Debug Challenges in Low-Power Design and Verification

Durgesh Prasad, Madhur Bhargava, Jitesh Bansal, Chuck Seeley

Mentor Graphics Corp. 8005 SW Boeckman Rd. Wilsonville, OR 97070

Abstract

Various studies have shown that a significant amount of engineering time and effort for a project is typically spent on debug. For a low-power design and verification, these debug challenges are further complicated as a result of the sophisticated power management architectures and techniques that are used. Moreover the traditional debug technology and methods focus on issues found in a design working in always-on mode and fails to address the new and complex power-related issues thereby consuming more engineering time. In this paper, we will provide an in-depth analysis of various debug challenges and problems faced in low-power design and verification. By taking relevant examples we will demonstrate how these issues can be either avoided or solved. We will also highlight some of the common pitfalls that low-power designers can avoid which otherwise can lead to complex low-power issues that are difficult to debug at later stages of the design cycle.

Problems in Power Intent Specification

UPF 2.0 Migration Issues

Problem: UPF supplies default to OFF state with UPF 2.0:
Solution: Utilize UPF package functions to explicitly turn on all necessary supplies

```
module tb;
import UPF::*;
…
initial begin
   supply_on ("tb/dut_inst/VDD", 1.1);
   supply_on ("tb/dut_inst/GND", 0.0);
end
…
dut dut_inst (…);
…
endmodule
```

Wildcard expansion issues

Problem: HDL/IP block placed in incorrect power domain

```
HDL :
module dut;
…
Ip_module my_ip1();
Ip_module my_ip2();
Ip_module my_ip3(); // my_ip3 is powered separately from my_ip1 and my_ip2
endmodule

UPF: create_power_domain pd_dut –elements {my_ip*}
```

Solution: Use save_upf command to create interpreted UPF code or use find_objects command to print out expanded TCL variable

Incorrect or missing VCT Specification

Problem: HDL GND pin driven to a logic "1" when connected to UPF GND supply
Solution: Use proper VCT option connect_supply_net command OR Use set_port_attribute command to designate HDL GND pin as primary_ground

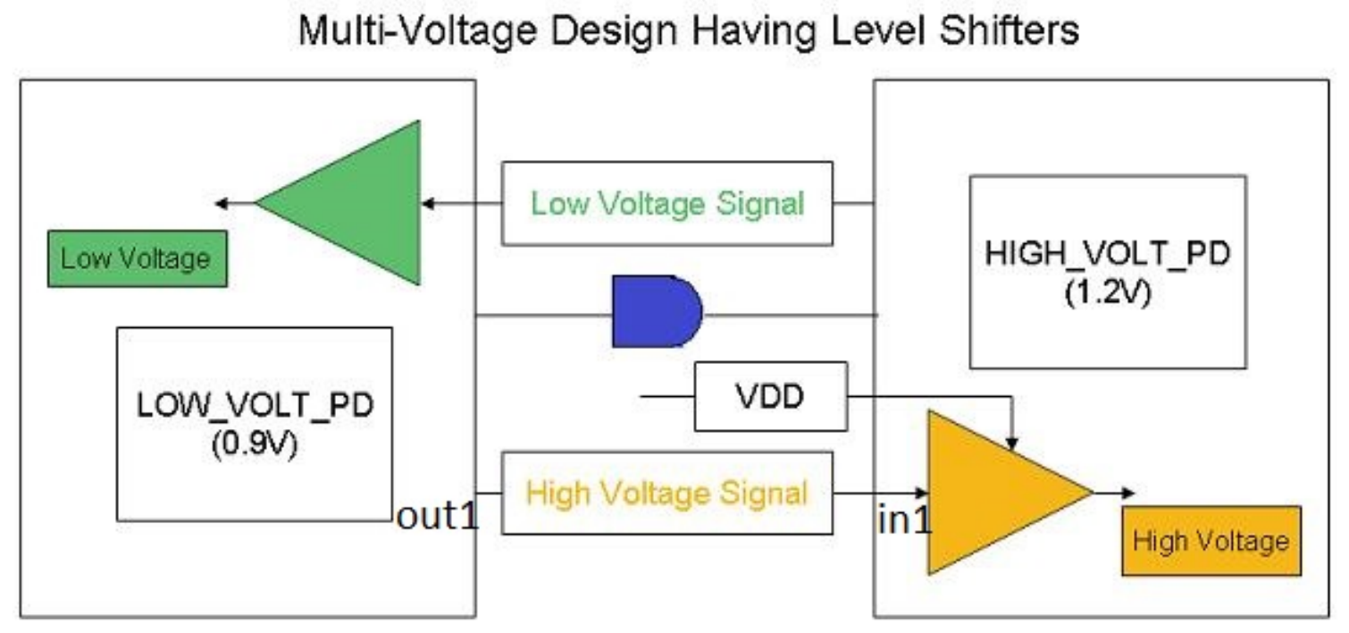
```
connect_supply_net upf_GND –ports {hm_inst/GND} –vct UPF_GNDZERO2SV_LOGIC
set_port_attributes –pg_type primary_ground –ports {hm_inst/GND}
```

