2024
DESIGN AND VERIFICATION™
DVCON
CONFERENCE AND EXHIBITION
UNITED STATES
SAN JOSE, CA, USA
MARCH 4-7, 2024

Variation-Aware Modeling Method for MRAM Behavior Model using System-Verilog

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

Samsung Electronics, Foundry Division

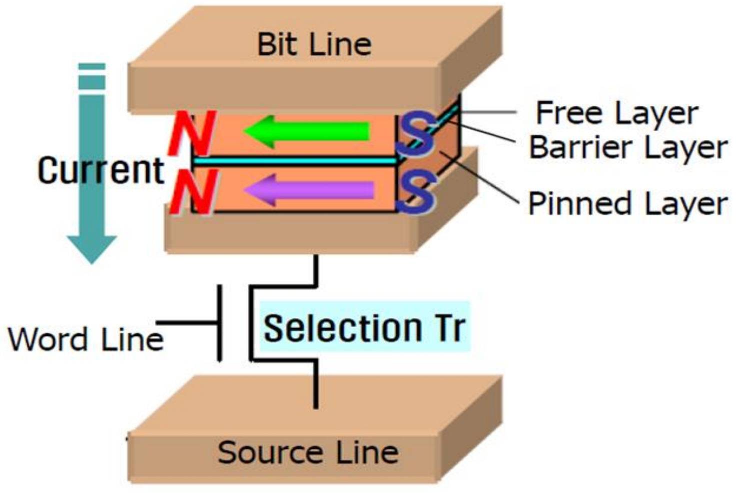
Samsung Foundry

accellera
SYSTEMS INITIATIVE

Index

MRAM Bit-cell

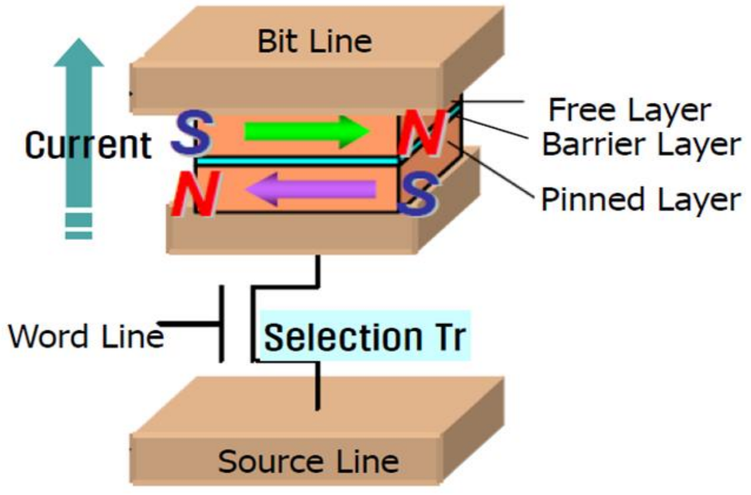
Write "0"



Electrons flow from pinned layer to free layer.

→ Low Resistance state

Write "1"



Electrons flow from free layer to pinned layer.

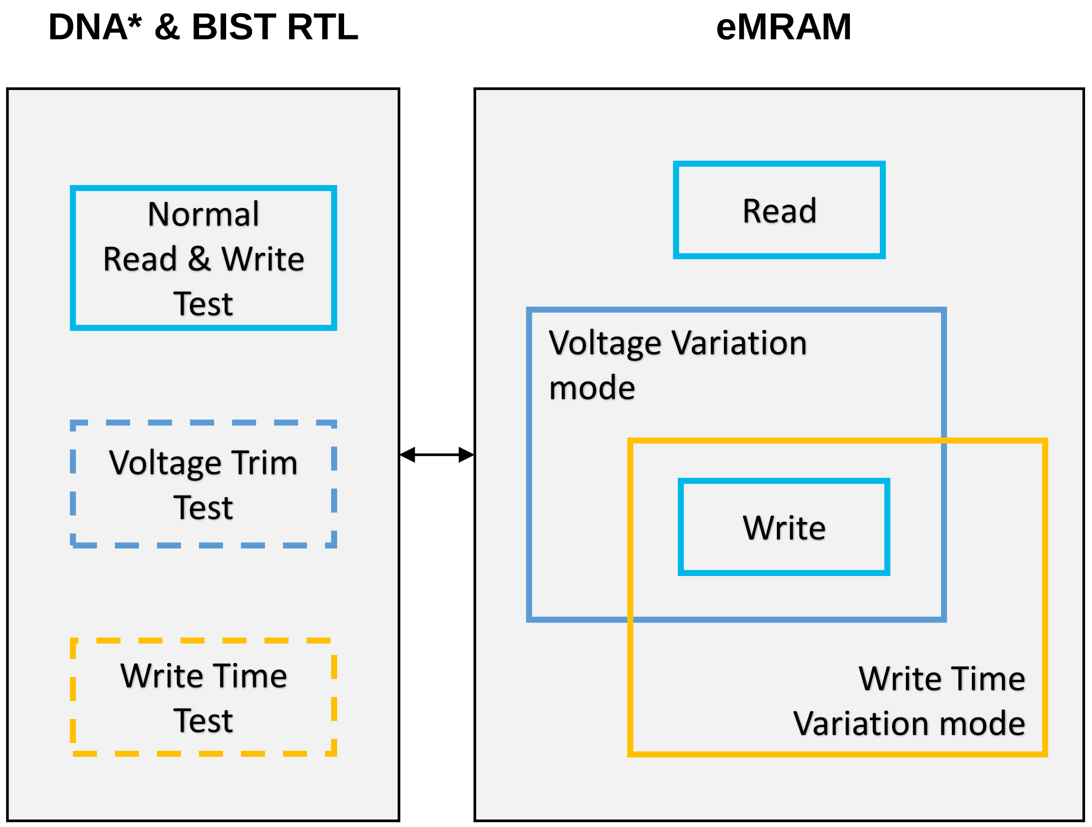
→ High Resistance state

Special Characteristics of Write Possibility

- Time Depend Possibility
  - Single Write Operation does not guarantee write operation
  - Probabilistic write operation modeling does not guarantee exact write operation for simulation.
  - After write guarantee specification time, write should be guarantee.
- Voltage Depend Possibility
  - Reference voltage characteristic is individual properties for each MRAM bit-cell.
  - Voltage characteristic follow the process variation.
  - If write failed, saved data could be reserve or inversed or crashed.

