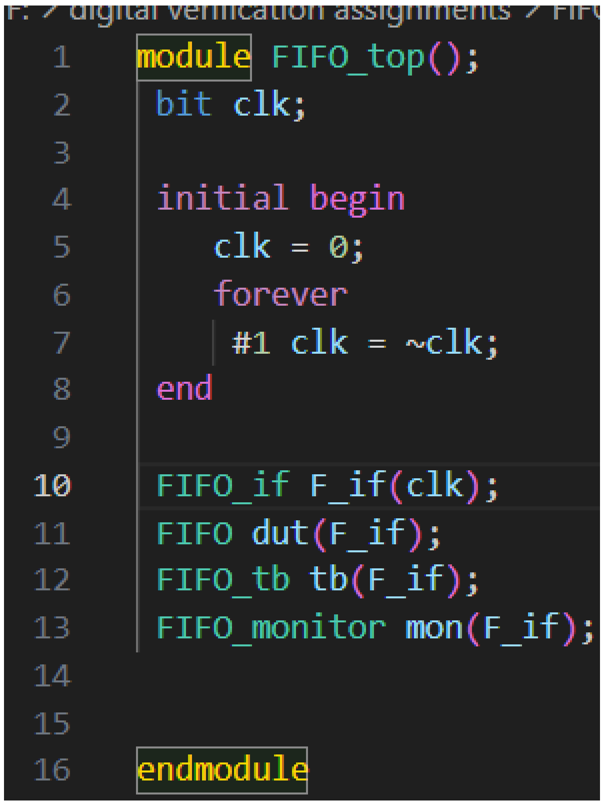
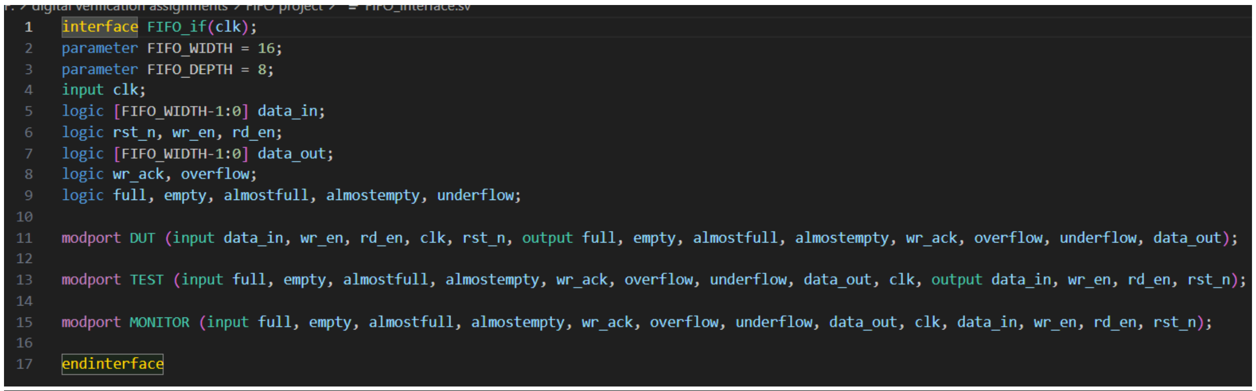
Project 1

Top code:



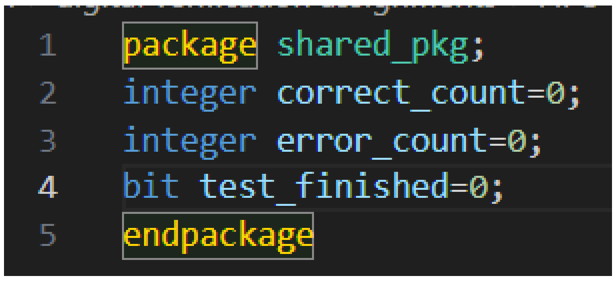
```
module FIFO_top();
 bit clk;

 initial begin
    clk = 0;
    forever
     #1 clk = ~clk;
 end

 FIFO_if F_if(clk);
 FIFO dut(F_if);
 FIFO_tb tb(F_if);
 FIFO_monitor mon(F_if);


endmodule
```

Interface code:



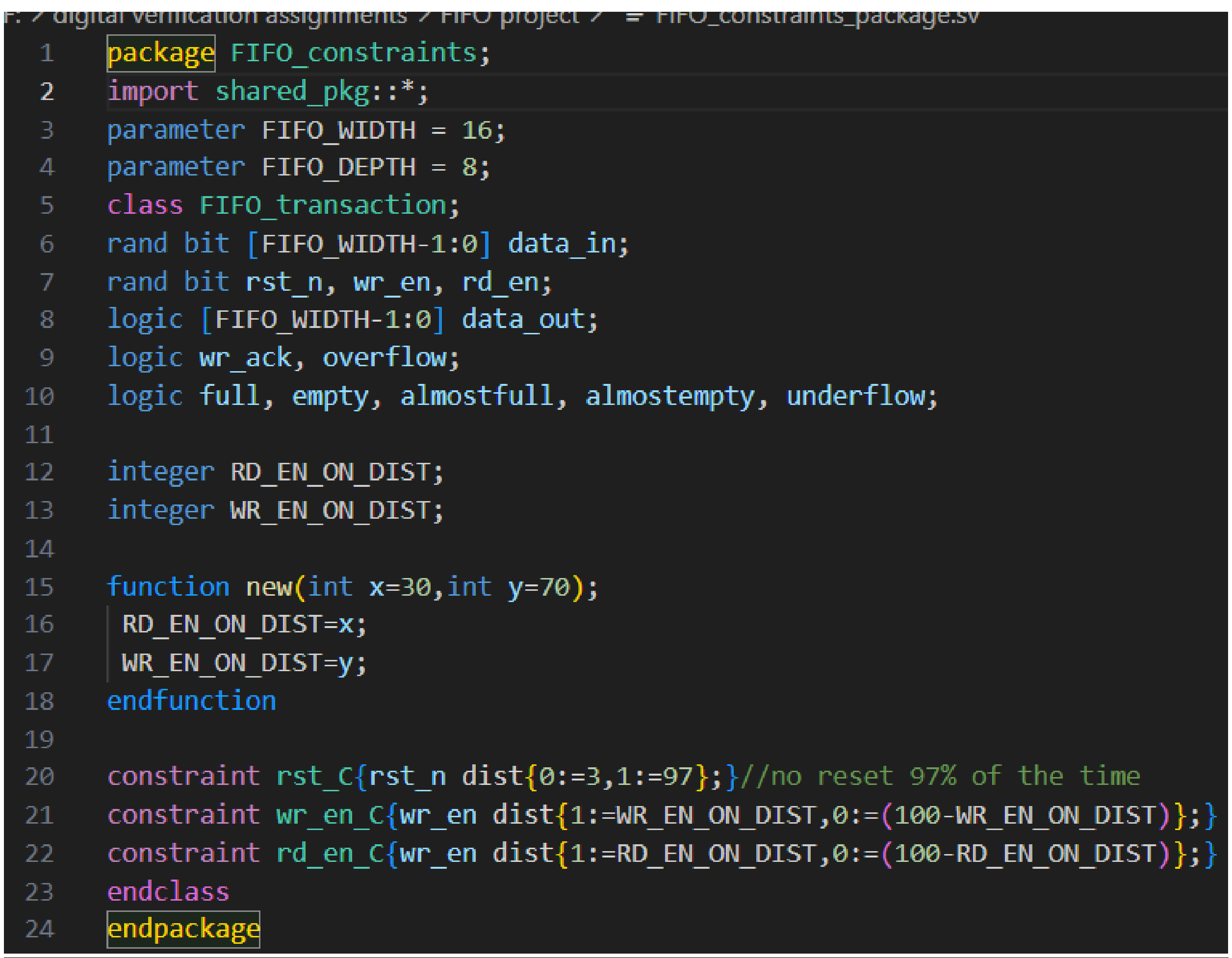
```
interface FIFO_if(clk);
parameter FIFO_WIDTH = 16;
parameter FIFO_DEPTH = 8;
input clk;
logic [FIFO_WIDTH-1:0] data_in;
logic rst_n, wr_en, rd_en;
logic [FIFO_WIDTH-1:0] data_out;
logic wr_ack, overflow;
logic full, empty, almostfull, almostempty, underflow;

modport DUT (input data_in, wr_en, rd_en, clk, rst_n, output full, empty, almostfull, almostempty, wr_ack, overflow, underflow, data_out);

modport TEST (input full, empty, almostfull, almostempty, wr_ack, overflow, underflow, data_out, clk, output data_in, wr_en, rd_en, rst_n);

modport MONITOR (input full, empty, almostfull, almostempty, wr_ack, overflow, underflow, data_out, clk, data_in, wr_en, rd_en, rst_n);

endinterface
```

Shared Package code:



```
package shared_pkg;
integer correct_count=0;
integer error_count=0;
bit test_finished=0;
endpackage
```

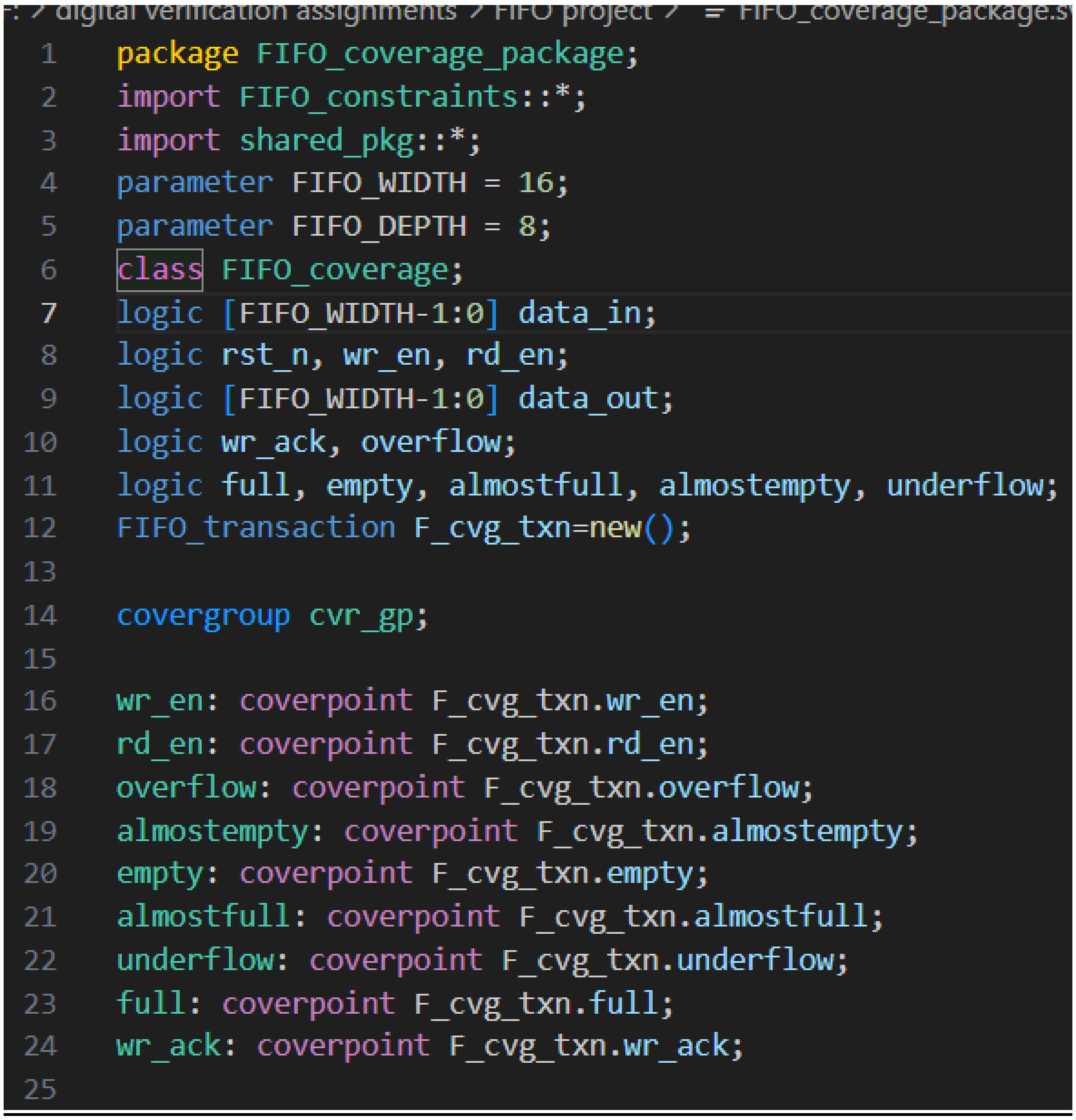
FIFO transaction Package code:



F: > digital verification assignments > FIFO project > FIFO_constraints_package.sv

