Scan Insertion Tutorial

1.Source: source /proj/cad/startup/profile.synopsys_2018_vcs_2021

2.mkdir ~/SYNOPSYS/WORK

3.copy library file:

```
{engnx03a:~/SYNOPSYS/WORK} cp  /home/cad/synopsys_2007.12/syn/packages/gtech/src_ver/gtech_lib.v .
```

4. WRITE CODE:

```
`timescale 1ns / 1ps
module DFT_example(clk,a,b,c,d,e,F);
input clk; //Scan clk
input a,b,c,d,e;
output F;
wire u, v, w,x,y,z;
GTECH_NOT NOT(e,y);
GTECH_NOR2 NOR2 (w,x,z);
GTECH_FD1 DFF1 (u, clk, F); //DFF which will become scan
GTECH_NAND2 NAND2 (y,z,u);
GTECH_AND2 AND2 (a,b,v);
GTECH_FD1 DFF2 (v, clk, w); //DFF which will become scan
GTECH_OR2 OR (c,d,x);
Endmodule
```

5. WRITE Tb:

```
`timescale 1ns / 1ps
module testbench();
  // Inputs
  reg clk;
  reg a, b, c, d, e;
  // Outputs
  wire F;
  // Instantiate the DFT_example module
  DFT_example dut (
    .clk(clk),
    .a(a),
    .b(b),
    .c(c),
    .d(d),
    .e(e),
    .F(F)
  );
  // Clock generation
  always #5 clk = ~clk;
  // Stimulus
  initial begin
    clk = 0;
    a = 0; b = 0; c = 0; d = 0; e = 0;
    #10; // Initial stabilization
    // Test case 1
    a = 1; b = 0; c = 0; d = 1; e = 0;
    #10
    // Test case 2
    a = 0; b = 1; c = 1; d = 0; e = 1;
    #10
    // Add more test cases as needed
    // End simulation
    $finish;
  end
 // Monitor
  always @(posedge clk) begin
```

```
      $display("Time=%0t a=%b b=%b c=%b d=%b e=%b F=%b", $time, a, b, c, d, e, F);
   end
endmodule
```

6. Compile:

vlogan scan.v scan_tb.v

7. design_vision&

8. Open design_vision environment. In Setup, use class.db, class.db, and class.sdb as the link, target and symbol libraries, respectively.

9. File > Analyze > ADD scan.v >Select>OK

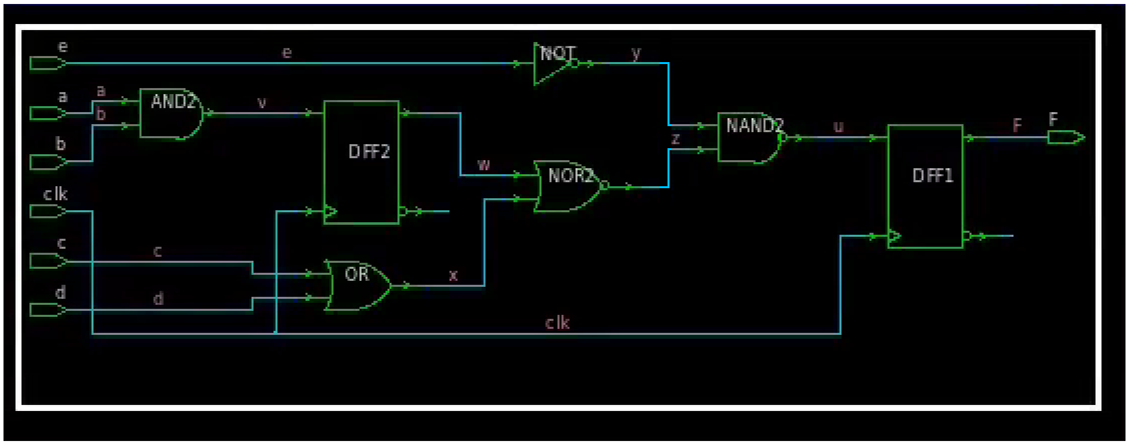
10. File > Elaborate > _ library: click on WORK; Design: click on DFT_example(verilog)> OK

