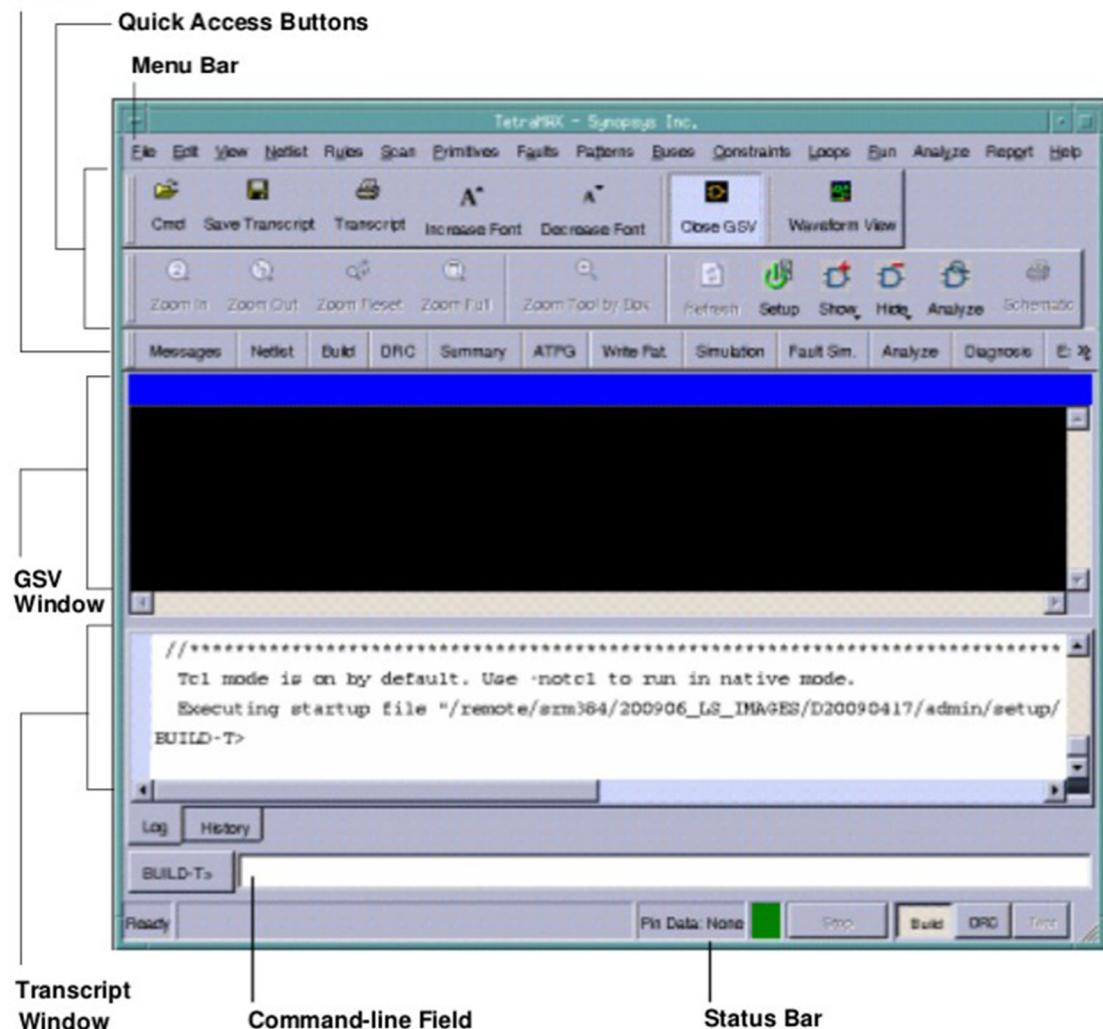


VLSI Design Verification and Testing (CMPE 418/646)
University of Maryland Baltimore County
Tutorial: Fault collapsing
Tutorial: Synopsys Tetramax

How to use TetraMAX?

