2022
DESIGN AND VERIFICATION™
DVCON
CONFERENCE AND EXHIBITION
UNITED STATES
SAN JOSE, CA, USA
FEBRUARY 28 - MARCH 3, 2022

Mixed-Signal Functional Verification Methodology for Embedded Non-Volatile Memory Using ESP Simulation

S. Do, J. Park, D. Kim, J. Jang

SAMSUNG FOUNDRY

accellera
SYSTEMS INITIATIVE

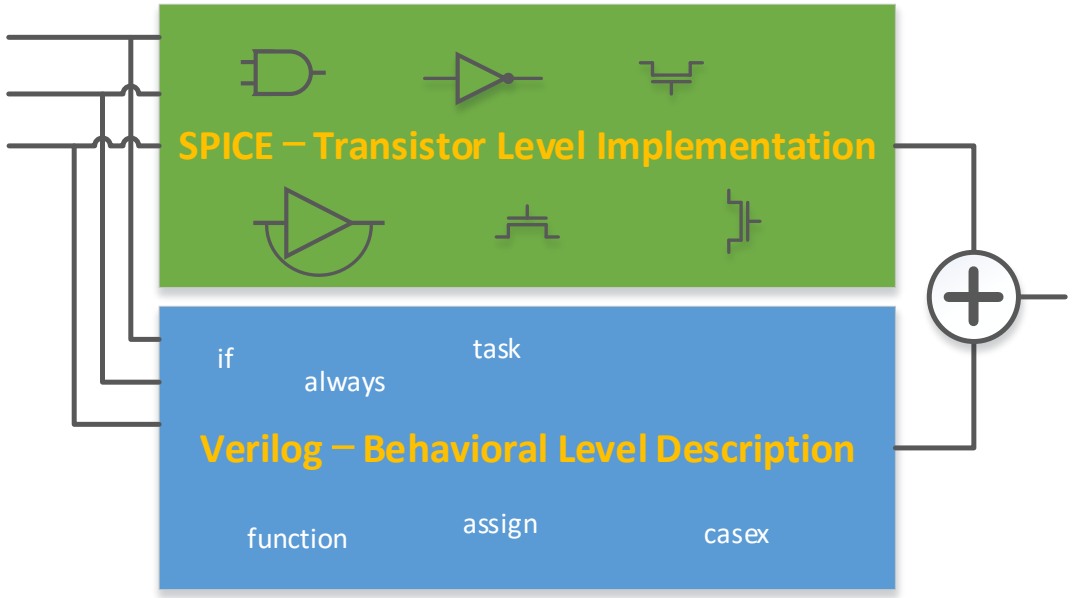
Index

- Functional Verification
- ESP Simulation
- ESP vs Our Proposed Verification Methodology
- Netlist Extraction & Module Conversion
  - Example 1 – Transistor replacement
  - Example 2 – Loop cutting
  - Example 3 – Behavior modeling
- Experimental Result
- Conclusion
- Question

