2023 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION INDIA

Paradigm Shift in Power Aware Simulation Using Formal Techniques

Sachin Bansal, Girish Marudwar, Sandeep Jana, Kamalesh Ghosh, Yogananda Mesa
sachinba@synopsys.com, girishm@synopsys.com sjana@synopsys.com kghosh@synopsys.com
mesa@synopsys.com

***Abstract*- In recent times, there has been an increasing need to exhaustively verify power aware connectivity and functional properties in designs using formal verification tools to shift left the traditional simulation based low power functional verification flow. This has created a cross technology flow using Synopsys low power static checker tool VCLP and Synopsys formal verification tool VC-Formal where, a power aware formal property is used to describe a low power design connection/function and the verification flow uses low power model (also called power network model (PNM)) from VCLP and exhaustive analysis engines of VC-Formal to prove correctness of these power aware properties. If design does not adhere to the behavior described by a power aware property, the power aware formal property is falsified. The results on customer and in-house designs has established the shift-left notion by significantly reducing the turn-around-time (TAT) of overall low power functional verification by catching bugs very early in the verification flow.**

I. MOTIVATION

In the low power functional verification paradigm, designers typically want to verify the following:

1) *Connectivity properties:* Prove that the chip level design/power network connections which cross block (tile) boundaries are not broken in presence of low power intent (say using UPF [3]).
2) *Functional properties:* Prove that the functional behavior of the design does not change in presence of the low power intent.

For (1), a design path which is structurally connected can get broken in presence of power network and states. For example, in Fig 1, there is a design connection from FF1/Q to FF2/D which is going through a buffer BUF. FF2, FF1 and BUF have different supplies namely VDD1, VDD2 and VDDO respectively. This makes the connection broken if there is a power state in the power intent (e.g., UPF) where supply VDD1 and VDD2 are ON but supply VDDO goes OFF (Fig. 1).

Similar examples can be provided for other power connectivity scenarios and functional behaviors. Even though, these scenarios can be theoretically verified using a power aware simulation tool like VCS-NLP [4], simulation comes late in the flow, simulation is expensive, and simulation does NOT guarantee exhaustive (100%) verification. Therefore, power aware low power formal verification is the need of the time to shift left the tradition functional verification flow and thereby significantly reduce the overall turn-around-time (TAT).

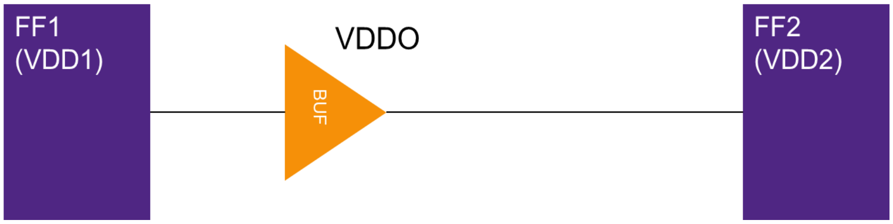


Figure 1. An example of a connection going through blocks powered by different power supplies.

II. RELATED WORK

There were some previous works on modeling power networks that can be fed to formal tools [5]. This invention models the power network as specified in the UPF by doing heavy design instrumentation. The heavy design instrumentation did not work well for the formal tools for the following reasons – a) Instrumenting the design lead to numerous modeling errors, b) Inefficient instrumentation lead to very big models, c) Debugging issues took huge amount of time because of absence of proper annotation in the instrumented data.

In the proposed model (PNM), we decided to take care of the above-mentioned issues which are summarized in Section III.

III. CONTRIBUTION

This paper introduces a simple, light weight Power Network Model (PNM) which is dumped in a Synthesizable Verilog format. This model is directly consumed by the formal tool to verify the power aware properties through corruption MUXes and minimal instrumentation (Flow is depicted in Fig. 2). Modeling of power semantics and dumping in synthesizable Verilog helps us in the following way:

1) Clean separation between design (DUT), testbench (TB) and PNM.
2) Well defined boundaries and interaction models for PNM.
3) Enhanced debuggability and model verification for PNM and its interaction with DUT/TB.