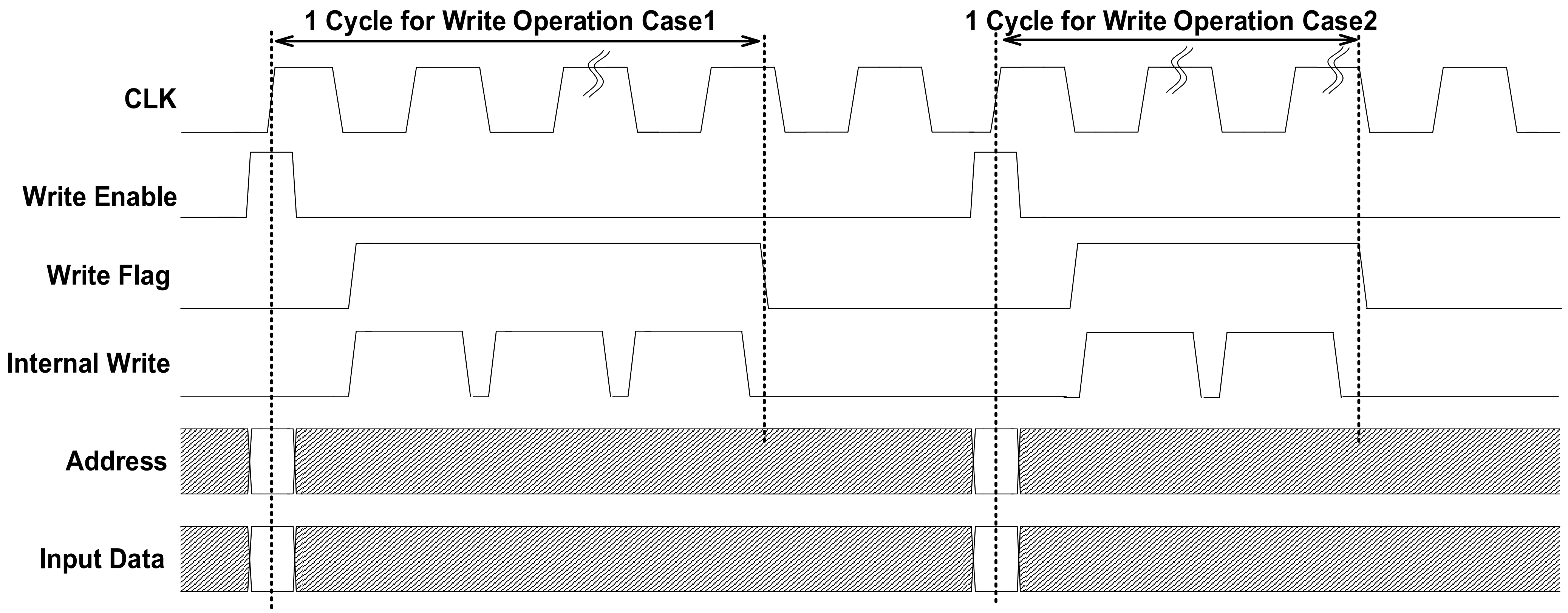
Why need & Where to use ?

- Enables Design Verification for BIST RTL
  - For Memory BIST, commercial tool does not support MRAM
  - MRAM use custom BIST RTL to verify the memory operation
  - Probabilistic MRAM model support verification of the DNA RTL & custom BIST RTL.

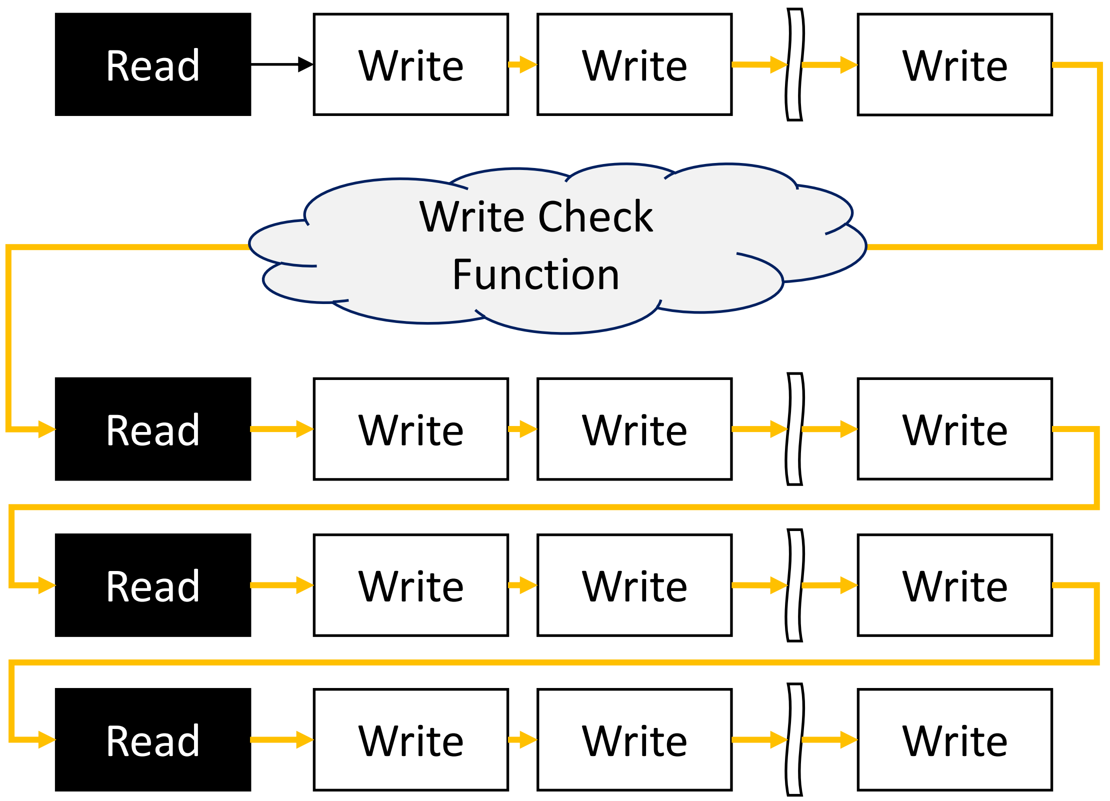


DNA* : Digital Non-volatile Assist

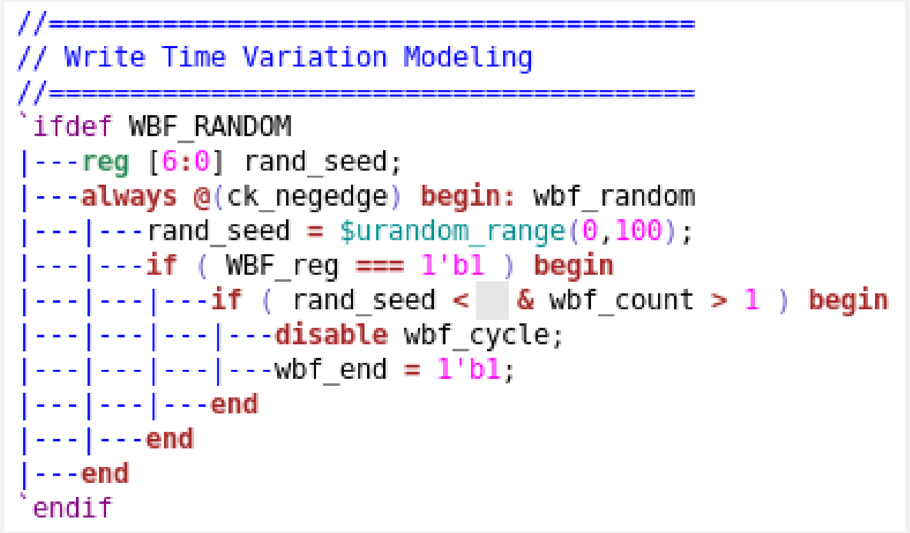
Write Operation Timing Diagram


1 Cycle for Write Operation Case1
1 Cycle for Write Operation Case2
CLK
Write Enable
Write Flag
Internal Write
Address
Input Data