Functional Verification

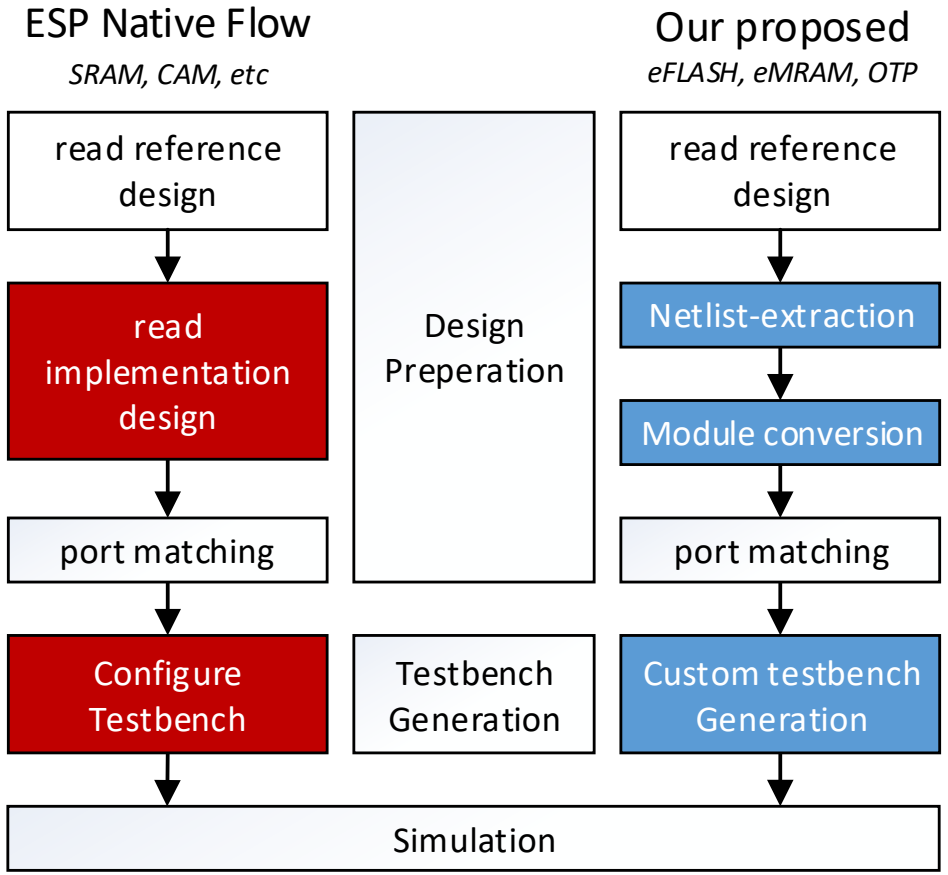
- IP verification is one of the key feature of Foundry business.
- Functional verification is essential part of IP Verification.
- Various EDA tools support functional verification for digital IPs.
  - Formality(Synopsys), Conformal(Cadence)
- Analog circuits are verified using SPICE simulation for validating functional operation.
- Normally, Mixed-signal IPs are big and unable to run SPICE simulation.

ESP



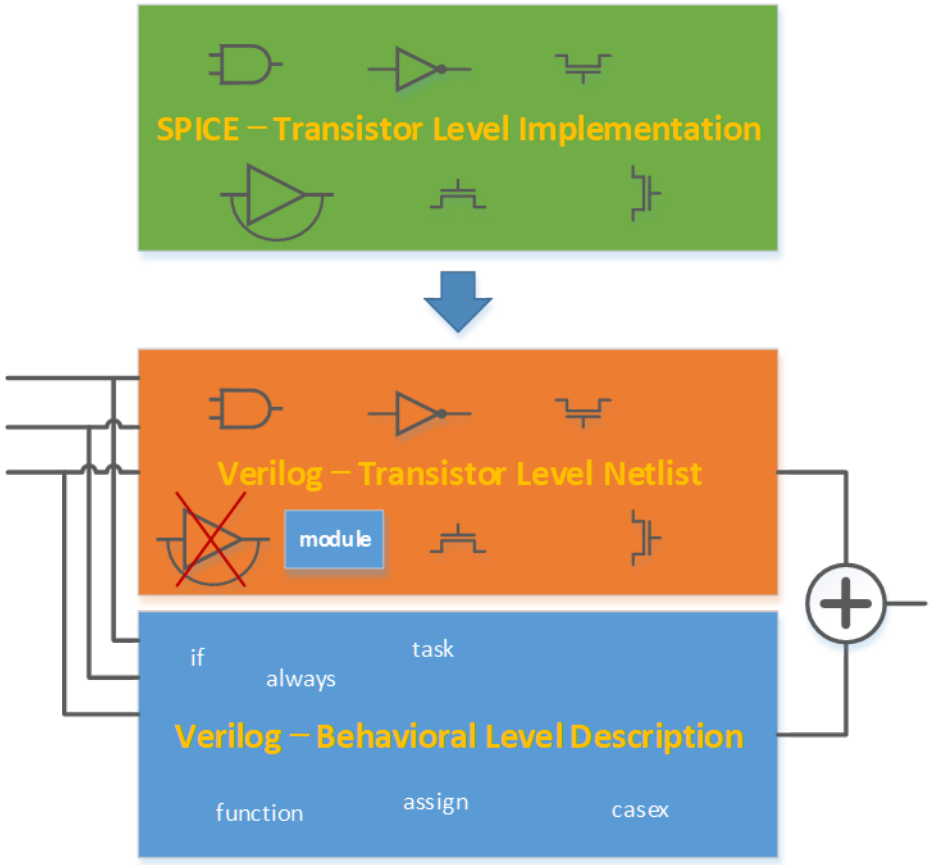
- The only EDA tool which can support direct functional comparison between SPICE netlist & Verilog model.
- Using ‘Symbol’ data type, compare the verification result between dynamic & static verification.