TetraMAX is a high-speed, high-capacity automatic test pattern generation (ATPG) tool. It can generate test patterns that maximize test coverage while using a minimum number of test vectors for a wide variety of design types and design flows. It is well suited for designs of all sizes up to millions of gates. Information about the tool is available in the Tetramax User Guide. The main components of the GUI are shown in figure.

**Command
Toolbar**



The required files are present in the “banckmarks” folder in HW5. You need to copy those file into your home directory. For your convenience, the files included in the “benchmarks” directory

are also available in the following path
[/afs/umbc.edu/users/r/a/rahman2/pub/cmpe646/benchmarks](http://afs/umbc.edu/users/r/a/rahman2/pub/cmpe646/benchmarks)

and you can copy them as described below. I suggest you create a work folder and copy these files onto it.

Preliminary Steps

To create a working directory in your home folder
\$ mkdir work_folder
\$ cd work_folder
\$ mkdir lab1
\$ cd lab1

To copy files:

\$ cp -r /afs/umbc.edu/users/r/a/rahman2/pub/cmpe646/benchmarks/* .

Do not forget to put the DOT at the end of last command.

To invoke TetraMax

\$ tmax

For Synthesized netlist

1. Reading library files

In GUI, Select Netlist Read Netlist, Browse and select → GSCLib.v, click Run

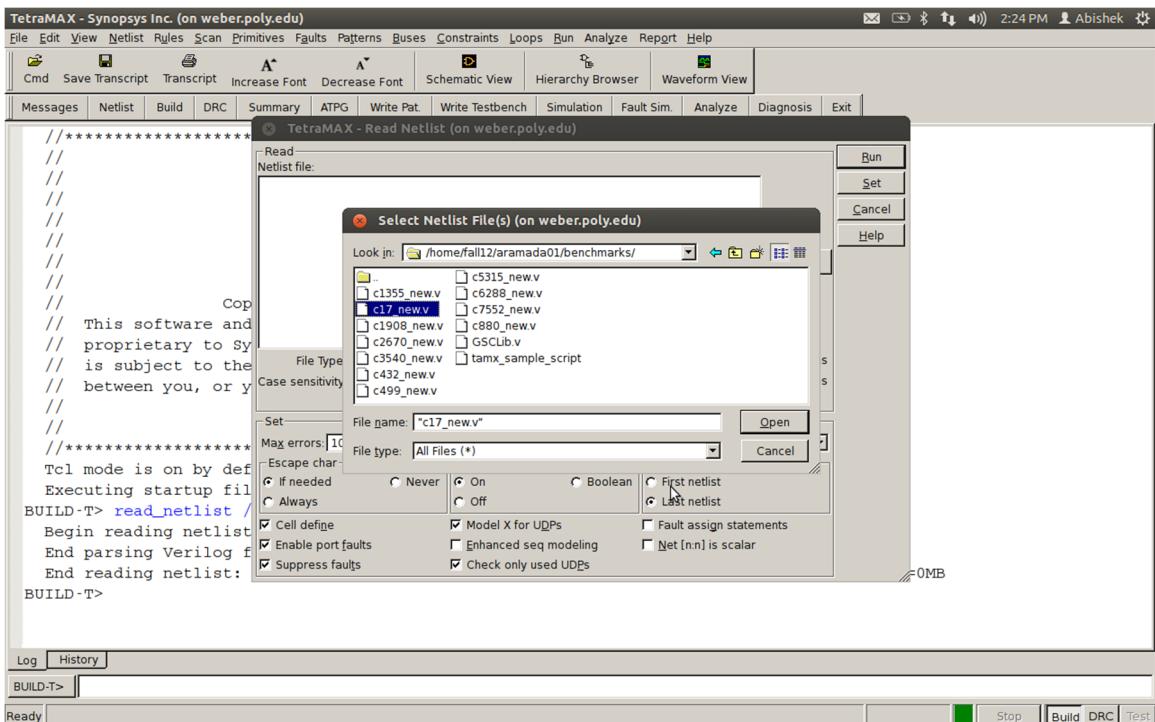
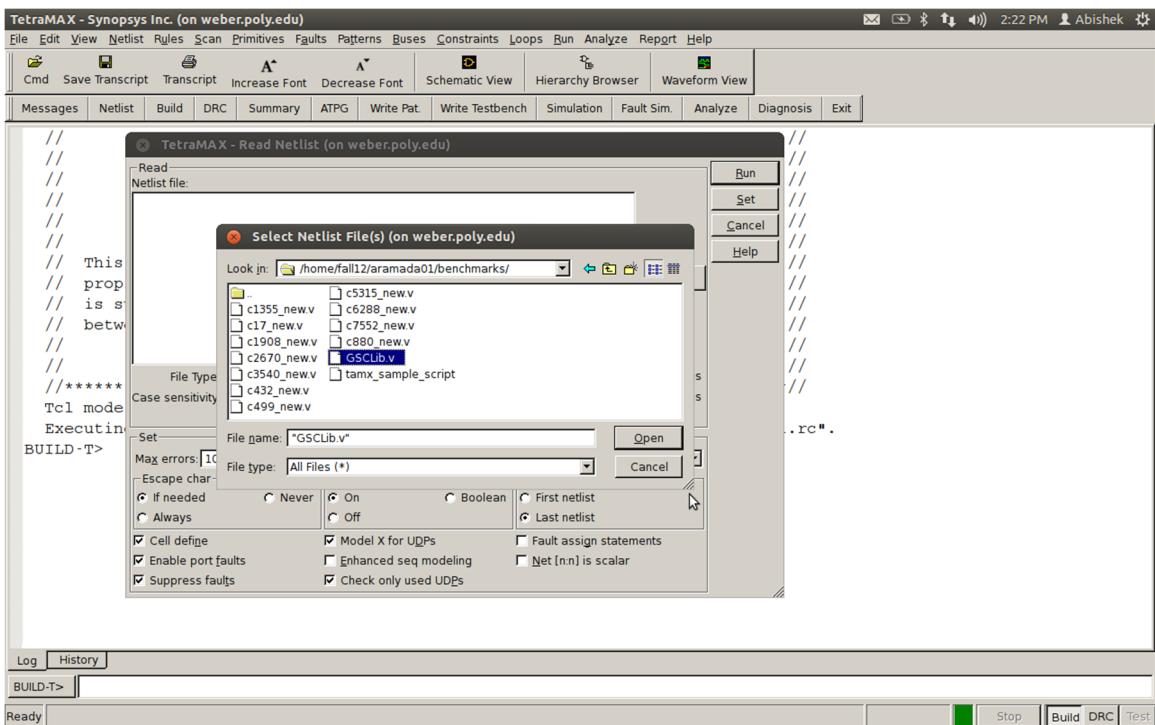
2. Reading the synthesized netlist – c17_new.v

