



# Verification of SPI slave IP with systemVerilog

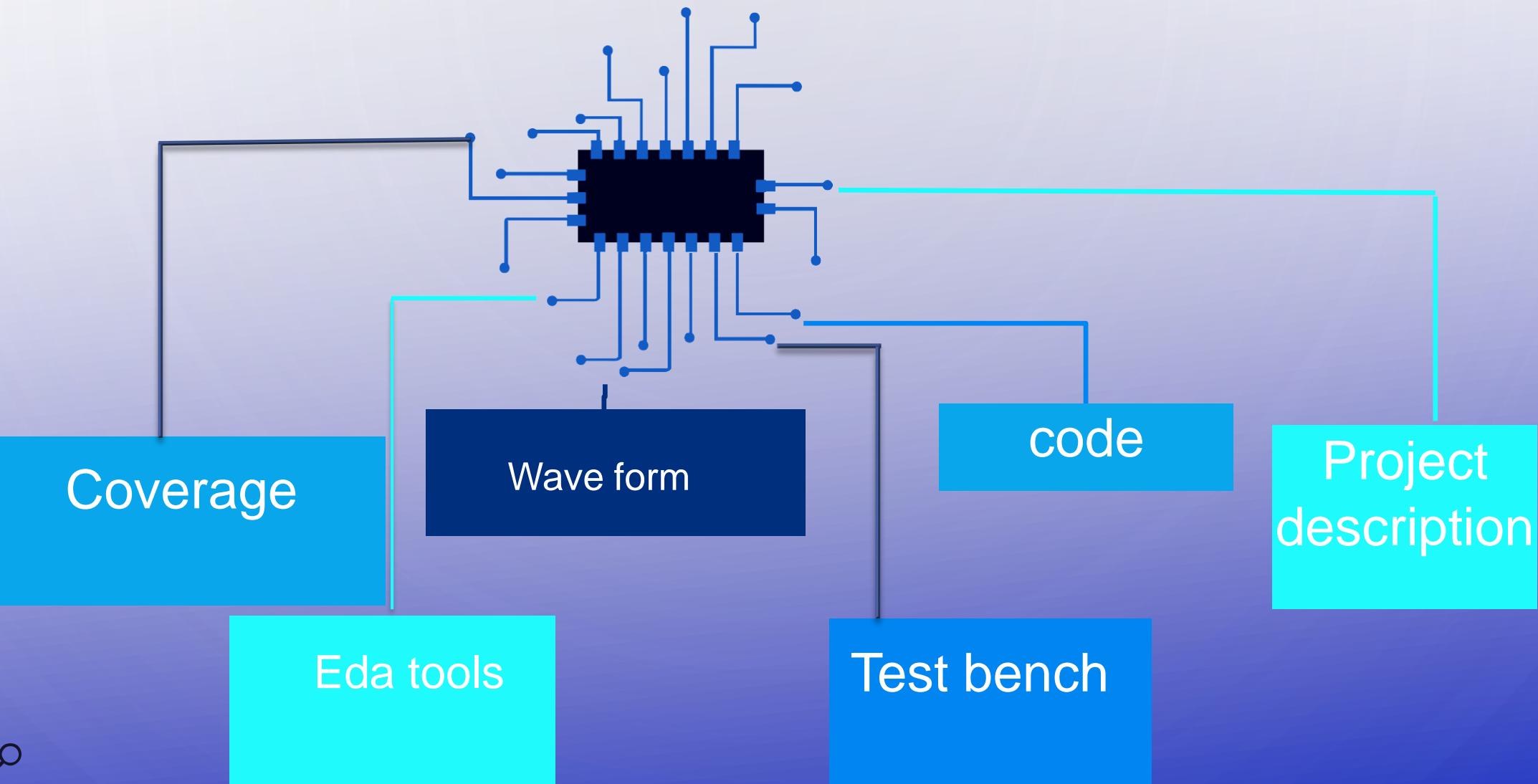
CND 212 : Digital Testing and Verification



# TEAM MEMBERS

Student Name	ID
Ahmed mohamed mohamed shawky Elattar	V23010568
Muhammed Khaled Omar Orabi	V23010695
Mohammed Salah Aboshosha Mohammed	V23010141
Omar Dyaa Mohammed Mohamed	V23010501
Amr mohamed ali hassan	V23010177

# CONTENT



# PROJECT DESCRIPTION

➤ In this project, we will test an SPI slave IP using systemVerilog. This will involve using an actual IP or an abstract model as the test will assume a black box approach. The testing will be incorporating new features offered by SystemVerilog such as interfaces, classes, constrained-random stimulus generation, and coverage-driven verification methodologies. You will be required to show different test scenarios for the Slave IP showing modularity and reusability of the test environment.

# PROJECT DESCRIPTION

Testing an SPI slave IP using SystemVerilog is a comprehensive task that involves various advanced features of the language. Here's a high-level overview of how you might approach creating different test scenarios:

- 1- Define the Interface:** Use SystemVerilog interfaces to encapsulate the signals related to the SPI protocol (like mosi, miso, sclk, and ss).
- 2- Create the Testbench:** Develop a testbench that instantiates the SPI interface, the DUT (Device Under Test), and the test environment.
- 3- Develop the Test Environment:** Build a test environment with classes that represent different components like drivers, monitors, and scoreboards.
- 4- Generate Stimulus:** Use constrained-random stimulus generation to create a variety of valid and invalid scenarios.
- 5- Implement Coverage:** Define coverage points to ensure that all interesting scenarios are being tested.
- 6- Create Scenarios:** Write scenarios that test the SPI slave IP under different conditions, such as varying clock speeds, data patterns, and error conditions.
- 7- Modularity and Reusability:** Ensure that your test components are modular and reusable. For example, you can create a generic SPI driver that can be used with different types of SPI devices.
- 8- Run the Tests:** Execute the tests and analyze the results. Use the coverage data to identify any scenarios that have not been tested.
- 9- Iterate:** Refine the test scenarios based on the results and coverage data to improve the test environment and achieve better verification.