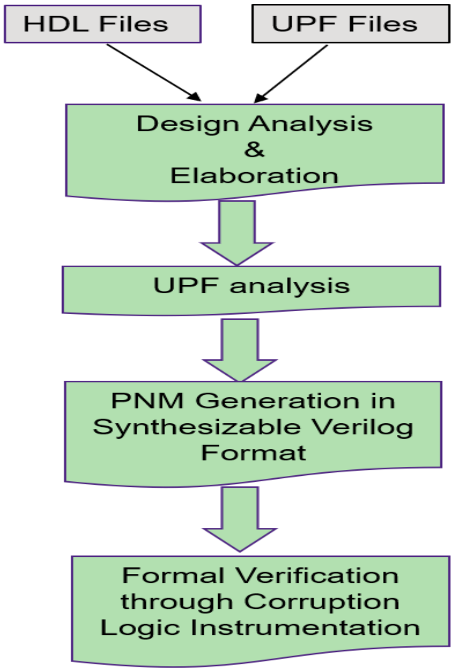

HDL Files
UPF Files
Design Analysis & Elaboration
UPF analysis
PNM Generation in Synthesizable Verilog Format
Formal Verification through Corruption Logic Instrumentation

Figure 2. Proposed tool flow for formal analysis with PNM.

A. PNM Model Generation

The absence of PST as indicator of different power states and transitions among various supplies specified in UPF necessitates the functional modeling of these supplies considering Power Switches, Fine grain macros, power down functions, add power state, resolution functions etc.The above functional modeling will help to capture the FULL_ON/OFF state modeling for different supplies specified in the UPF. In addition to this, the user needs to model Formal power aware testbench for Undriven Supplies (which are primary input ports).

Example Modelling of Power Switch:

```
create_power_switch PSW1
-domain PD1
-output_supply_port {VDDO VDDO}
-input_supply_port {VDDI VDDI}
-control_port {EN1 en1}
-control_port {EN2 en2}
-ack_port {ACK1 ack1}
-off_state {off {en1&&en2}}
-on_state {on VDDI {!(en1&&en2))}}
```

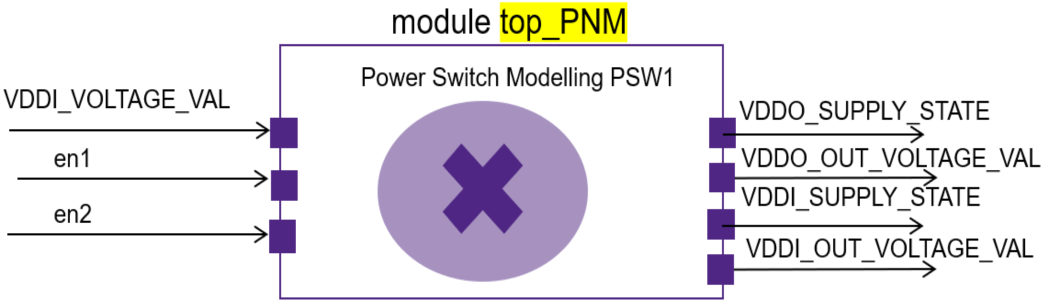
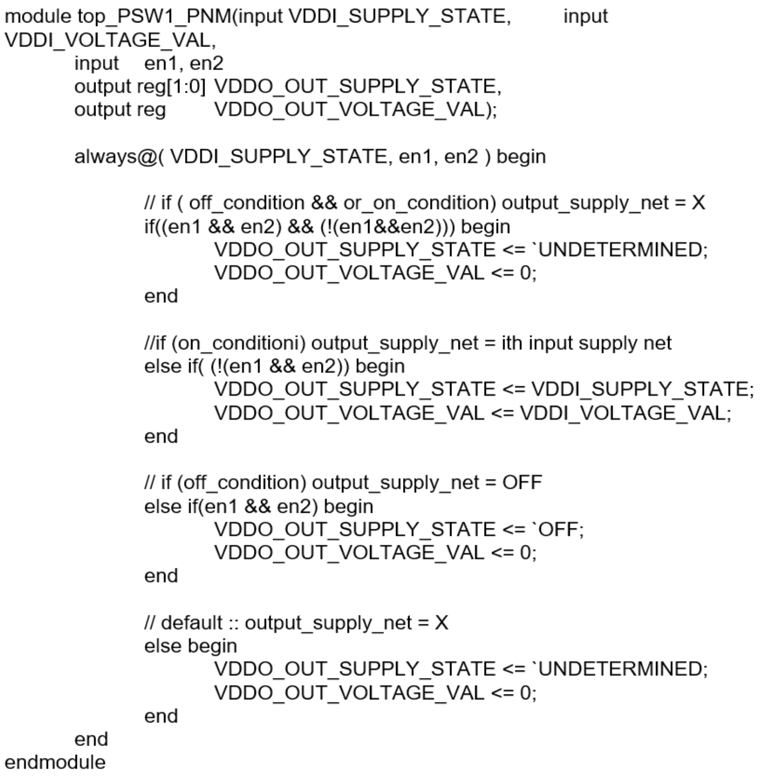


Figure 3. A power switch declaration in UPF and the corresponding modelling in the PNM.

