Digital Verification

SPI_RAM_FIFO Verification

By: Hassan Khaled
Ahmed Ashraf
Alaa Saleh
Eslam Hekal

RAM

- ## Test Plan

Function	How To Test
reset	constraint on input
reset impact	assertions
Write ADD	constraint on input & cover point
Read ADD	constraint on input & cover point
Write DATA	constraint on input & cover point
Read DATA	constraint on input & cover point

- ## Constraint Randomized Inputs

Reset
rx_valid
Din[9:8]
Din[7:0]

- # Code Coverage

```
#             ignore_bin transition_0                          448                    -
# Statement Coverage:
#     Enabled Coverage              Bins      Hits    Misses  Coverage
#     ----------------              ----      ----    ------  --------
#     Statements                       2         2         0   100.00%
#
# ================================Statement Details================================
#
# Statement Coverage for instance /ram3_tb_sv_unit --
#
#     Line         Item                      Count     Source
#     ----         ----                      -----     ------
#   File RAM3_C.sv
#     1                                                class C_RAM3;
#
#     2
#
#     3                                                rand logic [9:0] din;
```

```
#
# Toggle Coverage:
#     Enabled Coverage              Bins      Hits    Misses  Coverage
#     ----------------              ----      ----    ------  --------
#     Toggles                         96        92         4    95.83%
#
# ================================Toggle Details================================
#
# Toggle Coverage for instance /spi_ram_tb --
#
#                                           Node      1H->0L      0L->1H  "Coverage"
#                                           ----------------------------------------
#                                    address[7-0]           1           1      100.00
#                                          clk              1           1      100.00
#                                data_expected[7-0]           1           1      100.00
#                                       din[9-0]              1           1      100.00
#                                      dout[7-0]              1           1      100.00
#                                          flag               0           0        0.00
#                            mem_data_expected[7-0]           1           1      100.00
#                                        rst_n                1           1      100.00
#                                     rx_valid                1           1      100.00
#                                     tx_valid                0           0        0.00
#                                tx_valid_expected            1           1      100.00
#
# Total Node Count     =         48
# Toggled Node Count   =         46
# Untoggled Node Count =          2
#
# Toggle Coverage      =     95.83% (92 of 96 bins)
```

```
#                                        1405     All False Count
# Branch totals: 2 hits of 2 branches = 100.00%
#
# ------------------------------------IF Branch------------------------------------
#     60                                 2000     Count coming in to IF
#     60             1                    548     if(r_obj.din[9:8]== 2'b11)begin
#
#                                        1452     All False Count
# Branch totals: 2 hits of 2 branches = 100.00%
#
# ------------------------------------IF Branch------------------------------------
#     71                                  249     Count coming in to IF
#     71             1                    140     if(mem_data_expected !== dout) //$display("RIGHT !!,ac
pected);
#
#                                         109     All False Count
# Branch totals: 2 hits of 2 branches = 100.00%
#
#
# Condition Coverage:
#     Enabled Coverage              Bins   Covered    Misses  Coverage
#     ----------------              ----      ----    ------  --------
#     Conditions                       6         6         0   100.00%
#
# ================================Condition Details================================
#
# Condition Coverage for instance /spi_ram_tb --
```

- # Functional Coverage

```
# /spi_ram_tb/cover__add_in_range          spi_ram_tb Verilog  SVA  ram3_tb.sv(86)    856 Covered
# /spi_ram_tb/ram_dut/ram_sva_instance/cover__read_op
#                                          ram_sva Verilog  SVA  RAM3_AS.sv(38)   394 Covered
# /spi_ram_tb/ram_dut/ram_sva_instance/cover__write_op
#                                          ram_sva Verilog  SVA  RAM3_AS.sv(30)   448 Covered
# /spi_ram_tb/ram_dut/ram_sva_instance/cover__tx_deassertion
#                                          ram_sva Verilog  SVA  RAM3_AS.sv(22)     0 ZERO
# /spi_ram_tb/ram_dut/ram_sva_instance/cover__tx_assertion
#                                          ram_sva Verilog  SVA  RAM3_AS.sv(14)     0 ZERO
# /spi_ram_tb/ram_dut/ram_sva_instance/rst_out_cov
#                                          ram_sva Verilog  SVA  RAM3_AS.sv(47)    16 Covered
#
# TOTAL DIRECTIVE COVERAGE: 66.66%  COVERS: 6
#
# ASSERTION RESULTS:
# ---------------------------------------------------------------------
# Name                 File(Line)                    Failure      Pass
#                                                    Count        Count
# ---------------------------------------------------------------------
# /spi_ram_tb/assert__add_in_range
#                      ram3_tb.sv(85)                       0          1
# /spi_ram_tb/#anonblk#109154386#42#4#/#ublk#109154386#42/immed__44
#                      ram3_tb.sv(44)                       0          1
# /spi_ram_tb/ram_dut/ram_sva_instance/assert__read_op
#                      RAM3_AS.sv(37)                       0          1
# /spi_ram_tb/ram_dut/ram_sva_instance/assert__write_op
#                      RAM3_AS.sv(29)                       0          1
# /spi_ram_tb/ram_dut/ram_sva_instance/assert__tx_deassertion
#                      RAM3_AS.sv(21)                       1          0
# /spi_ram_tb/ram_dut/ram_sva_instance/assert__tx_assertion
#                      RAM3_AS.sv(13)                     534          0
# /spi_ram_tb/ram_dut/ram_sva_instance/rst_out
#                      RAM3_AS.sv(46)                       0          1
#
# Total Coverage By Instance (filtered view): 88.81%
#
# End time: 08:21:13 on May 02,2023, Elapsed time: 0:00:00
```

```
=== Instance: /ram3_tb_sv_unit
=== Design Unit: work.ram3_tb_sv_unit
================================================================================

Covergroup Coverage:
    Covergroups                 1        na        na    90.49%
        Coverpoints/Crosses     6        na        na        na
            Covergroup Bins   778       741        37    95.24%
--------------------------------------------------------------------------------
Covergroup                                          Metric       Goal       Bins    Status
--------------------------------------------------------------------------------
 TYPE /ram3_tb_sv_unit/C_RAM3/r_cg                  90.49%        100          -    Uncovered
    covered/total bins:                               741        778          -
    missing/total bins:                                37        778          -
    % Hit:                                         95.24%        100          -
    Coverpoint di_9_8                             100.00%        100          -    Covered
        covered/total bins:                             4          4          -
        missing/total bins:                             0          4          -
        % Hit:                                    100.00%        100          -
    Coverpoint di_7_0                             100.00%        100          -    Covered
        covered/total bins:                           256        256          -
        missing/total bins:                             0        256          -
        % Hit:                                    100.00%        100          -
    Coverpoint rx                                 100.00%        100          -    Covered
        covered/total bins:                             2          2          -
        missing/total bins:                             0          2          -
        % Hit:                                    100.00%        100          -
    Coverpoint tx                                  50.00%        100          -    Uncovered
        covered/total bins:                             1          2          -
        missing/total bins:                             1          2          -
        % Hit:                                     50.00%        100          -
    Coverpoint dot                                100.00%        100          -    Covered
        covered/total bins:                             2          2          -
        missing/total bins:                             0          2          -
```