Power UP Failures

Missing or Incorrect isolation/level shifter

Problem: X Values represent floating inputs whose voltage values can cause undesirable design behavior
Solution: Utilize vendors static checks to verify UPF contains the proper isolation/level_shfiter constructs

Multi-Voltage Design Having Level Shifters



Power State Table

States		Voltage Domains	
		LOW_VOLT_PD	HIGH_VOLT_PD
	norm_st	low_on (0.9V)	high_on (1.2V)
	off_st	low_off	high_on (1.2V)

Initial block Re-evaluation

Problem: For certain models, such as a ROM memory, initial block may need to be re-executed on power –up after time 0
Solution: Utilize vendors individual solutions to specify which modules or blocks need to have initial blocks re-executed at power up or exclude object from PA semantics

```
module rom_mem ( input [7:0] addr, input re, clk, output reg[7:0] data_out);

reg[7:0] mem[255:0];

initial begin
$readmemh("rom.mem", mem);
end

assign data_out = (re == 1) ? mem[addr] : 8'hzz;

endmodule
```

Incorrect or Missing Retention or incorrect Retention Protocol

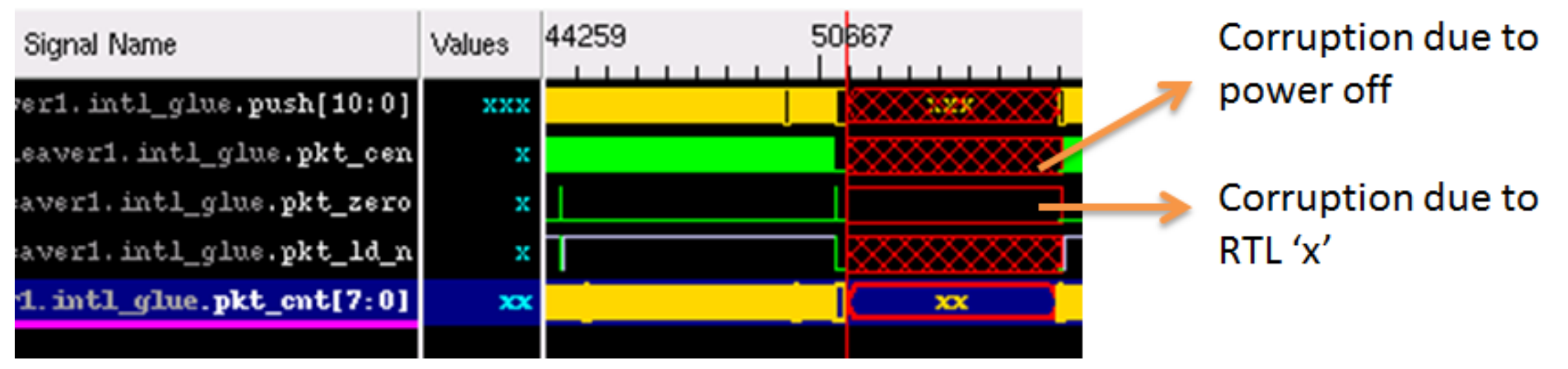
Problem: Registers remain X after power-up. Could be missing retention in UPF or non-retention register needing reset on power-up
Solution: Utilize vendors automated retention assertion to detect incorrect retention protocol or non-retention registers that need to be reset

```
Error: (vsim-8912) MSPA_NRET_ASYNCFF: Time: 12 ns, Asynchronous (set/reset) control for the following flop(s) of power domain 'PD1' is not asserted at power up:
/tb/top_inst/out1odule rom_mem ( input [3:0] addr, input re, clk, output reg[1:0] data_out);

Error: (vsim-8903) MSPA_RET_OFF_PSO: Time: 64 ns, Retention control (0) for the following retention elements in scope '/tb/top_vl' of power domain 'pd' is not asserted during power shut down:
```

