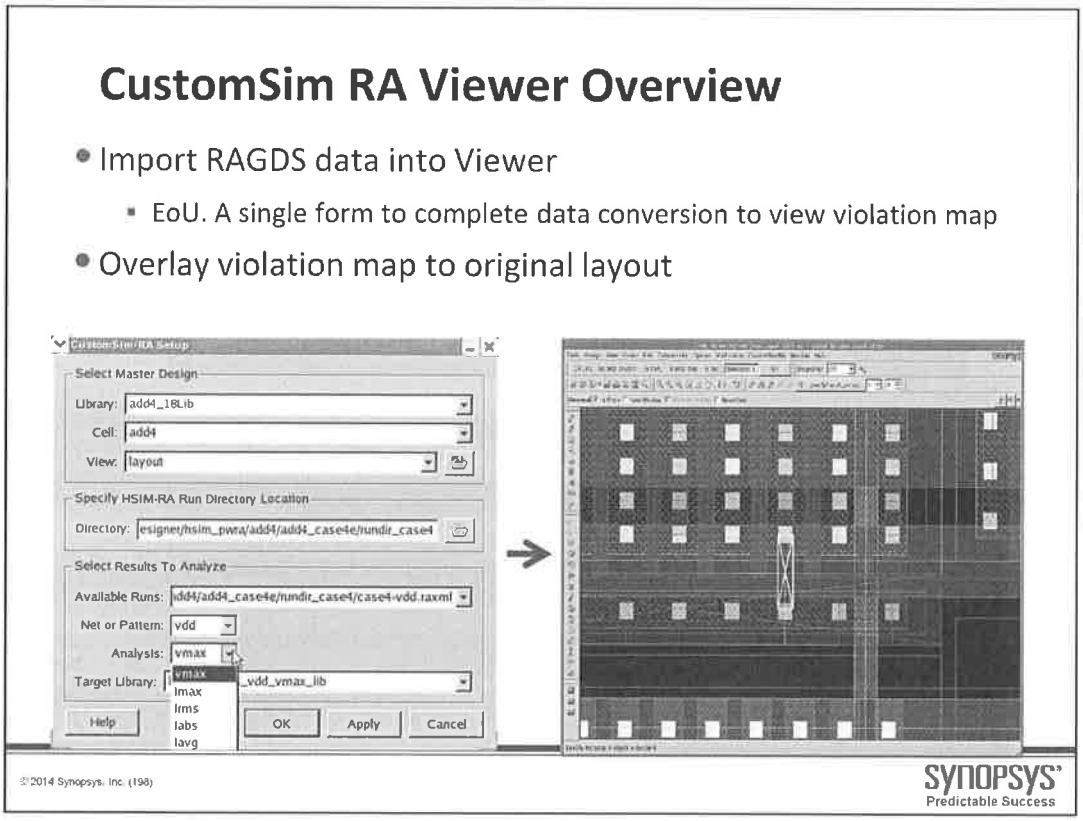
```
ragds -reflib DemoPLL -refcell vco -legend
```

- For more accurate violation display, recommended to use the following StarRC command:

```
EXTRA_GEOMETRY_INFO:  RES NODE
R1 VDD:1 VDD:4 0.443923 $l=0.092
$w=0.424 $lvl=21 $llx=6.868
$lly=26.113 $urx=6.96 $ury=26.537
$dir=0
```

© 2014 Synopsys, Inc. (197)

SYNOPSYS® Predictable Success



Loading SPRES GDSII Files into Layout Editors

You can import SPRES GDSII data in the following layout editors:

- Laker/ Synopsys Galaxy Custom Designer
- Cadence Virtuoso

Violation Map Visualization

Visualization marks different values of IR drop or EM current density with different colors producing a violation map. More precisely, a GDSII file is generated containing geometry elements, e.g. rectangles, tracks, etc., with

different colors ranging from Forest Green for minor violations to Blinking Red for values exceeding a specified threshold.

Generating a Violation Map

Assuming that ra vmax is specified in the .ratcl command file and the maximum IR drop (vmax) had been calculated and stored into .radb files during the power net analysis phase, a violation map showing maximum IR drop (vmax) can be generated using the following syntax:

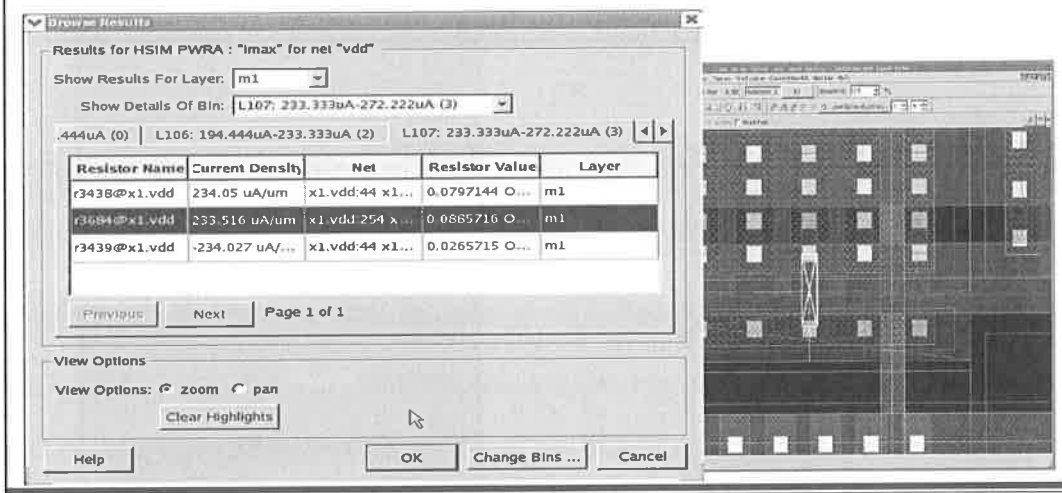
xa -ralayout vmax test-vdd

The resulting test-vdd_vmax.ragds file is created. The layout is generated in the vdd_vmax.ragds file.

© 2014 Synopsys, Inc. (CustomSim)

CustomSim RA Viewer Overview (cont.)

- Step through violations using Result Browser

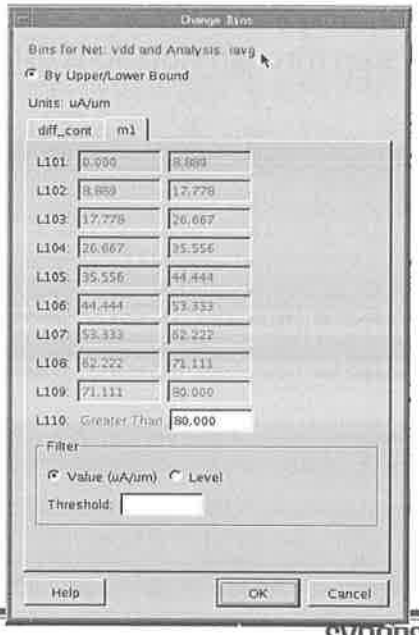


© 2014 Synopsys, Inc. (199)

SYNOPSYS®
Predictable Success

CustomSim RA Viewer Overview (cont.)