- # Sequential Domain Coverage

```
# COVERGROUP COVERAGE:
# ----------------------------------------------------------------------------------------------------
# Covergroup                                          Metric       Goal       Bins    Status
#
# ----------------------------------------------------------------------------------------------------
#  TYPE /ram3_tb_sv_unit/C_RAM3/r_cg                  88.63%        100          -    Uncovered
#     covered/total bins:                               684        778          -
#     missing/total bins:                                94        778          -
#     % Hit:                                         87.91%        100          -
#     Coverpoint di_9_8                             100.00%        100          -    Covered
#         covered/total bins:                             4          4          -
#         missing/total bins:                             0          4          -
#         % Hit:                                    100.00%        100          -
#     Coverpoint di_7_0                             100.00%        100          -    Covered
#         covered/total bins:                           256        256          -
#         missing/total bins:                             0        256          -
#         % Hit:                                    100.00%        100          -
#     Coverpoint rx                                 100.00%        100          -    Covered
#         covered/total bins:                             2          2          -
#         missing/total bins:                             0          2          -
#         % Hit:                                    100.00%        100          -
#     Coverpoint tx                                  50.00%        100          -    Uncovered
#         covered/total bins:                             1          2          -
#         missing/total bins:                             1          2          -
#         % Hit:                                     50.00%        100          -
#     Coverpoint dot                                100.00%        100          -    Covered
#         covered/total bins:                             2          2          -
```

- # Bug Report

Bug	Design_input_bug	Expected_behaviour	Observed_behaviour
• Tx_valid remains high during all the operation.	• When Tx_valid asserted still high.	• Must fall after read operation is done.	• Tx_valid remains high all simulation
• Write Operation within resetting	• Rst_n = 0	• No operations done while resetting	• Data is written into ram
• read_address doesn't reseted.	• Rst_n = 0	• Must fall when reset is asserted.	• Read address keeps its value
• write_address doesn't reseted.	• Rst_n = 0	Must fall when reset is asserted.	• Write address keeps its value

- # Waveform

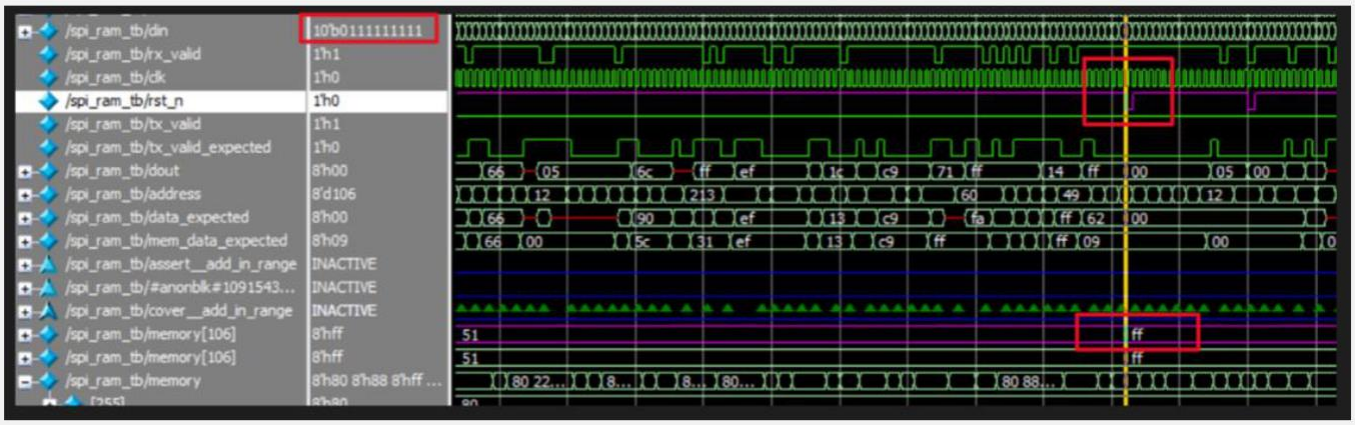


Figure 1. writing data during Resetting

Assertions

Feature	Assertion
When read data expecting tx_valid to be high	@(posedge clk) disable iff(!rst_n) (din[9:8] == 2'b11) |=> ##[1:2] $rose(tx_valid);
When tx_valid is high it still stable until read finish	@(posedge clk) disable iff(!rst_n) $rose(tx_valid) |=> ((!tx_valid) throughout din[9:8] != 2'b11);
When even write add operates expect the next operation to be write data	@(posedge clk) disable iff(!rst_n) (din[9:8] == 2'b00) |-> ##1 (din[9:8] == 2'b01);
When even read add operates expect the next operation to be read data	@(posedge clk) disable iff(!rst_n) (din[9:8] == 2'b10) |-> ##1 (din[9:8] == 2'b11);
When ever reset is asserted expect out to be zero	@(posedge clk) $fell(rst_n) |-> (dout == 0);