In GUI, Select Netlist → Read Netlist, Browse and select c17_new.v, click Run

Alternatively, You can use the read_netlist command in the command line.

Example:

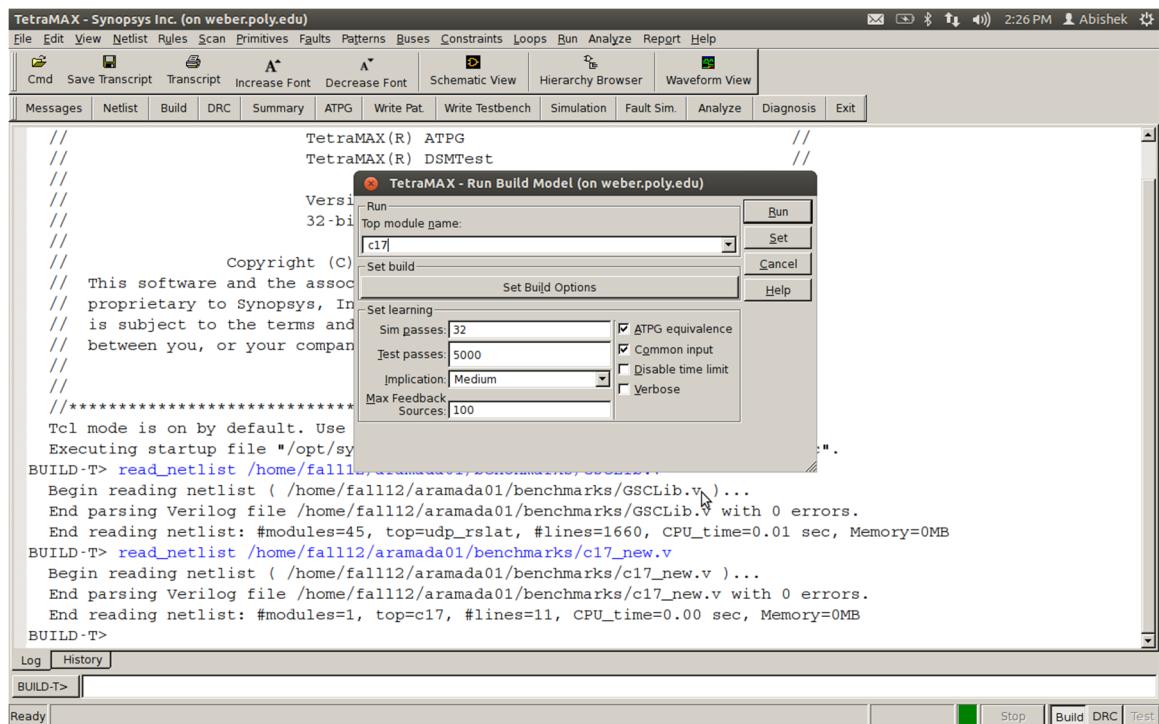
>read_netlist \$location_to/GSCLib.v



3. Build Model

You need to build the model for TetraMAX to work on the synthesized file.

In GUI, select Netlist Run Build Model, A window pops up, click Run with → default options.



Alternatively you can use the “run_build_model top_module” command
Example: run_build_model c17

4. Running DRC – To check if your design has any problems we need to run the design rule checker program. In GUI, select Rules → Run DRC, A window pops up, click “Run” with default options, see the transcript windows for a successful DRC.

TetraMAX - Synopsys Inc. (on weber.poly.edu)

File Edit View Netlist Rules Scan Primitives Faults Patterns Buses Constraints Loops Run Analyze Report Help

Cmd Save Transcript Transcript Increase Font Decrease Font Schematic View Hierarchy Browser Waveform View

Messages Netlist Build DRC Summary ATPG Write Pat Write Testbench Simulation Fault Sim. Analyze Diagnosis Exit

```
DRC-T> run_drc
-----
Begin scan design rules checking...
-----
Begin simulating test protocol procedures...
Test protocol simulation completed, CPU time=0.00 sec.
-----
Begin scan chain operation checking...
Scan chain operation checking completed, 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.
-----
TEST-T>
Log History
TEST-T> [ ] Stop Build DRC Test
Ready
```

Alternatively, you can type run_drc in the command line.

5. Set Fault Options

In the command line, type the following command to select the stuck at fault model

>set_faults -model stuck (press enter)

The model will be set as seen in the transcript window.

TetraMAX - Synopsys Inc. (on weber.poly.edu)

```
DRC-T> run_drc
-----
Begin scan design rules checking...
-----
Begin simulating test protocol procedures...
Test protocol simulation completed, CPU time=0.00 sec.
-----
Begin scan chain operation checking...
Scan chain operation checking completed, 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.
-----
TEST-T> set_faults -model stuck
TEST-T>
```

Log History

TEST-T> []