# EDA TOOLS

eda playground  
(synopsys VCS  
2023.03)

**vcs**

**dve**

**verdi**

# MODULE SLAVE

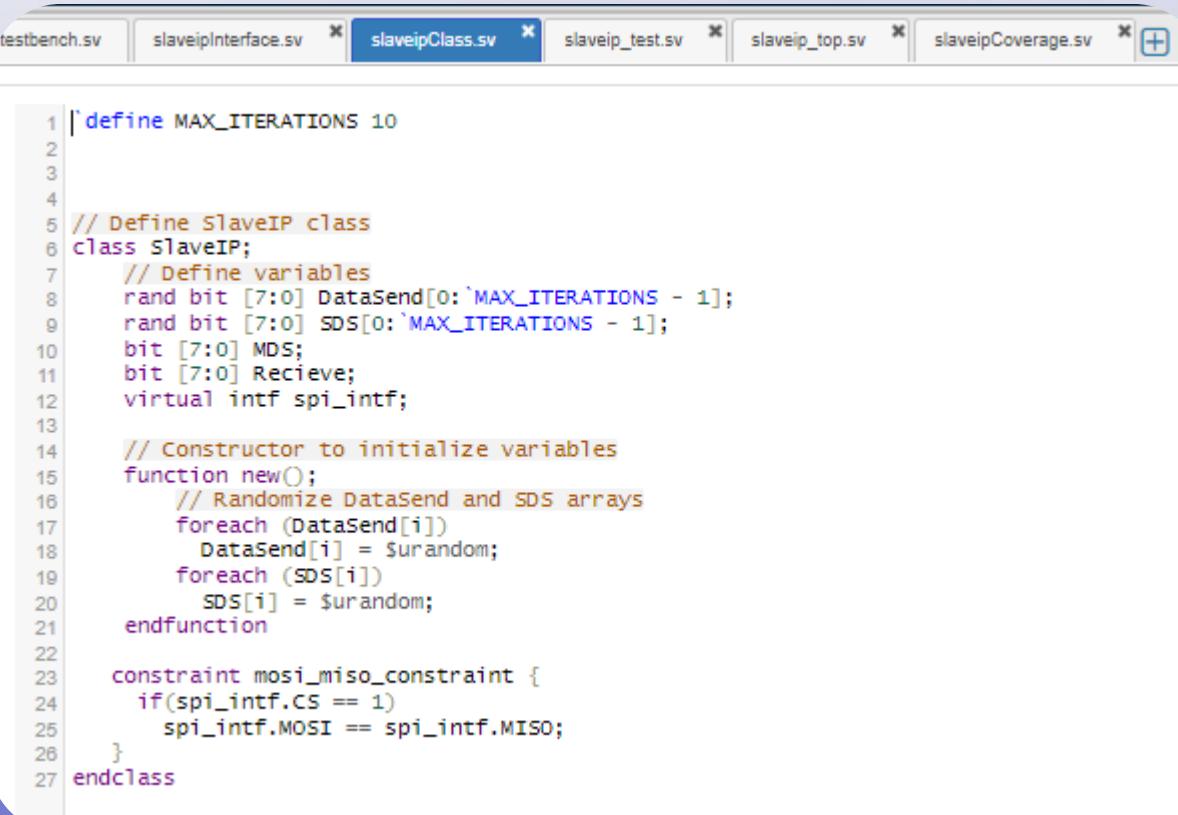
```
1 module slave
2 (
3     input reset,
4     input [7:0]slaveDataToSend,
5     output [7:0]slaveDataReceived,
6     input SCLK, CS, MOSI,
7     input wr_en ,rd_en,
8     output reg MISO
9 );
10 integer counter;
11 reg [7:0] SDS;
12 reg [7:0] SDR;
13
14 wire [7:0] SDS_next;
15 wire [7:0] SDR_next;
16
17 always @ (negedge CS)
18 begin
19     counter = 0;
20     SDS = slaveDataToSend;
21 end
22
23 always @(posedge SCLK , posedge reset)
24 begin
25 if(reset==1'b1)
26 begin
27     SDS=slaveDataToSend;
28     SDR=0;
29     counter = 0;
30 end
31 else
32 begin
33     SDR=SDR_next;
34     SDS=SDS_next;
35     counter = counter+1;
36 end
37
38 end
39 assign SDS_next = (CS==1'b0&&counter<=8 &&counter >0 && rd_en)?{ SDS[6:0],1'b0 }:SDS;
40 assign SDR_next = (CS==1'b0&&counter<=8 && wr_en )?[SDR[6:0],MOSI]:SDR;
41 assign MISO = (CS==1'b0 && rd_en)? SDS[7]:1'bz;
42 assign slaveDataReceived = SDR;
43
44
45 endmodule
```

# SLAVE IP INTERFACE



```
1 // Define interface
2 interface intf;
3     logic reset;
4     logic [7:0] slaveDataToSend;
5     logic [7:0] slaveDataReceived;
6     logic SCLK, CS, MOSI;
7     logic wr_en ,rd_en;
8     logic MISO;
9
10    clocking spi_clocking @(posedge SCLK);
11        input reset;
12        output MOSI, CS;
13        input MISO;
14        input wr_en ,rd_en;
15    endclocking
16 endinterface
```

# SLAVE IP CLASS



```
1 `define MAX_ITERATIONS 10
2
3
4
5 // Define SlaveIP class
6 class SlaveIP;
7     // Define variables
8     rand bit [7:0] DataSend[0:`MAX_ITERATIONS - 1];
9     rand bit [7:0] SDS[0:`MAX_ITERATIONS - 1];
10    bit [7:0] MDS;
11    bit [7:0] Recieve;
12    virtual intf spi_intf;
13
14    // Constructor to initialize variables
15    function new();
16        // Randomize DataSend and SDS arrays
17        foreach (DataSend[i])
18            DataSend[i] = $urandom;
19        foreach (SDS[i])
20            SDS[i] = $urandom;
21    endfunction
22
23    constraint mosi_miso_constraint {
24        if(spi_intf.CS == 1)
25            spi_intf.MOSI == spi_intf.MISO;
26    }
27 endclass
```

15 s25fjdcn5
16 }
17 !OSIM,47NTR\_rds == !SOM,47NTR\_rds
18 T == S,47NTR\_rds(T)
19 }\_introduction\_of\_iml\_from\_introduction
20 introduction
21 introduction