ESP vs Our proposed verification Methodology



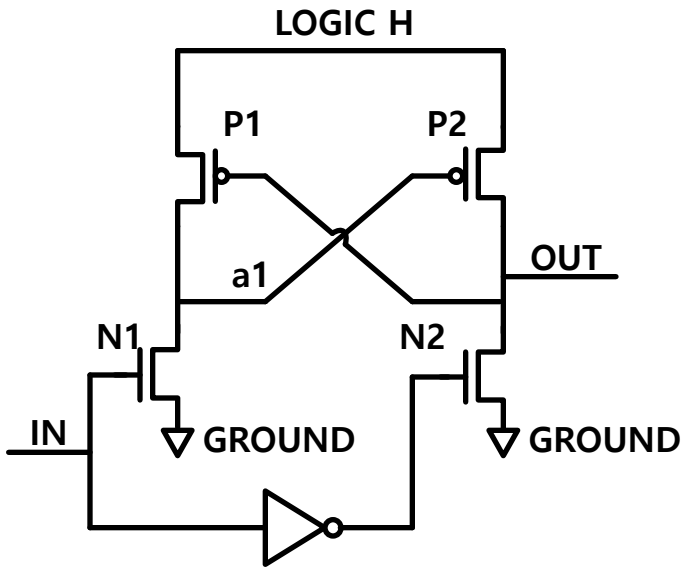
- ESP unsupported spice model
  - BJT, NVM bit-cell, etc
- ESP does not support analog simulation.
- ESP does not support asynchronous IPs
- Two steps enables unsupported IP
  - Netlist-Extraction
  - Module conversion
- Custom testbench should be prepared by verification engineer.

Netlist Extraction



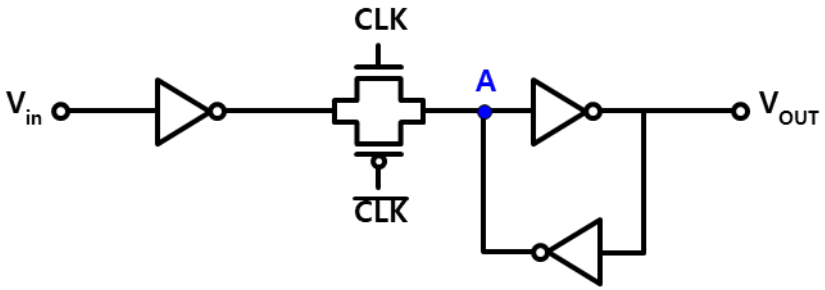
- For extracting the base netlist, use Cadence Virtuoso netlist extraction feature from schematic.
- Base netlist cannot use directly for simulation, and to be modified by engineer which did not work properly when directly converted for Verilog by EDA tools.
- Following module conversion enable the extracted netlist to run the simulation.

Module conversion - Ex1) TR replacement



- As the signals change, there is a conflict between the logic high and the ground, resulting in an unknown (X) value.
- Solve the problem by using rnmos/rpmos that changes the voltage priority so that even if a conflict occurs, it follows the high priority voltage.