Determining Sources of Unwanted X Values

Problem: Is X value on a signal due to power domain corrupting
Solution: Use Waveform highlighting to distinguish X value is caused by power domain corruption

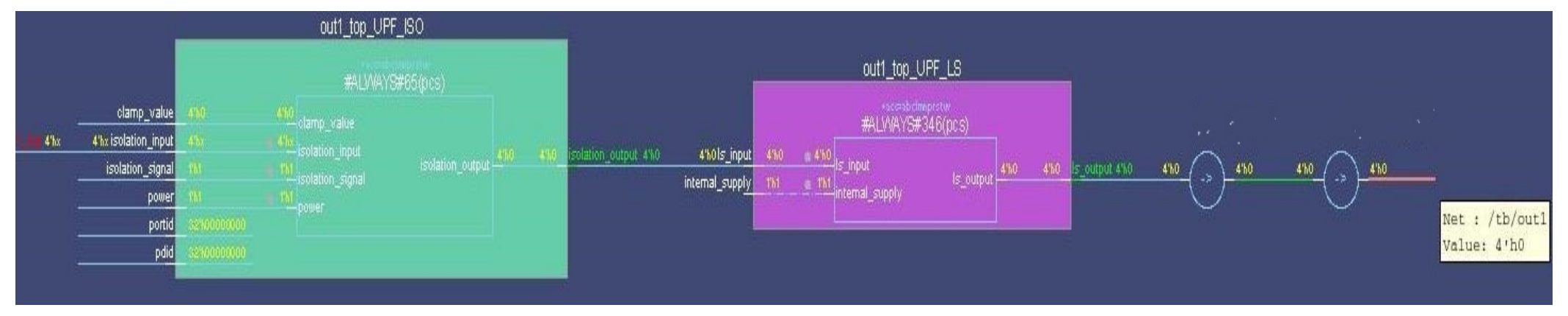


Signal Name	Values
er1.intl_glue.push[10:0]	xxx
eaver1.intl_glue.pkt_cen	x
aver1.intl_glue.pkt_zero	x
aver1.intl_glue.pkt_ld_n	x
1.intl_glue.pkt_cnt[7:0]	xx

Corruption due to power off

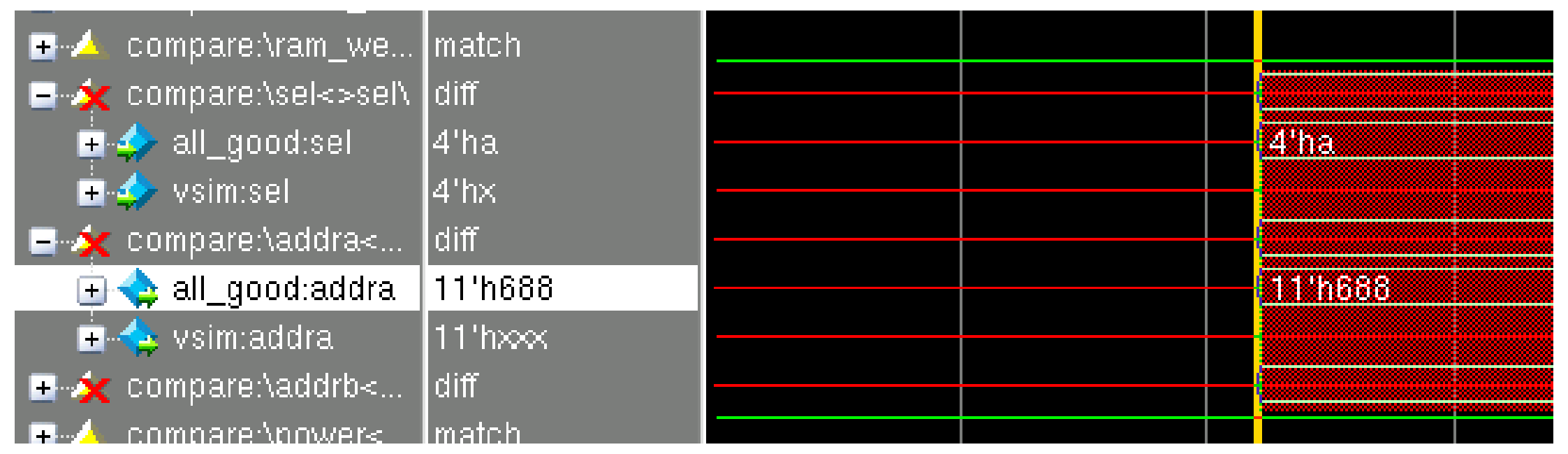
Corruption due to RTL 'x'