Write Possibility Modeling

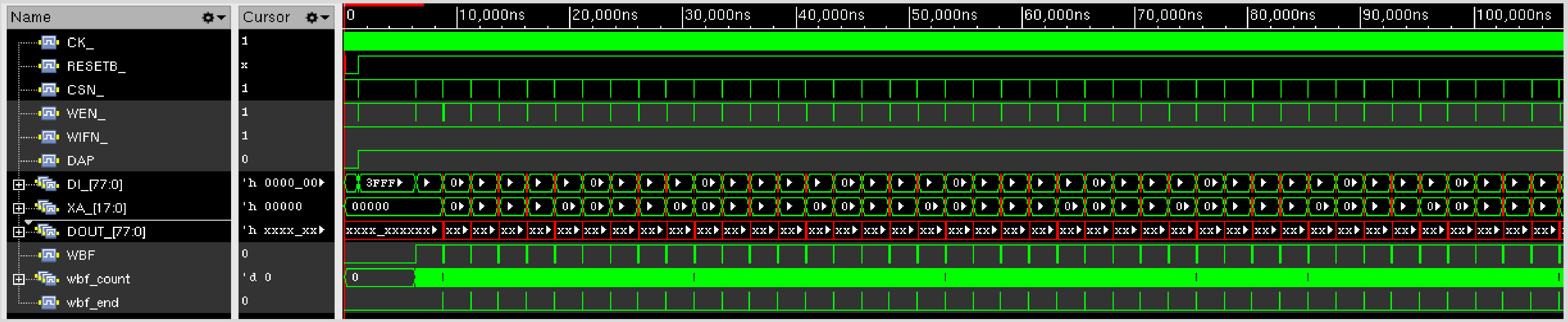


- Verilog modeling code
  - Decide write pass for every single internal write operation.
  - Decision is occur between WBF min/max spec.
  - Independent for bit-cell location or address

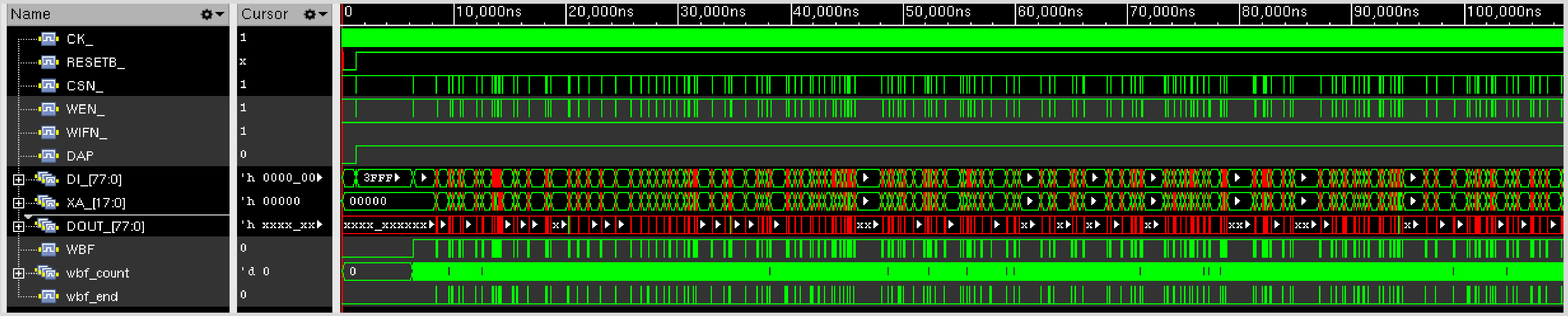


```
//========================================
// Write Time Variation Modeling
//========================================
`ifdef WBF_RANDOM
|---reg [6:0] rand_seed;
|---always @(ck_negedge) begin: wbf_random
|---|---rand_seed = $urandom_range(0,100);
|---|---if ( WBF_reg === 1'b1 ) begin
|---|---|---if ( rand_seed <    & wbf_count > 1 ) begin
|---|---|---|---disable wbf_cycle;
|---|---|---|---wbf_end = 1'b1;
|---|---|---end
|---|---end
|---end
`endif
```

Simulation Result

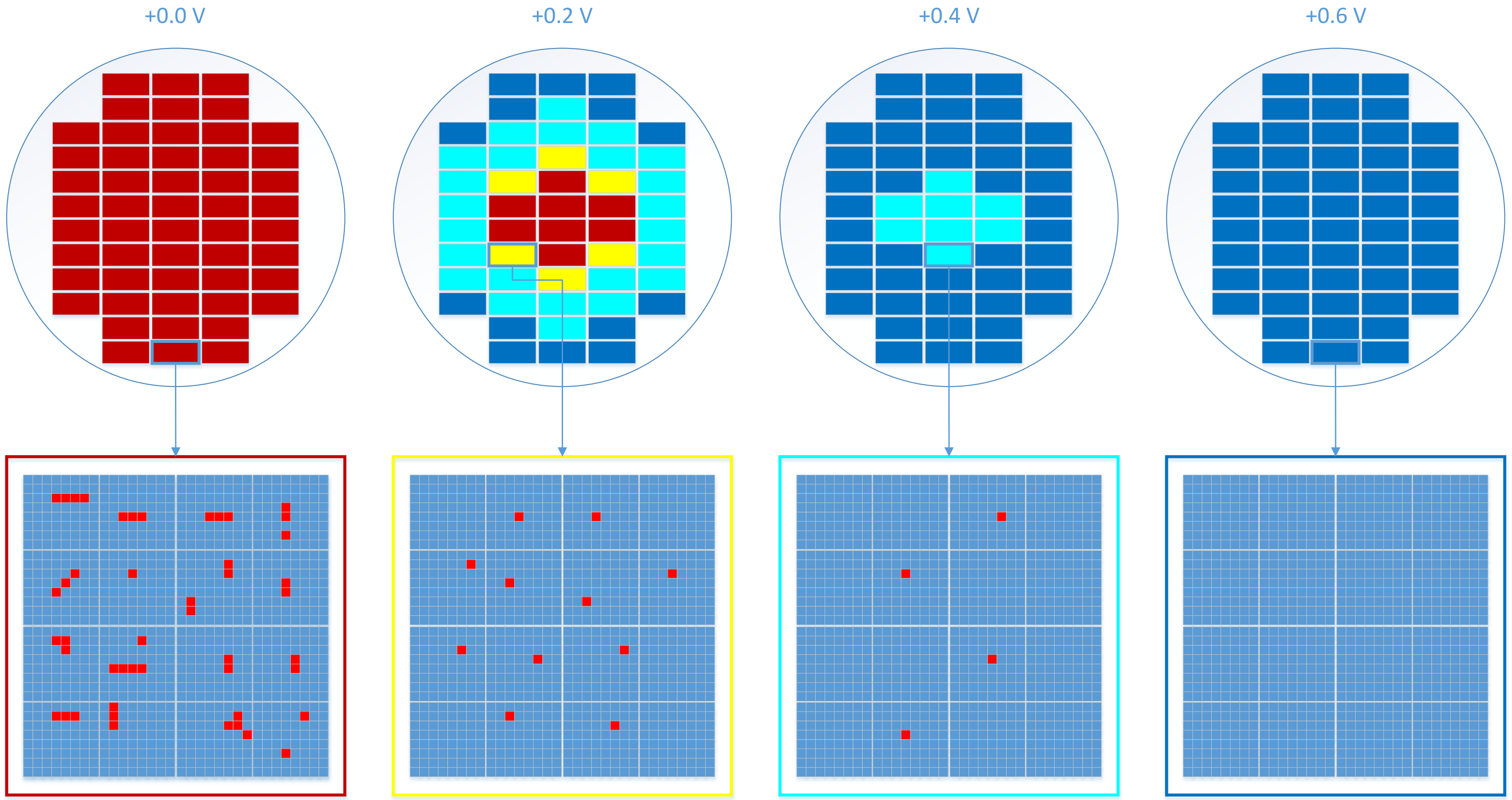


Name	Cursor
CK_	1
RESETB_	x
CSN_	1
WEN_	1
WIFN_	1
DAP	0
DI_[77:0]	'h 0000_00
XA_[17:0]	'h 00000
DOUT_[77:0]	'h xxxx_xx
WBF	0
wbf_count	'd 0
wbf_end	0