11. File > Save

12. File > read > DFT_example.ddc > OK

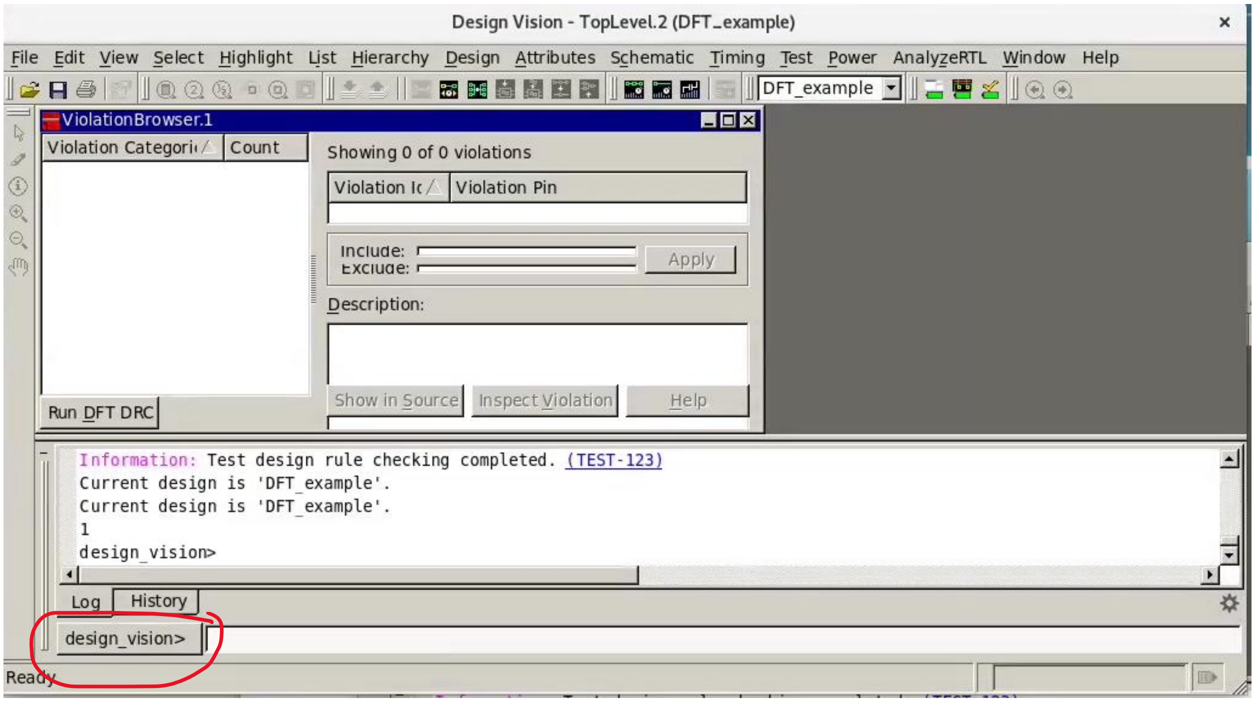
13. When the system level box is shown, double click on icon until logic is displayed.