Modelling of power switch in a synthesizable Verilog format models input supply VDDI & enable signals en1 and en2 as inputs. The output supply VDDO supply state is modelled as output of the Verilog module (Fig. 4).



```
module top_PSW1_PNM(input VDDI_SUPPLY_STATE,        input
VDDI_VOLTAGE_VAL,
        input    en1, en2
        output reg[1:0]  VDDO_OUT_SUPPLY_STATE,
        output reg       VDDO_OUT_VOLTAGE_VAL);

        always@( VDDI_SUPPLY_STATE, en1, en2 ) begin

                // if ( off_condition && or_on_condition) output_supply_net = X
                if((en1 && en2) && (!(en1&&en2))) begin
                        VDDO_OUT_SUPPLY_STATE <= `UNDETERMINED;
                        VDDO_OUT_VOLTAGE_VAL <= 0;
                end

                //if (on_conditioni) output_supply_net = ith input supply net
                else if( (!(en1 && en2)) begin
                        VDDO_OUT_SUPPLY_STATE <= VDDI_SUPPLY_STATE;
                        VDDO_OUT_VOLTAGE_VAL <= VDDI_VOLTAGE_VAL;
                end

                // if (off_condition) output_supply_net = OFF
                else if(en1 && en2) begin
                        VDDO_OUT_SUPPLY_STATE <= `OFF;
                        VDDO_OUT_VOLTAGE_VAL <= 0;
                end

                // default :: output_supply_net = X
                else begin
                        VDDO_OUT_SUPPLY_STATE <= `UNDETERMINED;
                        VDDO_OUT_VOLTAGE_VAL <= 0;
                end
        end
endmodule
```

Figure 4. Verilog dump of the power switch model in the PNM.

B. Corruption Login Instrumentation

To verify the functional connectivity between FF1 and FF2 (Fig. 5), Formal tool needs to prove that whenever source FF1 and destination FF2 are in FULL_ON state, buffer in path should not be in OFF state. This will break the functional connectivity.

To prove this, Formal tool will instrument corruption MUXes in the path to model X propagation logic.

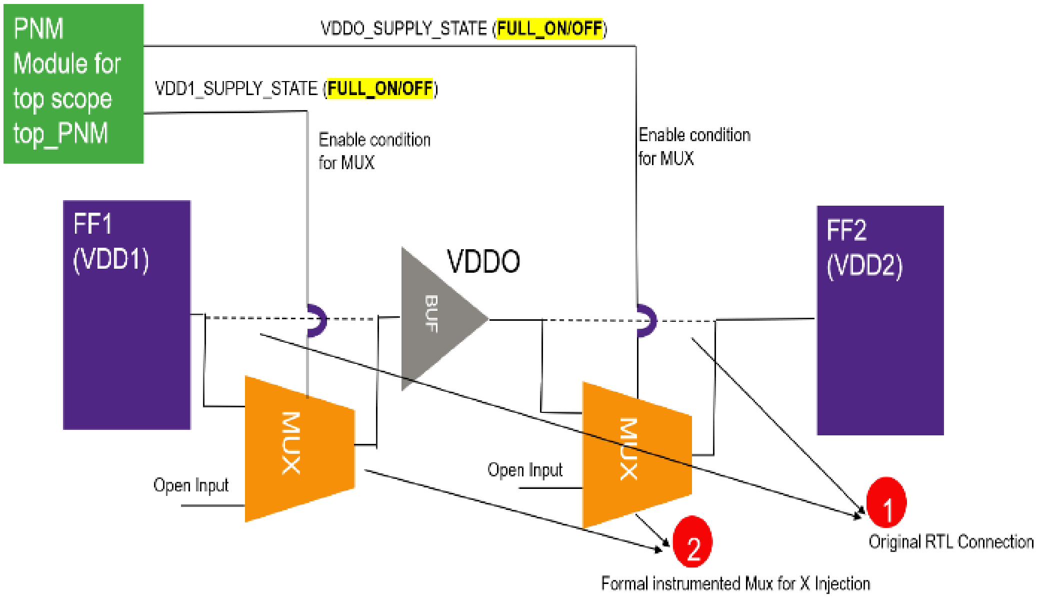

PNM Module for top scope top_PNM
VDDO_SUPPLY_STATE (FULL_ON/OFF)
VDD1_SUPPLY_STATE (FULL_ON/OFF)
Enable condition for MUX
Enable condition for MUX
FF1 (VDD1)
VDDO
BUF
FF2 (VDD2)
MUX
MUX
Open Input
Open Input
1
Original RTL Connection
2
Formal instrumented Mux for X Injection