Problem: Is X value on a signal due to missing UPF inserted logic or is power to the UPF inserted logic not turned on
Solution: Trace drivers in schematic/dataflow window



```
Net : /tb/out1
Value: 4'h0
```

Problem: Low power simulation failures occur after power up of a domain
Solution: Use Waveform compare feature to easily detect simulation differences where X values remain after power up also enable low power messages and or assertion checks



```
compare:\ram_we...     match
compare:\sel<>sel\     diff
  all_good:sel         4'ha
  vsim:sel             4'hx
compare:\addra<...     diff
  all_good:addra       11'h688
  vsim:addra           11'hxxx
compare:\addrb<...     diff
compare:\power<...     match
```

Problem: Power control signals are unexpectedly getting corrupted when certain domains are powered down
Solution: Ensure that any buffers on these "always on" nets have PA semantics disabled via:

1. Manually exclude these cells using vendor specific exclusion mechanism such a exclude files or setting DON'T_TOUCH attributes on them
2. Leverage Liberty files which contain always_on attributes to auto exclude them from corruption

** Note: Power Aware simulation semantics disabled for chip_top/u_hm_top_0/u_ip_1

Power States

Illegal Power States

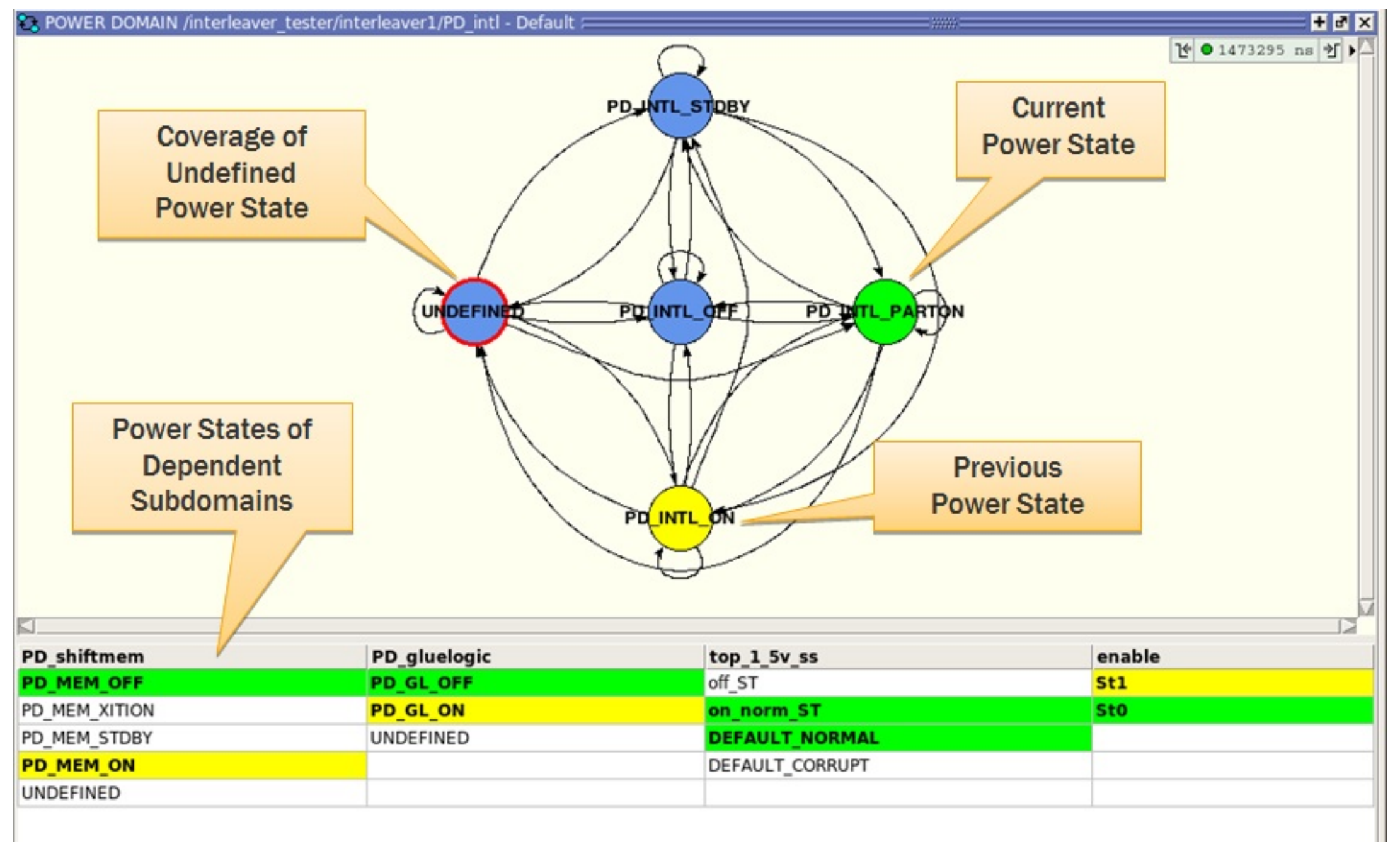
Problem: How to ensure that unwanted power State and power state transitions don't occur
Solution: Leverage UPF 2.0 add_power_state and describe_state_transition commands to declare illegal power states and state transitions

```
add_power_state PD_ALU_SS -state ON4 { -logic_expr { !pwr_alu && !pwr_ram } -simstate CORRUPT -illegal}

# ** Error: UPF_ILLEGAL_STATE_REACHED: Time: 129 ns, Supply set 'PD_ALU_SS' reached an illegal power state 'ON4'.
# File: src/parser_test22/demo.upf, Line: 73, Power state:ON4
```