- **Re-binning(change bins)**
  - **Generate a new violation map with a new set of user-defined thresholds**



© 2014 Synopsys, Inc. (200)

SYNOPSYS®
Predictable Success

Summary

CustomSim Power-net Reliability Analysis (PWRA) and Signal-net Reliability Analysis (SIGRA) deliver very accurate dynamic reliability solution

- True dynamic IR-drop (power only) and EM
- Minimizes over-design
- Let's designers detect and fix reliability problems before tape out

SPRES provides a fast sanity check for Power Rail reliability issues

© 2014 Synopsys, Inc. (201)

SYNOPSYS® Predictable Success

Providing realistic violation maps (meaning less over-designing the power network); and accurately capturing timing skews related to IR-drop

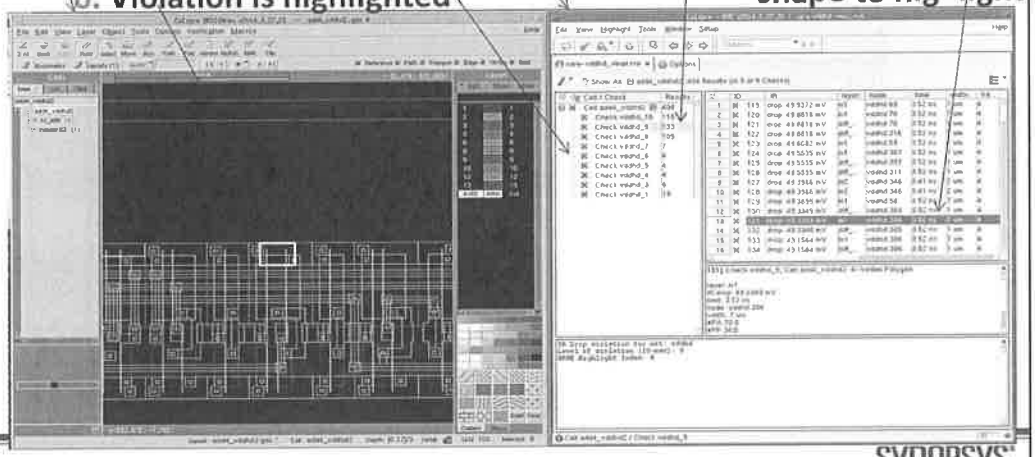
Debug CustomSim RA Result Using RVE

Appendix

© 2014 Synopsys, Inc. (CustomSim)

Debug CustomSim RA Violations in RVE

1. Open Calibre DESIGNrev
2. Open Calibre RVE
3. RVE data has 10 bins, just like RA GDSII's
4. Select a bin for highlighting
5. Select a violation shape to highlight
6. Violation is highlighted

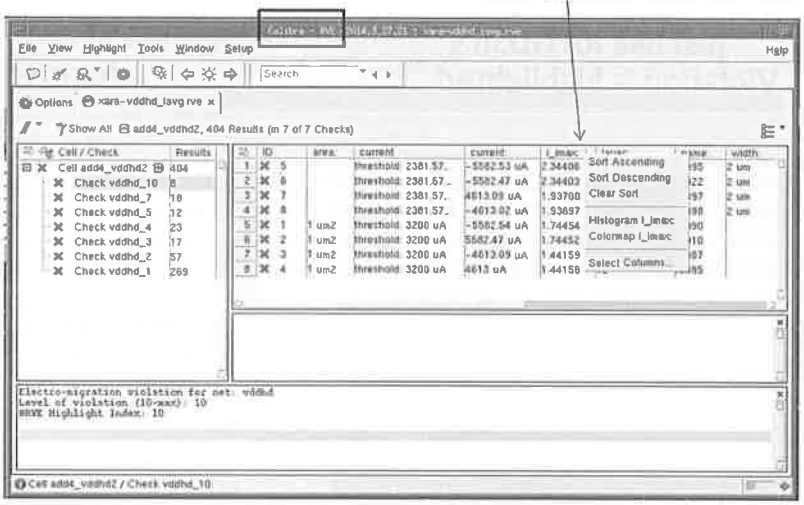


© 2014 Synopsys, Inc. (203)

SYNOPSYS
Predictable Success

Sorting of Each Column in RVE

Sorting of each column can be Controlled within RVE



© 2014 Synopsys, Inc. (204)

SYNOPSYS® Predictable Success

Filtering Of RVE Output Data

Available RA Commands control output filtering

rvemapi

```
rvemapi{max | avg | rms | abs} [-l] [<default_val>]
   <val_1>@<layer_1> <val_2>@<layer_2> …
```

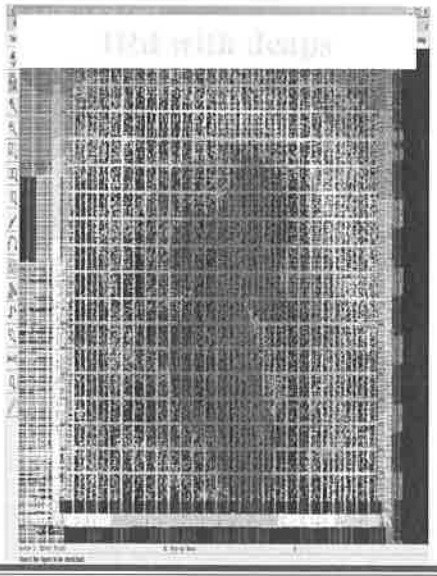
`rvemapi` <max|avg|rms|abs> specifies the filter for the number of resistors to be stored in the RVE output file as a result of any of the following EM analysis: imax, irms, iavg, iabs. This command supports two types of filtering: by by level of violation or value of current density. To request filtering by level of violation -l parameter must precede all other parameters. In this case < default_val> and all other <val_i> must be integers specifying the number of most violated levels to be outputted into RVE file. In case of filtering by value <default_val> and all <val_i> the value must be in uA/um specifying minimum value of resistor current density to be outputted into RVE file.

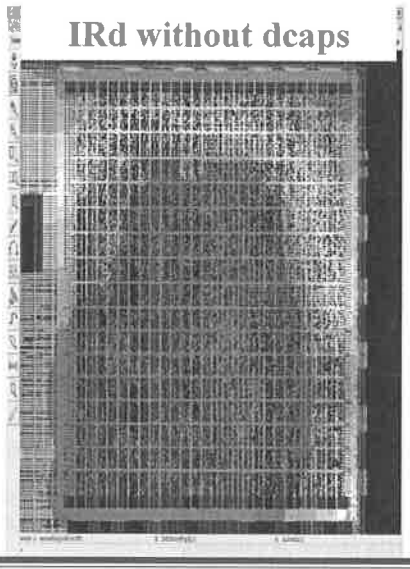
Each layer can have its own <val>@<layer> threshold specification, where <layer> is layer name/number. If <default_val> is specified, layers without individual specifications are treated as they have this <default_val> value. If <default_val> is not specified and a layer does not have its own specification, all resistors on this layer are listed in the output file. If rvemapi is skipped, PWRA stores all resistors.