(14) File > Print Schematic



(15) Click Design→ Compile, check “Fix design rules only” → OK

(16) write all the command lines in the circled portion below:



```
Information: Test design rule checking completed. (TEST-123)
Current design is 'DFT_example'.
Current design is 'DFT_example'.
1
design_vision>
```

```
design_vision> set_scan_configuration -style multiplexed_flip_flop
Accepted scan configuration for modes: all_dft
1
design_vision> set_scan_configuration -chain_count 1
Accepted scan configuration for modes: all_dft
1
design_vision> set_dft_signal -view existing_dft -type ScanClock -port clk -timing {1 10}
Accepted dft signal specification for modes: all_dft
1
design_vision> create_test_protocol -capture_procedure single_clock
In mode: all_dft...

Information: Starting test protocol creation. (TEST-219)
  ...reading user specified clock signals...
Information: Identified system/test clock port clk (1.0,10.0). (TEST-265)
  ...reading user specified asynchronous signals...
1
design_vision> insert_dft

Information: Starting test design rule checking. (TEST-222)
Test Design rule checking did not find violations
Information: Test design rule checking completed. (TEST-123)
  Architecting Scan Chains
  Routing Scan Chains
  Routing Global Signals
  Mapping New Logic
Resetting current test mode
  Beginning Mapping Optimizations
  -------------------------------


                    TOTAL
  ELAPSED          WORST NEG   SETUP    DESIGN
```

```
   TIME     AREA     SLACK     COST    RULE COST        ENDPOINT
  --------- --------- --------- --------- --------- -------------------------


1
design_vision> current_design
Current design is 'DFT_example'.
{DFT_example}
design_vision> dft_drc -verbose
In mode: Internal_scan...
  Design has scan chains in this mode
  Design is scan routed
  Post-DFT DRC enabled

Information: Starting test design rule checking. (TEST-222)
  Loading test protocol
  ...basic checks...
  ...basic sequential cell checks...
  ...checking vector rules...
  ...checking clock rules...
  ...checking scan chain rules...
  ...checking scan compression rules...
  ...checking X-state rules...
  ...checking tristate rules...
  ...extracting scan details...

-----------------------------------------------------------------
  DRC Report
  Total violations: 0 // check this
-----------------------------------------------------------------
Test Design rule checking did not find violations
-----------------------------------------------------------------
  Sequential Cell Report
  0 out of 2 sequential cells have violations
-----------------------------------------------------------------
SEQUENTIAL CELLS WITHOUT VIOLATIONS
      *   2 cells are valid scan cells
         DFF2
         DFF1
....Inferring feed-through connections....
Information: Test design rule checking completed. (TEST-123)
Current design is 'DFT_example'.
Current design is 'DFT_example'.
1
design_vision> check_design -multiple_designs
1
1
design_vision> write_test_protocol -o file_scan.stil
Writing test protocol file '/home/eng/n/nxa200042/cad/ee6303/work/file_scan.stil' for mode 'Internal_scan'...
Warning: The multi-clock protocol requires that the strobe time be before a clock's pulse if it is used for transition fault testing.
(TESTXG-56)
1
design_vision> write -hierarchy -format verilog -output file_scan.v
Writing verilog file '/home/eng/n/nxa200042/cad/ee6303/work/file_scan.v'.
1
```

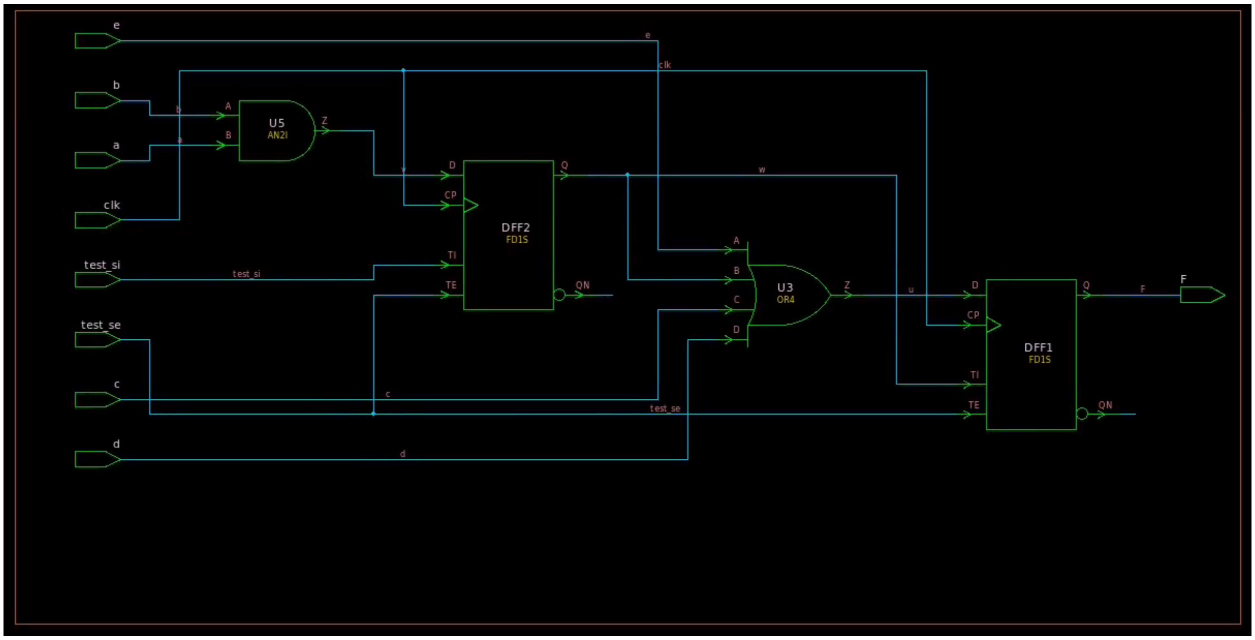
File_scan.v

```
/////////////////////////////////////////////////////////////
// Created by: Synopsys DC Expert(TM) in wire load mode
// Version   : O-2018.06-SP1
// Date      : Tue Apr  9 13:07:05 2024
/////////////////////////////////////////////////////////////


module DFT_example ( clk, a, b, c, d, e, F, test_si, test_se );
  input clk, a, b, c, d, e, test_si, test_se;
  output F;
  wire   w, u, v;

  FD1S DFF2 ( .D(v), .TI(test_si), .TE(test_se), .CP(clk), .Q(w) );
  FD1S DFF1 ( .D(u), .TI(w), .TE(test_se), .CP(clk), .Q(F) );
  AN2I U5 ( .A(b), .B(a), .Z(v) );
  OR4 U3 ( .A(e), .B(w), .C(c), .D(d), .Z(u) );
endmodule
```