SPI Wrapper

- ## Test Plan

Function	How To Test
Write ADD	Constrains on input && cover point
Read ADD	Constrains on input && cover point
Write DATA	Constrains on input && cover point
Read DATA	Constrains on input && cover point
Write OPERATION	Constrains on input
Read OPERATION	Constrains on input
DATA & ADD to be written	Constrains on input && cover point
Write in all ADD	Cross coverage
Write all DATA	Cross coverage
Read from all ADD	Cross coverage
Reset toggle	Constrains on input && cover point
Slave Select toggle	Cover point
Reset impact	Assertion
Slave Select impact	Assertion
MISO behaviour	Assertion
Test seqeucnce	Declare dynamic array “data_to_write” to store constrained randomized operations to make sure every read add follow by read data and every write add followed by write data Then looping on the array at every iteration we update a declared associative array to be a reference to compare output with . After updating the associative array we assign ss_n to 0 then iterate to send the operation on MOSI. And at every iteration of dynamic array we check and save add if write or read add and if it read data we assign expected out for wanted add . Then iterate on it and compare with MISO signal

- # Constraint Randomized Inputs

Reset
MOSI

- # Code Coverage

```
#
# Statement Coverage:
#     Enabled Coverage              Bins      Hits    Misses  Coverage
#     ----------------              ----      ----    ------  --------
#     Statements                      30        30         0   100.00%
#
# ================================Statement Details================================
#
# Statement Coverage for instance /spi_wrapper_tb --
#
#     Line         Item                      Count     Source
#     ----         ----                      -----     ------
#   File spi_wrapper_tb.sv
#     3                                              module spi_wrapper_tb();
#
#     4
#
#     5                                              bit MOSI,clk,rst_n,SS_n;
#
#     6                                              bit MISO;
#
#     7                                              bit MISO_expected;
#
#     8
#
```

```
#     Toggles                         28        27         1    96.42%
#
# ================================Toggle Details================================
#
# Toggle Coverage for instance /spi_wrapper_tb --
#
#                                             Node      1H->0L      0L->1H  "Coverage"
#                                             ----------------------------------------
#                                             MISO           1           1      100.00
#                                     MISO_expected          1           1      100.00
#                                             MOSI           1           1      100.00
#                                             SS_n           1           1      100.00
#                                              clk           1           1      100.00
#                                 data_expected[7-0]         1           1      100.00
#                                            rst_n           0           1       50.00
#
# Total Node Count     =         14
# Toggled Node Count   =         13
# Untoggled Node Count =          1
#
# Toggle Coverage      =      96.42% (27 of 28 bins)
#
```

```
# Branch totals: 2 hits of 2 branches = 100.00%
#
# ------------------------------------IF Branch------------------------------------
#     64                                     2000     Count coming in to IF
#     64              1                       329     if(data_to_write[i][10:8] == 3'b110)begin
#
#                                            1671     All False Count
# Branch totals: 2 hits of 2 branches = 100.00%
#
# ------------------------------------IF Branch------------------------------------
#     78                                     2000     Count coming in to IF
#     78              1                       329     if(data_to_write[i][10:8] == 3'b111)
#
#                                            1671     All False Count
# Branch totals: 2 hits of 2 branches = 100.00%
#
#
# Condition Coverage:
#     Enabled Coverage              Bins   Covered    Misses  Coverage
#     ----------------              ----      ----    ------  --------
#     Conditions                       3         3         0   100.00%
#
# ================================Condition Details================================
#
```

```
#
# DIRECTIVE COVERAGE:
# --------------------------------------------------------------------------------------------------
# Name                                     Design Design   Lang File(Line)      Hits Status
#                                          Unit   UnitType
# --------------------------------------------------------------------------------------------------
# /spi_wrapper_tb/spi_wr_dut/sw_sva_dut/cover__ssn_prop
#                                          spi_wrapper_sva Verilog  SVA  spi_wrapper_AS.sv(21)
#                                                                                329 Covered
# /spi_wrapper_tb/spi_wr_dut/sw_sva_dut/cover__rst_prop
#                                          spi_wrapper_sva Verilog  SVA  spi_wrapper_AS.sv(12)
#                                                                                654 Covered
# Toggle Coverage:
#     Enabled Coverage              Bins      Hits    Misses  Coverage
#     ----------------              ----      ----    ------  --------
#     Toggles                         10         9         1    90.00%
#
# ================================Toggle Details================================
#
# Toggle Coverage for instance /spi_wrapper_tb/spi_wr_dut/sw_sva_dut --
```

```
DIRECTIVE COVERAGE:
--------------------------------------------------------------------------------------------------
Name                                     Design Design   Lang File(Line)      Hits Status
                                         Unit   UnitType
--------------------------------------------------------------------------------------------------
/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/cover__ssn_prop
                                         spi_wrapper_sva Verilog  SVA  spi_wrapper_AS.sv(21)
                                                                               329 Covered
/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/cover__rst_prop
                                         spi_wrapper_sva Verilog  SVA  spi_wrapper_AS.sv(12)
                                                                               654 Covered

TOTAL DIRECTIVE COVERAGE: 100.00%  COVERS: 2

Total Coverage By Instance (filtered view): 50.00%
```

- # Functional Coverage

```
Covergroup Coverage:
    Covergroups                 1        na        na   100.00%
        Coverpoints/Crosses     3        na        na        na
            Covergroup Bins   774       774         0   100.00%
--------------------------------------------------------------------------------------------------
Covergroup                                          Metric       Goal       Bins    Status
--------------------------------------------------------------------------------------------------
 TYPE /spi_wrapper_tb_sv_unit/SW_C/cg              100.00%        100          -    Covered
    covered/total bins:                                774        774          -
    missing/total bins:                                  0        774          -
    % Hit:                                          100.00%        100          -
    Coverpoint joker_control                        100.00%        100          -    Covered
        covered/total bins:                                4          4          -
        missing/total bins:                                0          4          -
        % Hit:                                      100.00%        100          -
        bin w_ad                                       9587          1          -    Covered
        bin w_dt                                       9586          1          -    Covered
        bin r_ad                                       4732          1          -    Covered
        bin r_dt                                       4732          1          -    Covered
    Coverpoint joker_data                           100.00%        100          -    Covered
        covered/total bins:                              256        256          -
        missing/total bins:                                0        256          -
        % Hit:                                      100.00%        100          -
        bin f_z                                        132          1          -    Covered
        bin f_o                                        127          1          -    Covered
        bin other[1]                                   120          1          -    Covered
        bin other[2]                                   119          1          -    Covered
        bin other[3]                                   107          1          -    Covered
        bin other[4]                                   104          1          -    Covered
        bin other[5]                                   109          1          -    Covered
        bin other[6]                                   118          1          -    Covered
```

