IIT JODHPUR
Department of Electrical Engineering
Final Report

Academic Year: 2022-23
Semester: 7th

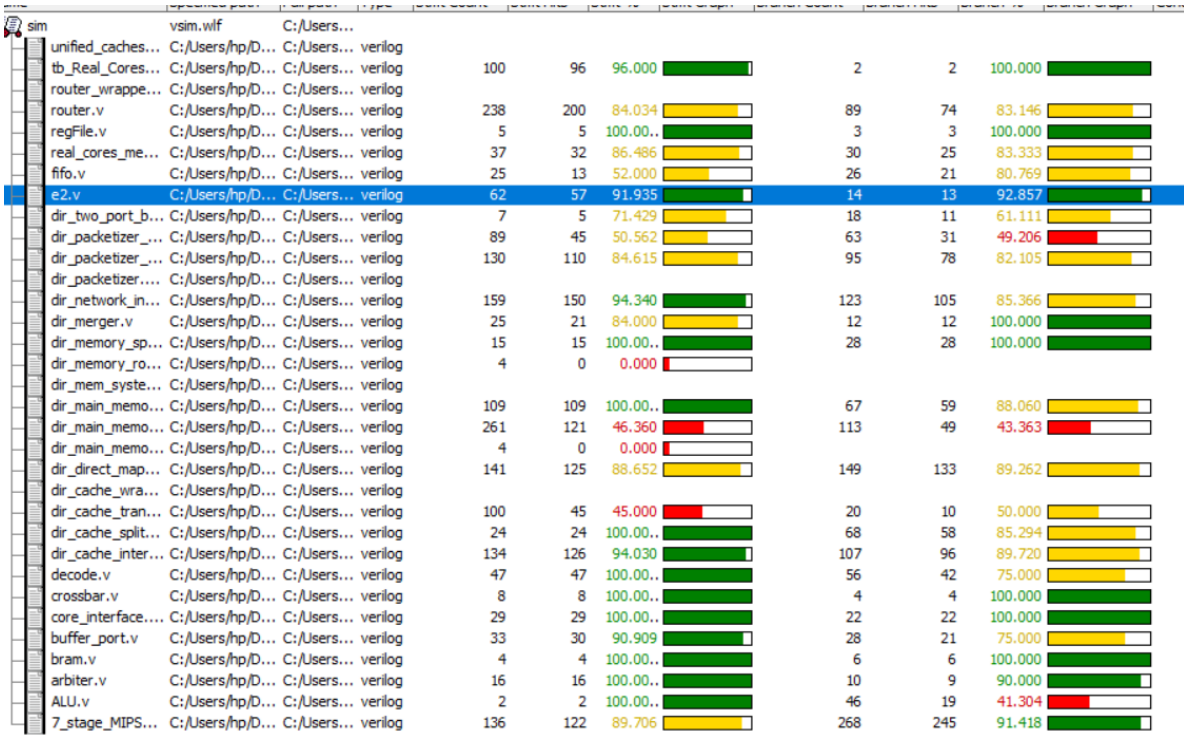
1. **Name of the Student:** Rahul Dhayal
2. **Roll Number:** B19EE069
3. **Title of the Project:** Simulation-based Methodology of Trojan Detection
4. **Project Category: 3**
5. **Targeted Deliverables:**
    a) Development of simulation-based methodology for trojan
    b) Detection & Documentation.
6. **Work Done:**
    In our project, we have tried to highlight the importance of development in the domain of Hardware Trojan detection. We ran Modelsim simulations on the design for Fibonacci series with different types of self-made Trojans. Our task in this project is to check the impact of these different types of Trojans on the simulation and code coverage of the design and then compare it with Trojan free design. The procedure, Results, and conclusions of one such trojan with design in modelsim have been discussed below:-
    - To begin with, we selected two Verilog files for trojan insertion which are Real_mesh_core.v and e2.v. both the files are easy to understand and there is huge ground for trojan insertion.
    - Trojan condition for e2.v file

```
138
139   //--------------Code Starts Here-----------------------
140   assign a = read_data[31] & read_data[21] & read_data[11] &read_data[1] ;
141   assign b = prog_address[31] & prog_address[21]  & prog_address[11] & prog
142   assign c =  data_out[31] & data_out[21] & data_out[11] & data_out[1];
143   assign trigger = a &b & c;
144
```