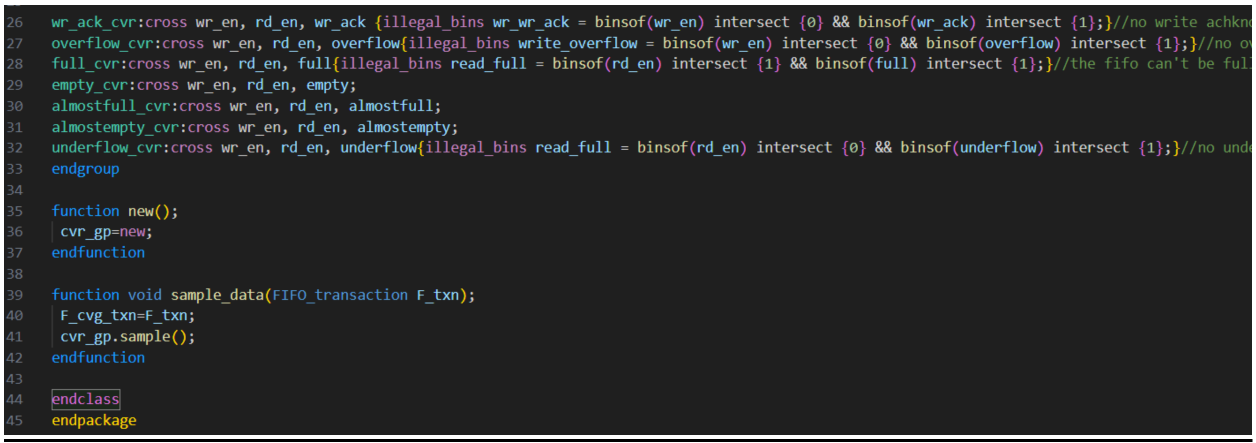
```
package FIFO_constraints;
import shared_pkg::*;
parameter FIFO_WIDTH = 16;
parameter FIFO_DEPTH = 8;
class FIFO_transaction;
rand bit [FIFO_WIDTH-1:0] data_in;
rand bit rst_n, wr_en, rd_en;
logic [FIFO_WIDTH-1:0] data_out;
logic wr_ack, overflow;
logic full, empty, almostfull, almostempty, underflow;

integer RD_EN_ON_DIST;
integer WR_EN_ON_DIST;

function new(int x=30,int y=70);
  RD_EN_ON_DIST=x;
  WR_EN_ON_DIST=y;
endfunction

constraint rst_C{rst_n dist{0:=3,1:=97};}//no reset 97% of the time
constraint wr_en_C{wr_en dist{1:=WR_EN_ON_DIST,0:=(100-WR_EN_ON_DIST)};}
constraint rd_en_C{wr_en dist{1:=RD_EN_ON_DIST,0:=(100-RD_EN_ON_DIST)};}
endclass
endpackage
```

FIFO coverage Package code:



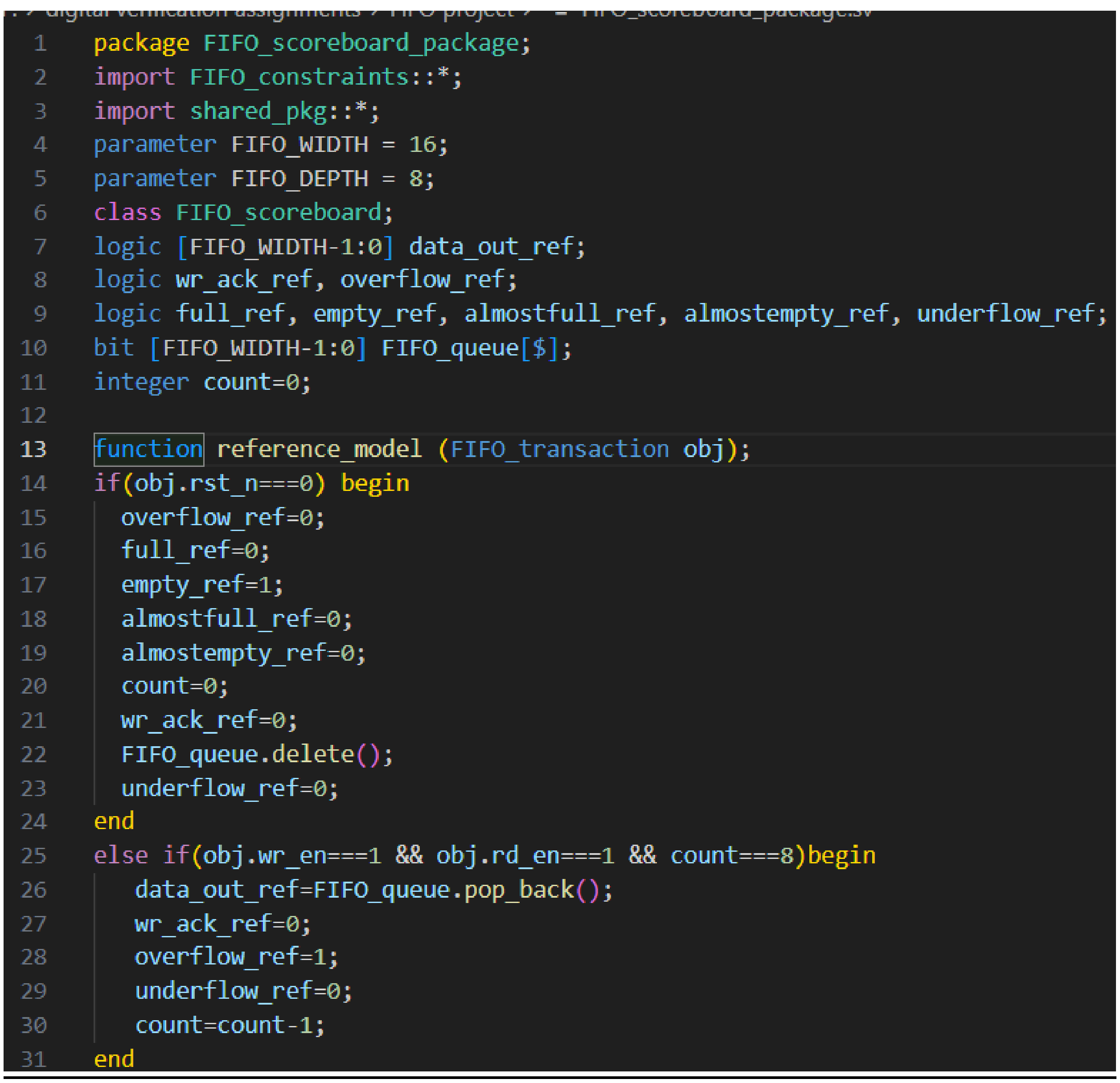
F: > digital verification assignments > FIFO project > ≡ FIFO_coverage_package.s

```
package FIFO_coverage_package;
import FIFO_constraints::*;
import shared_pkg::*;
parameter FIFO_WIDTH = 16;
parameter FIFO_DEPTH = 8;
class FIFO_coverage;
logic [FIFO_WIDTH-1:0] data_in;
logic rst_n, wr_en, rd_en;
logic [FIFO_WIDTH-1:0] data_out;
logic wr_ack, overflow;
logic full, empty, almostfull, almostempty, underflow;
FIFO_transaction F_cvg_txn=new();

covergroup cvr_gp;

wr_en: coverpoint F_cvg_txn.wr_en;
rd_en: coverpoint F_cvg_txn.rd_en;
overflow: coverpoint F_cvg_txn.overflow;
almostempty: coverpoint F_cvg_txn.almostempty;
empty: coverpoint F_cvg_txn.empty;
almostfull: coverpoint F_cvg_txn.almostfull;
underflow: coverpoint F_cvg_txn.underflow;
full: coverpoint F_cvg_txn.full;
wr_ack: coverpoint F_cvg_txn.wr_ack;

```

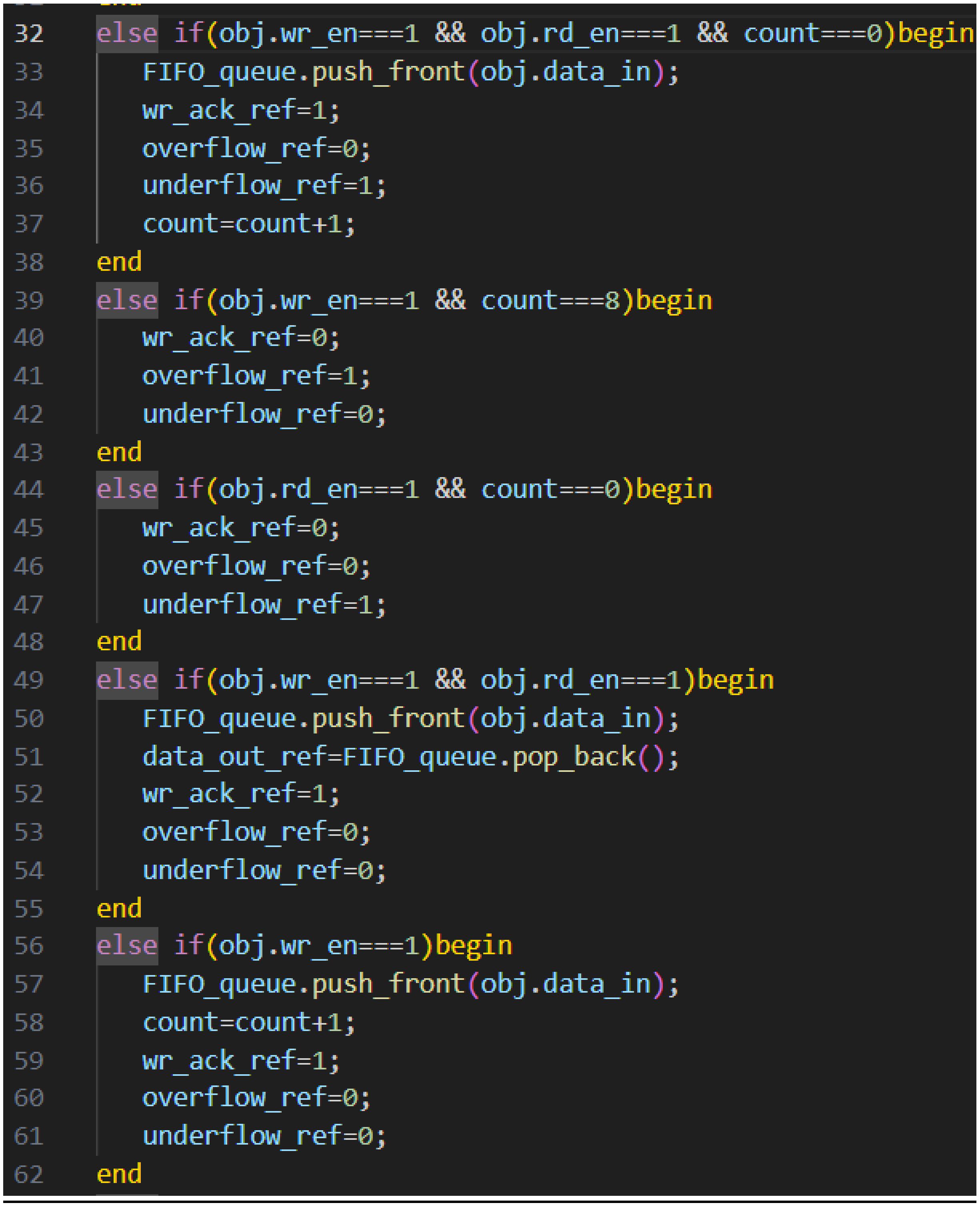


```
wr_ack_cvr:cross wr_en, rd_en, wr_ack {illegal_bins wr_wr_ack = binsof(wr_en) intersect {0} && binsof(wr_ack) intersect {1};}//no write achkn
overflow_cvr:cross wr_en, rd_en, overflow{illegal_bins write_overflow = binsof(wr_en) intersect {0} && binsof(overflow) intersect {1};}//no o
full_cvr:cross wr_en, rd_en, full{illegal_bins read_full = binsof(rd_en) intersect {1} && binsof(full) intersect {1};}//the fifo can't be ful
empty_cvr:cross wr_en, rd_en, empty;
almostfull_cvr:cross wr_en, rd_en, almostfull;
almostempty_cvr:cross wr_en, rd_en, almostempty;
underflow_cvr:cross wr_en, rd_en, underflow{illegal_bins read_full = binsof(rd_en) intersect {0} && binsof(underflow) intersect {1};}//no unde
endgroup

function new();
 cvr_gp=new;
endfunction

function void sample_data(FIFO_transaction F_txn);
 F_cvg_txn=F_txn;
 cvr_gp.sample();
endfunction

endclass
endpackage
```

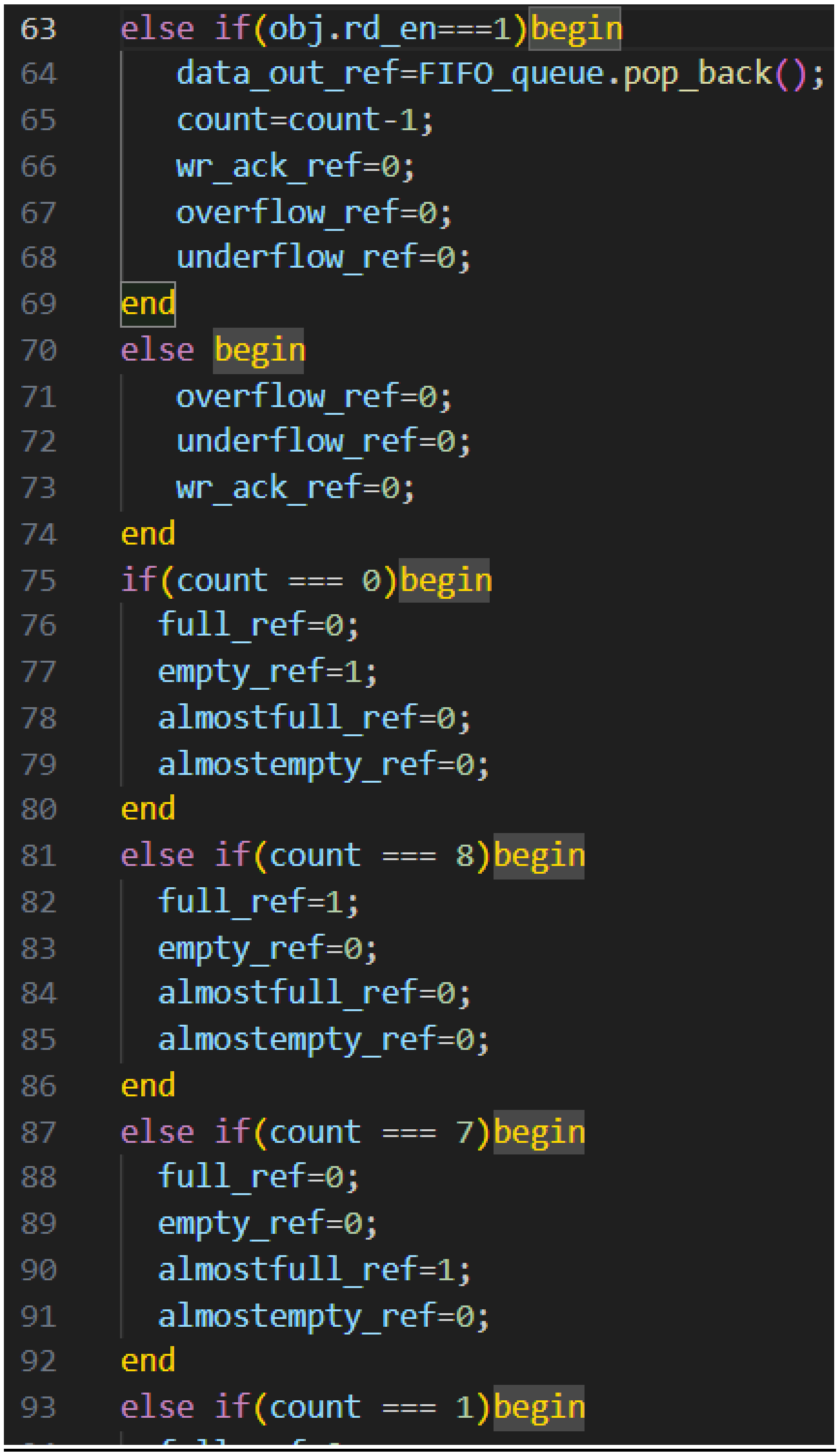
FIFO scoreboard Package code:



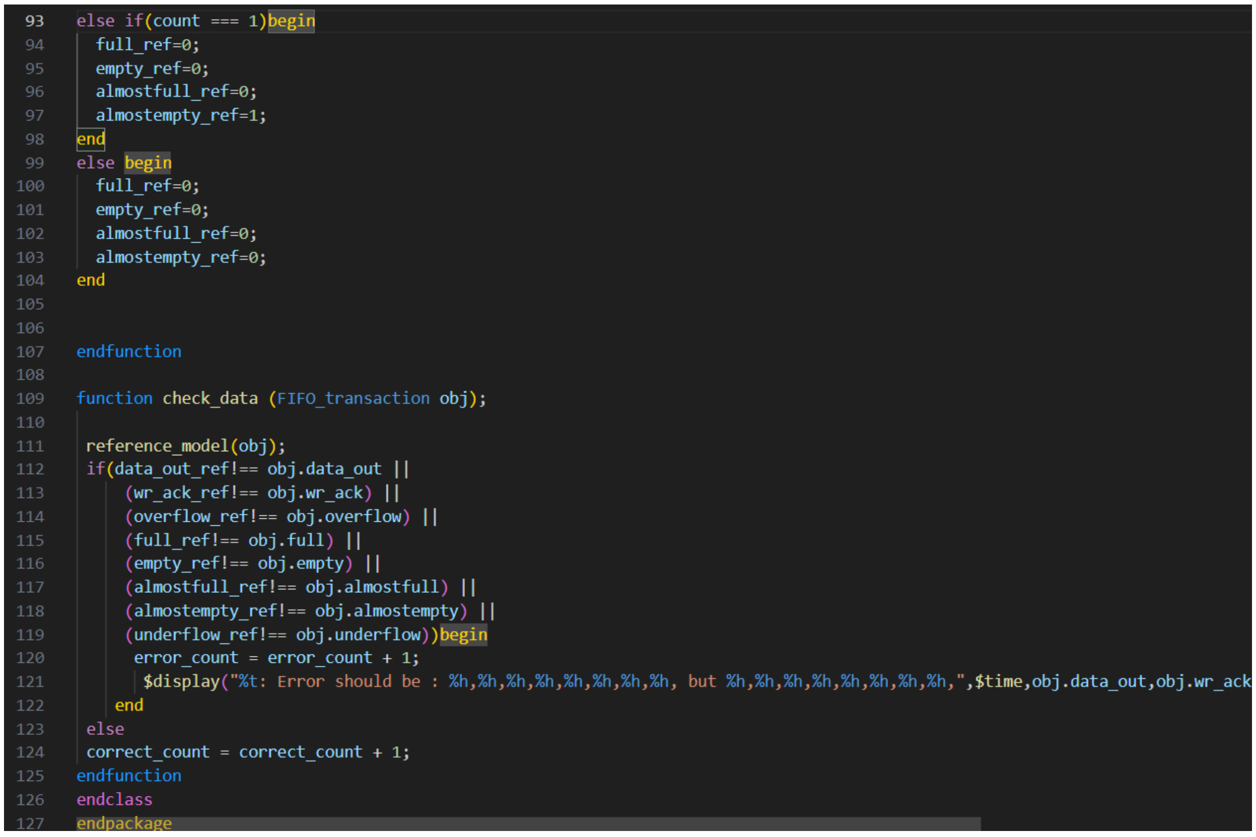
```
package FIFO_scoreboard_package;
import FIFO_constraints::*;
import shared_pkg::*;
parameter FIFO_WIDTH = 16;
parameter FIFO_DEPTH = 8;
class FIFO_scoreboard;
logic [FIFO_WIDTH-1:0] data_out_ref;
logic wr_ack_ref, overflow_ref;
logic full_ref, empty_ref, almostfull_ref, almostempty_ref, underflow_ref;
bit [FIFO_WIDTH-1:0] FIFO_queue[$];
integer count=0;

function reference_model (FIFO_transaction obj);
if(obj.rst_n===0) begin
  overflow_ref=0;
  full_ref=0;
  empty_ref=1;
  almostfull_ref=0;
  almostempty_ref=0;
  count=0;
  wr_ack_ref=0;
  FIFO_queue.delete();
  underflow_ref=0;
end
else if(obj.wr_en===1 && obj.rd_en===1 && count===8)begin
   data_out_ref=FIFO_queue.pop_back();
   wr_ack_ref=0;
   overflow_ref=1;
   underflow_ref=0;
   count=count-1;
end
```



```
else if(obj.wr_en===1 && obj.rd_en===1 && count===0)begin
   FIFO_queue.push_front(obj.data_in);
   wr_ack_ref=1;
   overflow_ref=0;
   underflow_ref=1;
   count=count+1;
end
else if(obj.wr_en===1 && count===8)begin
   wr_ack_ref=0;
   overflow_ref=1;
   underflow_ref=0;
end
else if(obj.rd_en===1 && count===0)begin
   wr_ack_ref=0;
   overflow_ref=0;
   underflow_ref=1;
end
else if(obj.wr_en===1 && obj.rd_en===1)begin
   FIFO_queue.push_front(obj.data_in);
   data_out_ref=FIFO_queue.pop_back();
   wr_ack_ref=1;
   overflow_ref=0;
   underflow_ref=0;
end
else if(obj.wr_en===1)begin
   FIFO_queue.push_front(obj.data_in);
   count=count+1;
   wr_ack_ref=1;
   overflow_ref=0;
   underflow_ref=0;
end
```

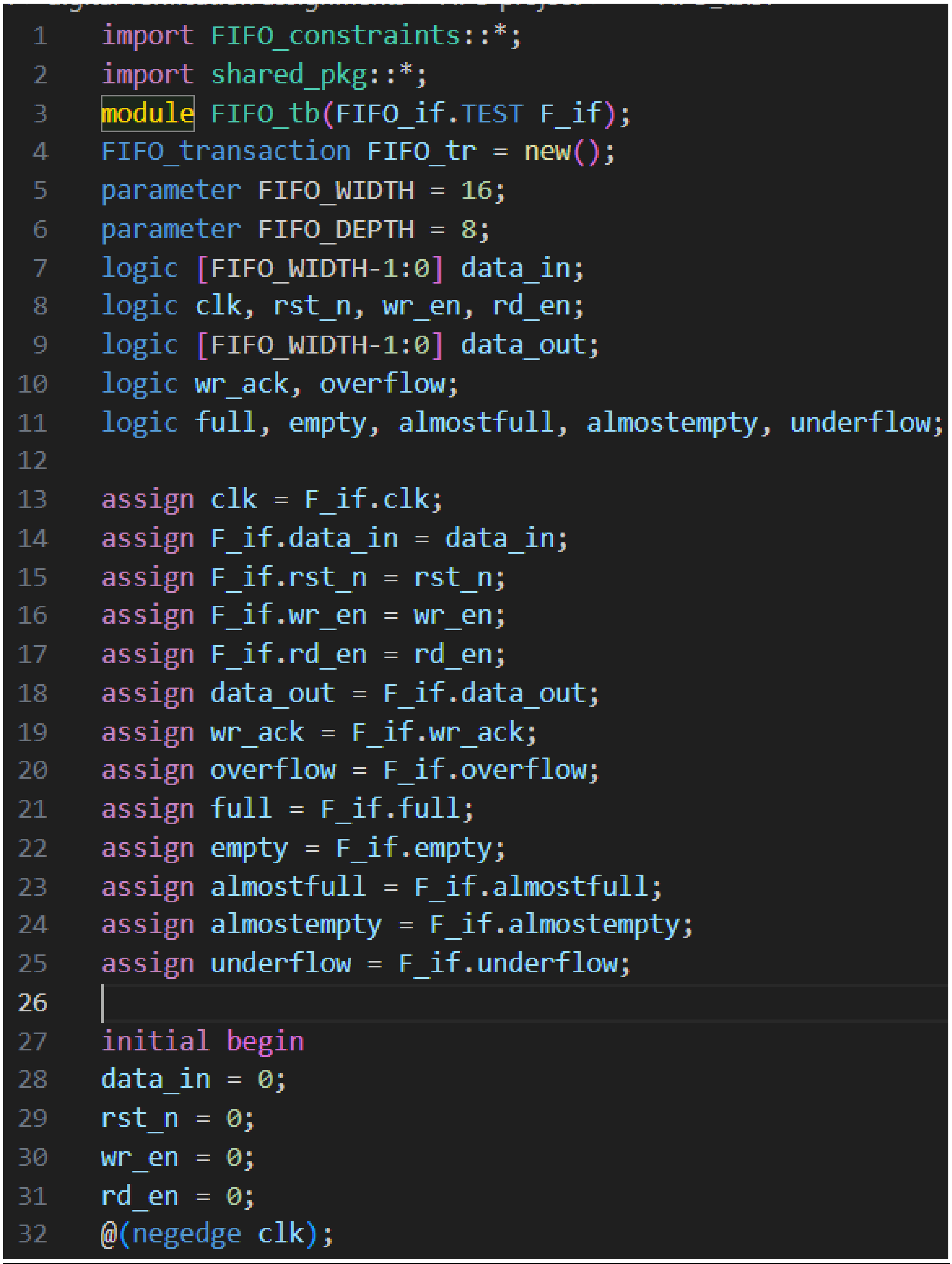


```
else if(obj.rd_en===1)begin
   data_out_ref=FIFO_queue.pop_back();
   count=count-1;
   wr_ack_ref=0;
   overflow_ref=0;
   underflow_ref=0;
end
else begin
   overflow_ref=0;
   underflow_ref=0;
   wr_ack_ref=0;
end
if(count === 0)begin
  full_ref=0;
  empty_ref=1;
  almostfull_ref=0;
  almostempty_ref=0;
end
else if(count === 8)begin
  full_ref=1;
  empty_ref=0;
  almostfull_ref=0;
  almostempty_ref=0;
end
else if(count === 7)begin
  full_ref=0;
  empty_ref=0;
  almostfull_ref=1;
  almostempty_ref=0;
end
else if(count === 1)begin
```

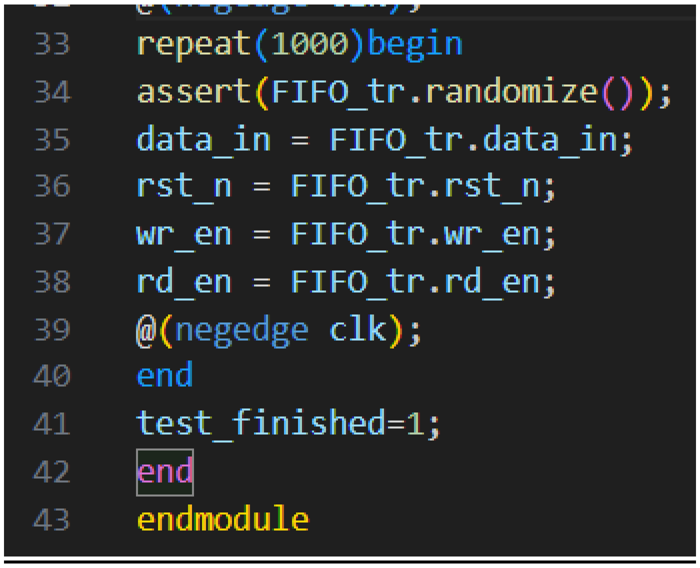


```
else if(count === 1)begin
  full_ref=0;
  empty_ref=0;
  almostfull_ref=0;
  almostempty_ref=1;
end
else begin
  full_ref=0;
  empty_ref=0;
  almostfull_ref=0;
  almostempty_ref=0;
end


endfunction

function check_data (FIFO_transaction obj);

 reference_model(obj);
 if(data_out_ref!== obj.data_out ||
     (wr_ack_ref!== obj.wr_ack) ||
     (overflow_ref!== obj.overflow) ||
     (full_ref!== obj.full) ||
     (empty_ref!== obj.empty) ||
     (almostfull_ref!== obj.almostfull) ||
     (almostempty_ref!== obj.almostempty) ||
     (underflow_ref!== obj.underflow))begin
      error_count = error_count + 1;
       $display("%t: Error should be : %h,%h,%h,%h,%h,%h,%h,%h, but %h,%h,%h,%h,%h,%h,%h,%h,",$time,obj.data_out,obj.wr_ack
     end
 else
 correct_count = correct_count + 1;
endfunction
endclass
endpackage
```