(17) Do analyze and elaborate again on the new “file_scan.v” file to see the new netlist with scan cells.

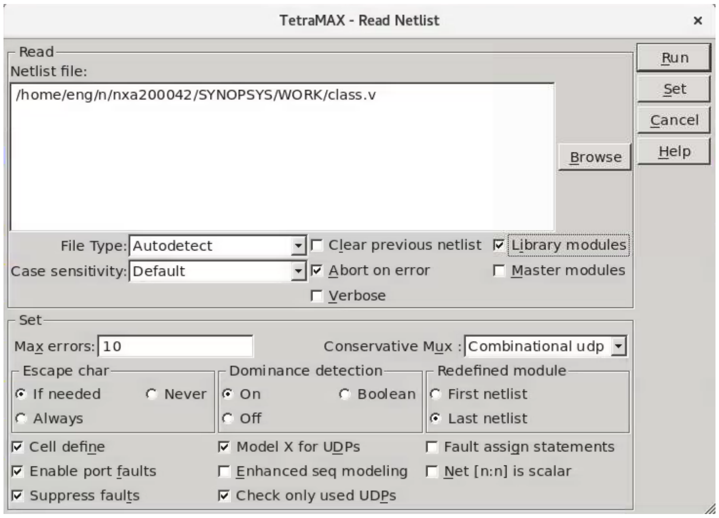


(18) copy file_scan.v and the stil file to SYNOPSYS/WORK

(19) copy class.v file to SYNOPSYS/WORK

```
engnx01a:~/SYNOPSYS/WORK} cp /proj/cad/synopsys/synopsys_2018/syn_vO-2018.06-SP1/doc/syn/dft_tutorial/LIB/class.v ./
```

(20) follow step 2.3 from test tutorial: 2.3 Read Library Models



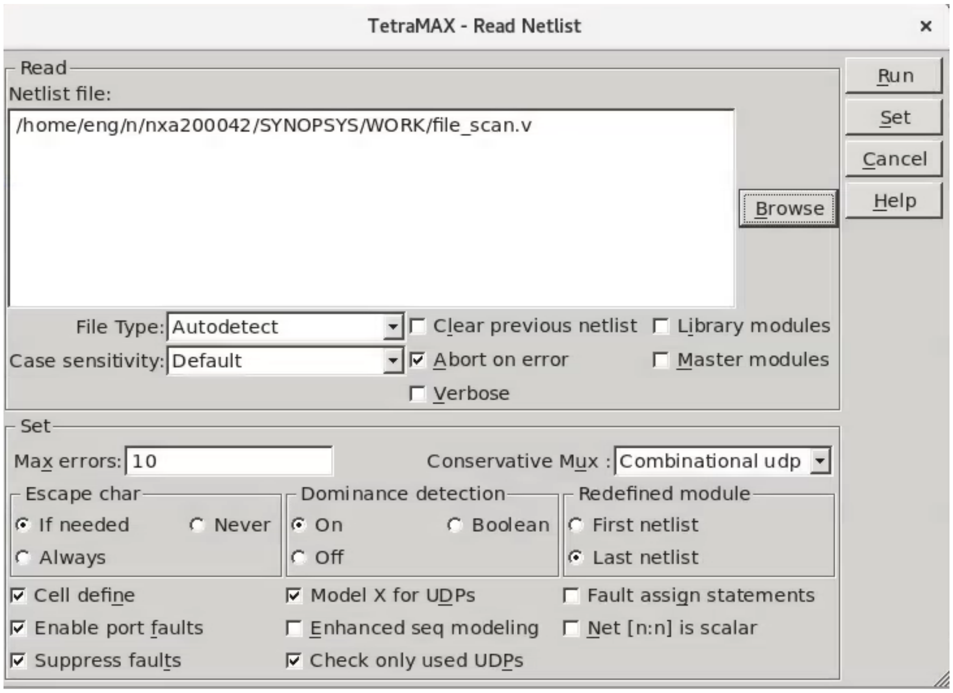
TetraMAX - Read Netlist

Read
Netlist file: /home/eng/n/nxa200042/SYNOPSYS/WORK/class.v

File Type: Autodetect
Case sensitivity: Default
□ Clear previous netlist
☑ Abort on error
□ Verbose
☑ Library modules
□ Master modules