# SLAVECOVERAGE(INTFSPI\_INTF)

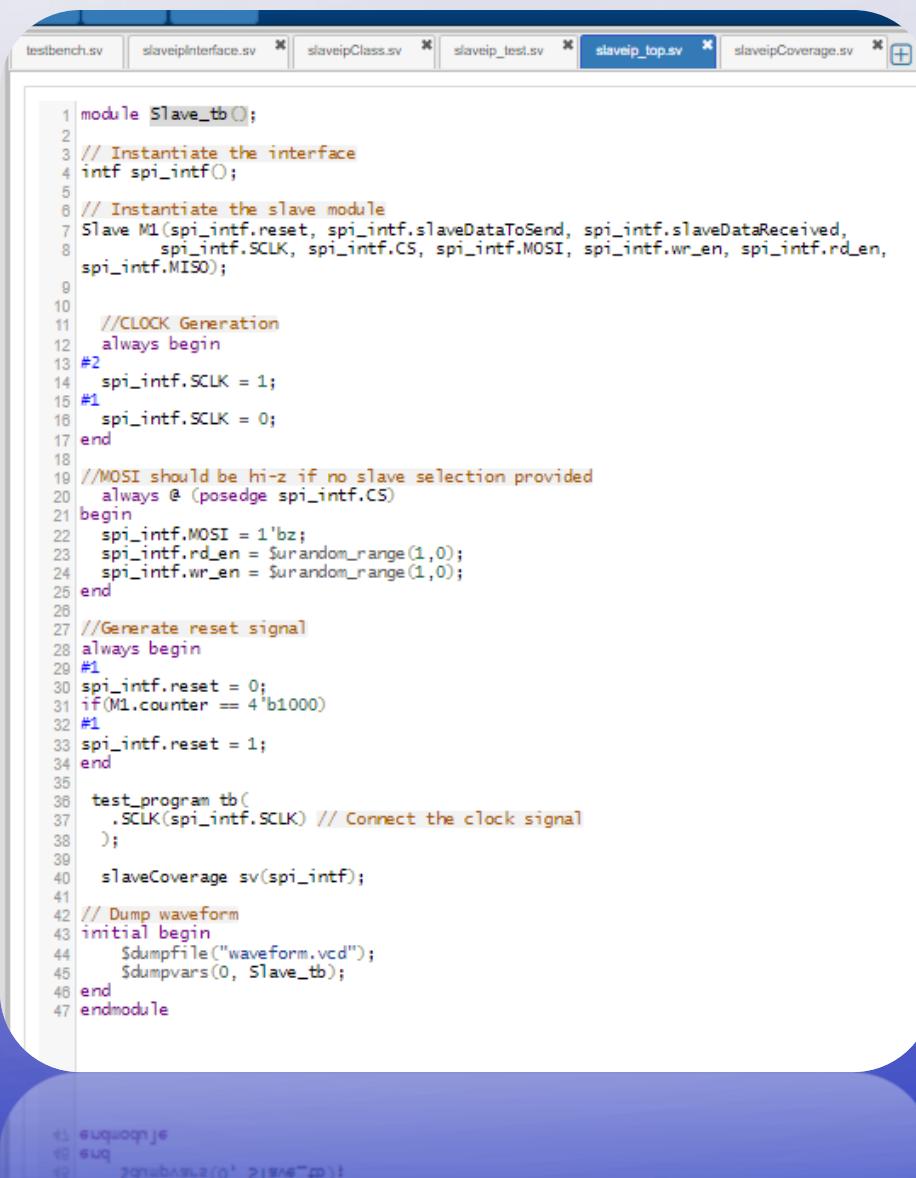


```
testbench.sv slaveipInterface.sv * slaveipClass.sv * slaveip_test.sv * slaveip_top.sv * slaveipCoverage.sv *
```

```
1 module slaveCoverage(intf spi_intf);
2   // Coverage group definition
3   covergroup spi_coverage;
4     option.per_instance = 1; // Allows per-instance coverage
5
6   // Coverage points for Full and Empty flags
7   coverpoint(spi_intf.MOSI) {
8     bins MOSI_is_true = {1'b1}; // Bin for full being true
9     bins MOSI_is_false = {1'b0}; // Bin for full being false
10 }
11
12  coverpoint(spi_intf.MISO) {
13    bins MISO_is_true = {1'b1}; // Bin for empty being true
14    bins MISO_is_false = {1'b0}; // Bin for empty being false
15 }
16
17  coverpoint(spi_intf.wr_en) {
18    bins wr_is_true = {1'b1}; // Bin for empty being true
19    bins wr_is_false = {1'b0}; // Bin for empty being false
20 }
21
22  coverpoint(spi_intf.rd_en) {
23    bins rd_is_true = {1'b1}; // Bin for empty being true
24    bins rd_is_false = {1'b0}; // Bin for empty being false
25 }
26 endgroup
27
28   spi_coverage fc = new;
29 endmodule
```

```
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
```

# SLAVE\_TB()



```
1 module Slave_tb();
2 // Instantiate the interface
3 intf spi_intf();
4
5 // Instantiate the slave module
6 Slave M1(spi_intf.reset, spi_intf.slaveDataToSend, spi_intf.slaveDataReceived,
7           spi_intf.SCLK, spi_intf.CS, spi_intf.MOSI, spi_intf.wr_en, spi_intf.rd_en,
8           spi_intf.MISO);
9
10 //CLOCK Generation
11 always begin
12 #2
13   spi_intf.SCLK = 1;
14 #1
15   spi_intf.SCLK = 0;
16 end
17
18 //MOSI should be hi-z if no slave selection provided
19 always @ (posedge spi_intf.CS)
20 begin
21   spi_intf.MOSI = 1'bz;
22   spi_intf.rd_en = $urandom_range(1,0);
23   spi_intf.wr_en = $urandom_range(1,0);
24 end
25
26 //Generate reset signal
27 always begin
28 #1
29 if(M1.counter == 4'b1000)
30   spi_intf.reset = 0;
31 #1
32 if(M1.counter == 4'b1000)
33   spi_intf.reset = 1;
34 end
35
36 test_program tb(
37   .SCLK(spi_intf.SCLK) // Connect the clock signal
38 );
39
40 slaveCoverage sv(spi_intf);
41
42 // Dump waveform
43 initial begin
44   $dumpfile("waveform.vcd");
45   $dumpvars(0, Slave_tb);
46 end
47 endmodule
```

# TEST\_PROGRAM

testbench.sv slaveipInterface.sv slaveipClass.sv slaveip\_test.sv slaveip\_top.sv slaveipCoverage.sv

```

1 program test_program(input SCLK);
2
3 // Instantiate the slave IP class
4 SlaveIP slave_inst = new();
5
6
7 // Define variables
8 integer i, j; // Declare i and j here
9 bit [7:0] MDS;
10 bit [7:0] Recieve;
11 int numIterations = `MAX_ITERATIONS;
12 -
13 task write_task();
14     int i;
15     spi_intf.MOSI = slave_inst.MDS[7];
16     if(!spi_intf.wr_en)
17         spi_intf.MOSI = 1'bz;
18     #2
19     for (i = 6; i >= 0; i = i - 1) begin
20         #1;
21         spi_intf.MOSI = slave_inst.MDS[i];
22         if(!spi_intf.wr_en)
23             spi_intf.MOSI = 1'bz;
24         #2;
25     end
26     #1;
27 endtask
28
29 task read_task();
30     int i;
31     #2
32     for (i = 6; i >= 0; i = i - 1) begin
33         #1;
34         slave_inst.Recieve[i+1] = spi_intf.MISO;
35         #2;
36     end
37     #1;
38     slave_inst.Recieve[0] = spi_intf.MISO;
39     Recieve = slave_inst.Recieve;
40 endtask
41
42
43 // Testbench code
44 initial begin
45     spi_intf.SCLK = 0;
46     spi_intf.MOSI = 1'bz;
47     spi_intf.reset = 1;
48     // Test data initialization
49     for (j = 0; j < numIterations; j = j + 1) begin
50         slave_inst.MDS = slave_inst.DataSend[j];
51         MDS = slave_inst.MDS;
52         spi_intf.slaveDataToSend = slave_inst.SDS[j];
53         spi_intf.CS = 1;
54         #6
55         spi_intf.CS = 0;
56
57         fork
58             write_task();
59             read_task();
60         join
61         disable fork;
62
63         // Assert slaveDataToSend and slaveDataReceived signals
64         if(spi_intf.rd_en) begin
65             assert (slave_inst.Recieve == spi_intf.slaveDataToSend)
66             $display("Master received Data %x from slave
successfully",slave_inst.Recieve);
67             if(spi_intf.wr_en) begin
68                 assert (spi_intf.slaveDataReceived == slave_inst.MDS)
69                 $display("Slave received Data %x from Master
successfully",spi_intf.slaveDataReceived );
70                 if(spi_intf.slaveDataReceived != slave_inst.MDS)
71                     $error("slaveDataReceived FAILED");
72             end
73         end
74
75         $finish;
76     end
77 endprogram
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99