Testbench code:

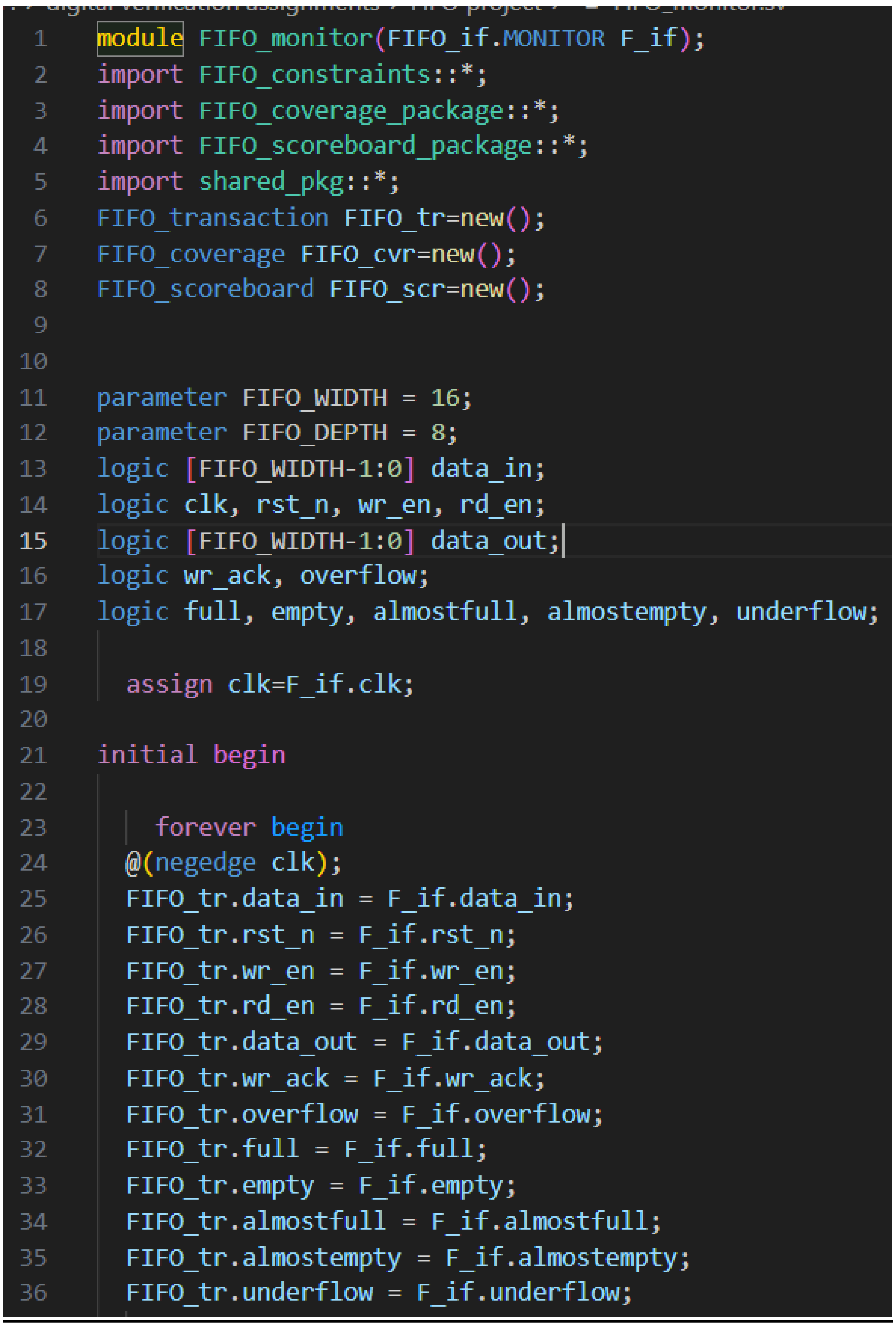


```
import FIFO_constraints::*;
import shared_pkg::*;
module FIFO_tb(FIFO_if.TEST F_if);
FIFO_transaction FIFO_tr = new();
parameter FIFO_WIDTH = 16;
parameter FIFO_DEPTH = 8;
logic [FIFO_WIDTH-1:0] data_in;
logic clk, rst_n, wr_en, rd_en;
logic [FIFO_WIDTH-1:0] data_out;
logic wr_ack, overflow;
logic full, empty, almostfull, almostempty, underflow;

assign clk = F_if.clk;
assign F_if.data_in = data_in;
assign F_if.rst_n = rst_n;
assign F_if.wr_en = wr_en;
assign F_if.rd_en = rd_en;
assign data_out = F_if.data_out;
assign wr_ack = F_if.wr_ack;
assign overflow = F_if.overflow;
assign full = F_if.full;
assign empty = F_if.empty;
assign almostfull = F_if.almostfull;
assign almostempty = F_if.almostempty;
assign underflow = F_if.underflow;

initial begin
data_in = 0;
rst_n = 0;
wr_en = 0;
rd_en = 0;
@(negedge clk);
```

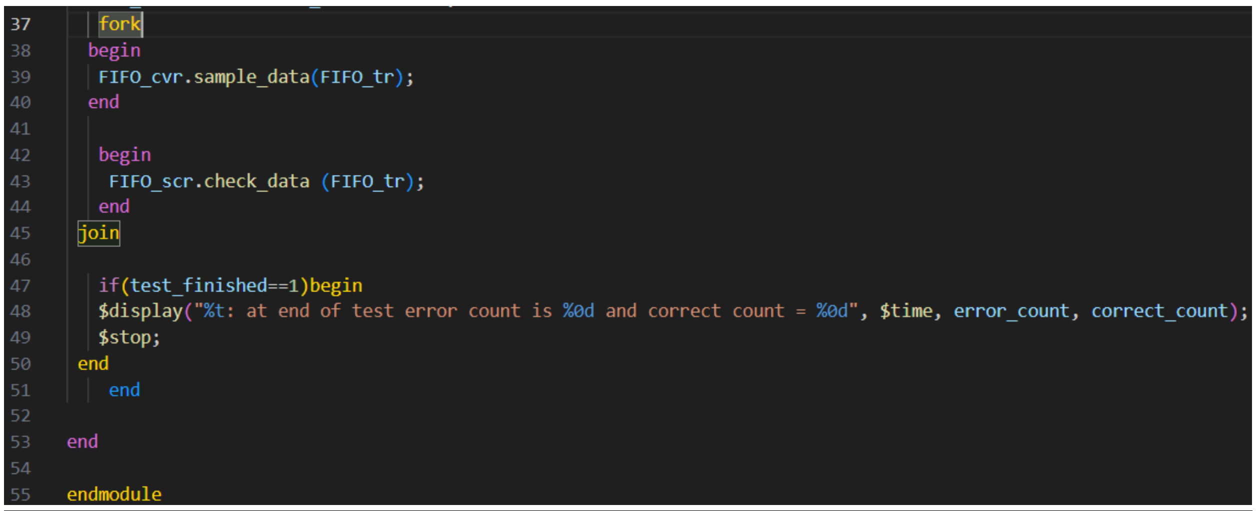


```
repeat(1000)begin
assert(FIFO_tr.randomize());
data_in = FIFO_tr.data_in;
rst_n = FIFO_tr.rst_n;
wr_en = FIFO_tr.wr_en;
rd_en = FIFO_tr.rd_en;
@(negedge clk);
end
test_finished=1;
end
endmodule
```

Monitor code:

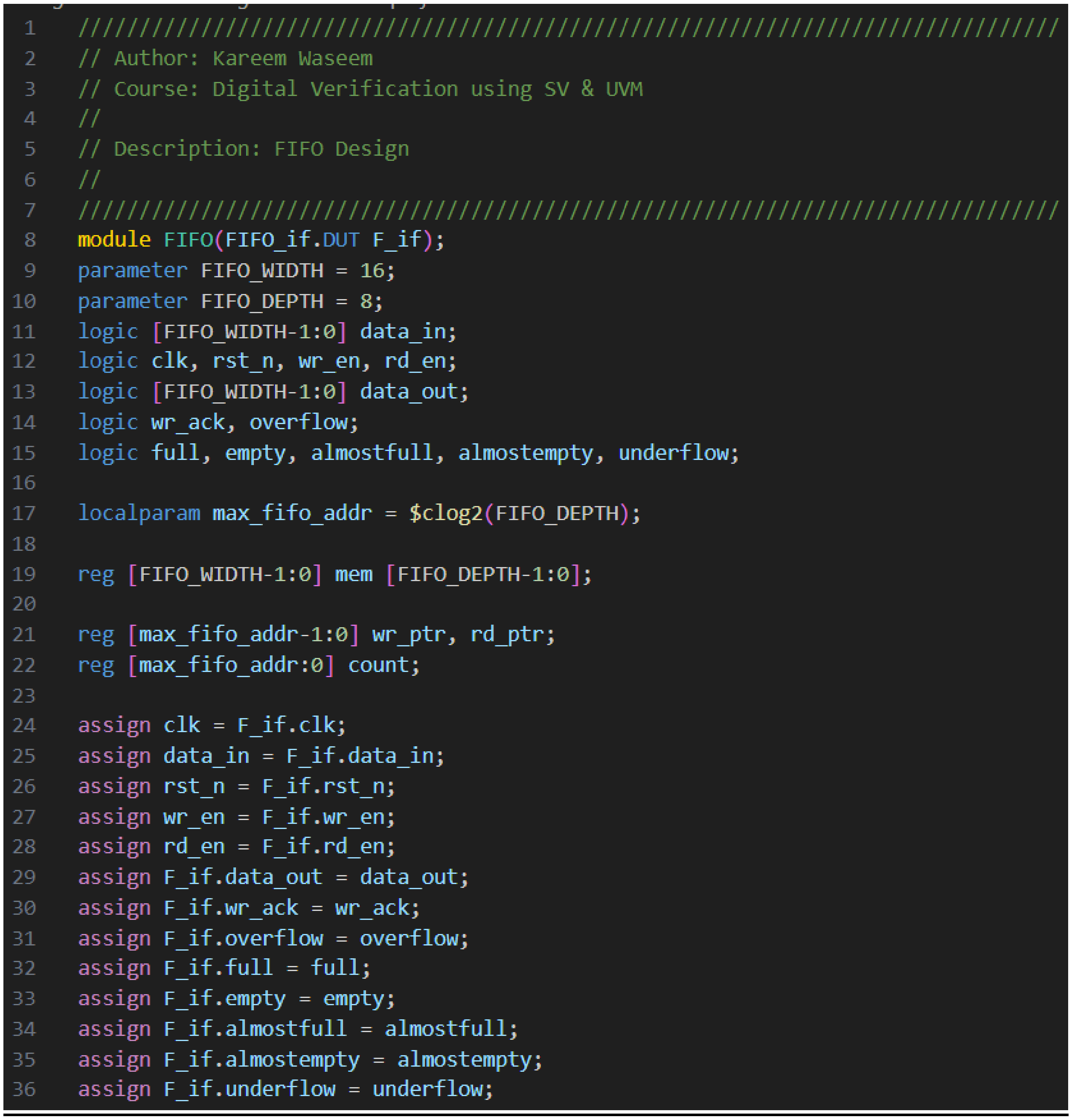


```
module FIFO_monitor(FIFO_if.MONITOR F_if);
import FIFO_constraints::*;
import FIFO_coverage_package::*;
import FIFO_scoreboard_package::*;
import shared_pkg::*;
FIFO_transaction FIFO_tr=new();
FIFO_coverage FIFO_cvr=new();
FIFO_scoreboard FIFO_scr=new();


parameter FIFO_WIDTH = 16;
parameter FIFO_DEPTH = 8;
logic [FIFO_WIDTH-1:0] data_in;
logic clk, rst_n, wr_en, rd_en;
logic [FIFO_WIDTH-1:0] data_out;
logic wr_ack, overflow;
logic full, empty, almostfull, almostempty, underflow;

  assign clk=F_if.clk;

initial begin

    forever begin
  @(negedge clk);
  FIFO_tr.data_in = F_if.data_in;
  FIFO_tr.rst_n = F_if.rst_n;
  FIFO_tr.wr_en = F_if.wr_en;
  FIFO_tr.rd_en = F_if.rd_en;
  FIFO_tr.data_out = F_if.data_out;
  FIFO_tr.wr_ack = F_if.wr_ack;
  FIFO_tr.overflow = F_if.overflow;
  FIFO_tr.full = F_if.full;
  FIFO_tr.empty = F_if.empty;
  FIFO_tr.almostfull = F_if.almostfull;
  FIFO_tr.almostempty = F_if.almostempty;
  FIFO_tr.underflow = F_if.underflow;
```