Coverage of Power States

Problem: How to ensure that desired power State and power state transitions occur during simulation
Solution: Leverage vendors capabilities in displaying and reporting power state and power state transition coverage data



Coverage of Undefined Power State

Current Power State

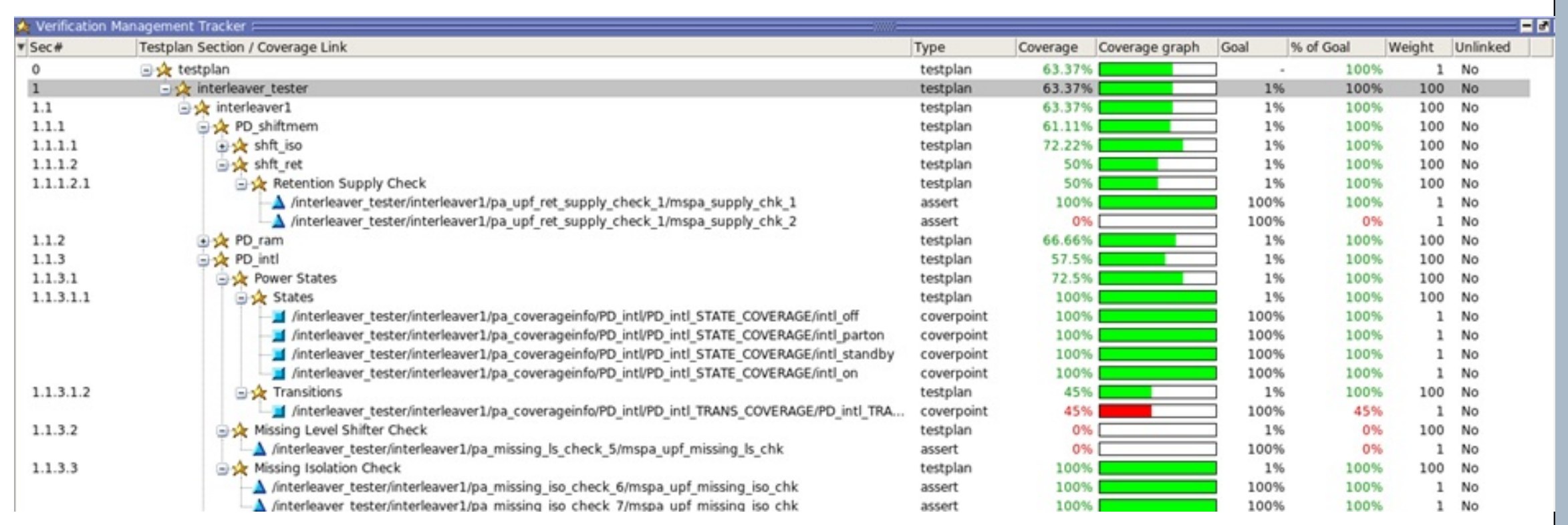
Power States of Dependent Subdomains

Previous Power State

PD_shiftmem	PD_gluelogic	top_1_5v_ss	enable
PD_MEM_OFF	PD_GL_OFF	off_ST	St1
PD_MEM_XITION	PD_GL_ON	on_norm_ST	St0
PD_MEM_STDBY	UNDEFINED	DEFAULT_NORMAL	
PD_MEM_ON		DEFAULT_CORRUPT	
UNDEFINED			

Achieving PA Coverage Data Closure

Problem: How to ensure coverage closure for power specific items
Solution: Utilize vendors solutions to report, display, and track all coverage data including PA coverage data



Verification Management Tracker

Sec#	Testplan Section / Coverage Link	Type	Coverage	Coverage graph	Goal	% of Goal	Weight	Unlinked
0	testplan	testplan	63.37%		-	100%	1	No
1	interleaver_tester	testplan	63.37%		1%	100%	100	No
1.1	interleaver1	testplan	63.37%		1%	100%	100	No
1.1.1	PD_shiftmem	testplan	61.11%		1%	100%	100	No
1.1.1.1	shft_iso	testplan	72.22%		1%	100%	100	No
1.1.1.2	shft_ret	testplan	50%		1%	100%	100	No
1.1.1.2.1	Retention Supply Check	testplan	50%		1%	100%	100	No
	/interleaver_tester/interleaver1/pa_upf_ret_supply_check_1/mspa_supply_chk_1	assert	100%		100%	100%	1	No