```

```

42 // Testbench code
43 initial begin
44     spi_intf.SCLK = 0;
45     spi_intf.MOSI = 1'bz;
46     spi_intf.reset = 1;
47     // Test data initialization
48     for (j = 0; j < numIterations; j = j + 1) begin
49         slave_inst.MDS = slave_inst.DataSend[j];
50         MDS = slave_inst.MDS;
51         spi_intf.slaveDataToSend = slave_inst.SDS[j];
52         spi_intf.CS = 1;
53         #6
54         spi_intf.CS = 0;
55
56         fork
57             write_task();
58             read_task();
59         join
50         disable fork;
51
52         // Assert slaveDataToSend and slaveDataReceived signals
53         if(spi_intf.rd_en) begin
54             assert (slave_inst.Recieve == spi_intf.slaveDataToSend)
55             $display("Master received Data %x from slave
successfully",slave_inst.Recieve);
56             if(spi_intf.wr_en) begin
57                 assert (spi_intf.slaveDataReceived == slave_inst.MDS)
58                 $display("Slave received Data %x from Master
successfully",spi_intf.slaveDataReceived );
59                 if(spi_intf.slaveDataReceived != slave_inst.MDS)
60                     $error("slaveDataReceived FAILED");
61             end
62         end
63
64         $finish;
65     end
66 endprogram
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99

```

# EDA PLAYGROUNGLINK

<https://www.edaplayground.com/x/iZZg>



# EDA RUN RESULT

The screenshot shows the EDA playground interface with several tabs open:

- UVM / OVM**: Selected library.
- testbench.sv**, **slaveipInterface.sv**, **slaveipClass.sv** (current tab), **slaveip\_test.sv**, **slaveip\_top.sv**, **slaveipCoverage.sv**.
- design.sv**: Shows the Slave module code.

**Log Output:**

```
Master received Data db from slave successfully
Slave received Data a8 from Master successfully
Master received Data 52 from slave successfully
Slave received Data a1 from Master successfully
Master received Data 0c from slave successfully
Slave received Data 20 from Master successfully
Master received Data 79 from slave successfully
$finish called from file "slaveip_test.sv", line 75.
$finish at simulation time 300
VCS Simulation Report
Time: 300 ns
CPU Time: 0.450 seconds; Data structure size: 0.2Mb
Mon May 13 12:04:36 2024
Finding VCD file...
./waveform.vcd
[2024-05-13 16:04:36 UTC] Opening EPWave...
Done
```

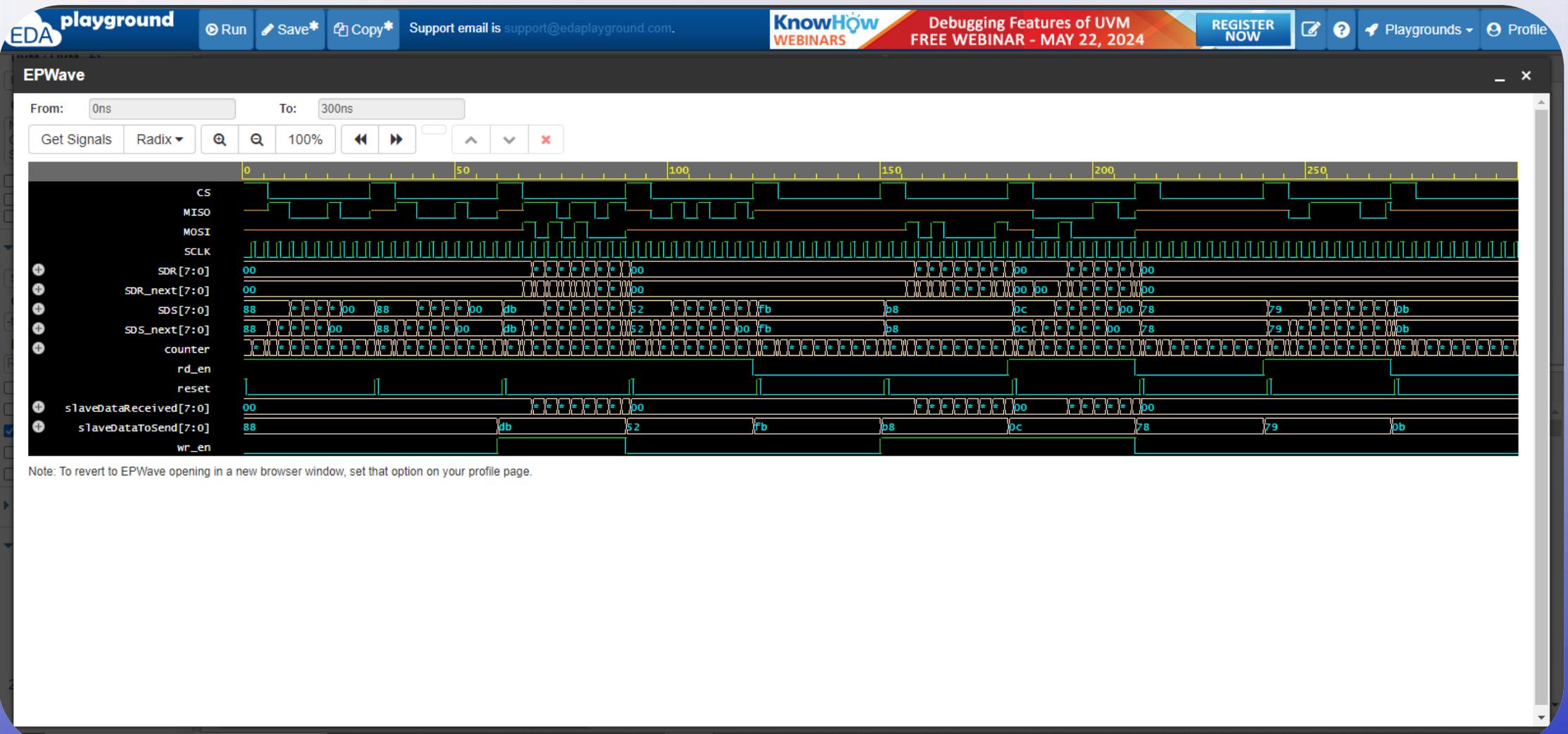
**EPWave Window:**

```
Done
[2024-05-13 16:04:36 UTC] Opening EPWave...
Done
```

**Left Sidebar:**

- playground**: Main title.
- Run**, **Save\***, **Copy\***: Action buttons.
- Support email is support@edaplayground.com.**
- KnowHow WEBINARS**: Banner.
- REGISTER NOW**: Call-to-action button.
- Playgrounds**, **Profile**: User navigation.
- UVM IEEE 1800.2-2017**: Standard dropdown.
- Other Libraries**: None, OVL, SVUnit.
- Tools & Simulators**: Synopsys VCS 2023.03.
- Compile Options**: -timescale=1ns/1ns +vcs+flush.
- Run Options**: Run Options, Use run.do Tcl file, Open EPWave after run (checked).
- Examples**.
- Community**: Collaborate, Forum, Follow @edaplayground.
- 208**: Recent activity count.