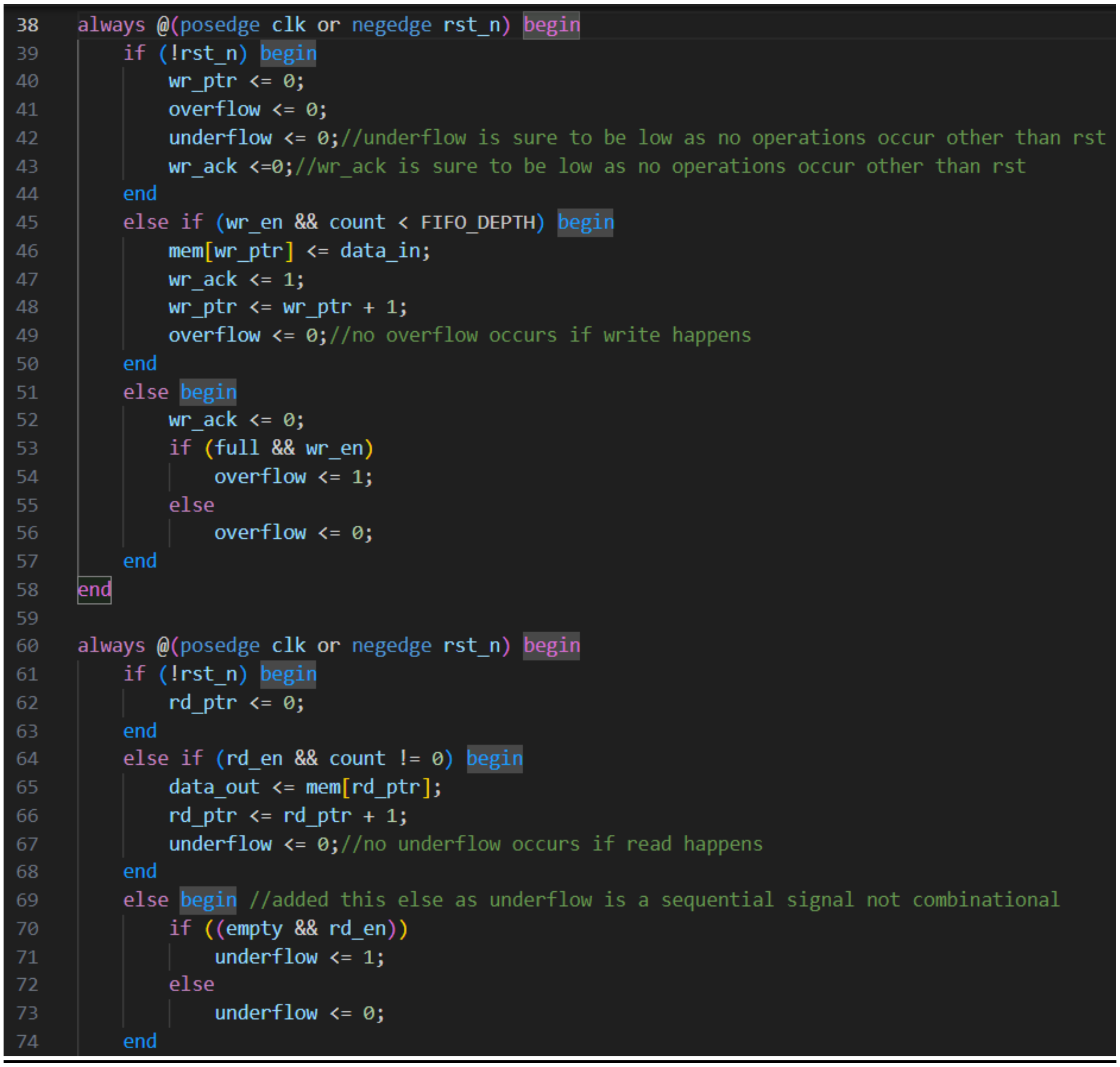


```
    fork
   begin
    FIFO_cvr.sample_data(FIFO_tr);
   end

    begin
     FIFO_scr.check_data (FIFO_tr);
    end
  join

    if(test_finished==1)begin
    $display("%t: at end of test error count is %0d and correct count = %0d", $time, error_count, correct_count);
    $stop;
  end
     end

end

endmodule
```

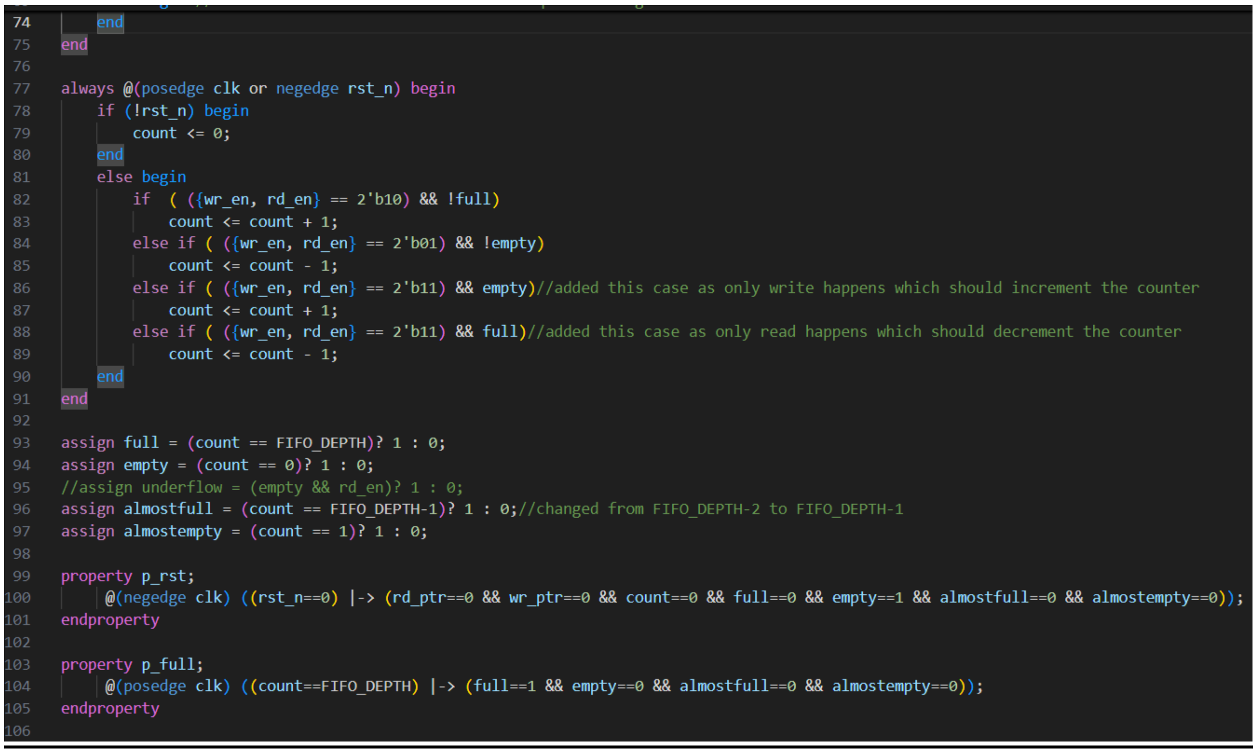
Design code:



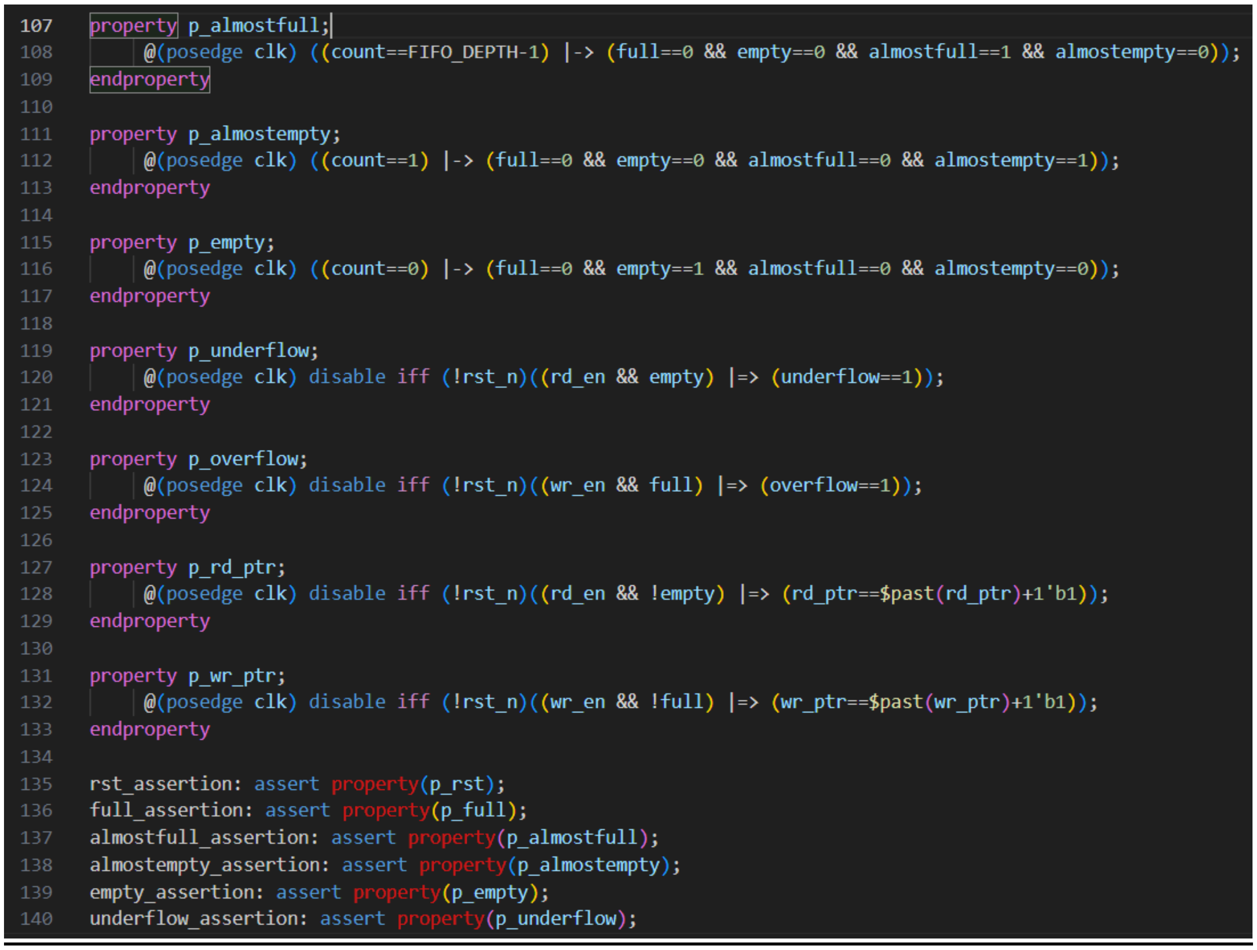
```
////////////////////////////////////////////////////////////////////////////////
// Author: Kareem Waseem
// Course: Digital Verification using SV & UVM
//
// Description: FIFO Design
//
////////////////////////////////////////////////////////////////////////////////
module FIFO(FIFO_if.DUT F_if);
parameter FIFO_WIDTH = 16;
parameter FIFO_DEPTH = 8;
logic [FIFO_WIDTH-1:0] data_in;
logic clk, rst_n, wr_en, rd_en;
logic [FIFO_WIDTH-1:0] data_out;
logic wr_ack, overflow;
logic full, empty, almostfull, almostempty, underflow;

localparam max_fifo_addr = $clog2(FIFO_DEPTH);

reg [FIFO_WIDTH-1:0] mem [FIFO_DEPTH-1:0];

reg [max_fifo_addr-1:0] wr_ptr, rd_ptr;
reg [max_fifo_addr:0] count;

assign clk = F_if.clk;
assign data_in = F_if.data_in;
assign rst_n = F_if.rst_n;
assign wr_en = F_if.wr_en;
assign rd_en = F_if.rd_en;
assign F_if.data_out = data_out;
assign F_if.wr_ack = wr_ack;
assign F_if.overflow = overflow;
assign F_if.full = full;
assign F_if.empty = empty;
assign F_if.almostfull = almostfull;
assign F_if.almostempty = almostempty;
assign F_if.underflow = underflow;
```