© 2014 Synopsys, Inc. (205)

SYNOPSYS® Predictable Success

IRd in GDS



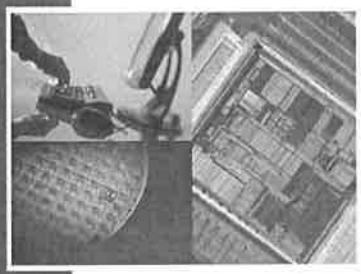


© 2014 Synopsys, Inc. (206)

SYNOPSYS®
Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Introduction to Synopsys China Support Center



Xiang Wei-Ping

Support Center Manager

SYNOPSYS®

Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Agenda

- China Support Center General Information
- Customer Support Models
- Summary

© 2014 Synopsys, Inc. (208)

SYNOPSYS®
Predictable Success

Our Mission

- Build Customer Loyalty
- Provide 90% of the Reactive Support for the Chinese Speaker Users
- Drive Change To Improve Synopsys Tools
- Enhance Synopsys Value To Customers

© 2014 Synopsys, Inc. (209)

SYNOPSYS®
Predictable Success

China Support Center Functions

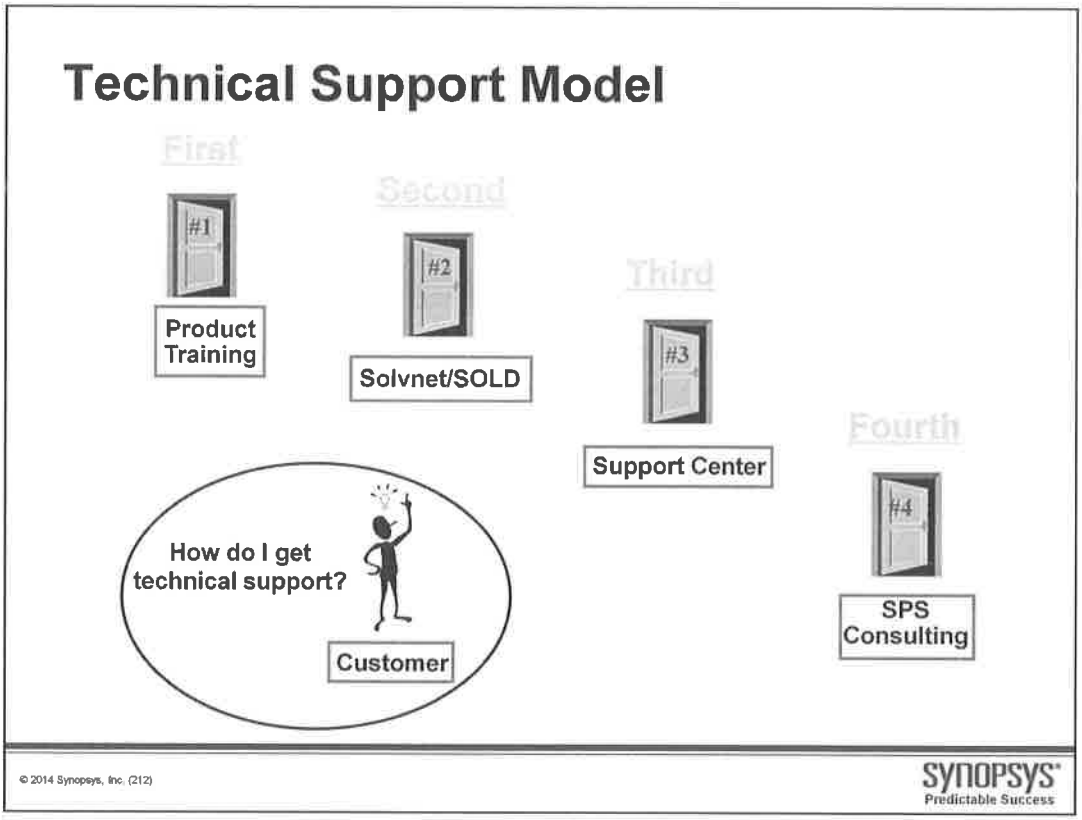
- Proactive Support
  - On-site installations for the new users
  - Difficult issues resolving on-site
- Reactive Support
  - Online support via SolvNet 'Enter a Call'
  - E-mail support via prchelp@synopsys.com
  - Hotline technical support via toll free number: 400-670-7677
- Customer Education
  - Training calendar at: http://asiapac.synopsys.com/china/training/training.html

© 2014 Synopsys, Inc. (210)

SYNOPSYS® Predictable Success

The Synopsys Support Model is a four tiered system. The model incorporates many levels of services to the end customer. These services include access to SolvNET; the Synopsys suite of tools which include our on-line Knowledgebase, and Product documentation, phone/web/ email access to the Support Center and in-direct access to the product CAE and R&D teams. Synopsys Service organization emphasizes teaching the customer to troubleshoot his own problems. Our philosophy is educating the customer is as valuable as solving the customer's issue. Above is the support model, we encourage our customers to use in working with Synopsys.

© 2014 Synopsys, Inc. (CustomSim)



Commendations:

•From Agere DTM (Adam Rosenberg):

I just wanted to pass on some positive feedback from the ACs supporting Front End Implementation at Agere. Bob Wiegand has told me that he would be absolutely buried with issues if it wasn't for the help that your team, specifically Anagha, has provided. Both he and the customer are very pleased with the level of support that they are getting and it has freed Bob up to work on competitive engagements.

Thank you, Adam R.

•From nVidia Design Engineer:

I am writing to make you aware of the outstanding service that we are receiving from Eva Coon. Eva is taking care of nearly all of my Support Center calls, and she has helped me and NVIDIA a great deal. [Three paragraphs of examples…]

This is just a short note and I'm sure I've missed mentioning some of her contributions. However, I am delighted to have her supporting NVIDIA and look forward to a continuing productive relationship. I wanted you to be aware of her great service and hope that she gets all the credit she deserves.