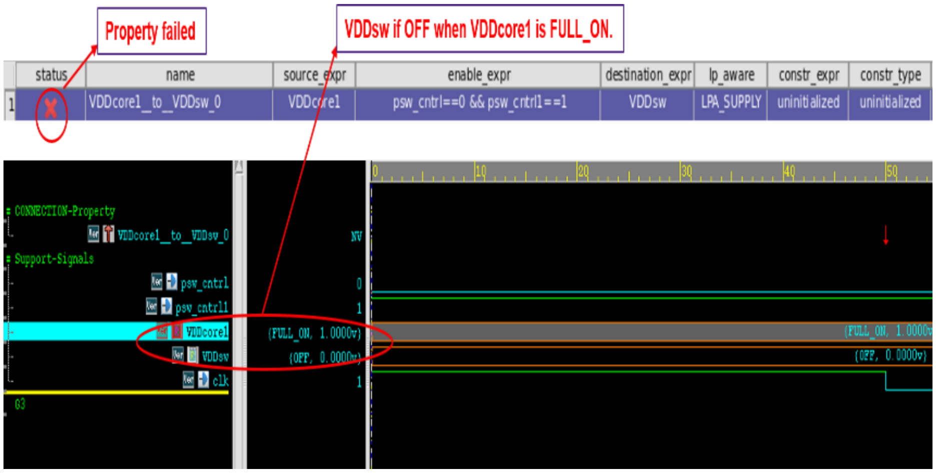
Figure 5.Introduction of Corruption Muxes for verification of functional connection.

The supply state of the buffer modelled from PNM will function as enable of the corruption muxes inserted in the path for the X propagation logic.

IV. RESULTS

Consider a Power Aware Connectivity property written between supplies VDDCore1 and VDDsw which are connected through a power switch in UPF. Formal verification needs to verify if both supplies have the same state when enable expression is true.

add_cc -dest VDDsw -src VDDcore1 -enable { psw_cntrl==0 && psw_cntrl1==1}



Property failed

VDDsw if OFF when VDDcore1 is FULL_ON.

	status	name	source_expr	enable_expr	destination_expr	lp_aware	constr_expr	constr_type
1	✗	VDDcore1_to_VDDsw_0	VDDcore1	psw_cntrl==0 && psw_cntrl1==1	VDDsw	LPA_SUPPLY	uninitialized	uninitialized

```
CONNECTION-Property
    VDDcore1_to_VDDsw_0      NV
Support-Signals
    psw_cntrl                0
    psw_cntrl1               1
    VDDcore1                 (FULL_ON, 1.0000v)
    VDDsw                    (OFF, 0.0000v)
    clk                      1
G3
```

Figure 6. Example waveform of the failing connection.

As shown in Fig. 6, VDDsw supply state is OFF when VDDCore1 supply is FULL_ON. The enable condition is also true as shown in timing diagram. The supplies are not functionally connected even they can be structurally connected.

Design Result:

The entire flow has been exercised on a customer design and all power aware formal properties were proven/falsified as per the expectation.

TABLE1
DESIGN RESULTS

Design Size	1M gate Design; 2 hour 10 minutes, 94 GB Peak Memory
Power Aware Formal Properties	77 total, 64 Proven, 13 Falsified; 40 minutes and 32 GB Peak memory

REFERENCES

[1] VC Low Power Documentation 2019.06. https://solvnetplus.synopsys.com/s/article/VC-Low-Power-User-Guide-P-2019-06-SP2
[2] VC Formal Documentation 2019.06. https://solvnetplus.synopsys.com/s/article/VC-Formal-User-Guide-P-2019-06-SP2
[3] IEEE UPF LRM . https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7445797
[4] VCS-NLP. https://solvnetplus.synopsys.com/s/article/VCS-Native-Low-Power-NLP-User-Guide
[5] K. M. Harer and P. Tiwari, Synopsys Inc, 2016. “System on chip I/O connectivity verification in presence of low power design considerations.” U.S. Patent No. 9514267.