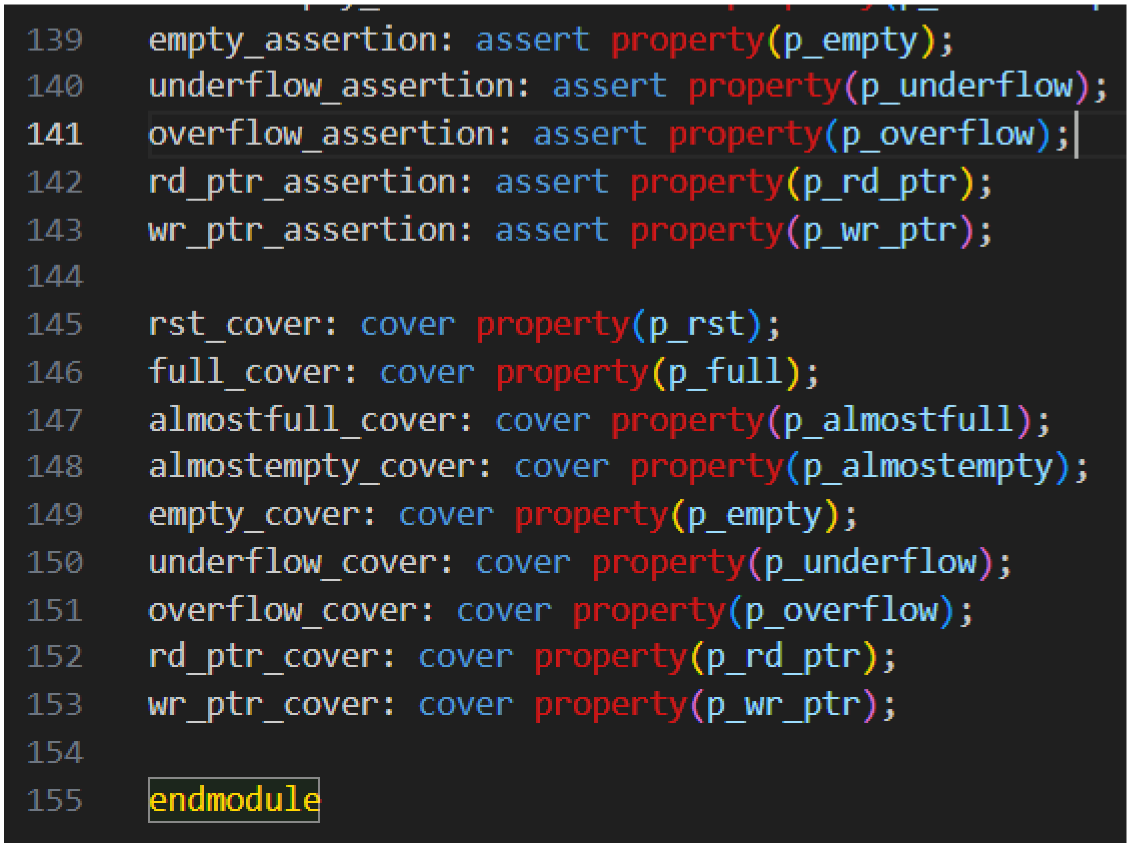
```
always @(posedge clk or negedge rst_n) begin
    if (!rst_n) begin
        wr_ptr <= 0;
        overflow <= 0;
        underflow <= 0;//underflow is sure to be low as no operations occur other than rst
        wr_ack <=0;//wr_ack is sure to be low as no operations occur other than rst
    end
    else if (wr_en && count < FIFO_DEPTH) begin
        mem[wr_ptr] <= data_in;
        wr_ack <= 1;
        wr_ptr <= wr_ptr + 1;
        overflow <= 0;//no overflow occurs if write happens
    end
    else begin
        wr_ack <= 0;
        if (full && wr_en)
            overflow <= 1;
        else
            overflow <= 0;
    end
end

always @(posedge clk or negedge rst_n) begin
    if (!rst_n) begin
        rd_ptr <= 0;
    end
    else if (rd_en && count != 0) begin
        data_out <= mem[rd_ptr];
        rd_ptr <= rd_ptr + 1;
        underflow <= 0;//no underflow occurs if read happens
    end
    else begin //added this else as underflow is a sequential signal not combinational
        if ((empty && rd_en))
            underflow <= 1;
        else
            underflow <= 0;
    end
```



```
    end
end

always @(posedge clk or negedge rst_n) begin
    if (!rst_n) begin
        count <= 0;
    end
    else begin
        if  ( ({wr_en, rd_en} == 2'b10) && !full)
            count <= count + 1;
        else if ( ({wr_en, rd_en} == 2'b01) && !empty)
            count <= count - 1;
        else if ( ({wr_en, rd_en} == 2'b11) && empty)//added this case as only write happens which should increment the counter
            count <= count + 1;
        else if ( ({wr_en, rd_en} == 2'b11) && full)//added this case as only read happens which should decrement the counter
            count <= count - 1;
    end
end

assign full = (count == FIFO_DEPTH)? 1 : 0;
assign empty = (count == 0)? 1 : 0;
//assign underflow = (empty && rd_en)? 1 : 0;
assign almostfull = (count == FIFO_DEPTH-1)? 1 : 0;//changed from FIFO_DEPTH-2 to FIFO_DEPTH-1
assign almostempty = (count == 1)? 1 : 0;

property p_rst;
    @(negedge clk) ((rst_n==0) |-> (rd_ptr==0 && wr_ptr==0 && count==0 && full==0 && empty==1 && almostfull==0 && almostempty==0));
endproperty

property p_full;
    @(posedge clk) ((count==FIFO_DEPTH) |-> (full==1 && empty==0 && almostfull==0 && almostempty==0));
endproperty

```



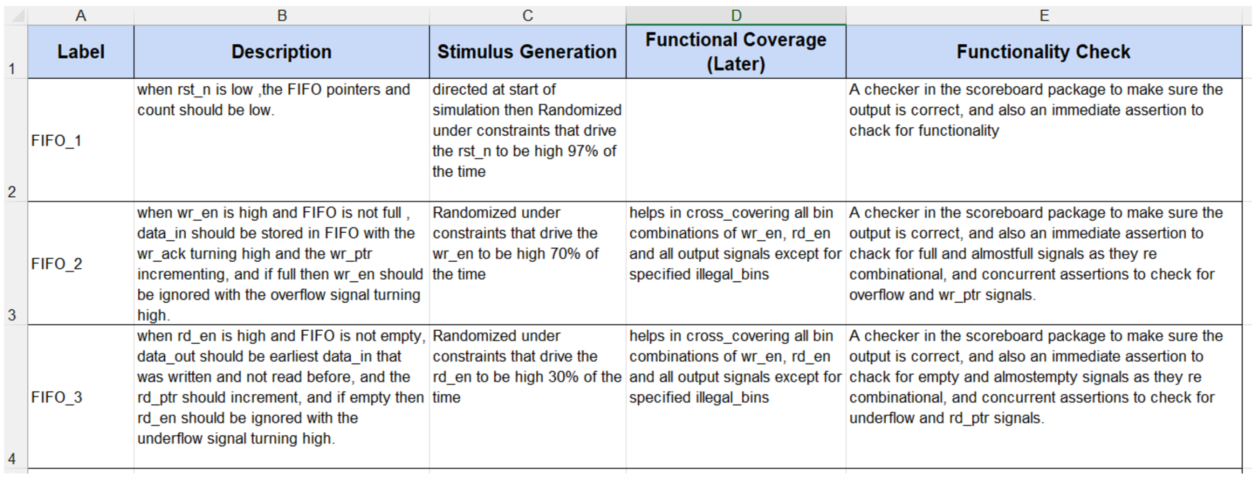
```
property p_almostfull;
    @(posedge clk) ((count==FIFO_DEPTH-1) |-> (full==0 && empty==0 && almostfull==1 && almostempty==0));
endproperty

property p_almostempty;
    @(posedge clk) ((count==1) |-> (full==0 && empty==0 && almostfull==0 && almostempty==1));
endproperty

property p_empty;
    @(posedge clk) ((count==0) |-> (full==0 && empty==1 && almostfull==0 && almostempty==0));
endproperty

property p_underflow;
    @(posedge clk) disable iff (!rst_n)((rd_en && empty) |=> (underflow==1));
endproperty

property p_overflow;
    @(posedge clk) disable iff (!rst_n)((wr_en && full) |=> (overflow==1));
endproperty

property p_rd_ptr;
    @(posedge clk) disable iff (!rst_n)((rd_en && !empty) |=> (rd_ptr==$past(rd_ptr)+1'b1));
endproperty

property p_wr_ptr;
    @(posedge clk) disable iff (!rst_n)((wr_en && !full) |=> (wr_ptr==$past(wr_ptr)+1'b1));
endproperty

rst_assertion: assert property(p_rst);
full_assertion: assert property(p_full);
almostfull_assertion: assert property(p_almostfull);
almostempty_assertion: assert property(p_almostempty);
empty_assertion: assert property(p_empty);
underflow_assertion: assert property(p_underflow);
```



```
139  empty_assertion: assert property(p_empty);
140  underflow_assertion: assert property(p_underflow);
141  overflow_assertion: assert property(p_overflow);
142  rd_ptr_assertion: assert property(p_rd_ptr);
143  wr_ptr_assertion: assert property(p_wr_ptr);
144
145  rst_cover: cover property(p_rst);
146  full_cover: cover property(p_full);
147  almostfull_cover: cover property(p_almostfull);
148  almostempty_cover: cover property(p_almostempty);
149  empty_cover: cover property(p_empty);
150  underflow_cover: cover property(p_underflow);
151  overflow_cover: cover property(p_overflow);
152  rd_ptr_cover: cover property(p_rd_ptr);
153  wr_ptr_cover: cover property(p_wr_ptr);
154
155  endmodule
```

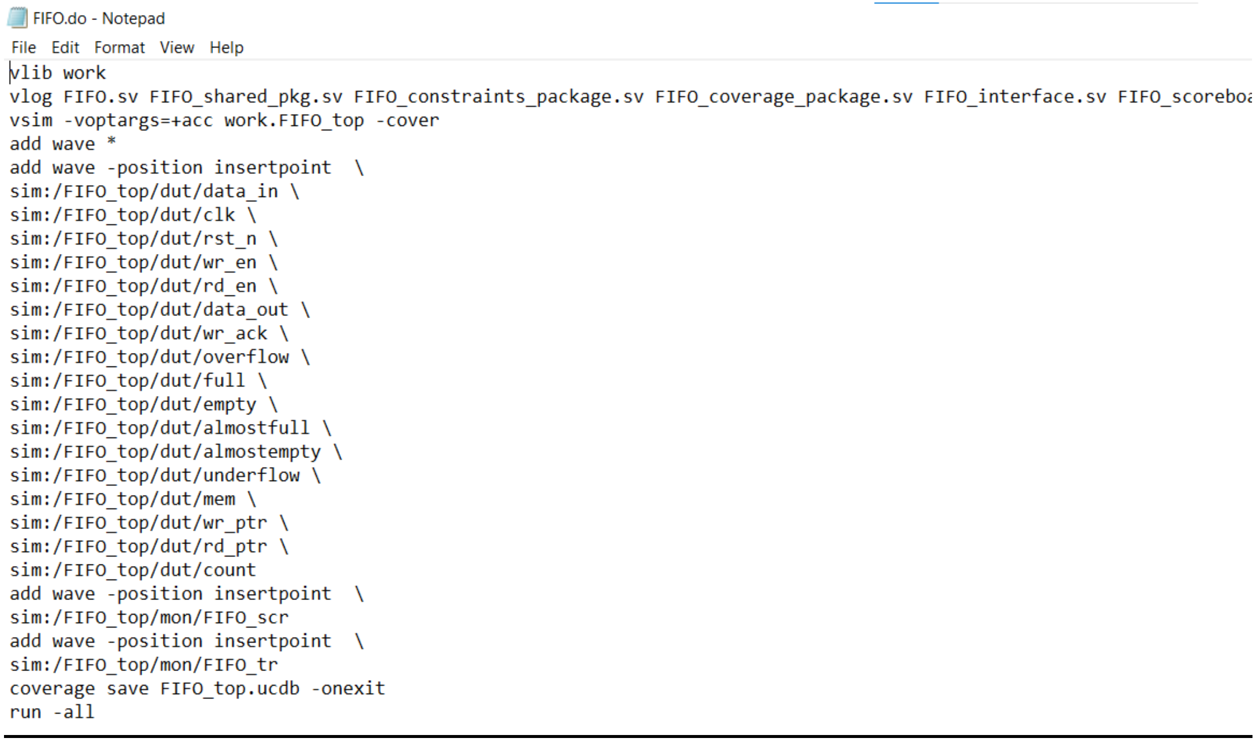
Bugs were found in lines 42,43,49,67,69,86-89,95 . comments are written in the code to explain the change and why it was required.

<u>Verification plan:</u>



Label	Description	Stimulus Generation	Functional Coverage (Later)	Functionality Check
FIFO_1	when rst_n is low ,the FIFO pointers and count should be low.	directed at start of simulation then Randomized under constraints that drive the the rst_n to be high 97% of the time		A checker in the scoreboard package to make sure the output is correct, and also an immediate assertion to check for functionality
FIFO_2	when wr_en is high and FIFO is not full , data_in should be stored in FIFO with the wr_ack turning high and the wr_ptr incrementing, and if full then wr_en should be ignored with the overflow signal turning high.	Randomized under constraints that drive the wr_en to be high 70% of the time	helps in cross_covering all bin combinations of wr_en, rd_en and all output signals except for specified illegal_bins	A checker in the scoreboard package to make sure the output is correct, and also an immediate assertion to check for full and almostfull signals as they re combinational, and concurrent assertions to check for overflow and wr_ptr signals.
FIFO_3	when rd_en is high and FIFO is not empty, data_out should be earliest data_in that was written and not read before, and the rd_ptr should increment, and if empty then rd_en should be ignored with the underflow signal turning high.	Randomized under constraints that drive the rd_en to be high 30% of the time	helps in cross_covering all bin combinations of wr_en, rd_en and all output signals except for specified illegal_bins	A checker in the scoreboard package to make sure the output is correct, and also an immediate assertion to check for empty and almostempty signals as they re combinational, and concurrent assertions to check for underflow and rd_ptr signals.