# EDA EP WAVE



# CMD COMMANDS

```
vcs -sverilog -ntb_opts dtm -cm line+cond+fsm+tgl Slaveip_Top.sv Slaveip-RTLdesign.v Slaveip_Test.sv  
Slaveip_Interface.sv Slaveip_Class.sv Slaveip_Coverage.sv
```

```
./simv -cm line+cond+fsm+tgl
```

```
urrg -dir simv.vdb
```

```
Verdi --cov --covdir simv.vdb &
```

# VCS RESULT

```
File Edit View Search Terminal Help
Parsing design file 'FINALPROJRCT/design.sv'
Parsing design file 'FINALPROJRCT/tb.sv'
Top Level Modules:
    Slave_tb
No TimeScale specified
VCS Coverage Metrics Release V-2023.12 Copyright (c) 1991-2023 by Synopsys Inc.

Warning-[INTFDV] VCD dumping of interface/program/package
FINALPROJRCT/tb.sv, 184
    Selective VCD dumping of interface 'intf' is not supported. Selective VCD
    dumping for interfaces, packages and programs is not supported.
    Use full VCD dumping '$dumpvars(0)', or use VPD or FSDB dumping, recompile
    with '-debug_access'.

Starting vcs inline pass...
3 modules and 0 UDP read.
    However, due to incremental compilation, no re-compilation is necessary.

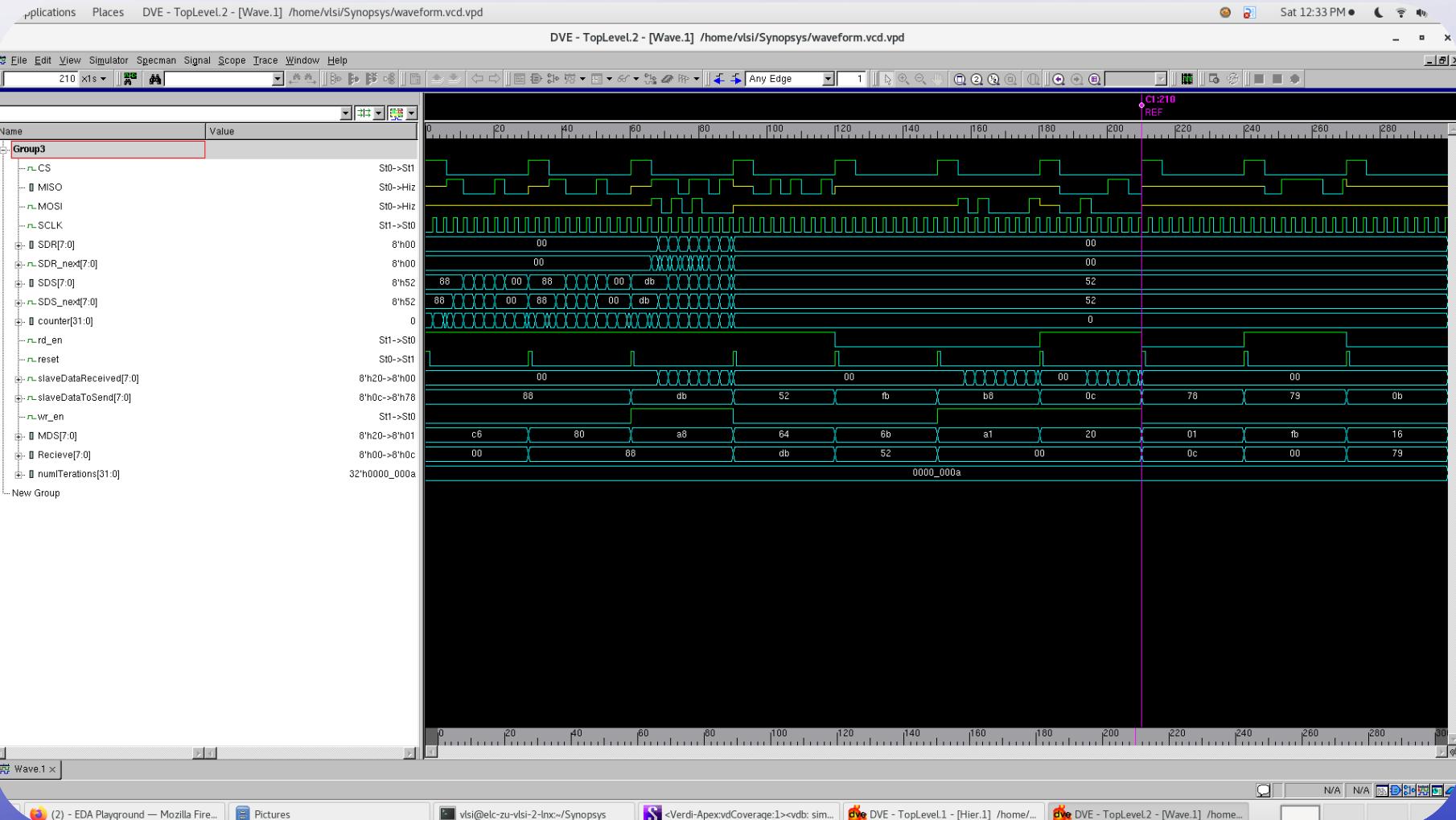
rm -f _cuarc*.so _csrc*.so pre_vcsobj_*.so share_vcsobj_*.so
if [ -x ..simv ]; then chmod a+x ..simv; fi
g++ -o ..simv -m32 -rdynamic -Wl,-rpath=$ORIGIN/simv.daidir -Wl,-rpath=/eda/synopsys/vcs/V-2023.12/linux/lib -Leda/synopsys/vcs/V-2023.12/linux/lib -Wl,-rpath-link=/
obj_amcQw_d.o __13555_archive_1.so __prev_archive_1.so SIM_l.o rmapats_mop.o rmapats_o rmar_o rmar_nd.o rmar_llvm_0_1.o rmar_llvm_0_0.o -lvirsim -lerrorinf -lsnpsmalloc -lvfs -lvcsnew -ls
improfile -lreader_common /eda/synopsys/vcs/V-2023.12/linux/lib/libBA.a -luclinative /eda/synopsys/vcs/V-2023.12/linux/lib/vcs_tls.o -Wl,-whole-archive -lvcsucli -Wl,-no-whole-archive /eda/synopsys/vcs/V-2023.12/linux/lib/vcs_save_restore_new.o /eda/synopsys/vcs/V-2023.12/linux/lib/ctype-stubs_32.a -ldl -lc -lm -lpthread -ldl
..simv up to date

CPU time: .182 seconds to compile + .216 seconds to elab + .132 seconds to link
[vlsi@elc-zu-vlsi-2-lnx Synopsys]$ 
[vlsi@elc-zu-vlsi-2-lnx Synopsys]$ 
[vlsi@elc-zu-vlsi-2-lnx Synopsys]$ 
[vlsi@elc-zu-vlsi-2-lnx Synopsys]$ ./simv -cm line+cond+fsm+tgl
Info: [VCS_SAVE_RESTORE_INFO] ASLR (Address Space Layout Randomization) is detected on the machine. To enable $save functionality, ASLR will be switched off and simv re-executed.
Please use '-no_save' simv switch to avoid re-execution or '-suppress=ASLR_DETECTED_INFO' to suppress this message.
Chronologic VCS simulator copyright 1991-2023
Contains Synopsys proprietary information.
Compiler version V-2023.12; Runtime version V-2023.12; May 11 12:21 2024

VCS Coverage Metrics Release V-2023.12 Copyright (c) 1991-2023 by Synopsys Inc.
Master received Data 88 from slave successfully
Master received Data 88 from slave successfully
Master received Data db from slave successfully
Slave received Data a8 from Master successfully
Master received Data 52 from slave successfully
Slave received Data a1 from Master successfully
Master received Data 0c from slave successfully
Slave received Data 20 from Master successfully
Master received Data 79 from slave successfully
$finish called from file "FINALPROJRCT/tb.sv", line 150.
$finish at simulation time 300

(v2) - EDA Playground — Mozilla Firefox
(v3) - EDA Playground — Mozilla Firefox
(v4) - EDA Playground — Mozilla Firefox
(v5) - EDA Playground — Mozilla Firefox
(v6) - EDA Playground — Mozilla Firefox
(v7) - EDA Playground — Mozilla Firefox
(v8) - EDA Playground — Mozilla Firefox
(v9) - EDA Playground — Mozilla Firefox
(v10) - EDA Playground — Mozilla Firefox
(v11) - EDA Playground — Mozilla Firefox
(v12) - EDA Playground — Mozilla Firefox
(v13) - EDA Playground — Mozilla Firefox
(v14) - EDA Playground — Mozilla Firefox
(v15) - EDA Playground — Mozilla Firefox
(v16) - EDA Playground — Mozilla Firefox
(v17) - EDA Playground — Mozilla Firefox
(v18) - EDA Playground — Mozilla Firefox
(v19) - EDA Playground — Mozilla Firefox
(v20) - EDA Playground — Mozilla Firefox
(v21) - EDA Playground — Mozilla Firefox
(v22) - EDA Playground — Mozilla Firefox
(v23) - EDA Playground — Mozilla Firefox
(v24) - EDA Playground — Mozilla Firefox
(v25) - EDA Playground — Mozilla Firefox
(v26) - EDA Playground — Mozilla Firefox
(v27) - EDA Playground — Mozilla Firefox
(v28) - EDA Playground — Mozilla Firefox
(v29) - EDA Playground — Mozilla Firefox
(v30) - EDA Playground — Mozilla Firefox
(v31) - EDA Playground — Mozilla Firefox
(v32) - EDA Playground — Mozilla Firefox
(v33) - EDA Playground — Mozilla Firefox
(v34) - EDA Playground — Mozilla Firefox
(v35) - EDA Playground — Mozilla Firefox
(v36) - EDA Playground — Mozilla Firefox
(v37) - EDA Playground — Mozilla Firefox
(v38) - EDA Playground — Mozilla Firefox
(v39) - EDA Playground — Mozilla Firefox
(v40) - EDA Playground — Mozilla Firefox
(v41) - EDA Playground — Mozilla Firefox
(v42) - EDA Playground — Mozilla Firefox
(v43) - EDA Playground — Mozilla Firefox
(v44) - EDA Playground — Mozilla Firefox
(v45) - EDA Playground — Mozilla Firefox
(v46) - EDA Playground — Mozilla Firefox
(v47) - EDA Playground — Mozilla Firefox
(v48) - EDA Playground — Mozilla Firefox
(v49) - EDA Playground — Mozilla Firefox
(v50) - EDA Playground — Mozilla Firefox