- # Sequential Domain Coverage

```
# DIRECTIVE COVERAGE:
# --------------------------------------------------------------------------------------------------
# Name                                    Design Design   Lang File(Line)      Hits Status
#                                         Unit   UnitType
# --------------------------------------------------------------------------------------------------
# /spi_wrapper_tb/spi_wr_dut/sw_sva_dut/cover__ssn_prop
#                                         spi_wrapper_sva Verilog  SVA  spi_wrapper_AS.sv(21)
#                                                                           329 Covered
# /spi_wrapper_tb/spi_wr_dut/sw_sva_dut/cover__rst_prop
#                                         spi_wrapper_sva Verilog  SVA  spi_wrapper_AS.sv(12)
#                                                                           654 Covered
#
# TOTAL DIRECTIVE COVERAGE: 100.00%  COVERS: 2
#
# ASSERTION RESULTS:
# ---------------------------------------------------------------------
# Name                File(Line)                   Failure      Pass
#                                                  Count        Count
# ---------------------------------------------------------------------
# /spi_wrapper_tb/#anonblk#2691354#44#4#/#ublk#2691354#44/immed__46
#                     spi_wrapper_tb.sv(46)              0          1
# /spi_wrapper_tb/spi_wr_dut/sw_sva_dut/assert__ssn_prop
#                     spi_wrapper_AS.sv(20)            329          1
# /spi_wrapper_tb/spi_wr_dut/sw_sva_dut/assert__rst_prop
#                     spi_wrapper_AS.sv(11)           1347          1
#
# Total Coverage By Instance (filtered view): 75.43%
#
# End time: 01:34:42 on May 02,2023, Elapsed time: 0:00:00
# Errors: 0, Warnings: 0
```

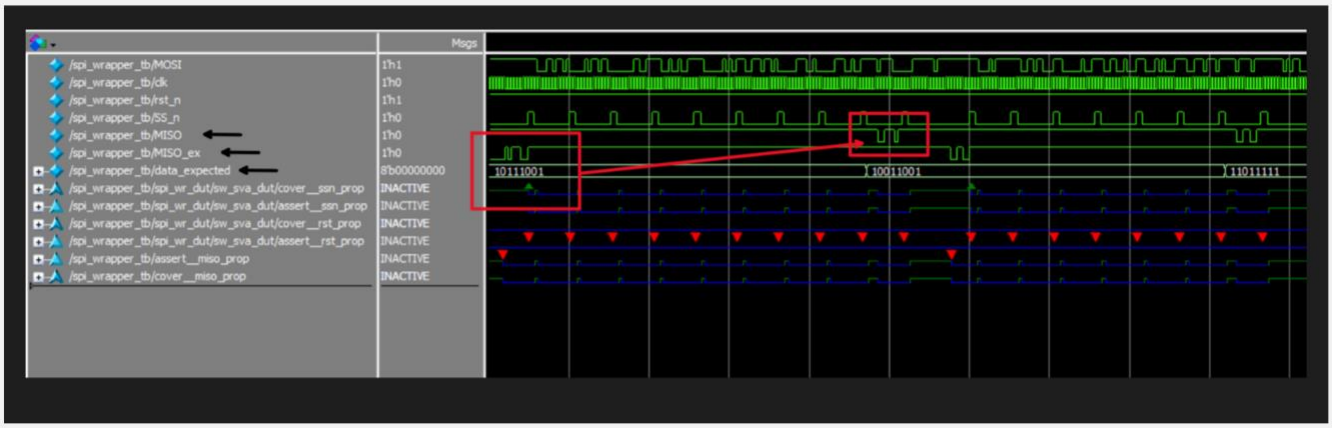
- # Bug Report

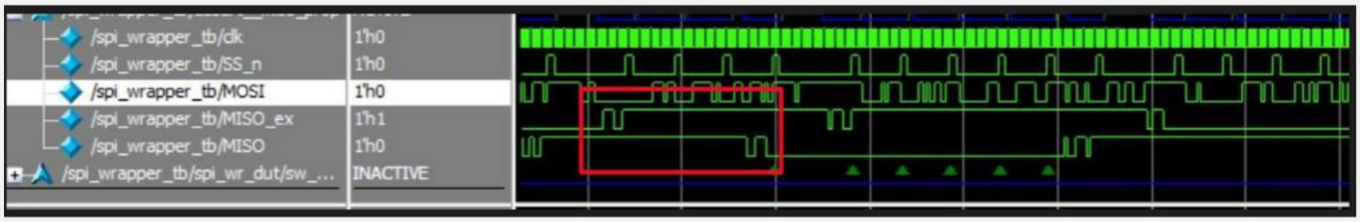
Bug	Design_input_bug	Expected_behaviour	Observed_behaviour
• Output from MISO during read operation from memory is reversed.	• MOSI with sequence 1->1->1 to read data.	Expected serial output starting From MSB.	• Serial Output in reversed way starting from LSB.
• Output from MISO has delay of two Read operation in case of multiple consecutive operations.	• MOSI with various consecutive sequence of different operations either write or read operations.	Expected serial output starting From MSB.	• Data out from MISO after read data command within 8 clk cycles after tx_valid flag.
• Output from MISO doesn't reseted.	• rst_n fell to zero.	MISO reseted to zero.	• MISO keeps its value even when reset happens.
• MISO doesn't rested in IDLE state after end of operation.	• SS_n rose to one.	MISO reseted to zero.	• MISO keeps its value even when state is IDLE.