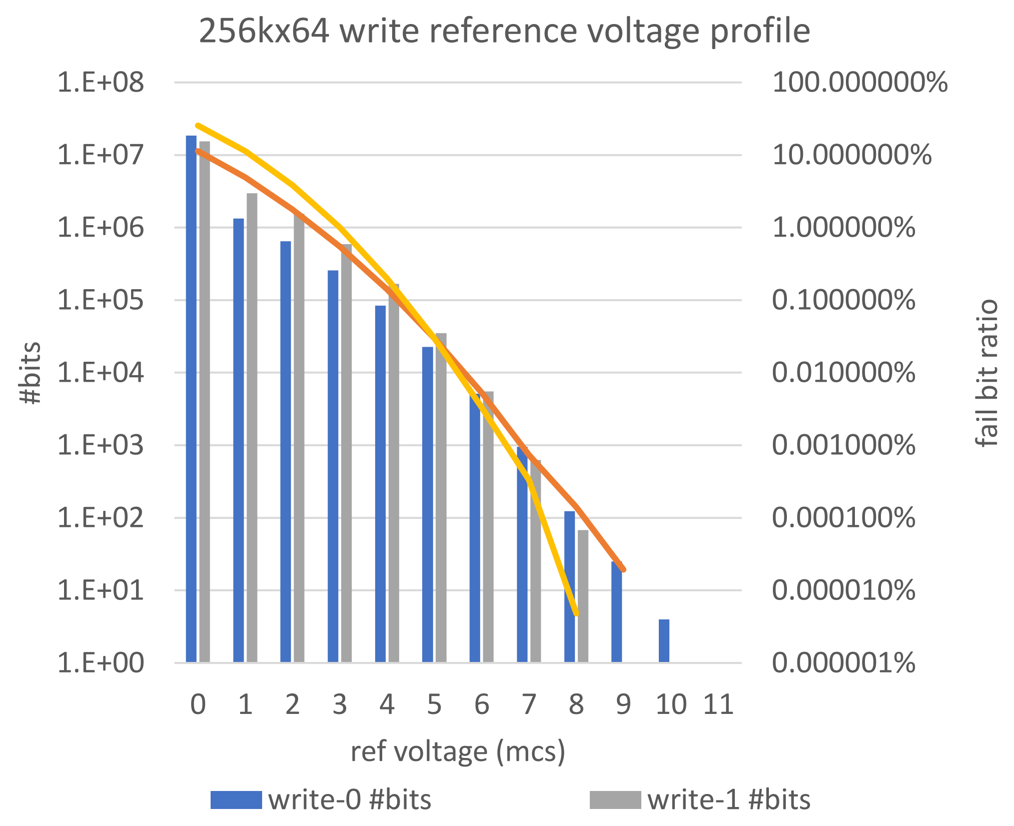
Name	Cursor
CK_	1
RESETB_	x
CSN_	1
WEN_	1
WIFN_	1
DAP	0
DI_[77:0]	'h 0000_00
XA_[17:0]	'h 00000
DOUT_[77:0]	'h xxxx_xx
WBF	0
wbf_count	'd 0
wbf_end	0

How Voltage Variation Effects Yield



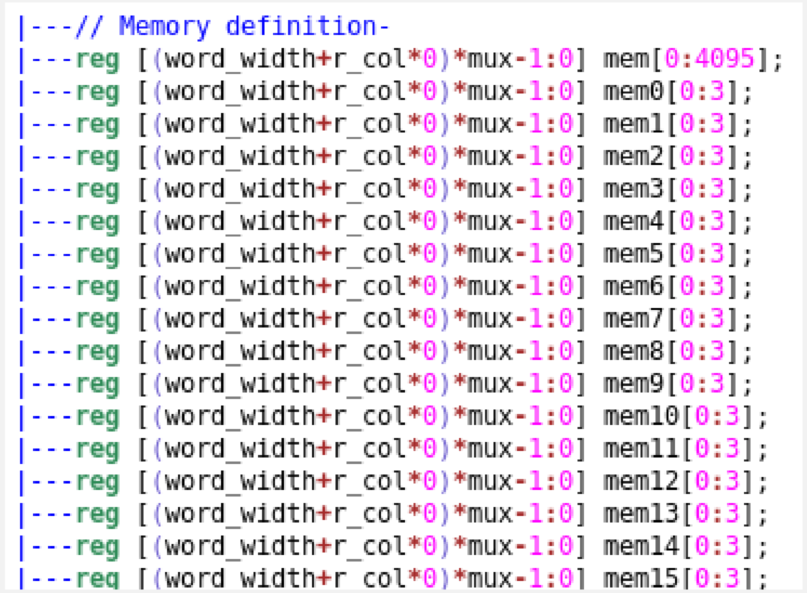
Write Voltage Trimming Configuration

Write-0	FAIL Ratio	Failure bit-cell # (ideal)		Write-1	FAIL Ratio	Failure bit-cell # (ideal)	
Trim-Level		16Mb	128Mb	Trim-Level		16Mb	128Mb
0	21.83500%	3,663,305	29,306,443	0	45.57641%	7,646,453	61,171,625
1	11.08118%	1,859,114	14,872,908	1	25.24925%	4,236,122	33,888,975
2	4.77904%	801,789	6,414,313	2	11.08118%	1,859,114	14,872,908
3	1.73814%	291,611	2,332,890	3	3.77202%	632,840	5,062,717
4	0.53009%	88,935	711,478	4	0.98153%	164,674	1,317,391
5	0.13499%	22,648	181,180	5	0.19330%	32,431	259,447
6	0.02861%	4,800	38,402	6	0.02861%	4,800	38,402
7	0.00504%	845	6,758	7	0.00317%	531	4,251
8	0.00073%	123	986	8	0.00026%	44	351
9	0.00009%	15	119	9	0.00002%	3	21
10	0.00001%	1	12	10	0.00000%	0	1
11	0.00000%	0	1	11	0.00000%	0	0
12	0.00000%	0	0	12	0.00000%	0	0
13	0.00000%	0	0	13	0.00000%	0	0
14	0.00000%	0	0	14	0.00000%	0	0
15	0.00000%	0	0	15	0.00000%	0	0