(v51) - EDA Playground — Mozilla Firefox
(v52) - EDA Playground — Mozilla Firefox
(v53) - EDA Playground — Mozilla Firefox
(v54) - EDA Playground — Mozilla Firefox
(v55) - EDA Playground — Mozilla Firefox
(v56) - EDA Playground — Mozilla Firefox
(v57) - EDA Playground — Mozilla Firefox
(v58) - EDA Playground — Mozilla Firefox
(v59) - EDA Playground — Mozilla Firefox
(v60) - EDA Playground — Mozilla Firefox
(v61) - EDA Playground — Mozilla Firefox
(v62) - EDA Playground — Mozilla Firefox
(v63) - EDA Playground — Mozilla Firefox
(v64) - EDA Playground — Mozilla Firefox
(v65) - EDA Playground — Mozilla Firefox
(v66) - EDA Playground — Mozilla Firefox
(v67) - EDA Playground — Mozilla Firefox
(v68) - EDA Playground — Mozilla Firefox
(v69) - EDA Playground — Mozilla Firefox
(v70) - EDA Playground — Mozilla Firefox
(v71) - EDA Playground — Mozilla Firefox
(v72) - EDA Playground — Mozilla Firefox
(v73) - EDA Playground — Mozilla Firefox
(v74) - EDA Playground — Mozilla Firefox
(v75) - EDA Playground — Mozilla Firefox
(v76) - EDA Playground — Mozilla Firefox
(v77) - EDA Playground — Mozilla Firefox
(v78) - EDA Playground — Mozilla Firefox
(v79) - EDA Playground — Mozilla Firefox
(v80) - EDA Playground — Mozilla Firefox
(v81) - EDA Playground — Mozilla Firefox
(v82) - EDA Playground — Mozilla Firefox
(v83) - EDA Playground — Mozilla Firefox
(v84) - EDA Playground — Mozilla Firefox
(v85) - EDA Playground — Mozilla Firefox
(v86) - EDA Playground — Mozilla Firefox
(v87) - EDA Playground — Mozilla Firefox
(v88) - EDA Playground — Mozilla Firefox
(v89) - EDA Playground — Mozilla Firefox
(v90) - EDA Playground — Mozilla Firefox
(v91) - EDA Playground — Mozilla Firefox
(v92) - EDA Playground — Mozilla Firefox
(v93) - EDA Playground — Mozilla Firefox
(v94) - EDA Playground — Mozilla Firefox
(v95) - EDA Playground — Mozilla Firefox
(v96) - EDA Playground — Mozilla Firefox
(v97) - EDA Playground — Mozilla Firefox
(v98) - EDA Playground — Mozilla Firefox
(v99) - EDA Playground — Mozilla Firefox
(v100) - EDA Playground — Mozilla Firefox
(v101) - EDA Playground — Mozilla Firefox
(v102) - EDA Playground — Mozilla Firefox
(v103) - EDA Playground — Mozilla Firefox
(v104) - EDA Playground — Mozilla Firefox
(v105) - EDA Playground — Mozilla Firefox
(v106) - EDA Playground — Mozilla Firefox
(v107) - EDA Playground — Mozilla Firefox
(v108) - EDA Playground — Mozilla Firefox
(v109) - EDA Playground — Mozilla Firefox
(v110) - EDA Playground — Mozilla Firefox
(v111) - EDA Playground — Mozilla Firefox
(v112) - EDA Playground — Mozilla Firefox
(v113) - EDA Playground — Mozilla Firefox
(v114) - EDA Playground — Mozilla Firefox
(v115) - EDA Playground — Mozilla Firefox
(v116) - EDA Playground — Mozilla Firefox
(v117) - EDA Playground — Mozilla Firefox
(v118) - EDA Playground — Mozilla Firefox
(v119) - EDA Playground — Mozilla Firefox
(v120) - EDA Playground — Mozilla Firefox
(v121) - EDA Playground — Mozilla Firefox
(v122) - EDA Playground — Mozilla Firefox
(v123) - EDA Playground — Mozilla Firefox
(v124) - EDA Playground — Mozilla Firefox
(v125) - EDA Playground — Mozilla Firefox
(v126) - EDA Playground — Mozilla Firefox
(v127) - EDA Playground — Mozilla Firefox
(v128) - EDA Playground — Mozilla Firefox
(v129) - EDA Playground — Mozilla Firefox
(v130) - EDA Playground — Mozilla Firefox
(v131) - EDA Playground — Mozilla Firefox
(v132) - EDA Playground — Mozilla Firefox
(v133) - EDA Playground — Mozilla Firefox
(v134) - EDA Playground — Mozilla Firefox
(v135) - EDA Playground — Mozilla Firefox
(v136) - EDA Playground — Mozilla Firefox
(v137) - EDA Playground — Mozilla Firefox
(v138) - EDA Playground — Mozilla Firefox
(v139) - EDA Playground — Mozilla Firefox
(v140) - EDA Playground — Mozilla Firefox
(v141) - EDA Playground — Mozilla Firefox
(v142) - EDA Playground — Mozilla Firefox
(v143) - EDA Playground — Mozilla Firefox
(v144) - EDA Playground — Mozilla Firefox
(v145) - EDA Playground — Mozilla Firefox
(v146) - EDA Playground — Mozilla Firefox
(v147) - EDA Playground — Mozilla Firefox
(v148) - EDA Playground — Mozilla Firefox
(v149) - EDA Playground — Mozilla Firefox
(v150) - EDA Playground — Mozilla Firefox
(v151) - EDA Playground — Mozilla Firefox
(v152) - EDA Playground — Mozilla Firefox
(v153) - EDA Playground — Mozilla Firefox
(v154) - EDA Playground — Mozilla Firefox
(v155) - EDA Playground — Mozilla Firefox
(v156) - EDA Playground — Mozilla Firefox
(v157) - EDA Playground — Mozilla Firefox
(v158) - EDA Playground — Mozilla Firefox
(v159) - EDA Playground — Mozilla Firefox
(v160) - EDA Playground — Mozilla Firefox
(v161) - EDA Playground — Mozilla Firefox
(v162) - EDA Playground — Mozilla Firefox
(v163) - EDA Playground — Mozilla Firefox
(v164) - EDA Playground — Mozilla Firefox
(v165) - EDA Playground — Mozilla Firefox
(v166) - EDA Playground — Mozilla Firefox
(v167) - EDA Playground — Mozilla Firefox
(v168) - EDA Playground — Mozilla Firefox
(v169) - EDA Playground — Mozilla Firefox
(v170) - EDA Playground — Mozilla Firefox
(v171) - EDA Playground — Mozilla Firefox
(v172) - EDA Playground — Mozilla Firefox
(v173) - EDA Playground — Mozilla Firefox
(v174) - EDA Playground — Mozilla Firefox
(v175) - EDA Playground — Mozilla Firefox
(v176) - EDA Playground — Mozilla Firefox