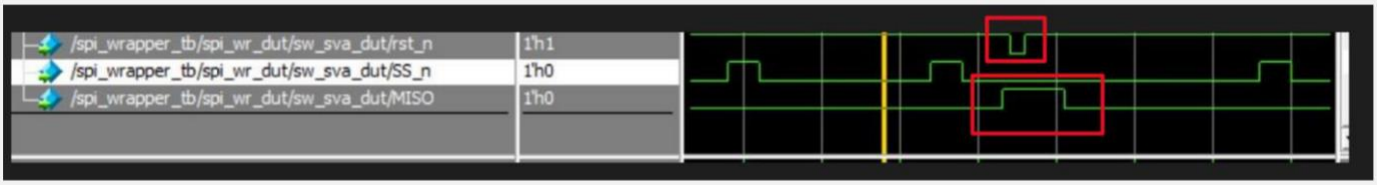
- # Direct Stimulus

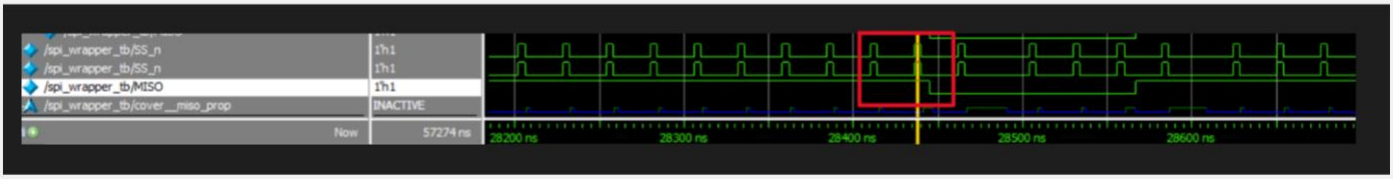
SS_n -> is directed in order to be asserted with the end of each operation and With different clock cycle delays according to type of operation.

- # Waveform


Msgs
/spi_wrapper_tb/MOSI
/spi_wrapper_tb/clk
/spi_wrapper_tb/rst_n
/spi_wrapper_tb/SS_n
/spi_wrapper_tb/MISO
/spi_wrapper_tb/MISO_ex
/spi_wrapper_tb/data_expected
/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/cover__ssn_prop
/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/assert__ssn_prop
/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/cover__rst_prop
/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/assert__rst_prop
/spi_wrapper_tb/assert__miso_prop
/spi_wrapper_tb/cover__miso_prop
10111001
10011001
11011111


/spi_wrapper_tb/clk
/spi_wrapper_tb/SS_n
/spi_wrapper_tb/MOSI
/spi_wrapper_tb/MISO_ex
/spi_wrapper_tb/MISO
/spi_wrapper_tb/spi_wr_dut/sw_...
INACTIVE


/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/rst_n
/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/SS_n
/spi_wrapper_tb/spi_wr_dut/sw_sva_dut/MISO


/spi_wrapper_tb/SS_n
/spi_wrapper_tb/SS_n
/spi_wrapper_tb/MISO
/spi_wrapper_tb/cover__miso_prop
INACTIVE
Now
57274 ns
28200 ns
28300 ns
28400 ns
28500 ns
28600 ns

- # Assertions

Feature	Assertion	Sequence
when reset is asserted | SS_n rise , MISO =0.	@(posedge clk) ($fell(rst_n) || $rose(SS_n)) |-> !MISO;	
when SS_n fall MOSI is high three times , SS_n remains low for 15 clk cycles then rise.	@(posedge clk) seq_read_data |=> !SS_n[*15] ##1 $rose(SS_n);	// sequence :: $fell(SS_n)##1 MOSI [*3];
when SS_n fall MOSI is high three times ,after 7 clk cycles MISO must remains as expected for 8 cycles	@ (posedge clk) seq_read_data |=> ##7 (MISO==MISO_ex)[*8]	// sequence :: $fell(SS_n)##1 MOSI [*3];

FIFO

- **Test Plan**

Function	How To Test
Data_in	Randomized
Wr_en	Randomized
Rd_en	Randomized
Rst_n	Randomized with constranse
Data_out	Compared to expected
Full	Cover bins & assertions
Almostfull	Cover bins & assertions
Empty	Cover bins & assertions
Almostempty	Cover bins & assertions
Overflow	Cover bins & assertions
Underflow	Cover bins & assertions
Test sequance	Write data to the fifo for n number with read stuck to zero and reset is randomized. Then read data for n with write stuck to zero and reset is randomized. Then write and read data randomized with reset. Then some constrained randomized sequence to highlight errors in data out
Reset impact	Assertion that all outs should be zero except empty signal
Write acknowledge	Cover bins & assertions

Code Coverage

-

```
Statement Coverage:
    Enabled Coverage              Bins      Hits    Misses  Coverage
    ----------------              ----      ----    ------  --------
    Statements                       4         4         0   100.00%

================================Statement Details================================

Statement Coverage for instance /top/f_if/asrt --

    Line         Item                      Count     Source
    ----         ----                      -----     ------
  File FIFO_ASER.sv
    2                                      1         module fifo_sva_a(FIFO_if.asert asrt
```

```
Toggle Coverage:
    Enabled Coverage              Bins      Hits    Misses  Coverage
    ----------------              ----      ----    ------  --------
    Toggles                        118       118         0   100.00%

================================Toggle Details================================

Toggle Coverage for instance /top/fif --

                                                 Node      1H->0L      0L->1H  "Coverage"
                                                 ----------------------------------------
                                           almostempty           1           1      100.00
                                            almostfull           1           1      100.00
                                                   clk           1           1      100.00
                                          data_in[15-0]          1           1      100.00
                                         data_out[15-0]          1           1      100.00
                                  data_out_expected[15-0]          1           1      100.00
                                                 empty           1           1      100.00
                                                  full           1           1      100.00
                                              overflow           1           1      100.00
                                                 rd_en           1           1      100.00
                                                 rst_n           1           1      100.00
                                             underflow           1           1      100.00
                                                 wr_ack          1           1      100.00
                                                 wr_en           1           1      100.00

Total Node Count        =        59
Toggled Node Count      =        59
Untoggled Node Count    =         0

Toggle Coverage         =    100.00% (118 of 118 bins)
```

```
Condition Coverage:
    Enabled Coverage              Bins    Covered     Misses  Coverage
    ----------------              ----    -------     ------  --------
    Conditions                       6          6          0   100.00%

================================Condition Details================================

Condition Coverage for instance /top/f_if/asrt --

  File FIFO_ASER.sv
----------------Focused Condition View-------------------
Line       10 Item    1  (asrt.wr_en && ~asrt.rd_en)
Condition totals: 2 of 2 input terms covered = 100.00%

  Input Term   Covered  Reason for no coverage   Hint
 -----------  --------  ----------------------  --------------
  asrt.wr_en         Y
```

```
Branch Coverage:
    Enabled Coverage              Bins      Hits    Misses  Coverage
    ----------------              ----      ----    ------  --------
    Branches                        10        10         0   100.00%

================================Branch Details================================

Branch Coverage for instance /top/f_if/asrt

    Line         Item                      Count     Source
    ----         ----                      -----     ------
  File FIFO_ASER.sv
------------------------------------IF Branch------------------------------------
    7                                      414     Count coming in to IF
    7              1                       53              if(~asrt.rst_n) beg

    9              1                       361       end else begin

Branch totals: 2 hits of 2 branches = 100.00%

------------------------------------IF Branch------------------------------------
```