Set
Max errors: 10
Conservative Mux : Combinational udp
Escape char: ● If needed ○ Never ○ Always
Dominance detection: ● On ○ Boolean ○ Off
Redefined module: ○ First netlist ● Last netlist
☑ Cell define
☑ Enable port faults
☑ Suppress faults
☑ Model X for UDPs
□ Enhanced seq modeling
☑ Check only used UDPs
□ Fault assign statements
□ Net [n:n] is scalar

(21) follow step 2.4 from test tutorial: 2.4 Read VHDL/Verilog File



TetraMAX - Read Netlist

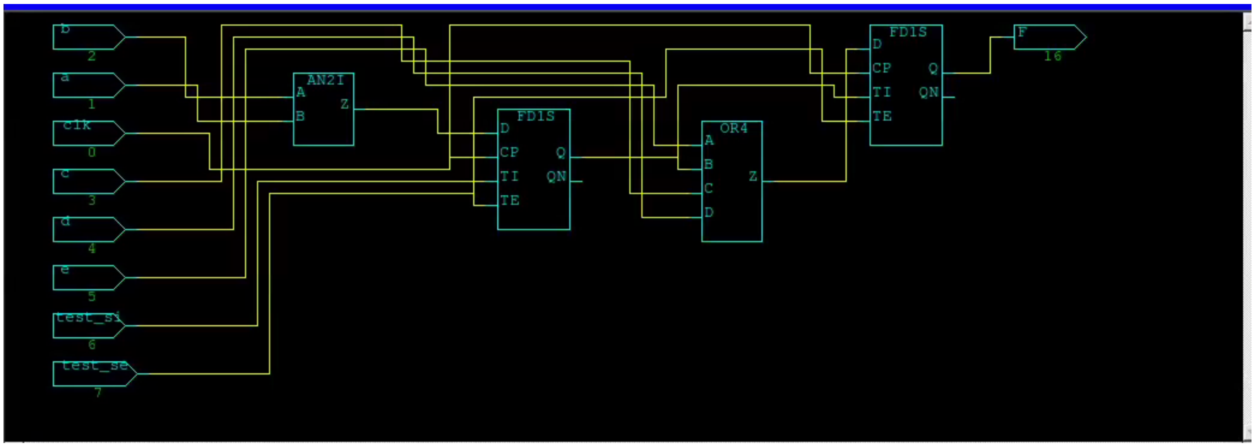
Read
Netlist file: /home/eng/n/nxa200042/SYNOPSYS/WORK/file_scan.v

File Type: Autodetect
Case sensitivity: Default
□ Clear previous netlist
☑ Abort on error
□ Verbose
□ Library modules
□ Master modules