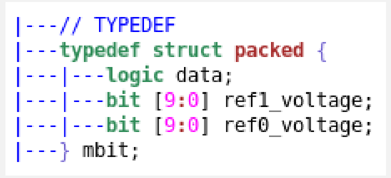
Bit-Cell Modeling Technique

- Previous MRAM bit-cell Array modeling

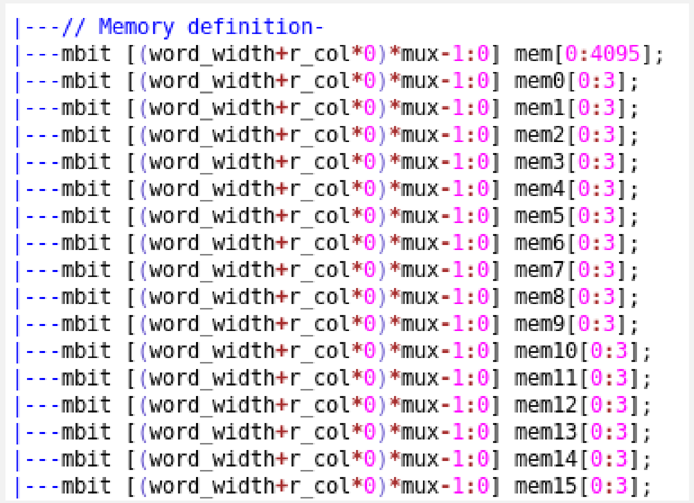


```
|---// Memory definition-
|---reg [(word_width+r_col*0)*mux-1:0] mem[0:4095];
|---reg [(word_width+r_col*0)*mux-1:0] mem0[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem1[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem2[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem3[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem4[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem5[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem6[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem7[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem8[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem9[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem10[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem11[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem12[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem13[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem14[0:3];
|---reg [(word_width+r_col*0)*mux-1:0] mem15[0:3];
```

- New MRAM Bit-cell and Array modeling

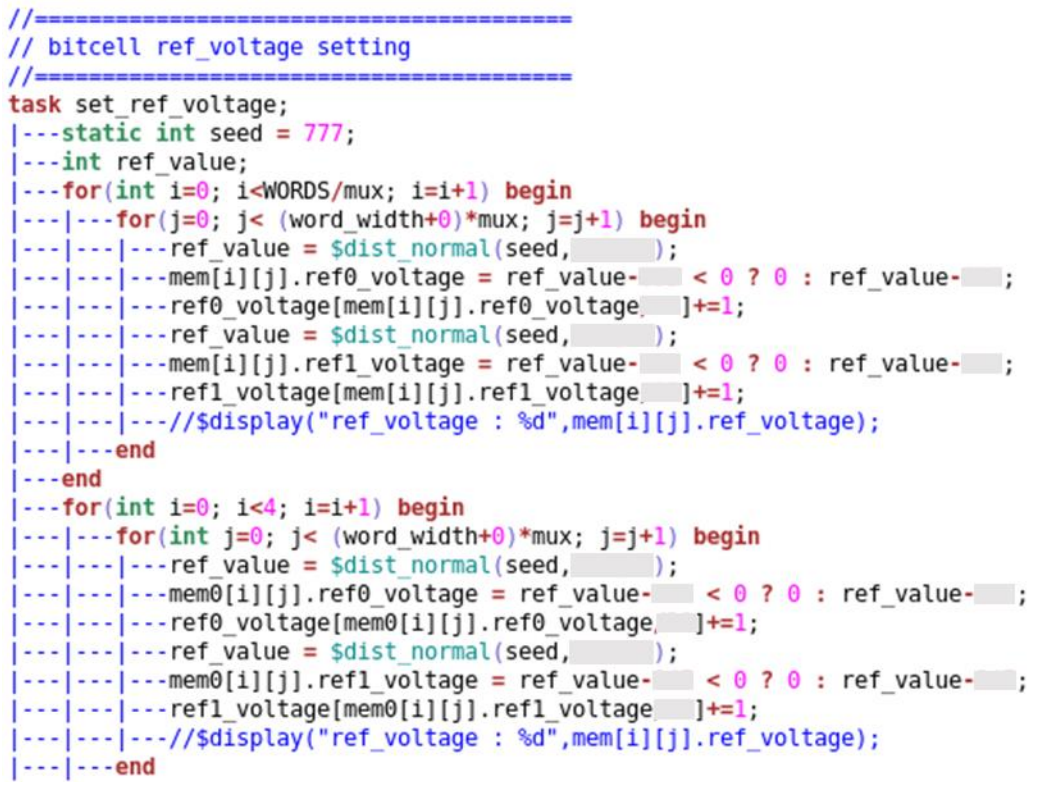


```
|---// TYPEDEF
|---typedef struct packed {
|---|---logic data;
|---|---bit [9:0] ref1_voltage;
|---|---bit [9:0] ref0_voltage;
|---} mbit;
```

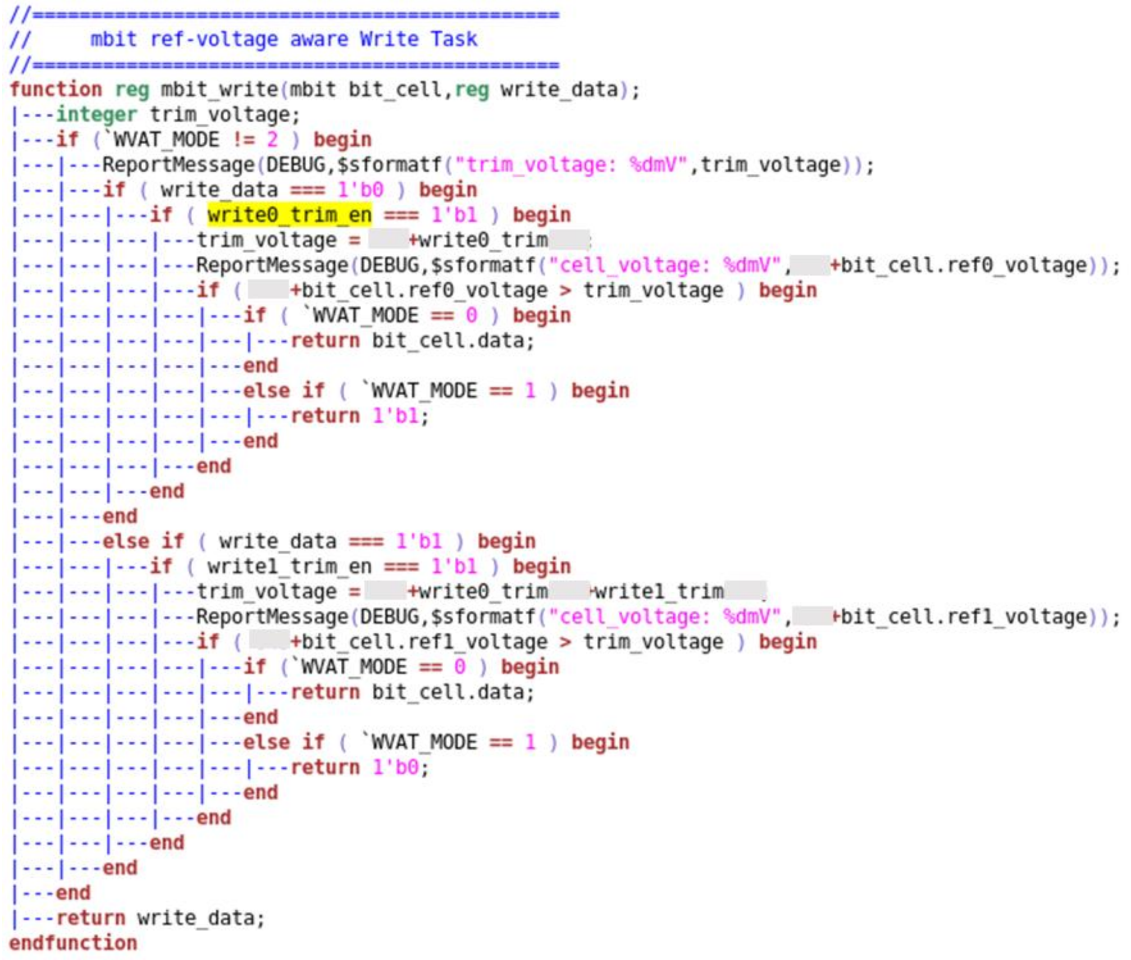


```
|---// Memory definition-
|---mbit [(word_width+r_col*0)*mux-1:0] mem[0:4095];
|---mbit [(word_width+r_col*0)*mux-1:0] mem0[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem1[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem2[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem3[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem4[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem5[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem6[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem7[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem8[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem9[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem10[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem11[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem12[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem13[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem14[0:3];
|---mbit [(word_width+r_col*0)*mux-1:0] mem15[0:3];
```