- # Functional Coverage

```
================================================================================
=== Instance: /pack_FIFO
=== Design Unit: work.pack_FIFO
================================================================================

Covergroup Coverage:
    Covergroups                     1        na        na   100.00%
        Coverpoints/Crosses        10        na        na        na
            Covergroup Bins        32        32         0   100.00%
--------------------------------------------------------------------------------
Covergroup                                              Metric       Goal       Bins

--------------------------------------------------------------------------------
 TYPE /pack_FIFO/fiffo/fif                             100.00%        100          -
    covered/total bins:                                     32         32          -
    missing/total bins:                                      0         32          -
    % Hit:                                              100.00%        100          -
    Coverpoint ack                                      100.00%        100          -
        covered/total bins:                                  1          1          -
        missing/total bins:                                  0          1          -
        % Hit:                                          100.00%        100          -
    Coverpoint wr                                       100.00%        100          -
        covered/total bins:                                  4          4          -
        missing/total bins:                                  0          4          -
        % Hit:                                          100.00%        100          -
    Coverpoint rd                                       100.00%        100          -
        covered/total bins:                                  4          4          -
        missing/total bins:                                  0          4          -
        % Hit:                                          100.00%        100          -
    Coverpoint f                                        100.00%        100          -
        covered/total bins:                                  3          3          -
```

- # Sequential Domain Coverage

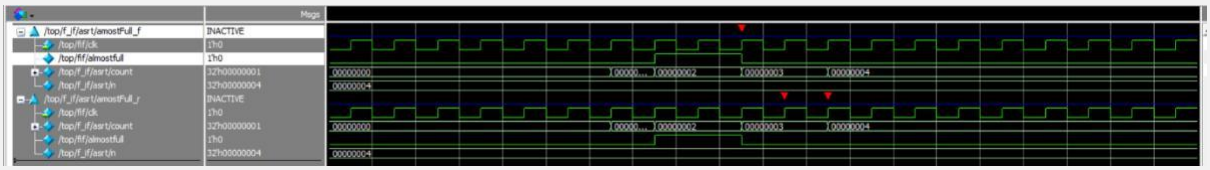
```
DIRECTIVE COVERAGE:
--------------------------------------------------------------------------------------------
Name                                    Design Design   Lang File(Line)      Hits Status
                                        Unit   UnitType
--------------------------------------------------------------------------------------------
/top/f_if/asrt/c_over_r                 fifo_sva_a Verilog  SVA  FIFO_ASER.sv(24)  34 Covered
/top/f_if/asrt/c_over_f                 fifo_sva_a Verilog  SVA  FIFO_ASER.sv(30)  26 Covered
/top/f_if/asrt/cfull_r                  fifo_sva_a Verilog  SVA  FIFO_ASER.sv(36) 116 Covered
/top/f_if/asrt/cfull_f                  fifo_sva_a Verilog  SVA  FIFO_ASER.sv(42)  17 Covered
/top/f_if/asrt/c_amostfull_r            fifo_sva_a Verilog  SVA  FIFO_ASER.sv(49)   0 ZERO
/top/f_if/asrt/c_amostfull_f            fifo_sva_a Verilog  SVA  FIFO_ASER.sv(54)  11 Covered
/top/f_if/asrt/c_amostempty_r           fifo_sva_a Verilog  SVA  FIFO_ASER.sv(62)  72 Covered
/top/f_if/asrt/c_amostempty_f           fifo_sva_a Verilog  SVA  FIFO_ASER.sv(68)  37 Covered
/top/f_if/asrt/cempty_r                 fifo_sva_a Verilog  SVA  FIFO_ASER.sv(76) 232 Covered
/top/f_if/asrt/cempty_f                 fifo_sva_a Verilog  SVA  FIFO_ASER.sv(81)  26 Covered
/top/f_if/asrt/cunder_r                 fifo_sva_a Verilog  SVA  FIFO_ASER.sv(88)  51 Covered
/top/f_if/asrt/cunder_f                 fifo_sva_a Verilog  SVA  FIFO_ASER.sv(94)  43 Covered
/top/f_if/asrt/c_wrAck_r                fifo_sva_a Verilog  SVA  FIFO_ASER.sv(102)
                                                                                  153 Covered
/top/f_if/asrt/c_wrAck_f                fifo_sva_a Verilog  SVA  FIFO_ASER.sv(108)
                                                                                   70 Covered
/top/f_if/asrt/C_rst_behave             fifo_sva_a Verilog  SVA  FIFO_ASER.sv(115)
                                                                                    1 Covered

TOTAL DIRECTIVE COVERAGE: 93.33%  COVERS: 15

ASSERTION RESULTS:
-----------------------------------------------------------------------
Name                 File(Line)                   Failure      Pass
                                                  Count        Count
-----------------------------------------------------------------------
/top/f_if/asrt/over_chk_r
                     FIFO_ASER.sv(23)                0          1
/top/f_if/asrt/over_chk_f
                     FIFO_ASER.sv(29)                3          1
/top/f_if/asrt/full_chk_r
                     FIFO_ASER.sv(35)                0          1
/top/f_if/asrt/full_chk_f
                     FIFO_ASER.sv(41)                0          1
/top/f_if/asrt/amostFull_r
                     FIFO_ASER.sv(48)               49          0
/top/f_if/asrt/amostFull_f
                     FIFO_ASER.sv(53)                3          1
/top/f_if/asrt/amostEmpty_r
                     FIFO_ASER.sv(61)                0          1
/top/f_if/asrt/amostEmpty_f
                     FIFO_ASER.sv(67)                0          1
/top/f_if/asrt/empty_chk_r
                     FIFO_ASER.sv(75)                0          1
/top/f_if/asrt/empty_chk_f
                     FIFO_ASER.sv(80)                0          1
/top/f_if/asrt/uner_chk_r
                     FIFO_ASER.sv(87)               37          1
/top/f_if/asrt/uner_chk_f
                     FIFO_ASER.sv(93)                2          1
/top/f_if/asrt/wrAck_chk1_r
                     FIFO_ASER.sv(101)               0          1
/top/f_if/asrt/wrAck_chk1_f
                     FIFO_ASER.sv(107)              20          1
/top/f_if/asrt/rst_behave_chk
                     FIFO_ASER.sv(114)              25          1
/top/tb_if/#anonblk#18214678#26#4#/#ublk#18214678#26/immed__28
                     FIFO_tb.sv(28)                  0          1
/top/tb_if/#anonblk#18214678#42#4#/#ublk#18214678#42/immed__44
                     FIFO_tb.sv(44)                  0          1
/top/tb_if/#anonblk#18214678#57#4#/#ublk#18214678#57/immed__59
                     FIFO_tb.sv(59)                  0          1
/top/tb_if/#anonblk#18214678#77#4#/#ublk#18214678#77/immed__80
                     FIFO_tb.sv(80)                  0          1
/top/print/compar    monitor.sv(8)                 361          1

Total Coverage By Instance (filtered view): 85.41%
```