(v177) - EDA Playground — Mozilla Firefox
(v178) - EDA Playground — Mozilla Firefox
(v179) - EDA Playground — Mozilla Firefox
(v180) - EDA Playground — Mozilla Firefox
(v181) - EDA Playground — Mozilla Firefox
(v182) - EDA Playground — Mozilla Firefox
(v183) - EDA Playground — Mozilla Firefox
(v184) - EDA Playground — Mozilla Firefox
(v185) - EDA Playground — Mozilla Firefox
(v186) - EDA Playground — Mozilla Firefox
(v187) - EDA Playground — Mozilla Firefox
(v188) - EDA Playground — Mozilla Firefox
(v189) - EDA Playground — Mozilla Firefox
(v190) - EDA Playground — Mozilla Firefox
(v191) - EDA Playground — Mozilla Firefox
(v192) - EDA Playground — Mozilla Firefox
(v193) - EDA Playground — Mozilla Firefox
(v194) - EDA Playground — Mozilla Firefox
(v195) - EDA Playground — Mozilla Firefox
(v196) - EDA Playground — Mozilla Firefox
(v197) - EDA Playground — Mozilla Firefox
(v198) - EDA Playground — Mozilla Firefox
(v199) - EDA Playground — Mozilla Firefox
(v200) - EDA Playground — Mozilla Firefox
(v201) - EDA Playground — Mozilla Firefox
(v202) - EDA Playground — Mozilla Firefox
(v203) - EDA Playground — Mozilla Firefox
(v204) - EDA Playground — Mozilla Firefox
(v205) - EDA Playground — Mozilla Firefox
(v206) - EDA Playground — Mozilla Firefox
(v207) - EDA Playground — Mozilla Firefox
(v208) - EDA Playground — Mozilla Firefox
(v209) - EDA Playground — Mozilla Firefox
(v210) - EDA Playground — Mozilla Firefox
(v211) - EDA Playground — Mozilla Firefox
(v212) - EDA Playground — Mozilla Firefox
(v213) - EDA Playground — Mozilla Firefox
(v214) - EDA Playground — Mozilla Firefox
(v215) - EDA Playground — Mozilla Firefox
(v216) - EDA Playground — Mozilla Firefox
(v217) - EDA Playground — Mozilla Firefox
(v218) - EDA Playground — Mozilla Firefox
(v219) - EDA Playground — Mozilla Firefox
(v220) - EDA Playground — Mozilla Firefox
(v221) - EDA Playground — Mozilla Firefox
(v222) - EDA Playground — Mozilla Firefox
(v223) - EDA Playground — Mozilla Firefox
(v224) - EDA Playground — Mozilla Firefox
(v225) - EDA Playground — Mozilla Firefox
(v226) - EDA Playground — Mozilla Firefox
(v227) - EDA Playground — Mozilla Firefox
(v228) - EDA Playground — Mozilla Firefox
(v229) - EDA Playground — Mozilla Firefox
(v230) - EDA Playground — Mozilla Firefox
(v231) - EDA Playground — Mozilla Firefox
(v232) - EDA Playground — Mozilla Firefox
(v233) - EDA Playground — Mozilla Firefox
(v234) - EDA Playground — Mozilla Firefox
(v235) - EDA Playground — Mozilla Firefox
(v236) - EDA Playground — Mozilla Firefox
(v237) - EDA Playground — Mozilla Firefox
(v238) - EDA Playground — Mozilla Firefox
(v239) - EDA Playground — Mozilla Firefox
(v240) - EDA Playground — Mozilla Firefox
(v241) - EDA Playground — Mozilla Firefox
(v242) - EDA Playground — Mozilla Firefox
(v243) - EDA Playground — Mozilla Firefox
(v244) - EDA Playground — Mozilla Firefox
(v245) - EDA Playground — Mozilla Firefox
(v246) - EDA Playground — Mozilla Firefox
(v247) - EDA Playground — Mozilla Firefox
(v248) - EDA Playground — Mozilla Firefox
(v249) - EDA Playground — Mozilla Firefox
(v250) - EDA Playground — Mozilla Firefox
(v251) - EDA Playground — Mozilla Firefox
(v252) - EDA Playground — Mozilla Firefox
(v253) - EDA Playground — Mozilla Firefox
(v254) - EDA Playground — Mozilla Firefox
(v255) - EDA Playground — Mozilla Firefox
(v256) - EDA Playground — Mozilla Firefox
(v257) - EDA Playground — Mozilla Firefox
(v258) - EDA Playground — Mozilla Firefox
(v259) - EDA Playground — Mozilla Firefox
(v260) - EDA Playground — Mozilla Firefox
(v261) - EDA Playground — Mozilla Firefox
(v262) - EDA Playground — Mozilla Firefox
(v263) - EDA Playground — Mozilla Firefox
(v264) - EDA Playground — Mozilla Firefox
(v265) - EDA Playground — Mozilla Firefox
(v266) - EDA Playground — Mozilla Firefox
(v267) - EDA Playground — Mozilla Firefox
(v268) - EDA Playground — Mozilla Firefox
(v269) - EDA Playground — Mozilla Firefox
(v270) - EDA Playground — Mozilla Firefox
(v271) - EDA Playground — Mozilla Firefox
(v272) - EDA Playground — Mozilla Firefox
(v273) - EDA Playground — Mozilla Firefox
(v274) - EDA Playground — Mozilla Firefox
(v275) - EDA Playground — Mozilla Firefox
(v276) - EDA Playground — Mozilla Firefox
(v277) - EDA Playground — Mozilla Firefox
(v278) - EDA Playground — Mozilla Firefox
(v279) - EDA Playground — Mozilla Firefox
(v280) - EDA Playground — Mozilla Firefox
(v281) - EDA Playground — Mozilla Firefox
(v282) - EDA Playground — Mozilla Firefox
(v283) - EDA Playground — Mozilla Firefox
(v284) - EDA Playground — Mozilla Firefox
(v285) - EDA Playground — Mozilla Firefox
(v286) - EDA Playground — Mozilla Firefox
(v287) - EDA Playground — Mozilla Firefox
(v288) - EDA Playground — Mozilla Firefox
(v289) - EDA Playground — Mozilla Firefox
(v290) - EDA Playground — Mozilla Firefox
(v291) - EDA Playground — Mozilla Firefox
(v292) - EDA Playground — Mozilla Firefox
(v293) - EDA Playground — Mozilla Firefox
(v294) - EDA Playground — Mozilla Firefox
(v295) - EDA Playground — Mozilla Firefox
(v296) - EDA Playground — Mozilla Firefox
(v297) - EDA Playground — Mozilla Firefox
(v298) - EDA Playground — Mozilla Firefox
(v299) - EDA Playground — Mozilla Firefox
(v300) - EDA Playground — Mozilla Firefox
```