Sincerely, John Busco

NVIDIA CORPORATION

•From nVidia DTM (Chris Lyon):

I know that I speak for all of the implementation ACs when I say that we have definitely felt the benefit of Eva's presence at nVidia. She is able to support John and his engineers very well and goes above and beyond the call of duty! Thank you for allowing one of your best engineers to help us out. -Chris Lyon

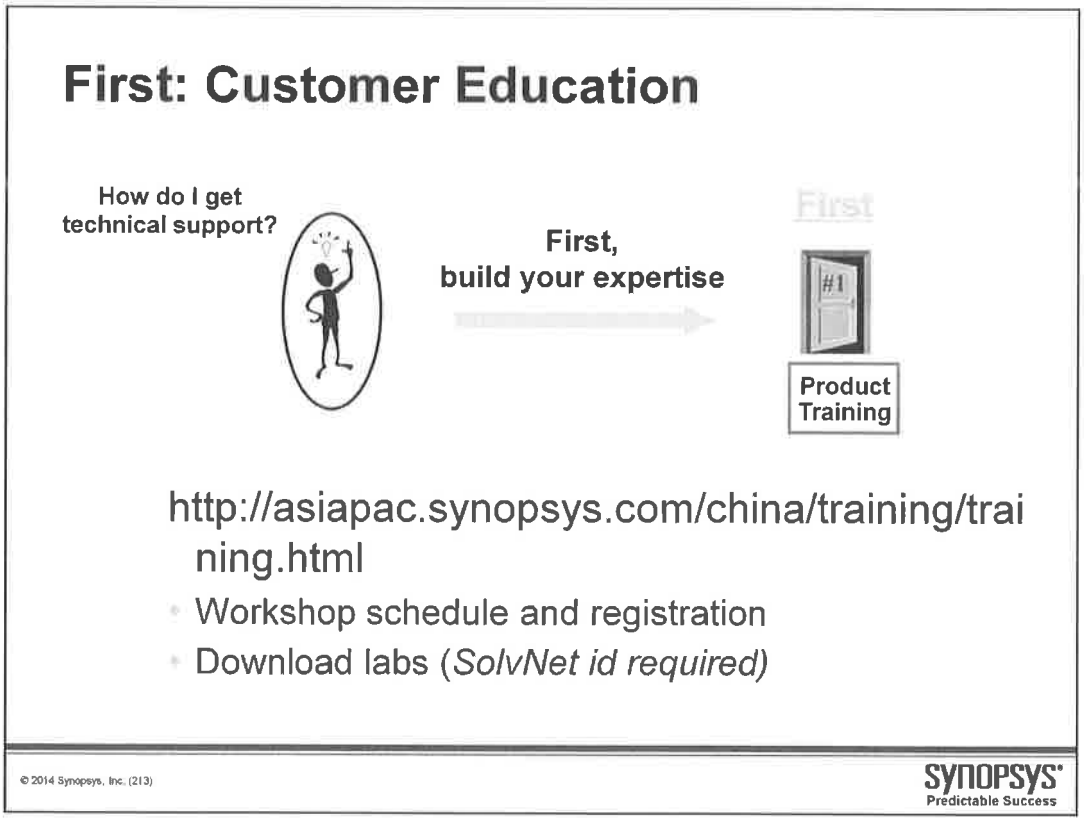
•From Intel AC (Kevin Croysdale):

Ajitha,

Thank you. I just wanted to say that you have been extremely helpful in supporting Nehalem. Your support over the last few weeks has allowed me to concentrate on other aspects of the Nehalem PC evaluation. More importantly, I learned from my conversations with Harsha today, that your support has really given Nehalem great confidence in Synopsys' ability to support PC.

We feel that with some work over the next few months, we will win significant PC business and we owe part of that to the great support that you have been giving these guys. Thanks!

- Kevin



Commendations:

•From Agere DTM (Adam Rosenberg):

I just wanted to pass on some positive feedback from the ACs supporting Front End Implementation at Agere. Bob Wiegand has told me that he would be absolutely buried with issues if it wasn't for the help that your team, specifically Anagha, has provided. Both he and the customer are very pleased with the level of support that they are getting and it has freed Bob up to work on competitive engagements.

Thank you, Adam R.

•From nVidia Design Engineer:

I am writing to make you aware of the outstanding service that we are receiving from Eva Coon. Eva is taking care of nearly all of my Support Center calls, and she has helped me and NVIDIA a great deal. [Three paragraphs of examples…]

This is just a short note and I'm sure I've missed mentioning some of her contributions. However, I am delighted to have her supporting NVIDIA and look forward to a continuing productive relationship. I wanted you to be aware of her great service and hope that she gets all the credit she deserves.

Sincerely, John Busco

NVIDIA CORPORATION

•From nVidia DTM (Chris Lyon):

I know that I speak for all of the implementation ACs when I say that we have definitely felt the benefit of Eva's presence at nVidia. She is able to support John and his engineers very well and goes above and beyond the call of duty! Thank you for allowing one of your best engineers to help us out. -Chris Lyon

•From Intel AC (Kevin Croysdale):

Ajitha,

Thank you. I just wanted to say that you have been extremely helpful in supporting Nehalem. Your support over the last few weeks has allowed me to concentrate on other aspects of the Nehalem PC evaluation. More importantly, I learned from my conversations with Harsha today, that your support has really given Nehalem great confidence in Synopsys' ability to support PC.

We feel that with some work over the next few months, we will win significant PC business and we owe part of that to the great support that you have been giving these guys. Thanks!

- Kevin

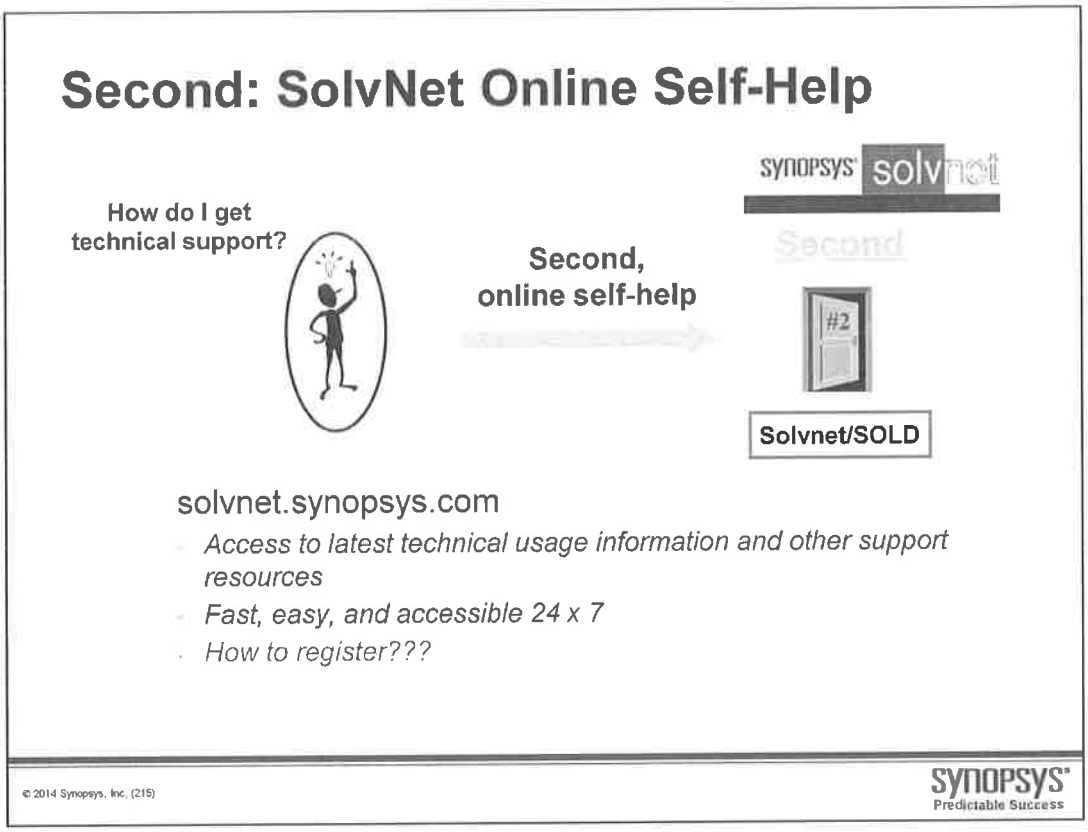
© 2014 Synopsys, Inc. (CustomSim)

How to Register?