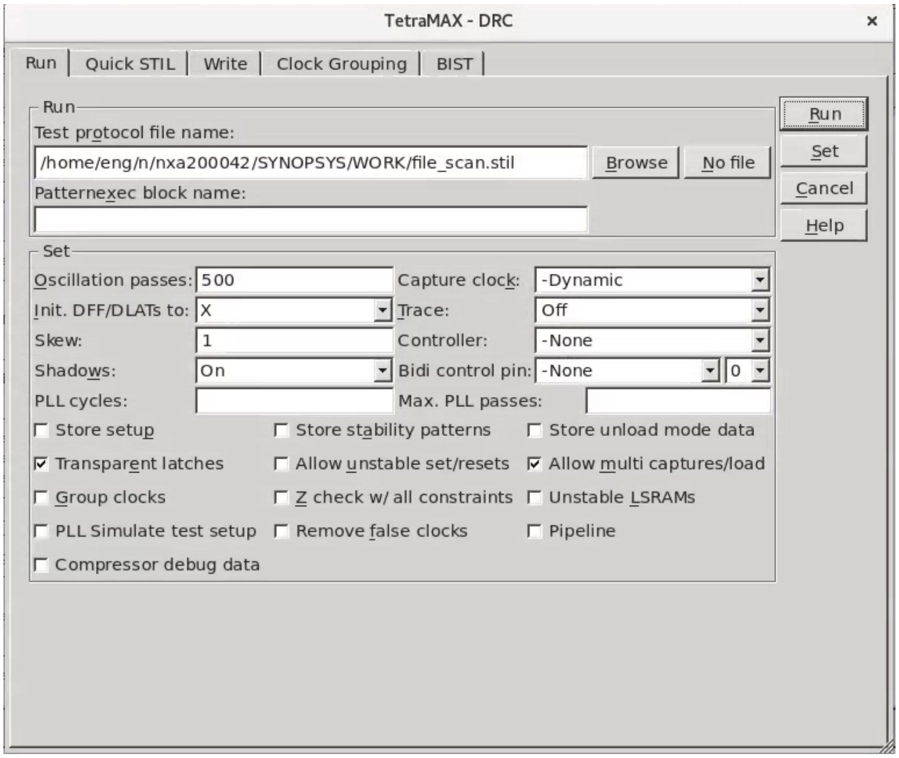
Set
Max errors: 10
Conservative Mux : Combinational udp
Escape char: ● If needed ○ Never ○ Always
Dominance detection: ● On ○ Boolean ○ Off
Redefined module: ○ First netlist ● Last netlist
☑ Cell define
☑ Enable port faults
☑ Suppress faults
☑ Model X for UDPs
□ Enhanced seq modeling
☑ Check only used UDPs
□ Fault assign statements
□ Net [n:n] is scalar

(22) follow step 2.5 from test tutorial: 2.5 Building the ATPG Model

(23) The Graphical Schematic Viewer (GSV) toolbar is used to display the schematic view of the circuit. Click SHOW → ALL.It shows the schematic view of the circuit, as shown in below:



(24) In DRC step of Tmax, you choose the test protocol (stil) file and get the fault coverage of the testable design. Note that if you have any violations in the design you won't be able to test it in Tetramax ATPG



Run

Test protocol file name: /home/eng/n/nxa200042/SYNOPSYS/WORK/file_scan.stil

Patternexec block name:

Set

Oscillation passes: 500

Capture clock: -Dynamic

Init. DFF/DLATs to: X

Trace: Off

Skew: 1

Controller: -None

Shadows: On

Bidi control pin: -None 0

PLL cycles:

Max. PLL passes:

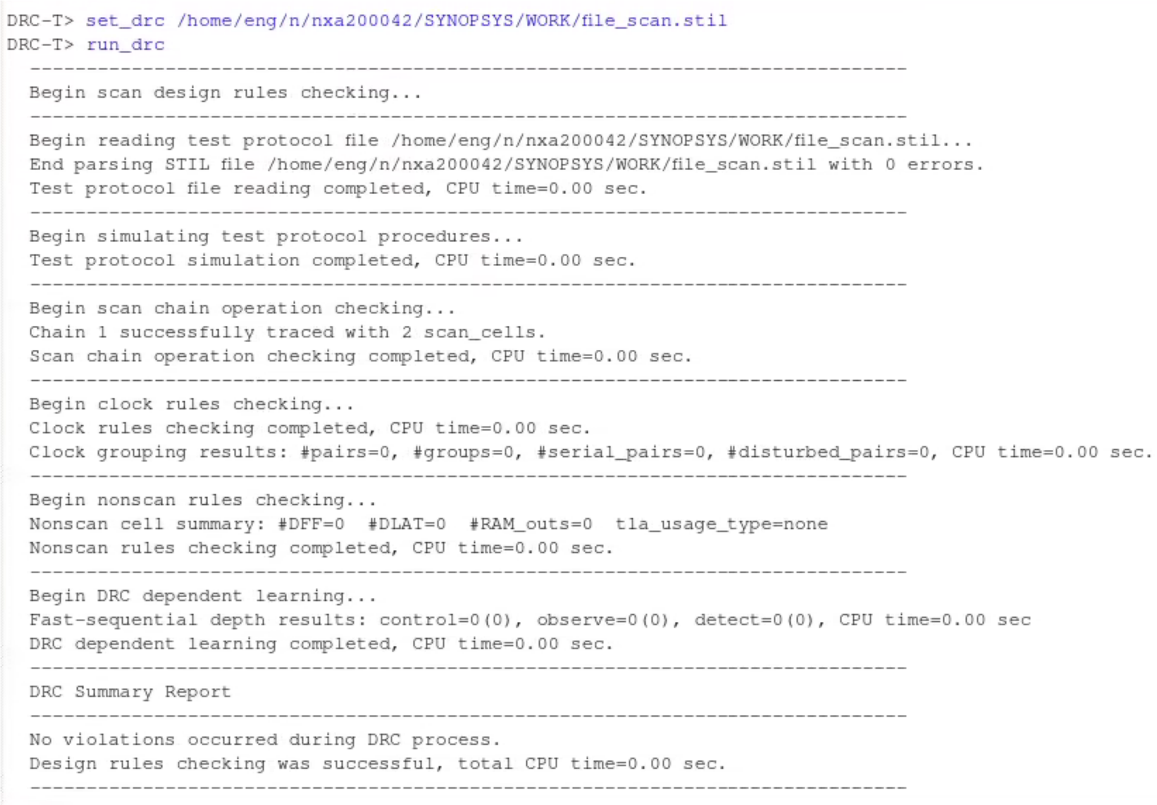
☐ Store setup ☐ Store stability patterns ☐ Store unload mode data