    - Trojan condition for real_core_mesh.v file

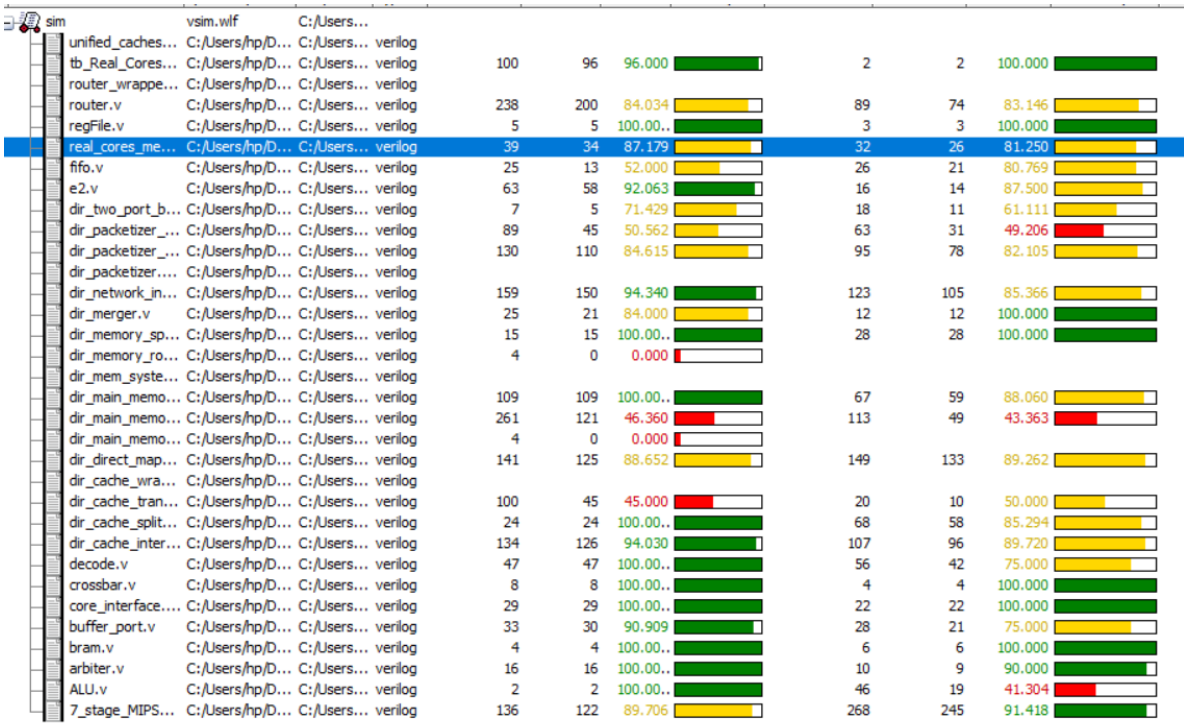
```
// code start here
assign a = from_peripheral_data[20] & from_peripheral_data[21]  &from_peripheral_data[0] & f
assign b = to_peripheral_data[20] & to_peripheral_data[21]  & to_peripheral_data[0] & to_per
assign c = prog_address[20] & prog_address[21]  & prog_address[0] & prog_address[15]& prog_a
assign trigger = a & b & c;
//----------------------code----------------------//
```

When Hardware Trojan Free simulation:



sim	vsim.wlf	C:/Users...										
unified_caches...	C:/Users/hp/D...	C:/Users...	verilog									
tb_Real_Cores...	C:/Users/hp/D...	C:/Users...	verilog	100	96	96.000		2	2	100.000		
router_wrappe...	C:/Users/hp/D...	C:/Users...	verilog									
router.v	C:/Users/hp/D...	C:/Users...	verilog	238	200	84.034		89	74	83.146		
regFile.v	C:/Users/hp/D...	C:/Users...	verilog	5	5	100.00..		3	3	100.000		
real_cores_me...	C:/Users/hp/D...	C:/Users...	verilog	37	32	86.486		30	25	83.333		
fifo.v	C:/Users/hp/D...	C:/Users...	verilog	25	13	52.000		26	21	80.769		
e2.v	C:/Users/hp/D...	C:/Users...	verilog	62	57	91.935		14	13	92.857		
dir_two_port_b...	C:/Users/hp/D...	C:/Users...	verilog	7	5	71.429		18	11	61.111		
dir_packetizer_...	C:/Users/hp/D...	C:/Users...	verilog	89	45	50.562		63	31	49.206		
dir_packetizer_...	C:/Users/hp/D...	C:/Users...	verilog	130	110	84.615		95	78	82.105		
dir_packetizer....	C:/Users/hp/D...	C:/Users...	verilog									
dir_network_in...	C:/Users/hp/D...	C:/Users...	verilog	159	150	94.340		123	105	85.366		
dir_merger.v	C:/Users/hp/D...	C:/Users...	verilog	25	21	84.000		12	12	100.000		
dir_memory_sp...	C:/Users/hp/D...	C:/Users...	verilog	15	15	100.00..		28	28	100.000		
dir_memory_ro...	C:/Users/hp/D...	C:/Users...	verilog	4	0	0.000						
dir_mem_syste...	C:/Users/hp/D...	C:/Users...	verilog									
dir_main_memo...	C:/Users/hp/D...	C:/Users...	verilog	109	109	100.00..		67	59	88.060		
dir_main_memo...	C:/Users/hp/D...	C:/Users...	verilog	261	121	46.360		113	49	43.363		
dir_main_memo...	C:/Users/hp/D...	C:/Users...	verilog	4	0	0.000						
dir_direct_map...	C:/Users/hp/D...	C:/Users...	verilog	141	125	88.652		149	133	89.262		
dir_cache_wra...	C:/Users/hp/D...	C:/Users...	verilog									
dir_cache_tran...	C:/Users/hp/D...	C:/Users...	verilog	100	45	45.000		20	10	50.000		
dir_cache_split...	C:/Users/hp/D...	C:/Users...	verilog	24	24	100.00..		68	58	85.294		
dir_cache_inter...	C:/Users/hp/D...	C:/Users...	verilog	134	126	94.030		107	96	89.720		
decode.v	C:/Users/hp/D...	C:/Users...	verilog	47	47	100.00..		56	42	75.000		
crossbar.v	C:/Users/hp/D...	C:/Users...	verilog	8	8	100.00..		4	4	100.000		
core_interface....	C:/Users/hp/D...	C:/Users...	verilog	29	29	100.00..		22	22	100.000		
buffer_port.v	C:/Users/hp/D...	C:/Users...	verilog	33	30	90.909		28	21	75.000		
bram.v	C:/Users/hp/D...	C:/Users...	verilog	4	4	100.00..		6	6	100.000		
arbiter.v	C:/Users/hp/D...	C:/Users...	verilog	16	16	100.00..		10	9	90.000		
ALU.v	C:/Users/hp/D...	C:/Users...	verilog	2	2	100.00..		46	19	41.304		
7_stage_MIPS...	C:/Users/hp/D...	C:/Users...	verilog	136	122	89.706		268	245	91.418		

When Hardware Trojan Inserted simulation:



sim	vsim.wlf	C:/Users...										
unified_caches...	C:/Users/hp/D...	C:/Users...	verilog									
tb_Real_Cores...	C:/Users/hp/D...	C:/Users...	verilog	100	96	96.000		2	2	100.000		
router_wrappe...	C:/Users/hp/D...	C:/Users...	verilog									
router.v	C:/Users/hp/D...	C:/Users...	verilog	238	200	84.034		89	74	83.146		
regFile.v	C:/Users/hp/D...	C:/Users...	verilog	5	5	100.00..		3	3	100.000		
real_cores_me...	C:/Users/hp/D...	C:/Users...	verilog	39	34	87.179		32	26	81.250		
fifo.v	C:/Users/hp/D...	C:/Users...	verilog	25	13	52.000		26	21	80.769		
e2.v	C:/Users/hp/D...	C:/Users...	verilog	63	58	92.063		16	14	87.500		
dir_two_port_b...	C:/Users/hp/D...	C:/Users...	verilog	7	5	71.429		18	11	61.111		
dir_packetizer_...	C:/Users/hp/D...	C:/Users...	verilog	89	45	50.562		63	31	49.206		
dir_packetizer_...	C:/Users/hp/D...	C:/Users...	verilog	130	110	84.615		95	78	82.105		
dir_packetizer....	C:/Users/hp/D...	C:/Users...	verilog									
dir_network_in...	C:/Users/hp/D...	C:/Users...	verilog	159	150	94.340		123	105	85.366		
dir_merger.v	C:/Users/hp/D...	C:/Users...	verilog	25	21	84.000		12	12	100.000		
dir_memory_sp...	C:/Users/hp/D...	C:/Users...	verilog	15	15	100.00..		28	28	100.000		
dir_memory_ro...	C:/Users/hp/D...	C:/Users...	verilog	4	0	0.000						
dir_mem_syste...	C:/Users/hp/D...	C:/Users...	verilog									
dir_main_memo...	C:/Users/hp/D...	C:/Users...	verilog	109	109	100.00..		67	59	88.060		
dir_main_memo...	C:/Users/hp/D...	C:/Users...	verilog	261	121	46.360		113	49	43.363		
dir_main_memo...	C:/Users/hp/D...	C:/Users...	verilog	4	0	0.000						
dir_direct_map...	C:/Users/hp/D...	C:/Users...	verilog	141	125	88.652		149	133	89.262		
dir_cache_wra...	C:/Users/hp/D...	C:/Users...	verilog									
dir_cache_tran...	C:/Users/hp/D...	C:/Users...	verilog	100	45	45.000		20	10	50.000		
dir_cache_split...	C:/Users/hp/D...	C:/Users...	verilog	24	24	100.00..		68	58	85.294		
dir_cache_inter...	C:/Users/hp/D...	C:/Users...	verilog	134	126	94.030		107	96	89.720		
decode.v	C:/Users/hp/D...	C:/Users...	verilog	47	47	100.00..		56	42	75.000		
crossbar.v	C:/Users/hp/D...	C:/Users...	verilog	8	8	100.00..		4	4	100.000		
core_interface....	C:/Users/hp/D...	C:/Users...	verilog	29	29	100.00..		22	22	100.000		
buffer_port.v	C:/Users/hp/D...	C:/Users...	verilog	33	30	90.909		28	21	75.000		
bram.v	C:/Users/hp/D...	C:/Users...	verilog	4	4	100.00..		6	6	100.000		
arbiter.v	C:/Users/hp/D...	C:/Users...	verilog	16	16	100.00..		10	9	90.000		
ALU.v	C:/Users/hp/D...	C:/Users...	verilog	2	2	100.00..		46	19	41.304		
7_stage_MIPS...	C:/Users/hp/D...	C:/Users...	verilog	136	122	89.706		268	245	91.418		