- Check the workshop schedule on the China Training Page
- Make sure you are qualified for this training
  - Please go to the link on that page http://www.synopsys.com/services/education/summary3.html for course details
    - Some courses even have tests
- Get your manager's approval and your manager will contact your Synopsys Account Manager (AM) to do the registration for you
- If you contact Synopsys admin, she will confirm your request with your manager via SNPS AM
- You will get a confirmation e-mail from Synopsys admin to inform you the success of the registration
  - Please cancel the registration immediately if you can not attend it

© 2014 Synopsys, Inc. (214)

SYNOPSYS® Predictable Success



Commendations:

•From Agere DTM (Adam Rosenberg):

I just wanted to pass on some positive feedback from the ACs supporting Front End Implementation at Agere. Bob Wiegand has told me that he would be absolutely buried with issues if it wasn't for the help that your team, specifically Anagha, has provided. Both he and the customer are very pleased with the level of support that they are getting and it has freed Bob up to work on competitive engagements.

Thank you, Adam R.

•From nVidia Design Engineer:

I am writing to make you aware of the outstanding service that we are receiving from Eva Coon. Eva is taking care of nearly all of my Support Center calls, and she has helped me and NVIDIA a great deal. [Three paragraphs of examples…]

This is just a short note and I'm sure I've missed mentioning some of her contributions. However, I am delighted to have her supporting NVIDIA and look forward to a continuing productive relationship. I wanted you to be aware of her great service and hope that she gets all the credit she deserves.

Sincerely, John Busco

NVIDIA CORPORATION

•From nVidia DTM (Chris Lyon):

I know that I speak for all of the implementation ACs when I say that we have definitely felt the benefit of Eva's presence at nVidia. She is able to support John and his engineers very well and goes above and beyond the call of duty! Thank you for allowing one of your best engineers to help us out. -Chris Lyon

•From Intel AC (Kevin Croysdale):

Ajitha,

Thank you. I just wanted to say that you have been extremely helpful in supporting Nehalem. Your support over the last few weeks has allowed me to concentrate on other aspects of the Nehalem PC evaluation. More importantly, I learned from my conversations with Harsha today, that your support has really given Nehalem great confidence in Synopsys' ability to support PC.

We feel that with some work over the next few months, we will win significant PC business and we owe part of that to the great support that you have been giving these guys. Thanks!

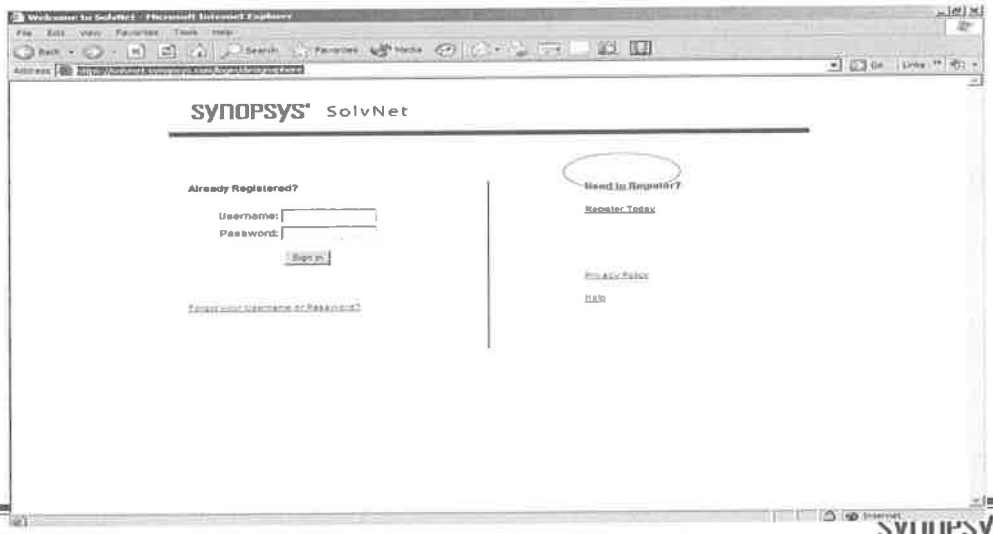
- Kevin

© 2014 Synopsys, Inc. (CustomSim)

SolvNet Registration Step 1:

Go to:

http://solvnet.synopsys.com/login/designsphere

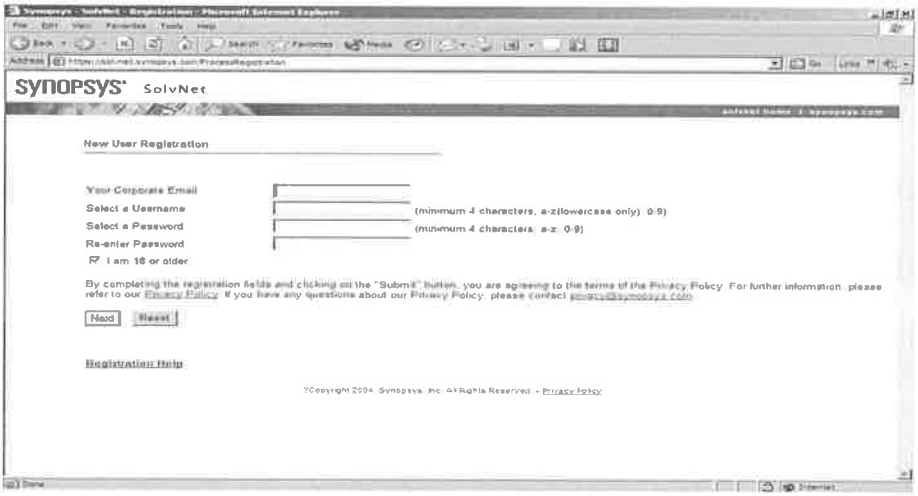


© 2014 Synopsys, Inc. (216)

SYNOPSYS
Predictable Success

SolvNet Registration Step 2:

You can only use your corporate e-mail to register the SolvNet!