☑ Transparent latches ☐ Allow unstable set/resets ☑ Allow multi captures/load

☐ Group clocks ☐ Z check w/ all constraints ☐ Unstable LSRAMs

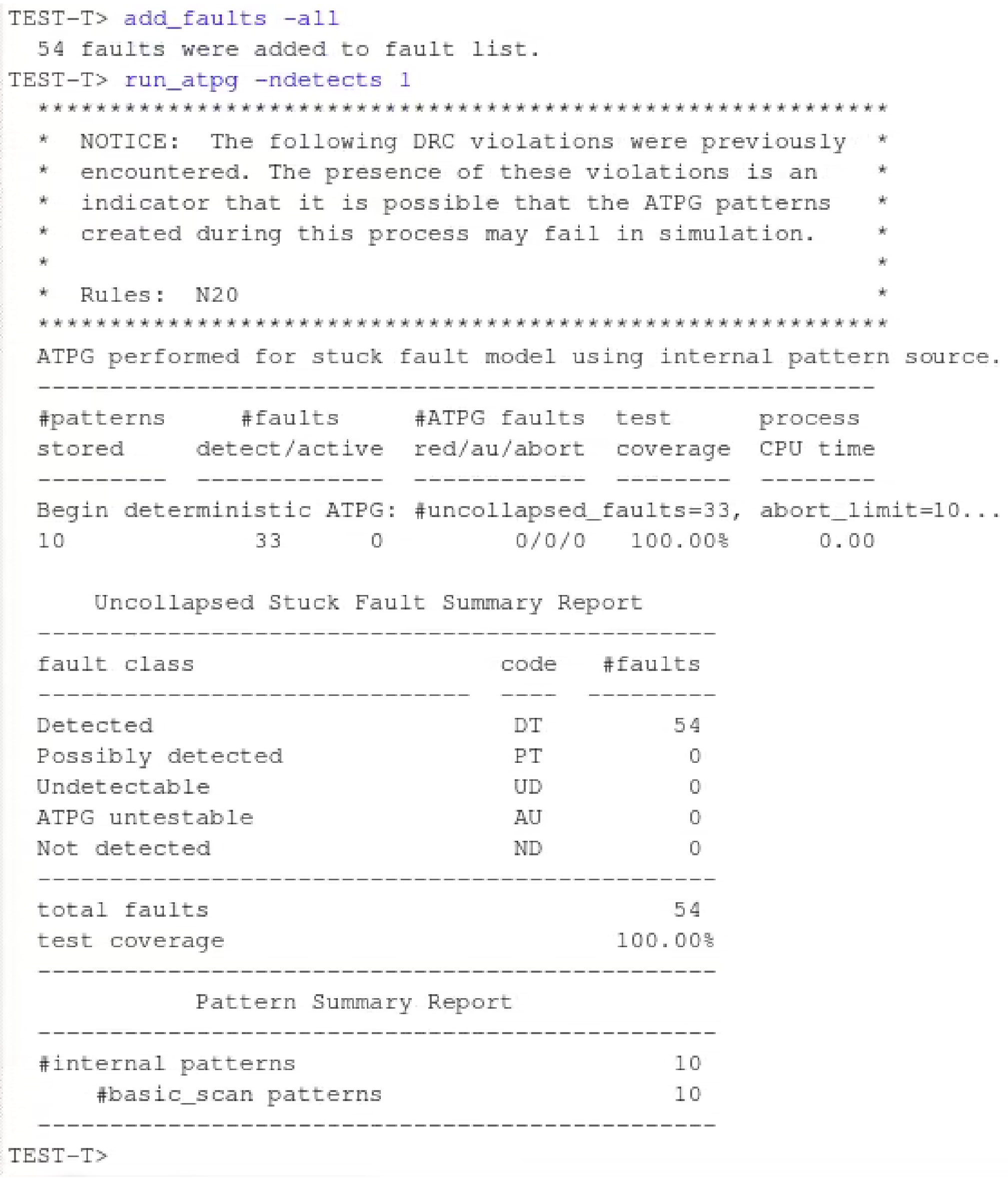
☐ PLL Simulate test setup ☐ Remove false clocks ☐ Pipeline