Real_core_messh.v (Trojan Free)

Design Unit Coverage Details:

Total Coverage:					49.28%	67.19%
Coverage Type	**Bins**	**Hits**	**Misses**	**Weight**	**% Hit**	**Coverage**
Statements	41	36	5	1	87.80%	**87.80%**
Branches	30	25	5	1	83.33%	**83.33%**
Toggles	138	42	96	1	30.43%	**30.43%**

e2.v (Trojan Free)

Design Unit Coverage Details:

Total Coverage:					27.37%	67.03%
Coverage Type	**Bins**	**Hits**	**Misses**	**Weight**	**% Hit**	**Coverage**
Statements	62	57	5	1	91.93%	**91.93%**
Branches	14	13	1	1	92.85%	**92.85%**
Toggles	324	27	297	1	8.33%	**8.33%**
FSMs	31	21	10	1	67.74%	**75.00%**
States	11	11	0	1	100.00%	**100.00%**
Transitions	20	10	10	1	50.00%	**50.00%**

Real_core_mesh.v (Trojan Inserted)

Design Unit Coverage Details:

Total Coverage:					37.19%	63.20%
Coverage Type	**Bins**	**Hits**	**Misses**	**Weight**	**% Hit**	**Coverage**
Statements	43	38	5	1	88.37%	**88.37%**
Branches	32	26	6	1	81.25%	**81.25%**
Toggles	210	42	168	1	20.00%	**20.00%**

e2.v (Trojan Inserted)

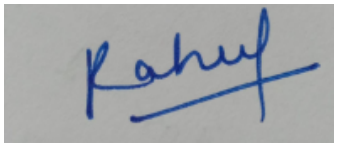
Design Unit Coverage Details:

Total Coverage:					27.14%	65.67%
Coverage Type	**Bins**	**Hits**	**Misses**	**Weight**	**% Hit**	**Coverage**
Statements	63	58	5	1	92.06%	**92.06%**
Branches	16	14	2	1	87.50%	**87.50%**
Toggles	332	27	305	1	8.13%	**8.13%**
FSMs	31	21	10	1	67.74%	**75.00%**
States	11	11	0	1	100.00%	**100.00%**
Transitions	20	10	10	1	50.00%	**50.00%**

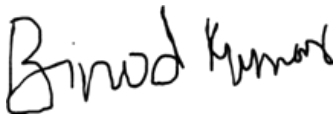
7. **Concluding Remarks:**
   As we can observe from the above data, There is a major difference in the code coverage analysis of the design. It is likely that Trojan can affect our design badly. And if it gets activated, it can change the whole result and scenario. We can see that from code coverage reports. These kinds of trojans will be easily detected by Code Coverage and FSM Analysis.

8. **References:**

9. **Declaration:** I declare that no part of this report is copied from other sources. All the references are properly cited in this report.



Signature of the Student



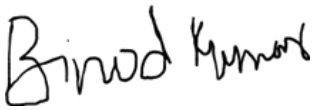
Signature of the Supervisor

Supervisor's Recommendation for the Evaluation

Please tick any one of the following

1. The work done is satisfactory, and sufficient time has been spent by the student. The submission by the student should be evaluated in this term.
2. The work is not complete. Continuity Grade should be given to the student. The student would need to be evaluated in the next semester for the same Design Project with me.
3. The work is not satisfactory. There is no need for evaluation. The students should look for another Design Credit Project for the next semester.
4. [Other Comment, if 1-3 are not valid]

_____



Signature of the Supervisor