# DVE WAVE FORM



# VERDI COVERAGE

The design 'simv.vdb' was loaded successfully.  
The following test is loaded from "simv.vdb",  
simv/test

The following flags are unrecognized for coverage and have been saved as 'verdi' debug flags:  
-c  
Please use Tools->Debug to launch the Verdi debugger with these flags.

Message

CovSrc.1: Slave\_tb.M1

CovDetail

File View Plan Exclusion Tools Window Help

<Verdi-Apex:vdCoverage:1><vdb: simv.vdb>

Sat 12:23 PM ● ☽ ☾

Applications Places <Verdi-Apex:vdCoverage:1><vdb: simv.vdb>

File View Plan Exclusion Tools Window Help

CovSrc.1: Slave\_tb.M1

CovDetail

Line Toggle FSM Condition Branch Assert

Uncovered CovSrc.1 Hvp CovDetail HvpDetail

Message

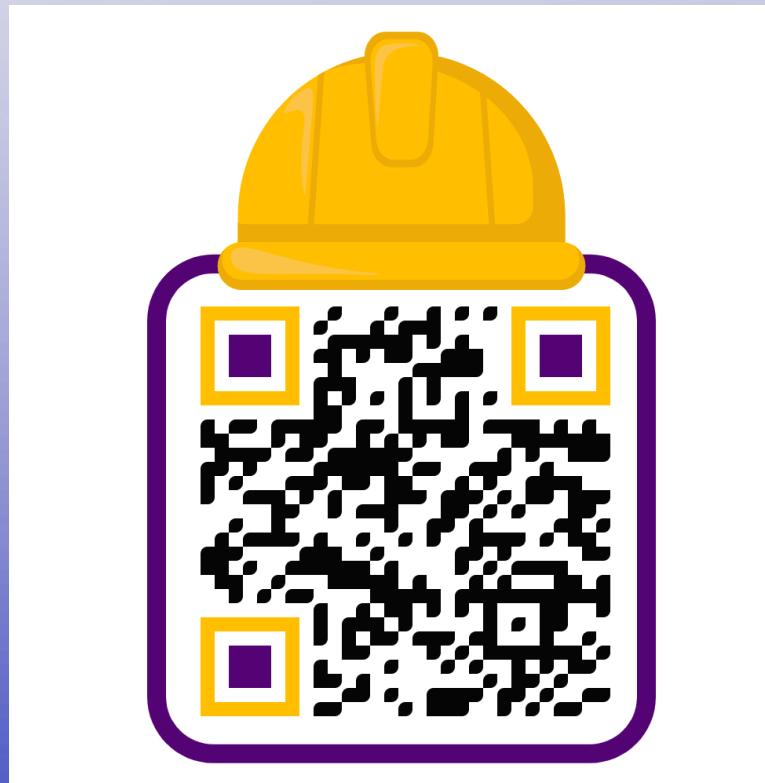
vlsi@elc-zu-vlsi-2-Inx:~\$ Synopsys

(2) - EDA Playground — Mozilla Fire... Pictures vlsi@elc-zu-vlsi-2-Inx:~\$ Synopsys <Verdi-Apex:vdCoverage:1><vdb: si...

(5) - EDA Playground — Mozilla Fire... Pictures vlsi@elc-zu-vlsi-2-Inx:~\$ Synopsys <Verdi-Apex:vdCoverage:1><vdb: si...

FOR MORE INFORMATION, CLICK ON THE GETHUB LINK  
OR SCAN THE QR

<https://github.com/tito642/Verification-of-SPI-slave-IP-with-systemVerilog>





THANKS