Ready | Stop Build DRC Test

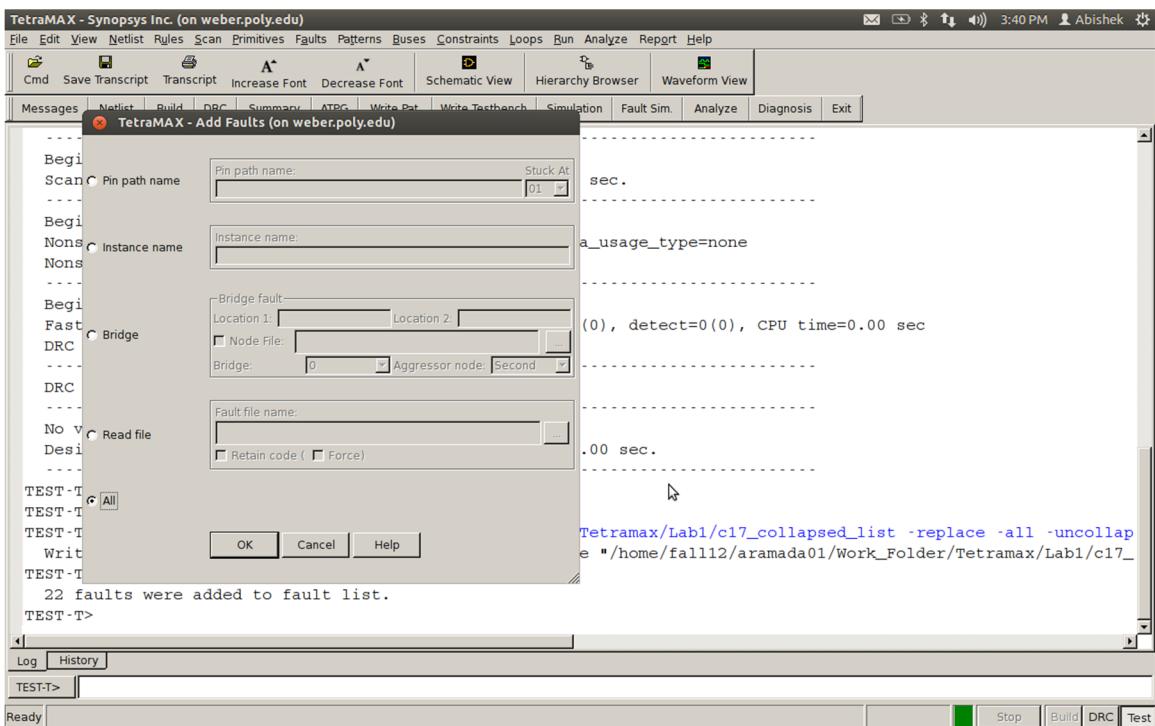
You can alternatively select the model from GUI from the “Faults” → “Set Fault Options” Menu.

6. Adding Faults

In GUI, select Faults → Add Faults, select "All", click ok.