	/interleaver_tester/interleaver1/pa_upf_ret_supply_check_1/mspa_supply_chk_2	assert	0%		100%	0%	1	No
1.1.2	PD_ram	testplan	66.66%		1%	100%	100	No
1.1.3	PD_intl	testplan	57.5%		1%	100%	100	No
1.1.3.1	Power States	testplan	72.5%		1%	100%	100	No
1.1.3.1.1	States	testplan	100%		1%	100%	100	No
	/interleaver_tester/interleaver1/pa_coverageinfo/PD_intl/PD_intl_STATE_COVERAGE/intl_off	coverpoint	100%		100%	100%	1	No
	/interleaver_tester/interleaver1/pa_coverageinfo/PD_intl/PD_intl_STATE_COVERAGE/intl_parton	coverpoint	100%		100%	100%	1	No
	/interleaver_tester/interleaver1/pa_coverageinfo/PD_intl/PD_intl_STATE_COVERAGE/intl_standby	coverpoint	100%		100%	100%	1	No
	/interleaver_tester/interleaver1/pa_coverageinfo/PD_intl/PD_intl_STATE_COVERAGE/intl_on	coverpoint	100%		100%	100%	1	No
1.1.3.1.2	Transitions	testplan	45%		1%	100%	100	No
	/interleaver_tester/interleaver1/pa_coverageinfo/PD_intl/PD_intl_TRANS_COVERAGE/PD_intl_TRA…	coverpoint	45%		100%	45%	1	No
1.1.3.2	Missing Level Shifter Check	testplan	0%		1%	0%	100	No
	/interleaver_tester/interleaver1/pa_missing_ls_check_5/mspa_upf_missing_ls_chk	assert	0%		100%	0%	1	No
1.1.3.3	Missing Isolation Check	testplan	100%		1%	100%	100	No
	/interleaver_tester/interleaver1/pa_missing_iso_check_6/mspa_upf_missing_iso_chk	assert	100%		100%	100%	1	No
	/interleaver_tester/interleaver1/pa_missing_iso_check_7/mspa_upf_missing_iso_chk	assert	100%		100%	100%	1	No