- # Bug Report

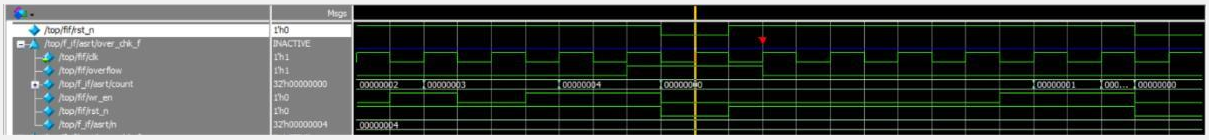
no	Bug	Describtion
1	Almost full _r flag	Rise when more than one write left
2	Overflow _f flag	Still high if reset and write enable achieved
3	Underflow _f flag	Doesn’t fell when reset it detects that read operation happens while reset
4	Underflow _r flag	Rise without posedge clk that detect it not synchronise
5	Write acknowledge	High while reset that detects write operation happens while reset
6	Data out	When Read&&Write [overlapping ones] then FIFO doesn't work well as expected it rotates the value to the Right in addition to rising overflow while write acknowledge fells down to zero.
7	Data out	FIFO works well if and only if its size equals to 2 to the power n where n is size of FIFO as monitoring for discovering the bug from wave the data out in some clk cycles may hold value of Zeros.
8	Data out	It’s value isn’t zero when reset

Waveform

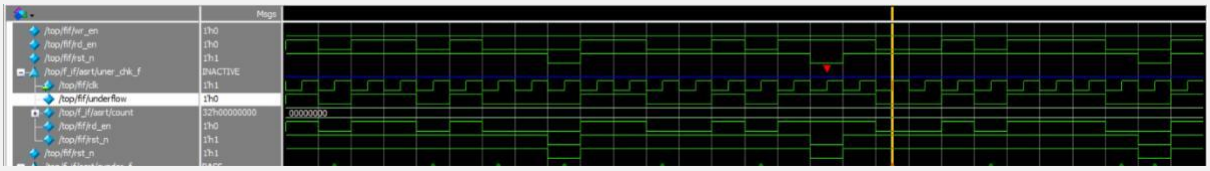
1)



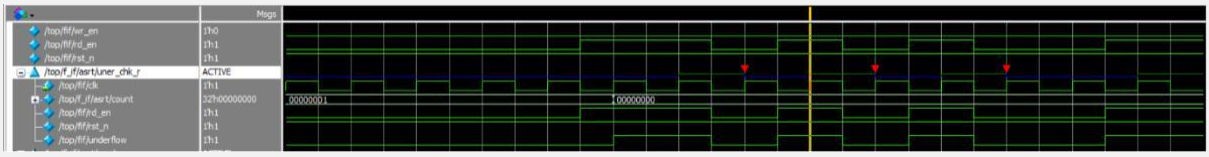
2)



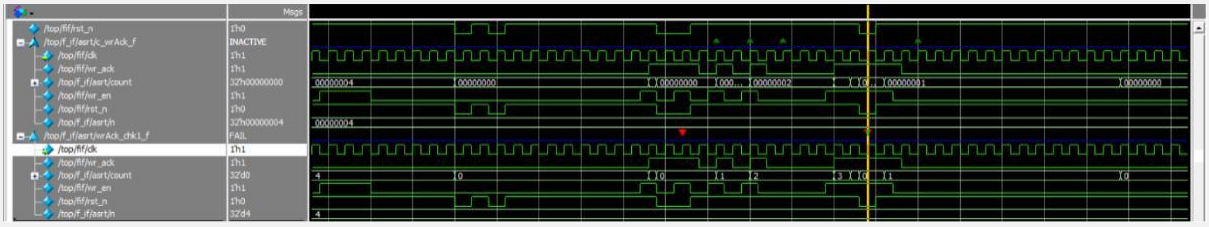
3)