Do file:

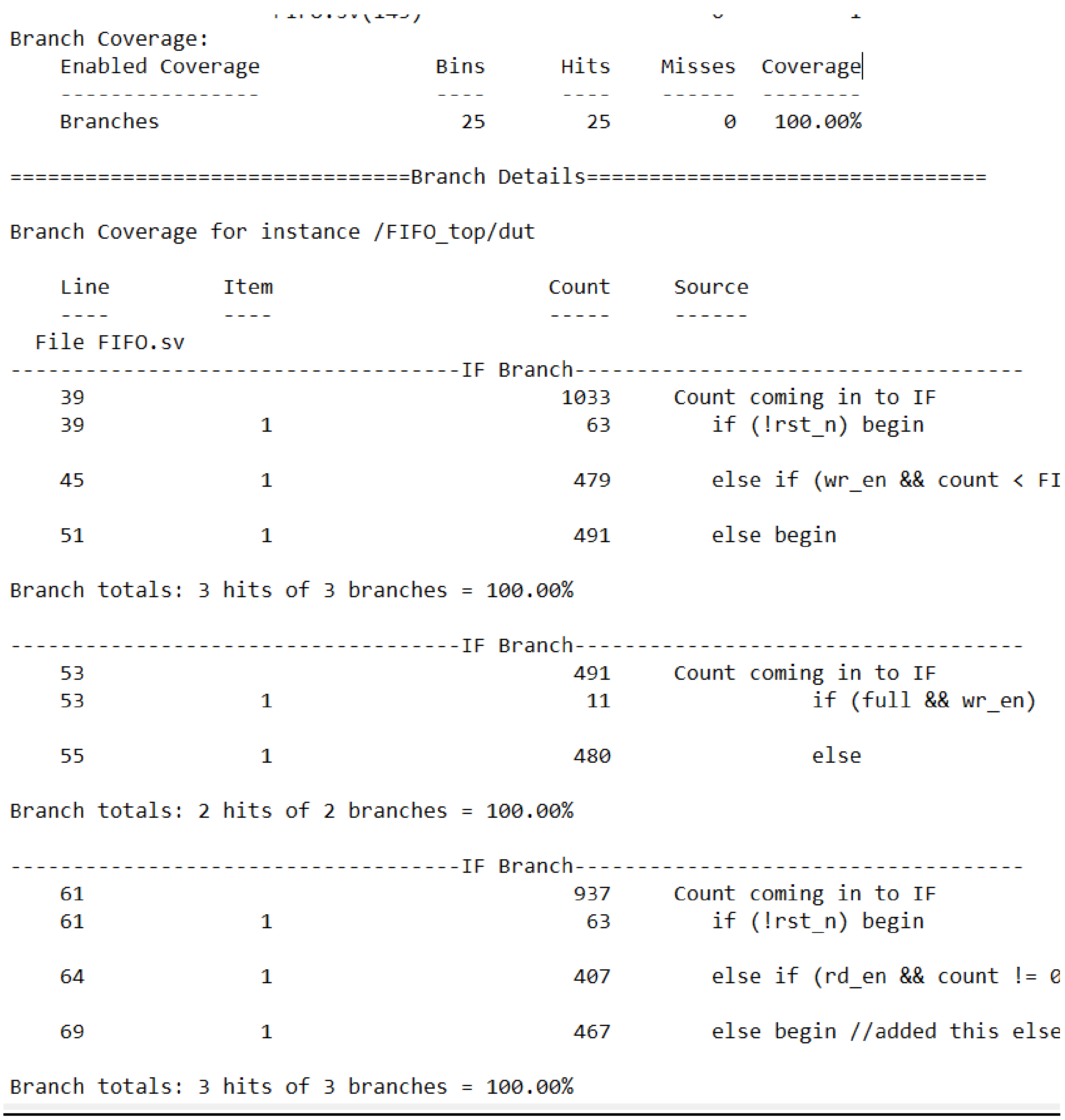

FIFO.do - Notepad

File Edit Format View Help

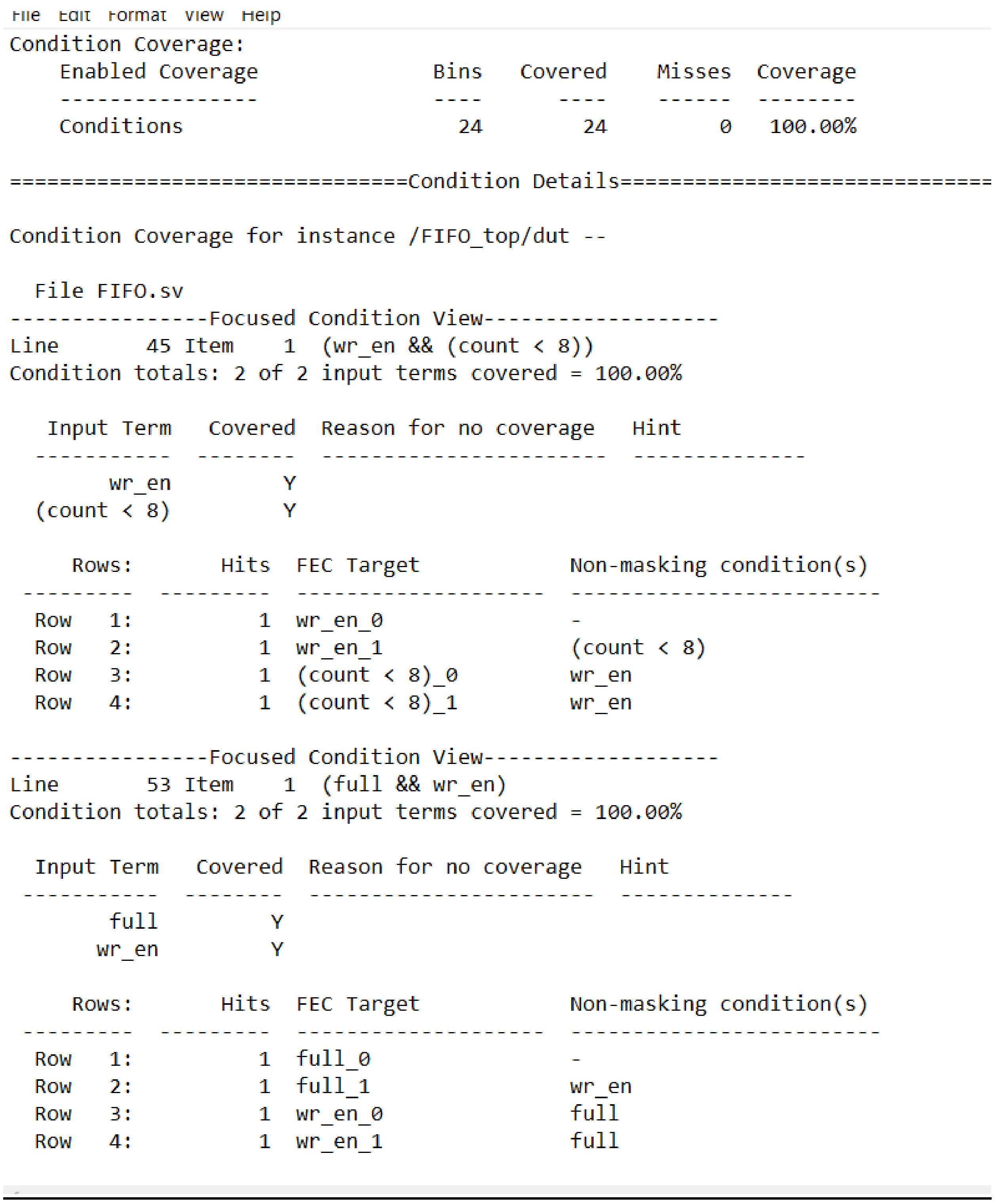
```
vlib work
vlog FIFO.sv FIFO_shared_pkg.sv FIFO_constraints_package.sv FIFO_coverage_package.sv FIFO_interface.sv FIFO_scoreboa
vsim -voptargs=+acc work.FIFO_top -cover
add wave *
add wave -position insertpoint  \
sim:/FIFO_top/dut/data_in \
sim:/FIFO_top/dut/clk \
sim:/FIFO_top/dut/rst_n \
sim:/FIFO_top/dut/wr_en \
sim:/FIFO_top/dut/rd_en \
sim:/FIFO_top/dut/data_out \
sim:/FIFO_top/dut/wr_ack \
sim:/FIFO_top/dut/overflow \
sim:/FIFO_top/dut/full \
sim:/FIFO_top/dut/empty \
sim:/FIFO_top/dut/almostfull \
sim:/FIFO_top/dut/almostempty \
sim:/FIFO_top/dut/underflow \
sim:/FIFO_top/dut/mem \
sim:/FIFO_top/dut/wr_ptr \
sim:/FIFO_top/dut/rd_ptr \
sim:/FIFO_top/dut/count
add wave -position insertpoint  \
sim:/FIFO_top/mon/FIFO_scr
add wave -position insertpoint  \
sim:/FIFO_top/mon/FIFO_tr
coverage save FIFO_top.ucdb -onexit
run -all
```

Coverage:

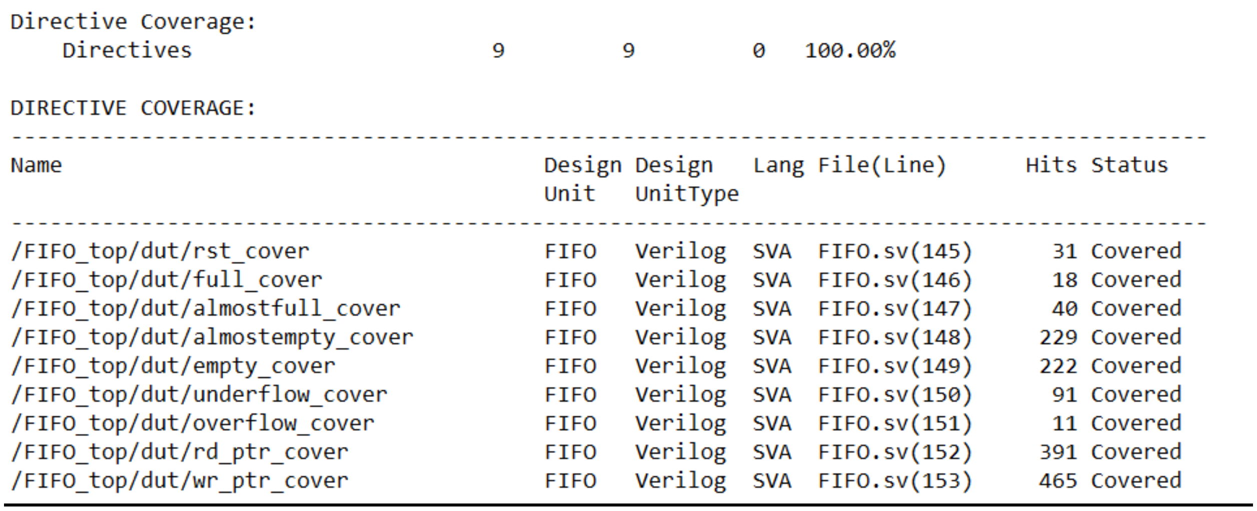
```
=================================================================================
=== Instance: /FIFO_top/dut
=== Design Unit: work.FIFO
=================================================================================

Assertion Coverage:
    Assertions                       9         9         0    100.00%
--------------------------------------------------------------------
Name                 File(Line)                   Failure      Pass
                                                  Count        Count
--------------------------------------------------------------------
/FIFO_top/dut/rst_assertion
                     FIFO.sv(135)                       0          1
/FIFO_top/dut/full_assertion
                     FIFO.sv(136)                       0          1
/FIFO_top/dut/almostfull_assertion
                     FIFO.sv(137)                       0          1
/FIFO_top/dut/almostempty_assertion
                     FIFO.sv(138)                       0          1
/FIFO_top/dut/empty_assertion
                     FIFO.sv(139)                       0          1
/FIFO_top/dut/underflow_assertion
                     FIFO.sv(140)                       0          1
/FIFO_top/dut/overflow_assertion
                     FIFO.sv(141)                       0          1
/FIFO_top/dut/rd_ptr_assertion
                     FIFO.sv(142)                       0          1
/FIFO_top/dut/wr_ptr_assertion
                     FIFO.sv(143)                       0          1
```



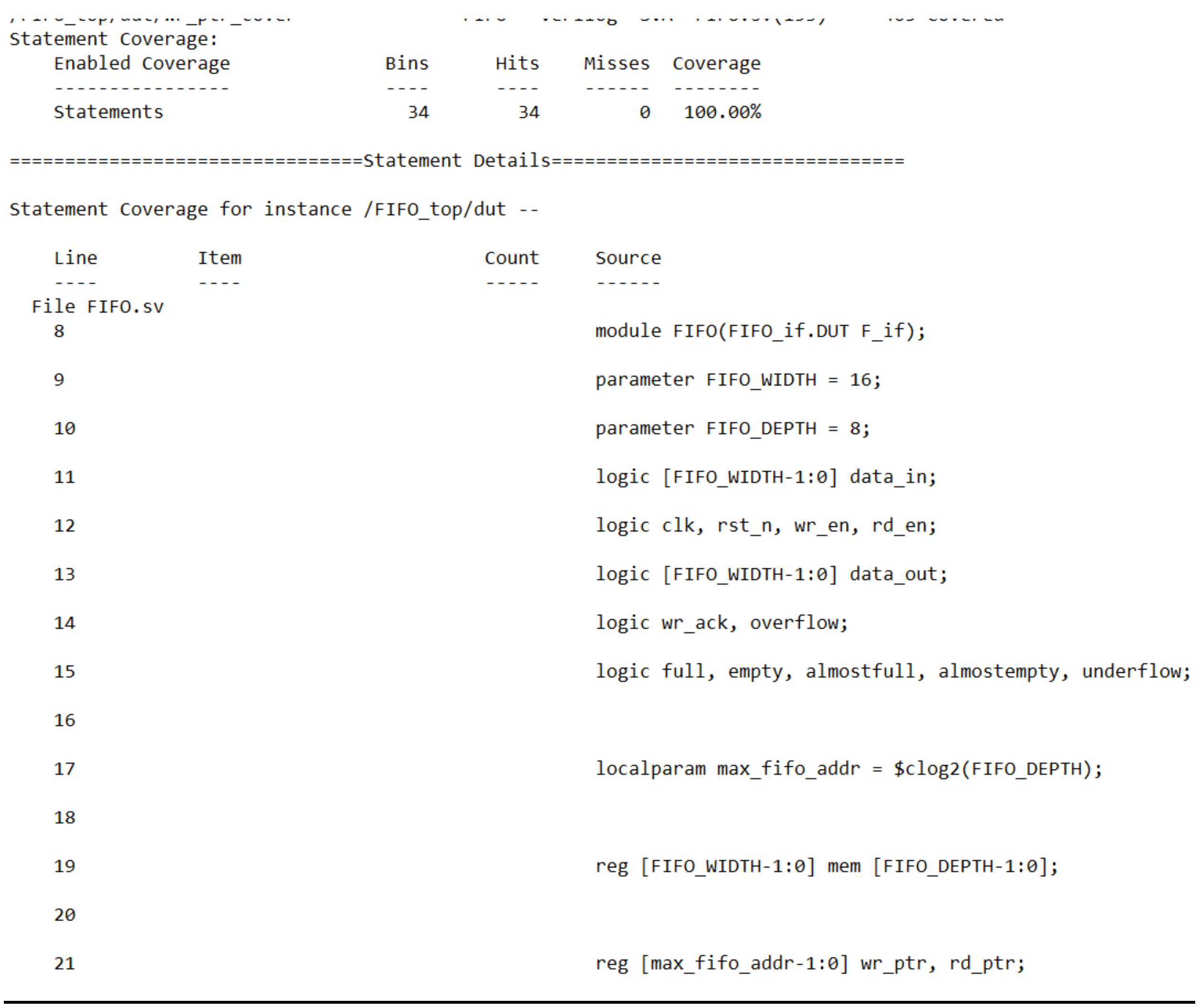
```
                 FIFO.sv(149)                        0          1
Branch Coverage:
    Enabled Coverage              Bins      Hits    Misses  Coverage
    ----------------              ----      ----    ------  --------
    Branches                        25        25         0   100.00%

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

Branch Coverage for instance /FIFO_top/dut

    Line         Item                      Count     Source
    ----         ----                      -----     ------
  File FIFO.sv
------------------------------------IF Branch------------------------------------
    39                                      1033     Count coming in to IF
    39              1                         63        if (!rst_n) begin

    45              1                        479        else if (wr_en && count < FI

    51              1                        491        else begin

Branch totals: 3 hits of 3 branches = 100.00%

------------------------------------IF Branch------------------------------------
    53                                       491     Count coming in to IF
    53              1                         11                if (full && wr_en)

    55              1                        480                else

Branch totals: 2 hits of 2 branches = 100.00%

------------------------------------IF Branch------------------------------------
    61                                       937     Count coming in to IF
    61              1                         63        if (!rst_n) begin

    64              1                        407        else if (rd_en && count != 0

    69              1                        467        else begin //added this else

Branch totals: 3 hits of 3 branches = 100.00%
```



```
Condition Coverage:
    Enabled Coverage              Bins    Covered    Misses  Coverage
    ----------------              ----       ----    ------  --------
    Conditions                      24         24         0   100.00%

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

Condition Coverage for instance /FIFO_top/dut --

  File FIFO.sv
----------------Focused Condition View-------------------
Line       45 Item    1  (wr_en && (count < 8))
Condition totals: 2 of 2 input terms covered = 100.00%

   Input Term   Covered  Reason for no coverage   Hint
  -----------  --------  -----------------------  --------------
        wr_en         Y
  (count < 8)         Y

     Rows:       Hits  FEC Target            Non-masking condition(s)
 ---------  ---------  --------------------  -------------------------
  Row   1:          1  wr_en_0               -
  Row   2:          1  wr_en_1               (count < 8)
  Row   3:          1  (count < 8)_0         wr_en
  Row   4:          1  (count < 8)_1         wr_en

----------------Focused Condition View-------------------
Line       53 Item    1  (full && wr_en)
Condition totals: 2 of 2 input terms covered = 100.00%

  Input Term   Covered  Reason for no coverage   Hint
 -----------  --------  -----------------------  --------------
        full         Y
       wr_en         Y

     Rows:       Hits  FEC Target            Non-masking condition(s)
 ---------  ---------  --------------------  -------------------------
  Row   1:          1  full_0                -
  Row   2:          1  full_1                wr_en
  Row   3:          1  wr_en_0               full
  Row   4:          1  wr_en_1               full
```