☐ Compressor debug data



```
DRC-T> set_drc /home/eng/n/nxa200042/SYNOPSYS/WORK/file_scan.stil
DRC-T> run_drc
  ------------------------------------------------------------------------------
  Begin scan design rules checking...
  ------------------------------------------------------------------------------
  Begin reading test protocol file /home/eng/n/nxa200042/SYNOPSYS/WORK/file_scan.stil...
  End parsing STIL file /home/eng/n/nxa200042/SYNOPSYS/WORK/file_scan.stil with 0 errors.
  Test protocol file reading completed, CPU time=0.00 sec.
  ------------------------------------------------------------------------------
  Begin simulating test protocol procedures...
  Test protocol simulation completed, CPU time=0.00 sec.
  ------------------------------------------------------------------------------
  Begin scan chain operation checking...
  Chain 1 successfully traced with 2 scan_cells.
  Scan chain operation checking completed, CPU time=0.00 sec.
  ------------------------------------------------------------------------------
  Begin clock rules checking...
  Clock rules checking completed, CPU time=0.00 sec.
  Clock grouping results: #pairs=0, #groups=0, #serial_pairs=0, #disturbed_pairs=0, CPU time=0.00 sec.
  ------------------------------------------------------------------------------
  Begin nonscan rules checking...
  Nonscan cell summary: #DFF=0  #DLAT=0  #RAM_outs=0  tla_usage_type=none
  Nonscan rules checking completed, CPU time=0.00 sec.
  ------------------------------------------------------------------------------
  Begin DRC dependent learning...
  Fast-sequential depth results: control=0(0), observe=0(0), detect=0(0), CPU time=0.00 sec
  DRC dependent learning completed, CPU time=0.00 sec.
  ------------------------------------------------------------------------------
  DRC Summary Report
  ------------------------------------------------------------------------------
  No violations occurred during DRC process.
  Design rules checking was successful, total CPU time=0.00 sec.
  ------------------------------------------------------------------------------
```

(25) Perform ATPG



```
TEST-T> add_faults -all
  54 faults were added to fault list.
TEST-T> run_atpg -ndetects 1
  ***********************************************************
  *  NOTICE:  The following DRC violations were previously  *
  *  encountered. The presence of these violations is an    *
  *  indicator that it is possible that the ATPG patterns   *
  *  created during this process may fail in simulation.    *
  *                                                         *
  *  Rules:  N20                                            *
  ***********************************************************
  ATPG performed for stuck fault model using internal pattern source.
  ----------------------------------------------------------
  #patterns     #faults     #ATPG faults  test      process
  stored     detect/active  red/au/abort  coverage  CPU time
  ---------  -------------  ------------  --------  --------
  Begin deterministic ATPG: #uncollapsed_faults=33, abort_limit=10...
  10             33      0         0/0/0   100.00%      0.00

      Uncollapsed Stuck Fault Summary Report
  -----------------------------------------------
  fault class                     code   #faults
  ------------------------------  ----  ---------
  Detected                         DT         54
  Possibly detected                PT          0
  Undetectable                     UD          0
  ATPG untestable                  AU          0
  Not detected                     ND          0
  -----------------------------------------------
  total faults                                54
  test coverage                           100.00%
  -----------------------------------------------
             Pattern Summary Report
  -----------------------------------------------
  #internal patterns                          10
      #basic_scan patterns                    10
  -----------------------------------------------
TEST-T>
```