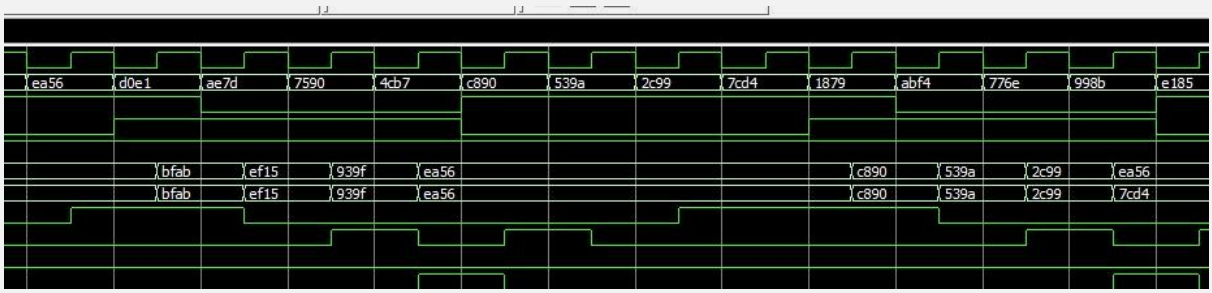
4)



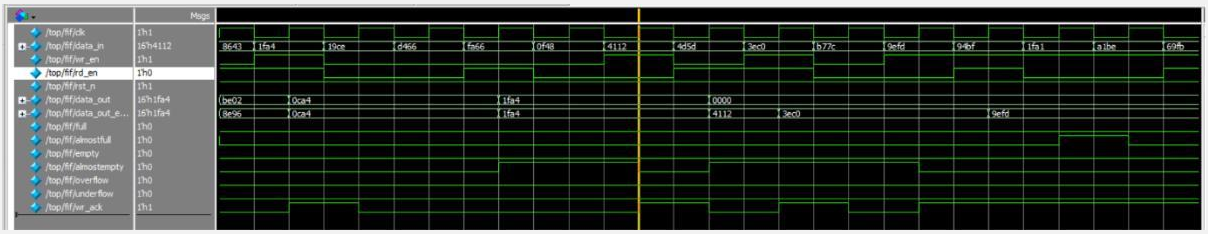
5)



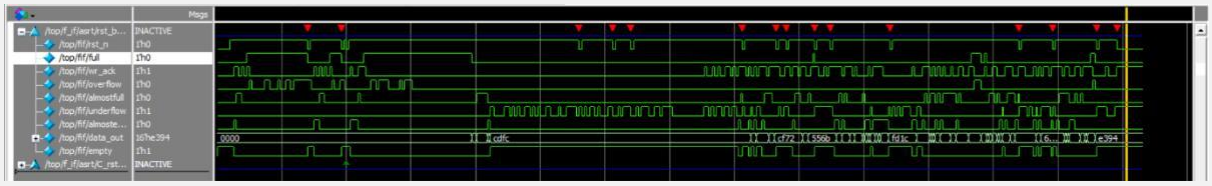
6)


ea56
d0e1
ae7d
7590
4cb7
c890
539a
2c99
7cd4
1879
abf4
776e
998b
e185
bfab
ef15
939f
ea56
c890
539a
2c99
ea56
bfab
ef15
939f
ea56
c890
539a
2c99
7cd4

7)


Msgs
/top/ff/clk
/top/ff/data_in
/top/ff/wr_en
/top/ff/rd_en
/top/ff/rst_n
/top/ff/data_out
/top/ff/data_out_e...
/top/ff/full
/top/ff/almostfull
/top/ff/empty
/top/ff/almostempty
/top/ff/overflow
/top/ff/underflow
/top/ff/wr_ack

8)


Msgs
/top/ff/full
/top/ff/wr_ack
/top/ff/overflow
/top/ff/almostfull
/top/ff/underflow
/top/ff/data_out
/top/ff/empty

- # Assertions

Feature	Assertion
Writing while queue is full, overflow =1.	@(posedge asrt.clk) (count==n && asrt.wr_en&& !asrt.rd_en && asrt.rst_n) |=> asrt.overflow ;
overflow =1 & queue size has decreased, overflow = 0 .	@(posedge asrt.clk) ($past(asrt.overflow) && (count < n\|\|$past(!asrt.wr_en)||!asrt.rst_n)) |-> !(asrt.overflow) ;
queue size is full, full=1.	@(posedge asrt.clk) (count==n) |-> (asrt.full) ;
full=1 & queue size has decresed , full =0.	@(posedge asrt.clk) ($past(asrt.full) && count < n) |-> !(asrt.full) ;
queue has only one free location, almostfull=1.	@(posedge asrt.clk) (count==n-1)|-> (asrt.almostfull);
almostfull=1 & queue size has decreased, almostfull = 0 .	@(posedge asrt.clk) ($past(asrt.almostfull)&&(count!=n-1))|-> !(asrt.almostfull);
queue has only one occupied loaction, almostempty=1.	@(posedge asrt.clk) (count==1)|->(asrt.almostempty);
almostempty=1 & queue size has decreased, almostempty = 0 .	@(posedge asrt.clk) ($past(asrt.almostempty)&&count!=1)|->!(asrt.almostempty);
queue is empty,empty=1.	@(posedge asrt.clk) (count==0) |-> (asrt.empty) ;
empty =1 & queue has occupied location , empty=0.	@(posedge asrt.clk) ($past(asrt.empty)&&count!=0) |-> !(asrt.empty) ;
queue is empty & read operation , underflow =1.	@(posedge asrt.clk) (count==0 && asrt.rd_en && asrt.rst_n)|=> (asrt.underflow) ;
underflow =1 & queue has occupied location & no read operation , underflow =0.	@(posedge asrt.clk) ($past(asrt.underflow)&&(count!=0 \|\| !asrt.rd_en\|\|$fell(asrt.rst_n)))|-> !(asrt.underflow) ;
queue has free locations & write operation , wr_ack =1.	@(posedge asrt.clk) (asrt.wr_en &&count!=n && asrt.rst_n) |=> (asrt.wr_ack);
(wr_ack=1 & queue is full)| (wr_en =0&rst_n =0) , wr_ack =0.	@(posedge asrt.clk) ($past(asrt.wr_ack) &&(count==n \|\|$past(!asrt.wr_en)||!asrt.rst_n)) |-> !(asrt.wr_ack);