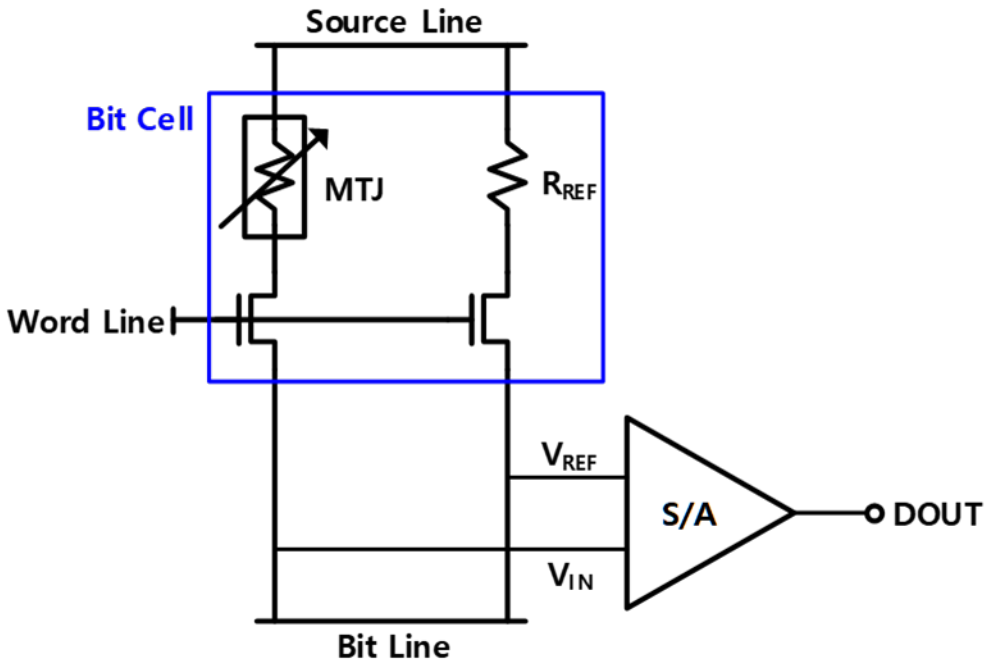
Module conversion - Ex2) Loop cutting




```
module Latch (VIN, VOUT, CLK, CLKB);
  output  VOUT;
  input  VIN, CLK, CLKB;
  always @(CLK==1'b1) begin
   if ( CLKB ==1'b0 ) begin
     VOUT = VIN;
   end
   else begin
    VOUT = 1'bx;
   end
  end
endmodule
```

- Unintended x states in circuits with feedback loop
- Methods to remove unclear state
  - Signal priority
  - Behavior modeling

Module conversion - Ex3) Behavior modeling



- Reason for bit cell modeling.
  - Using analog signals (SL, BL, VREF ..)
  - MTJ device doesn't support digital simulation.
- Write modeling
  - Address and data Check
  - Write data in a virtual cell
- Read modeling
  - Check written data in a virtual cell
  - Output (V_{REF}, V_{IN} according to written data)

Experimental Result – Symbolic coverage

Type		Non Symbolic Nets	Total Nets	Symbolic Net Coverage	Line Coverage
eFLASH	imp	90885	259865	65.03%	99.71%
	ref	3178	3869	17.86%	
eMRAM	imp	450248	948651	52.54%	99.81%
	ref	52855	86199	38.68%	
OTP	imp	740523	1206951	38.65%	99.27%
	ref	2228	2586	13.84%	

- Symbolic net coverage of imp. Model is higher than behavior model.
- For all IPs, line coverage is over than 99%
- Symbolic net coverage cannot be 100% due to the functional characteristic of IPs.

Experimental Result – Computing Power

< Our simulation result >

Memory Type	Instance Size	Memory Requirement	Runtime
eFLASH	8Mb	769MB	18min
eMRAM	44Mb	4.37GB	1hr 13min
OTP	32Kb	2.79GB	30min

< ARM's simulation result >

Memory Type	Instance Size	Memory Requirement	Runtime
eMRAM	8Mb	4GB	60min
	16Mb	8GB	~6hrs

- From the simulation runtime result, our proposed method works fast and proved by various types of NVM.
- Compared with ARM's simulation result, we can verify that our proposed methodology is much efficient in terms of computing power.

Conclusion

- Mixed-Signal functional verification is difficult problem.
- Using our proposed method, we can enable the unsupported IPs for ESP simulation.
  - Overcome limitation of ESP simulation (unsupported device / async)
- Our proposed methodology can adopt other mixed-signal IPs and it also support conventional Verilog simulation using Xcelium or VCS.
- Our proposed method is much efficient way to verify the mixed-signal IPs compared with ARM’s method.

Questions ?

- Contact : sanggi.do@samsung.com