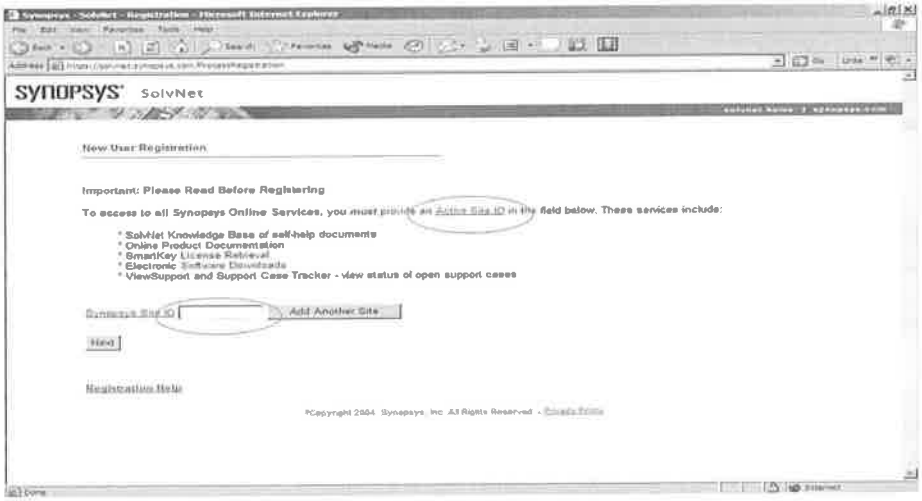


© 2014 Synopsys, Inc. (217)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

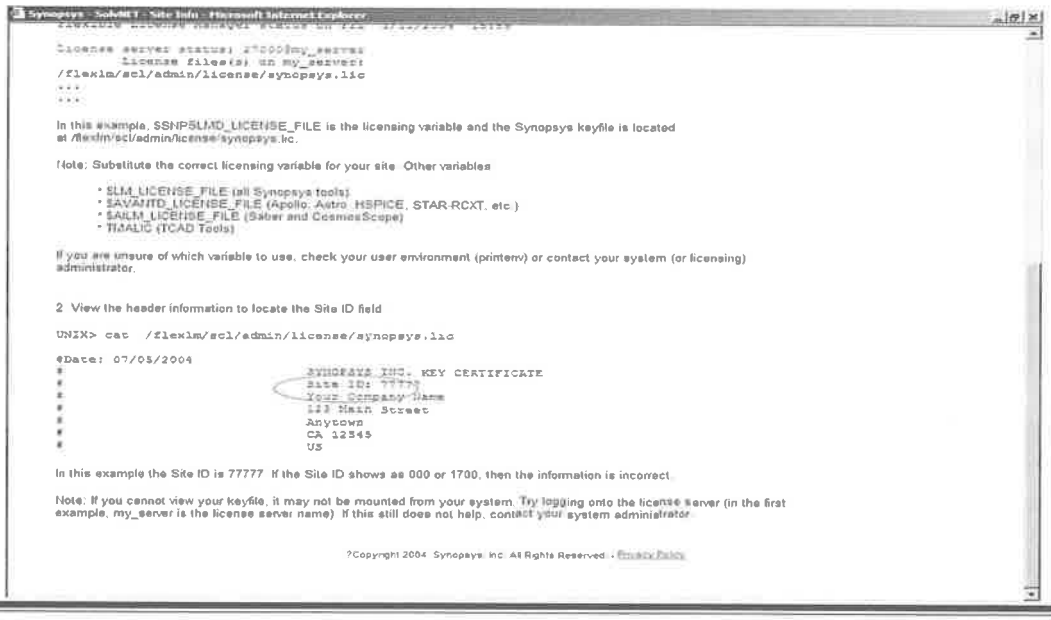
SolvNet Registration Step 3: Site ID



© 2014 Synopsys, Inc. (218)

SYNOPSYS
Predictable Success

How to Get Your Synopsys Site ID?



```
License server status: 27000@my_server
        License file(s) on my_server:
/flexlm/scl/admin/license/synopsys.lic
...
...
```

In this example, SSNP5LMD_LICENSE_FILE is the licensing variable and the Synopsys keyfile is located at /flexlm/scl/admin/license/synopsys.lic.

Note: Substitute the correct licensing variable for your site. Other variables

- SLM_LICENSE_FILE (all Synopsys tools)
- SAVANTD_LICENSE_FILE (Apollo, Astro, HSPICE, STAR-RCXT, etc.)
- SAILM_LICENSE_FILE (Saber and CosmosScope)
- TMALIC (TCAD Tools)

If you are unsure of which variable to use, check your user environment (printenv) or contact your system (or licensing) administrator.

2 View the header information to locate the Site ID field

```
UNIX> cat  /flexlm/scl/admin/license/synopsys.lic

#Date: 07/05/2004
#                              SYNOPSYS INC. KEY CERTIFICATE
#                              Site ID: 77777
#                              Your Company Name
#                              123 Main Street
#                              Anytown
#                              CA 12345
#                              US
```

In this example the Site ID is 77777. If the Site ID shows as 000 or 1700, then the information is incorrect.

Note: If you cannot view your keyfile, it may not be mounted from your system. Try logging onto the license server (in the first example, my_server is the license server name). If this still does not help, contact your system administrator.

?Copyright 2004 Synopsys, Inc. All Rights Reserved. - Privacy Policy

© 2014 Synopsys, Inc. (219)

SYNOPSYS®
Predictable Success

SolvNet Registration Step 4: Activate Your SolvNet Account

- An e-mail will send to your corporate e-mail box. You may be asked to confirm some additional information to activate the account
- If there are any problems regarding your SolvNet account, please send e-mail to: solvnetfeedback@synopsys.com or contact Support Center for help

© 2014 Synopsys, Inc. (220)

SYNOPSYS® Predictable Success

Response Time:

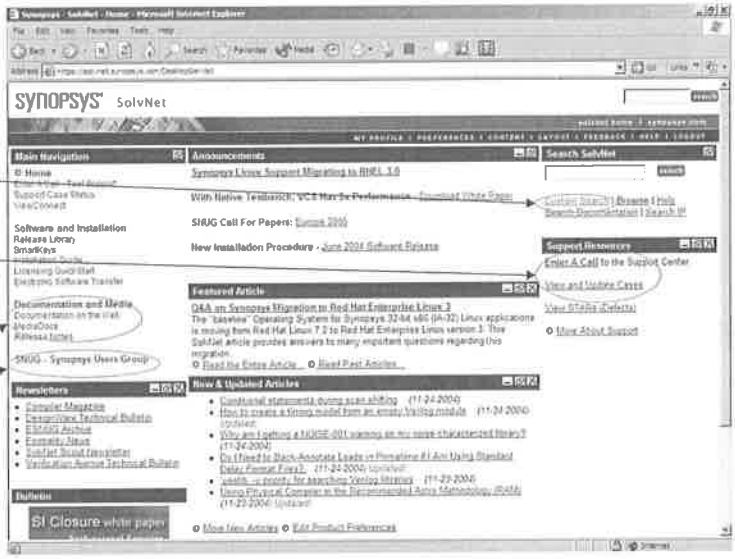
First contact with the Support Center is within 24 hours

Most issues are responded to within 4 hours

© 2014 Synopsys, Inc. (CustomSim)

SolvNet – How to Use It?