Ref_voltage profile setting

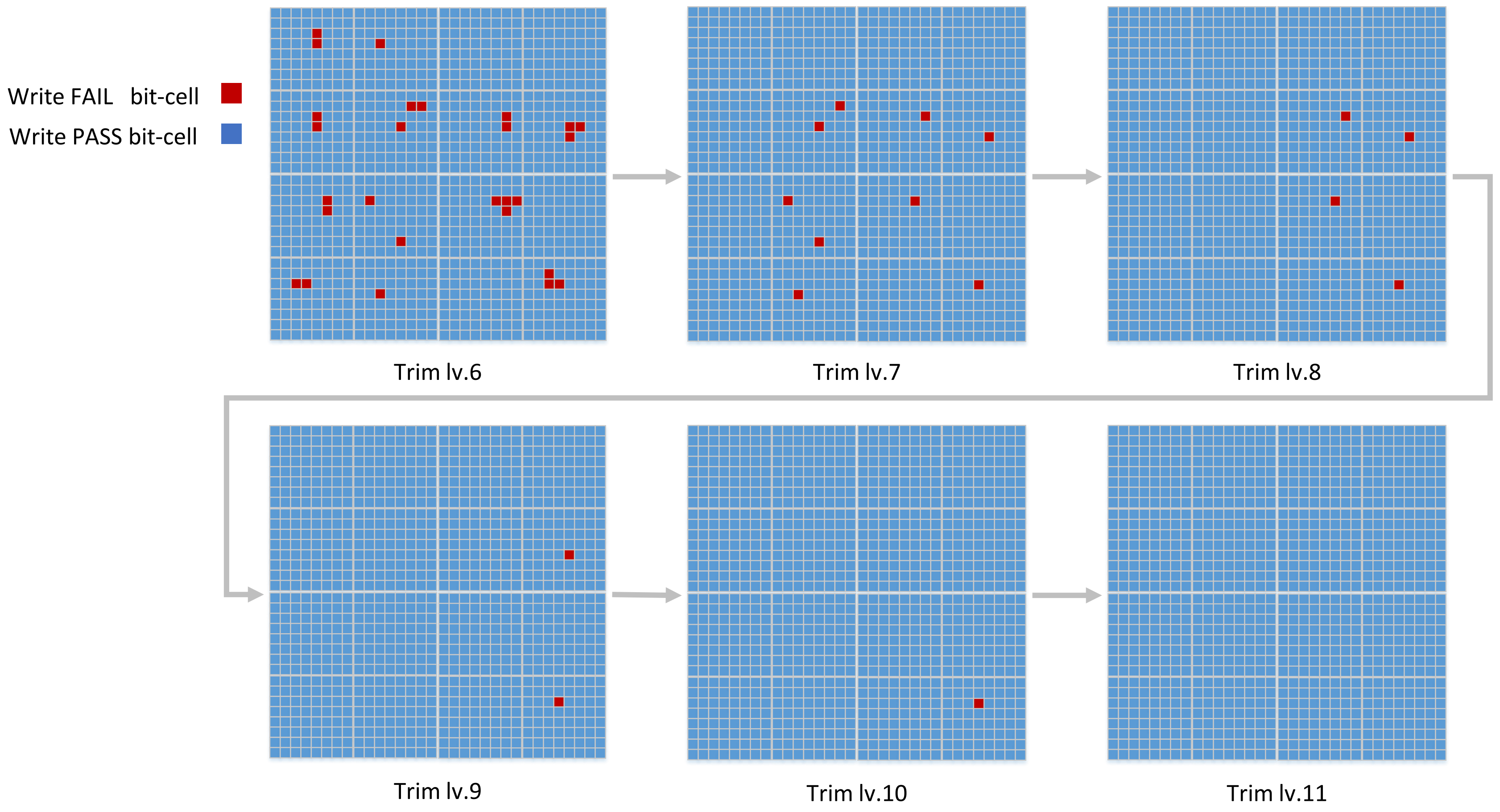


```
//=========================================
// bitcell ref_voltage setting
//=========================================
task set_ref_voltage;
|---static int seed = 777;
|---int ref_value;
|---for(int i=0; i<WORDS/mux; i=i+1) begin
|---|---for(j=0; j< (word_width+0)*mux; j=j+1) begin
|---|---|---ref_value = $dist_normal(seed,[redacted]);
|---|---|---mem[i][j].ref0_voltage = ref_value-[redacted] < 0 ? 0 : ref_value-[redacted];
|---|---|---ref0_voltage[mem[i][j].ref0_voltage[redacted]]+=1;
|---|---|---ref_value = $dist_normal(seed,[redacted]);
|---|---|---mem[i][j].ref1_voltage = ref_value-[redacted] < 0 ? 0 : ref_value-[redacted];
|---|---|---ref1_voltage[mem[i][j].ref1_voltage[redacted]]+=1;
|---|---|---//$display("ref_voltage : %d",mem[i][j].ref_voltage);
|---|---end
|---end
|---for(int i=0; i<4; i=i+1) begin
|---|---for(int j=0; j< (word_width+0)*mux; j=j+1) begin
|---|---|---ref_value = $dist_normal(seed,[redacted]);
|---|---|---mem0[i][j].ref0_voltage = ref_value-[redacted] < 0 ? 0 : ref_value-[redacted];
|---|---|---ref0_voltage[mem0[i][j].ref0_voltage[redacted]]+=1;
|---|---|---ref_value = $dist_normal(seed,[redacted]);
|---|---|---mem0[i][j].ref1_voltage = ref_value-[redacted] < 0 ? 0 : ref_value-[redacted];
|---|---|---ref1_voltage[mem0[i][j].ref1_voltage[redacted]]+=1;
|---|---|---//$display("ref_voltage : %d",mem[i][j].ref_voltage);
|---|---end
```