Alternatively, In the command line you can type

>add_faults -all



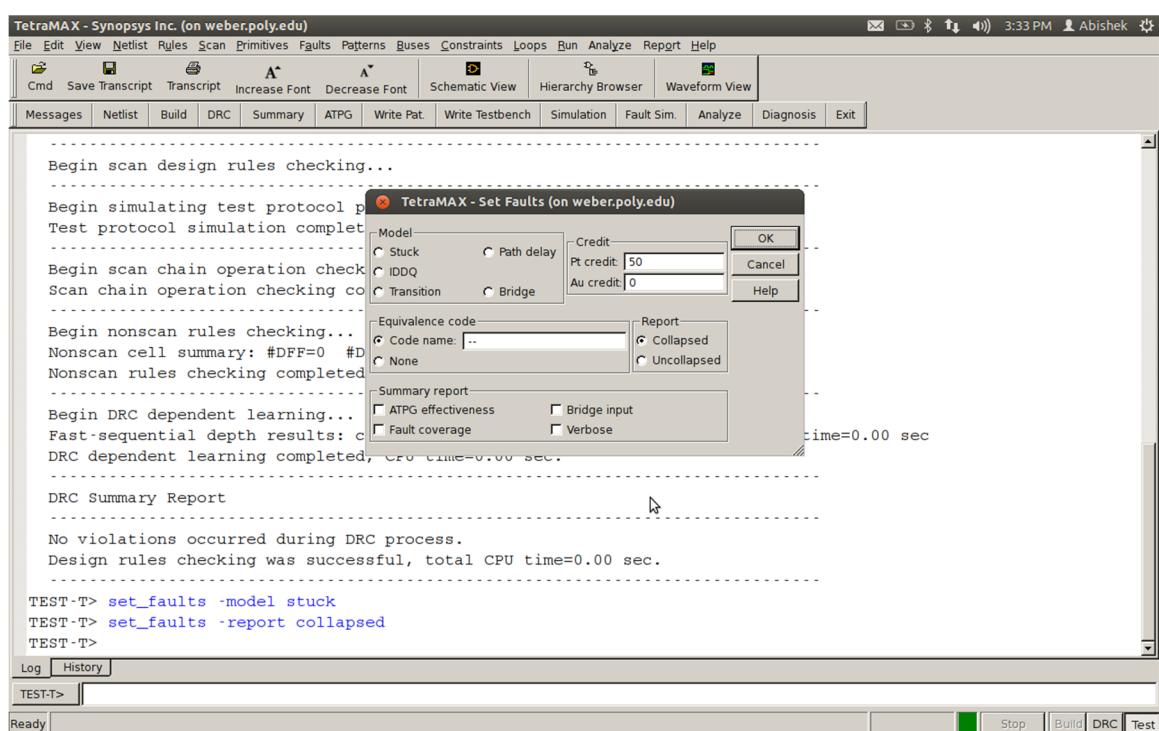
7. Generate collapsed faults

In command window type

```
>set_faults -report collapsed
```

Alternatively, In the GUI, Select "Faults" → "Set Fault Options", select the "Collapsed" → radio button in

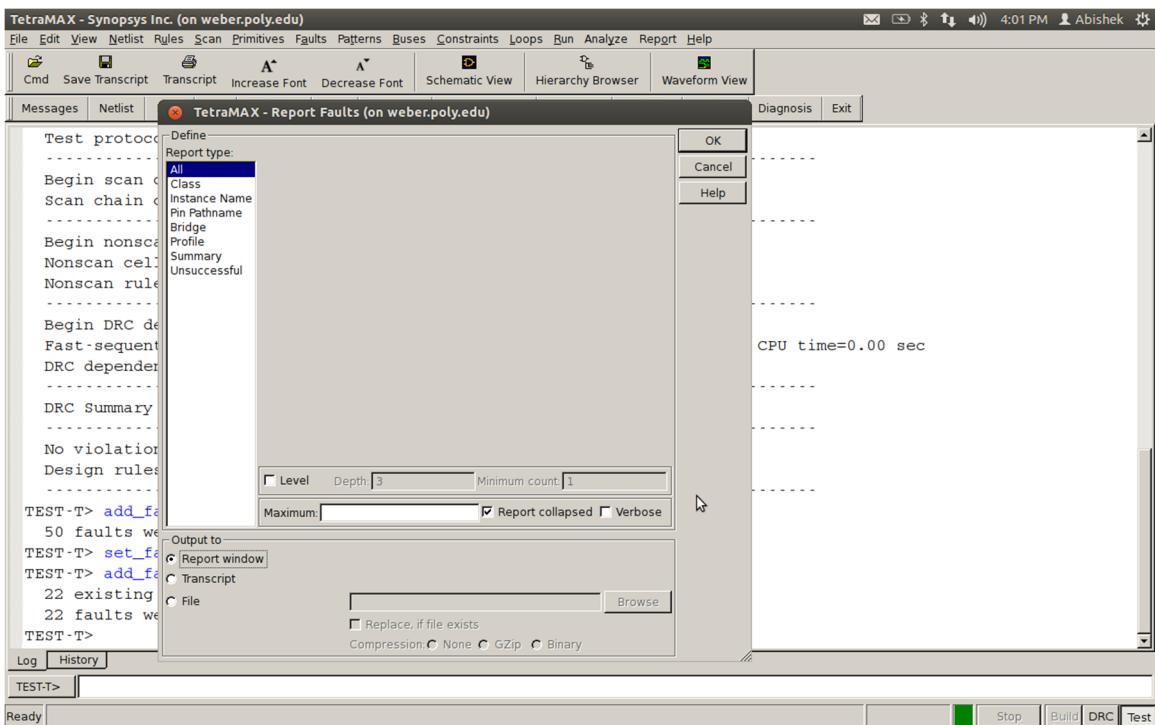
"Report pane, click ok"