- solvnet.synopsys.com/
- Search the database w/ custom search
- Enter-a-call to SNPS Support Center and monitor the progress of your cases
- Documentation, including Release Notes
- SNUG
- Lots more besides…!!

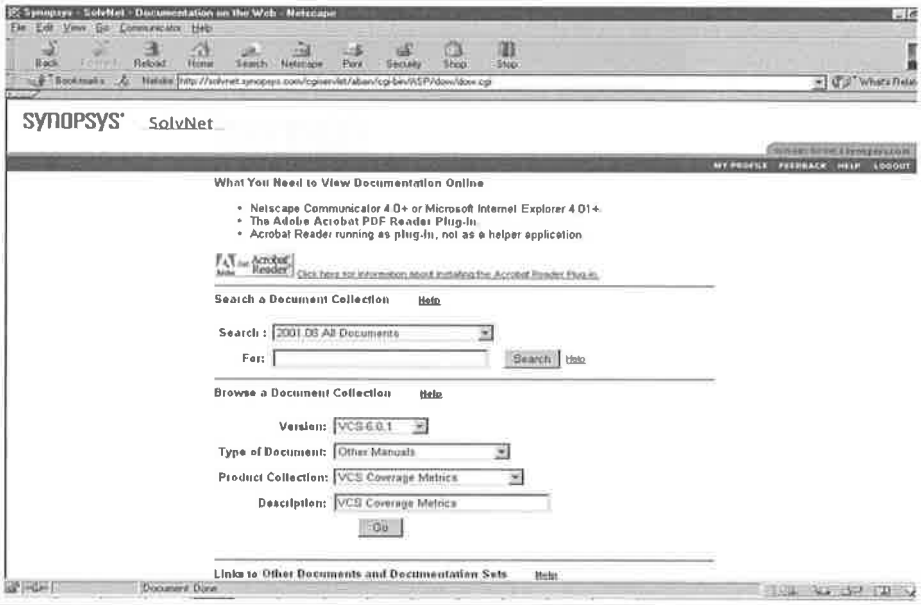


© 2014 Synopsys, Inc. (222)

SYNOPSYS Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

Documents on the Web



© 2014 Synopsys, Inc. (223)

SYNOPSYS® Predictable Success

© 2014 Synopsys, Inc. (CustomSim)

"SolvNet Search"

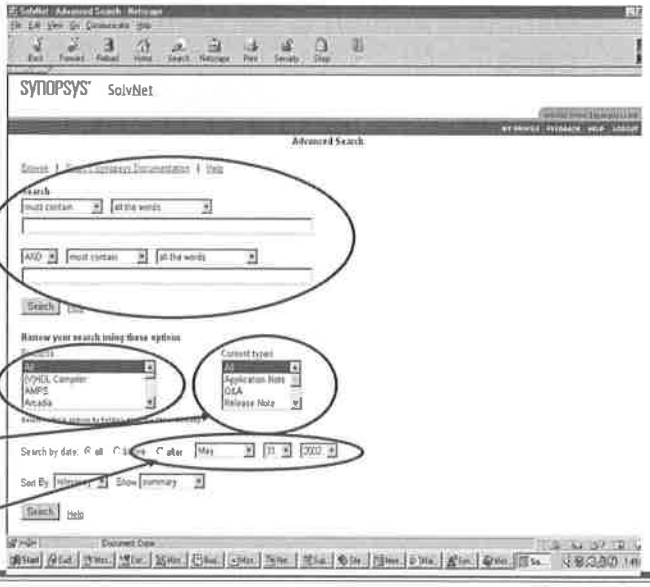
- Simplest search is performed from main SolvNet page
- "Search" examines a database of articles
- Enter keywords
- A list of articles is returned
- Click on an article to read it

© 2014 Synopsys, Inc. (224)

SYNOPSYS®
Predictable Success

SolvNet "Custom Search"

- Custom Search is also available for more specific searches
- Start your search by entering keywords
- Execute the search by product…
- …and/or by Content Types
- …with date constraints



© 2014 Synopsys, Inc. (225)

SYNOPSYS Predictable Success

Third: Support Center

How do I get technical support?

Third,
Support Center

Support Center

- *AE based technical support*
- *Access to internal knowledge database, BU AE's (CAE) and R&D via SC*
- *High availability*
  - *Monday via Friday (Public holidays are not included)*
  - *Working hours: 9:00 AM – 6:00 PM*
- *Access via Enter-a-Call webpage – The Best Way!*

© 2014 Synopsys, Inc. (226)

SYNOPSYS® Predictable Success

Commendations:

•From Agere DTM (Adam Rosenberg):

I just wanted to pass on some positive feedback from the ACs supporting Front End Implementation at Agere. Bob Wiegand has told me that he would be absolutely buried with issues if it wasn't for the help that your team, specifically Anagha, has provided. Both he and the customer are very pleased with the level of support that they are getting and it has freed Bob up to work on competitive engagements.

Thank you, Adam R.

•From nVidia Design Engineer:

I am writing to make you aware of the outstanding service that we are receiving from Eva Coon. Eva is taking care of nearly all of my Support Center calls, and she has helped me and NVIDIA a great deal. [Three paragraphs of examples…]

This is just a short note and I'm sure I've missed mentioning some of her contributions. However, I am delighted to have her supporting NVIDIA and look forward to a continuing productive relationship. I wanted you to be aware of her great service and hope that she gets all the credit she deserves.

Sincerely, John Busco

NVIDIA CORPORATION

•From nVidia DTM (Chris Lyon):

I know that I speak for all of the implementation ACs when I say that we have definitely felt the benefit of Eva's presence at nVidia. She is able to support John and his engineers very well and goes above and beyond the call of duty! Thank you for allowing one of your best engineers to help us out. -Chris Lyon

•From Intel AC (Kevin Croysdale):

Ajitha,

Thank you. I just wanted to say that you have been extremely helpful in supporting Nehalem. Your support over the last few weeks has allowed me to concentrate on other aspects of the Nehalem PC evaluation. More importantly, I learned from my conversations with Harsha today, that your support has really given Nehalem great confidence in Synopsys' ability to support PC.

We feel that with some work over the next few months, we will win significant PC business and we owe part of that to the great support that you have been giving these guys. Thanks!