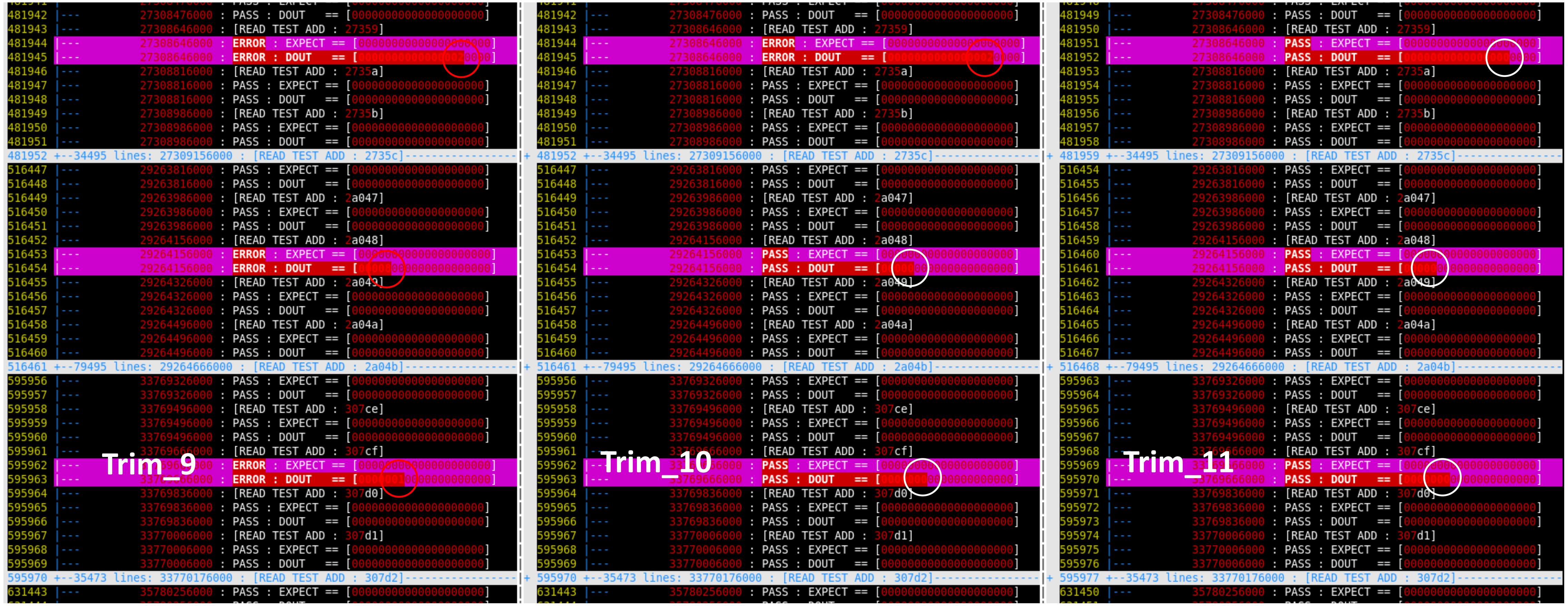


```
//==========================================
//     mbit ref-voltage aware Write Task
//==========================================
function reg mbit_write(mbit bit_cell,reg write_data);
|---integer trim_voltage;
|---if (`WVAT_MODE != 2 ) begin
|---|---ReportMessage(DEBUG,$sformatf("trim_voltage: %dmV",trim_voltage));
|---|---if ( write_data === 1'b0 ) begin
|---|---|---if ( write0_trim_en === 1'b1 ) begin
|---|---|---|---trim_voltage = [redacted]+write0_trim[redacted]
|---|---|---|---ReportMessage(DEBUG,$sformatf("cell_voltage: %dmV",[redacted]+bit_cell.ref0_voltage));
|---|---|---|---if ( [redacted]+bit_cell.ref0_voltage > trim_voltage ) begin
|---|---|---|---|---if ( `WVAT_MODE == 0 ) begin
|---|---|---|---|---|---return bit_cell.data;
|---|---|---|---|---end
|---|---|---|---|---else if ( `WVAT_MODE == 1 ) begin
|---|---|---|---|---|---return 1'b1;
|---|---|---|---|---end
|---|---|---|---end
|---|---|---end
|---|---end
|---|---else if ( write_data === 1'b1 ) begin
|---|---|---if ( write1_trim_en === 1'b1 ) begin
|---|---|---|---trim_voltage = [redacted]+write0_trim[redacted]+write1_trim[redacted]
|---|---|---|---ReportMessage(DEBUG,$sformatf("cell_voltage: %dmV",[redacted]+bit_cell.ref1_voltage));
|---|---|---|---if ( [redacted]+bit_cell.ref1_voltage > trim_voltage ) begin
|---|---|---|---|---if (`WVAT_MODE == 0 ) begin
|---|---|---|---|---|---return bit_cell.data;
|---|---|---|---|---end
|---|---|---|---|---else if ( `WVAT_MODE == 1 ) begin
|---|---|---|---|---|---return 1'b0;
|---|---|---|---|---end
|---|---|---|---end
|---|---|---end
|---|---end
|---end
|---return write_data;
endfunction
```

Cell Level Variation aware Simulation diagram



Simulation Result



Trim_9

```
481942 |---          27308476000 : PASS : DOUT   == [00000000000000000000]
481943 |---          27308646000 : [READ TEST ADD : 27359]
481944 |---          27308646000 : ERROR : EXPECT == [00000000000000000000]
481945 |---          27308646000 : ERROR : DOUT   == [00000000000000020000]
481946 |---          27308816000 : [READ TEST ADD : 2735a]
481947 |---          27308816000 : PASS : EXPECT == [00000000000000000000]
481948 |---          27308816000 : PASS : DOUT   == [00000000000000000000]
481949 |---          27308986000 : [READ TEST ADD : 2735b]
481950 |---          27308986000 : PASS : EXPECT == [00000000000000000000]
481951 |---          27308986000 : PASS : DOUT   == [00000000000000000000]
481952 +--34495 lines: 27309156000 : [READ TEST ADD : 2735c]-----------------
516447 |---          29263816000 : PASS : EXPECT == [00000000000000000000]
516448 |---          29263816000 : PASS : DOUT   == [00000000000000000000]
516449 |---          29263986000 : [READ TEST ADD : 2a047]
516450 |---          29263986000 : PASS : EXPECT == [00000000000000000000]
516451 |---          29263986000 : PASS : DOUT   == [00000000000000000000]
516452 |---          29264156000 : [READ TEST ADD : 2a048]
516453 |---          29264156000 : ERROR : EXPECT == [00000000000000000000]
516454 |---          29264156000 : ERROR : DOUT   == [00008000000000000000]
516455 |---          29264326000 : [READ TEST ADD : 2a049]
516456 |---          29264326000 : PASS : EXPECT == [00000000000000000000]
516457 |---          29264326000 : PASS : DOUT   == [00000000000000000000]
516458 |---          29264496000 : [READ TEST ADD : 2a04a]
516459 |---          29264496000 : PASS : EXPECT == [00000000000000000000]
516460 |---          29264496000 : PASS : DOUT   == [00000000000000000000]
516461 +--79495 lines: 29264666000 : [READ TEST ADD : 2a04b]-----------------
595956 |---          33769326000 : PASS : EXPECT == [00000000000000000000]
595957 |---          33769326000 : PASS : DOUT   == [00000000000000000000]
595958 |---          33769496000 : [READ TEST ADD : 307ce]
595959 |---          33769496000 : PASS : EXPECT == [00000000000000000000]
595960 |---          33769496000 : PASS : DOUT   == [00000000000000000000]
595961 |---          33769666000 : [READ TEST ADD : 307cf]
595962 |---          33769666000 : ERROR : EXPECT == [00000000000000000000]
595963 |---          33769666000 : ERROR : DOUT   == [00000010000000000000]
595964 |---          33769836000 : [READ TEST ADD : 307d0]
595965 |---          33769836000 : PASS : EXPECT == [00000000000000000000]
595966 |---          33769836000 : PASS : DOUT   == [00000000000000000000]
595967 |---          33770006000 : [READ TEST ADD : 307d1]
595968 |---          33770006000 : PASS : EXPECT == [00000000000000000000]
595969 |---          33770006000 : PASS : DOUT   == [00000000000000000000]
595970 +--35473 lines: 33770176000 : [READ TEST ADD : 307d2]-----------------
631443 |---          35780256000 : PASS : EXPECT == [00000000000000000000]
```