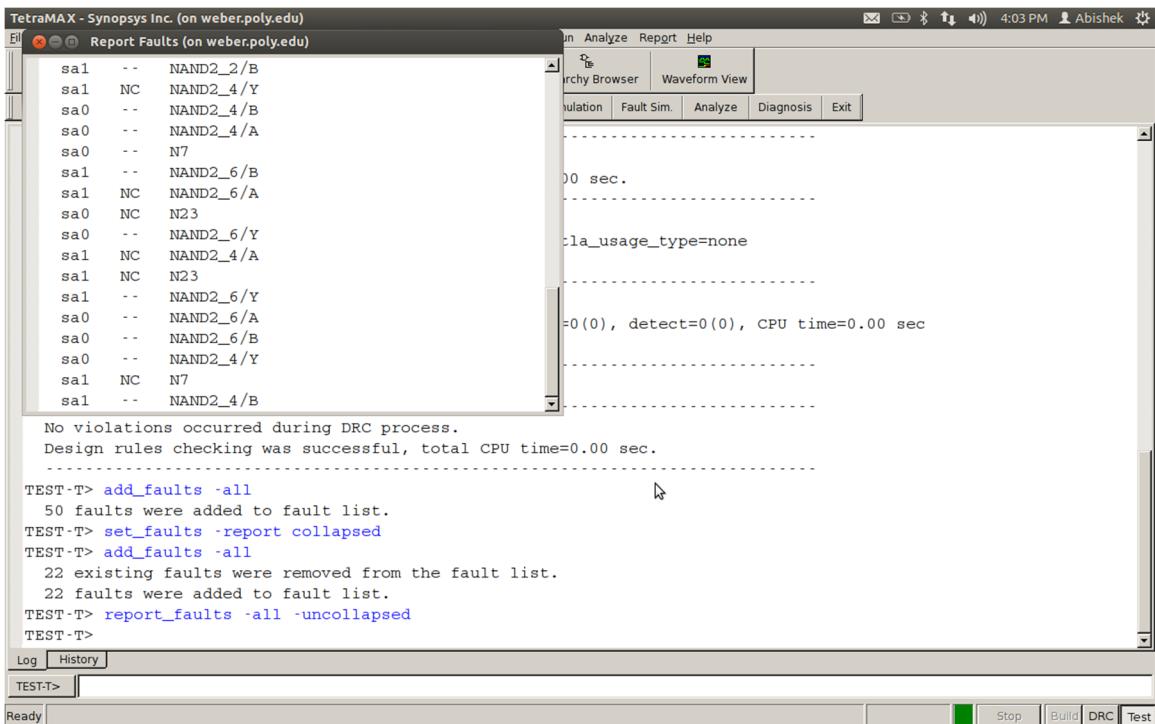
8. View the faults generated

Collapsed Faults

In the GUI, Select Faults → Report Faults, in "Report type" select "All", In output pane, select "Report Window"



The faults will be displayed on the report window.



Alternatively, you can

1. Write the uncollapsed faults in a file – In command line type
 >write_faults \$location/c17_all_faults -all -replace

(GUI: Faults → Report Faults, in "Report type" select "All", deselect "Report collapsed", In output pane, select file, browse to a convenient location and save the fault list in a file by name "c17_all_faults", Select "Replace, if file exists", click on ok)

2. Write collapsed faults in a file

```
write_faults $BASE/c17Collapsed_list -all -collapsed -replace
```

(GUI: Faults -> Report Faults, in "Report type" select "All", select "Report collapsed", In output pane, select file, browse to a convenient location and save the fault list in a file by name "c17Collapsed_list", Select "Replace, if file exists", click on ok)

3. Report the collapsed fault on the report window

```
report_faults -collapsed -all
```

(GUI: Report Faults, in "Report type" select "All", select "Report collapsed", In output pane, select "transcript", click on ok.)

4. Report Summary on the report window

```
report_summaries > $BASE/c17_report_summary
```

(GUI: Report Faults, in "Report type" select "Summary", In output pane, select "transcript". Click ok)

Reference

1. TetraMAX ATPG User Guide.
2. TetraMAX ATPG Quick Reference.