- Kevin

© 2014 Synopsys, Inc. (CustomSim)

Synopsys Support Center

- Three ways to get access to Synopsys Support Center:
  - *Enter a Call* on SolvNet
    - Preferred, the case directly route to the product specialist
  - Send e-mails to prchelp@synopsys.com
    - Needs manual dispatch to a product specialist
  - Make a call at: 400-670-7677
    - Slowest, need to route the case manually by dispatcher
- Always just reply the e-mail from Synopsys support center
  - The case ID (8000XXXXXX) will be included in the e-mail subject
  - Our system will handle your incoming e-mail and link it to the previous e-mails you sent to us in this case

© 2014 Synopsys, Inc. (227)

SYNOPSYS®
Predictable Success

Which Info You Need to Provide to Use the Services?

- Identify yourself
  - Your contact info, Company name & Site ID for 400 hotline support
  - Please use your corporate e-mail account
    - Please do not send us mails using your personal e-mail accounts from @hotmail.com, @263.net...etc.. They are blocked by our junk mail filter
- Describe your question as detailed as possible
- Be willing to provide test cases to help Synopsys AC to duplicate the problem
  - Necessary for better support to you!
  - If a NDA issue rises, please escalate to your manager. SNPS AE will also escalate to SC manager. The managers will resolve the issue

© 2014 Synopsys, Inc. (228)

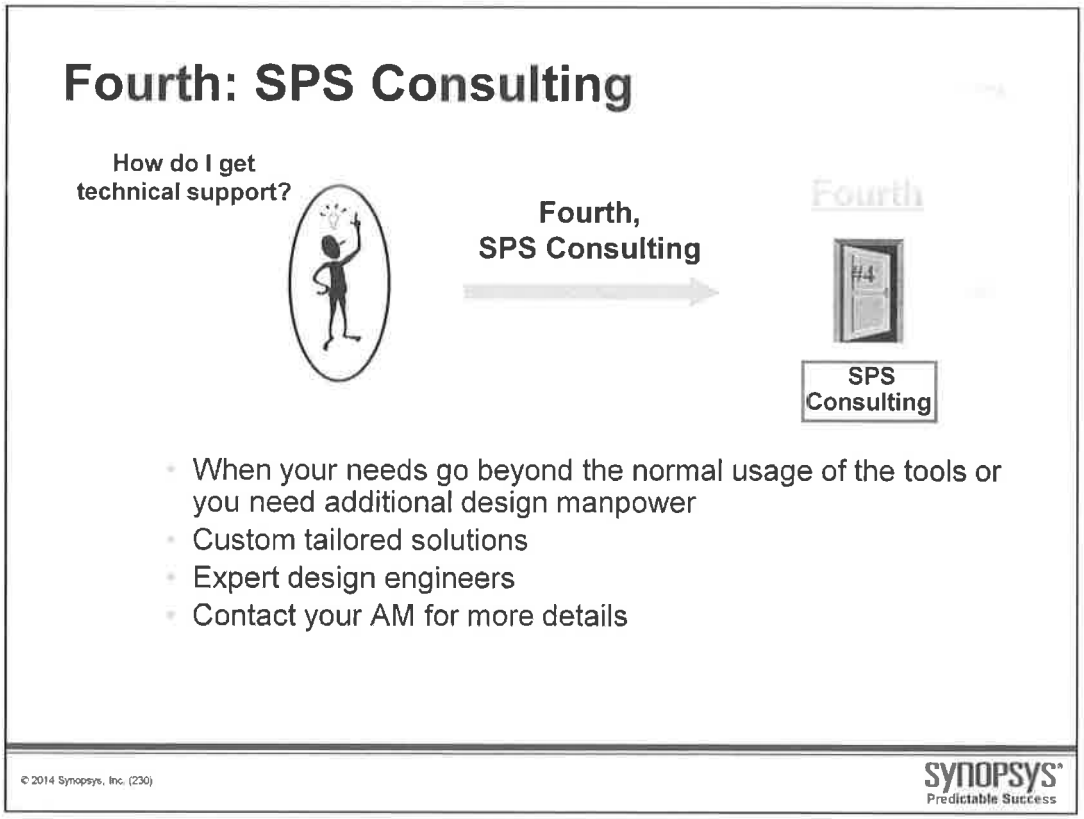
SYNOPSYS®
Predictable Success

Synopsys China Support Center Commitments

- First response to your question within 4 working hrs
  - At least the acknowledgement of receiving your question
- Final results to all questions
  - solutions/answers
  - bug recorders/workarounds
  - You can use SolvNet to monitor/update the processing of your cases

© 2014 Synopsys, Inc. (229)

SYNOPSYS® Predictable Success



Commendations:

•From Agere DTM (Adam Rosenberg):

I just wanted to pass on some positive feedback from the ACs supporting Front End Implementation at Agere. Bob Wiegand has told me that he would be absolutely buried with issues if it wasn't for the help that your team, specifically Anagha, has provided. Both he and the customer are very pleased with the level of support that they are getting and it has freed Bob up to work on competitive engagements.

Thank you, Adam R.

•From nVidia Design Engineer:

I am writing to make you aware of the outstanding service that we are receiving from Eva Coon. Eva is taking care of nearly all of my Support Center calls, and she has helped me and NVIDIA a great deal. [Three paragraphs of examples…]

This is just a short note and I'm sure I've missed mentioning some of her contributions. However, I am delighted to have her supporting NVIDIA and look forward to a continuing productive relationship. I wanted you to be aware of her great service and hope that she gets all the credit she deserves.

Sincerely, John Busco

NVIDIA CORPORATION

•From nVidia DTM (Chris Lyon):

I know that I speak for all of the implementation ACs when I say that we have definitely felt the benefit of Eva's presence at nVidia. She is able to support John and his engineers very well and goes above and beyond the call of duty! Thank you for allowing one of your best engineers to help us out. -Chris Lyon

•From Intel AC (Kevin Croysdale):

Ajitha,

Thank you. I just wanted to say that you have been extremely helpful in supporting Nehalem. Your support over the last few weeks has allowed me to concentrate on other aspects of the Nehalem PC evaluation. More importantly, I learned from my conversations with Harsha today, that your support has really given Nehalem great confidence in Synopsys' ability to support PC.

We feel that with some work over the next few months, we will win significant PC business and we owe part of that to the great support that you have been giving these guys. Thanks!

- Kevin

Snopsys Users Group (SNUG) Conference

- An open forum for Synopsys users to exchange ideas, discuss technical problems and learn new concepts
- An opportunity to discuss with Synopsys technical forks
- Local technical experts' presentations and Synopsys experts' technical workshops
- http://www.snug-universal.org/

© 2014 Synopsys, Inc. (231)

SYNOPSYS®
Predictable Success

And More…

- Need a training?
  - Please Visit http://www.synopsys.com.cn
- Want to exchange design experiences with other engineers?
  - Write a SNUG paper and win a big prize
  - Attend the annual SNUG conference

© 2014 Synopsys, Inc. (232)

SYNOPSYS®
Predictable Success