Trim_10

```
481942 |---          27308476000 : PASS : DOUT   == [00000000000000000000]
481943 |---          27308646000 : [READ TEST ADD : 27359]
481944 |---          27308646000 : ERROR : EXPECT == [00000000000000000000]
481945 |---          27308646000 : ERROR : DOUT   == [00000000000000020000]
481946 |---          27308816000 : [READ TEST ADD : 2735a]
481947 |---          27308816000 : PASS : EXPECT == [00000000000000000000]
481948 |---          27308816000 : PASS : DOUT   == [00000000000000000000]
481949 |---          27308986000 : [READ TEST ADD : 2735b]
481950 |---          27308986000 : PASS : EXPECT == [00000000000000000000]
481951 |---          27308986000 : PASS : DOUT   == [00000000000000000000]
481952 +--34495 lines: 27309156000 : [READ TEST ADD : 2735c]-----------------
516447 |---          29263816000 : PASS : EXPECT == [00000000000000000000]
516448 |---          29263816000 : PASS : DOUT   == [00000000000000000000]
516449 |---          29263986000 : [READ TEST ADD : 2a047]
516450 |---          29263986000 : PASS : EXPECT == [00000000000000000000]
516451 |---          29263986000 : PASS : DOUT   == [00000000000000000000]
516452 |---          29264156000 : [READ TEST ADD : 2a048]
516453 |---          29264156000 : PASS : EXPECT == [00000000000000000000]
516454 |---          29264156000 : PASS : DOUT   == [00000000000000000000]
516455 |---          29264326000 : [READ TEST ADD : 2a049]
516456 |---          29264326000 : PASS : EXPECT == [00000000000000000000]
516457 |---          29264326000 : PASS : DOUT   == [00000000000000000000]
516458 |---          29264496000 : [READ TEST ADD : 2a04a]
516459 |---          29264496000 : PASS : EXPECT == [00000000000000000000]
516460 |---          29264496000 : PASS : DOUT   == [00000000000000000000]
516461 +--79495 lines: 29264666000 : [READ TEST ADD : 2a04b]-----------------
595956 |---          33769326000 : PASS : EXPECT == [00000000000000000000]
595957 |---          33769326000 : PASS : DOUT   == [00000000000000000000]
595958 |---          33769496000 : [READ TEST ADD : 307ce]
595959 |---          33769496000 : PASS : EXPECT == [00000000000000000000]
595960 |---          33769496000 : PASS : DOUT   == [00000000000000000000]
595961 |---          33769666000 : [READ TEST ADD : 307cf]
595962 |---          33769666000 : PASS : EXPECT == [00000000000000000000]
595963 |---          33769666000 : PASS : DOUT   == [00000000000000000000]
595964 |---          33769836000 : [READ TEST ADD : 307d0]
595965 |---          33769836000 : PASS : EXPECT == [00000000000000000000]
595966 |---          33769836000 : PASS : DOUT   == [00000000000000000000]
595967 |---          33770006000 : [READ TEST ADD : 307d1]
595968 |---          33770006000 : PASS : EXPECT == [00000000000000000000]
595969 |---          33770006000 : PASS : DOUT   == [00000000000000000000]
595970 +--35473 lines: 33770176000 : [READ TEST ADD : 307d2]-----------------
631443 |---          35780256000 : PASS : EXPECT == [00000000000000000000]
```

Trim_11

```
481949 |---          27308476000 : PASS : DOUT   == [00000000000000000000]
481950 |---          27308646000 : [READ TEST ADD : 27359]
481951 |---          27308646000 : PASS : EXPECT == [00000000000000000000]
481952 |---          27308646000 : PASS : DOUT   == [00000000000000000000]
481953 |---          27308816000 : [READ TEST ADD : 2735a]
481954 |---          27308816000 : PASS : EXPECT == [00000000000000000000]
481955 |---          27308816000 : PASS : DOUT   == [00000000000000000000]
481956 |---          27308986000 : [READ TEST ADD : 2735b]
481957 |---          27308986000 : PASS : EXPECT == [00000000000000000000]
481958 |---          27308986000 : PASS : DOUT   == [00000000000000000000]
481959 +--34495 lines: 27309156000 : [READ TEST ADD : 2735c]-----------------
516454 |---          29263816000 : PASS : EXPECT == [00000000000000000000]
516455 |---          29263816000 : PASS : DOUT   == [00000000000000000000]
516456 |---          29263986000 : [READ TEST ADD : 2a047]
516457 |---          29263986000 : PASS : EXPECT == [00000000000000000000]
516458 |---          29263986000 : PASS : DOUT   == [00000000000000000000]
516459 |---          29264156000 : [READ TEST ADD : 2a048]
516460 |---          29264156000 : PASS : EXPECT == [00000000000000000000]
516461 |---          29264156000 : PASS : DOUT   == [00000000000000000000]
516462 |---          29264326000 : [READ TEST ADD : 2a049]
516463 |---          29264326000 : PASS : EXPECT == [00000000000000000000]
516464 |---          29264326000 : PASS : DOUT   == [00000000000000000000]
516465 |---          29264496000 : [READ TEST ADD : 2a04a]
516466 |---          29264496000 : PASS : EXPECT == [00000000000000000000]
516467 |---          29264496000 : PASS : DOUT   == [00000000000000000000]
516468 +--79495 lines: 29264666000 : [READ TEST ADD : 2a04b]-----------------
595963 |---          33769326000 : PASS : EXPECT == [00000000000000000000]
595964 |---          33769326000 : PASS : DOUT   == [00000000000000000000]
595965 |---          33769496000 : [READ TEST ADD : 307ce]
595966 |---          33769496000 : PASS : EXPECT == [00000000000000000000]
595967 |---          33769496000 : PASS : DOUT   == [00000000000000000000]
595968 |---          33769666000 : [READ TEST ADD : 307cf]
595969 |---          33769666000 : PASS : EXPECT == [00000000000000000000]
595970 |---          33769666000 : PASS : DOUT   == [00000000000000000000]
595971 |---          33769836000 : [READ TEST ADD : 307d0]
595972 |---          33769836000 : PASS : EXPECT == [00000000000000000000]
595973 |---          33769836000 : PASS : DOUT   == [00000000000000000000]
595974 |---          33770006000 : [READ TEST ADD : 307d1]
595975 |---          33770006000 : PASS : EXPECT == [00000000000000000000]
595976 |---          33770006000 : PASS : DOUT   == [00000000000000000000]
595977 +--35473 lines: 33770176000 : [READ TEST ADD : 307d2]-----------------
631450 |---          35780256000 : PASS : EXPECT == [00000000000000000000]
```

Summary

- To support MRAM probabilistic write, we use user define data type for MRAM memory array.
- Our modeling enables simulation base design verification for DNA & BIST RTL.
- Our modeling support to develop BIST algorithm for Samsung Foundry's MRAM.

Questions

- Contact : sanggi.do@samsung.com