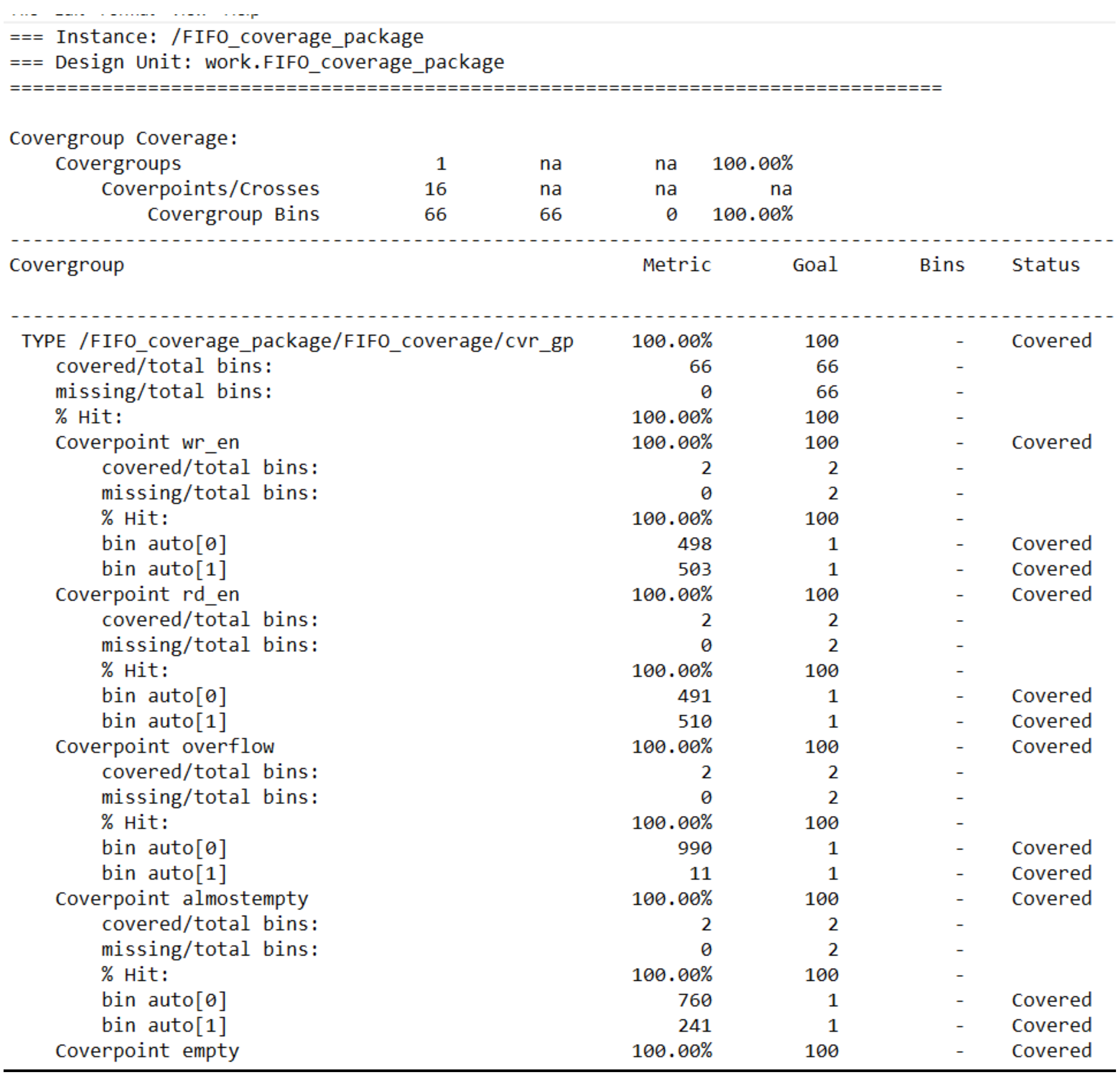
```
Directive Coverage:
    Directives                      9          9          0    100.00%

DIRECTIVE COVERAGE:
--------------------------------------------------------------------------------------------
Name                                    Design Design   Lang File(Line)      Hits Status
                                        Unit   UnitType
--------------------------------------------------------------------------------------------
/FIFO_top/dut/rst_cover                 FIFO   Verilog  SVA  FIFO.sv(145)      31 Covered
/FIFO_top/dut/full_cover                FIFO   Verilog  SVA  FIFO.sv(146)      18 Covered
/FIFO_top/dut/almostfull_cover          FIFO   Verilog  SVA  FIFO.sv(147)      40 Covered
/FIFO_top/dut/almostempty_cover         FIFO   Verilog  SVA  FIFO.sv(148)     229 Covered
/FIFO_top/dut/empty_cover               FIFO   Verilog  SVA  FIFO.sv(149)     222 Covered
/FIFO_top/dut/underflow_cover           FIFO   Verilog  SVA  FIFO.sv(150)      91 Covered
/FIFO_top/dut/overflow_cover            FIFO   Verilog  SVA  FIFO.sv(151)      11 Covered
/FIFO_top/dut/rd_ptr_cover              FIFO   Verilog  SVA  FIFO.sv(152)     391 Covered
/FIFO_top/dut/wr_ptr_cover              FIFO   Verilog  SVA  FIFO.sv(153)     465 Covered
```



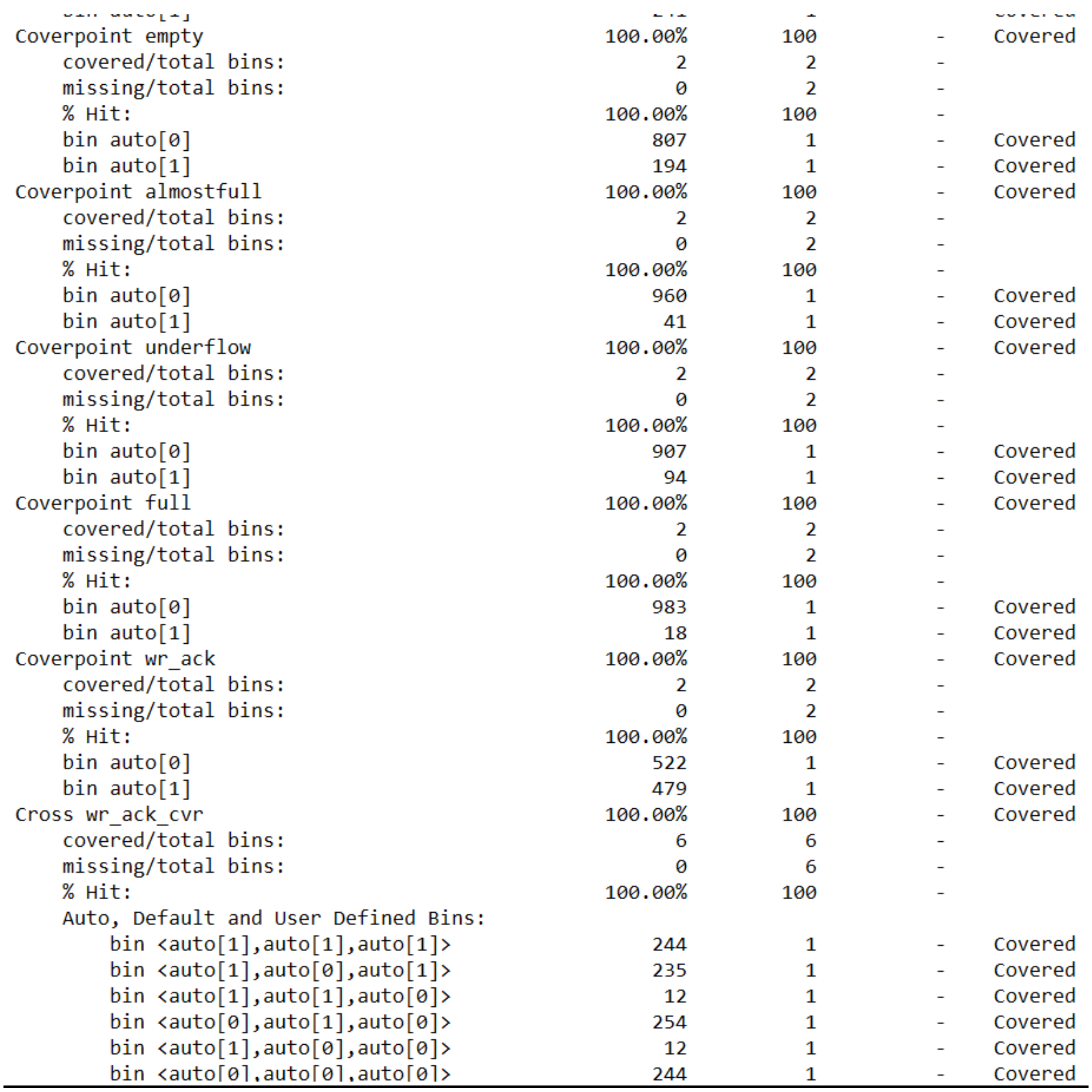
```
Statement Coverage:
    Enabled Coverage              Bins      Hits    Misses  Coverage
    ----------------              ----      ----    ------  --------
    Statements                      34        34         0   100.00%

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

Statement Coverage for instance /FIFO_top/dut --

    Line         Item                      Count     Source
    ----         ----                      -----     ------
  File FIFO.sv
    8                                                module FIFO(FIFO_if.DUT F_if);

    9                                                parameter FIFO_WIDTH = 16;

    10                                               parameter FIFO_DEPTH = 8;

    11                                               logic [FIFO_WIDTH-1:0] data_in;

    12                                               logic clk, rst_n, wr_en, rd_en;

    13                                               logic [FIFO_WIDTH-1:0] data_out;

    14                                               logic wr_ack, overflow;

    15                                               logic full, empty, almostfull, almostempty, underflow;

    16

    17                                               localparam max_fifo_addr = $clog2(FIFO_DEPTH);

    18

    19                                               reg [FIFO_WIDTH-1:0] mem [FIFO_DEPTH-1:0];

    20

    21                                               reg [max_fifo_addr-1:0] wr_ptr, rd_ptr;
```

```
Toggle Coverage:
    Enabled Coverage              Bins      Hits    Misses  Coverage
    ----------------              ----      ----    ------  --------
    Toggles                        106       106         0   100.00%

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

Toggle Coverage for instance /FIFO_top/dut --

                                              Node      1H->0L      0L->1H  "Coverage"
                                              ---------------------------------------
                                       almostempty           1           1      100.00
                                        almostfull           1           1      100.00
                                               clk           1           1      100.00
                                        count[3-0]           1           1      100.00
                                     data_in[15-0]           1           1      100.00
                                    data_out[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
                                       rd_ptr[2-0]           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
                                       wr_ptr[2-0]           1           1      100.00

Total Node Count     =         53
Toggled Node Count   =         53
Untoggled Node Count =          0

Toggle Coverage      =     100.00% (106 of 106 bins)
```

Functional coverage:

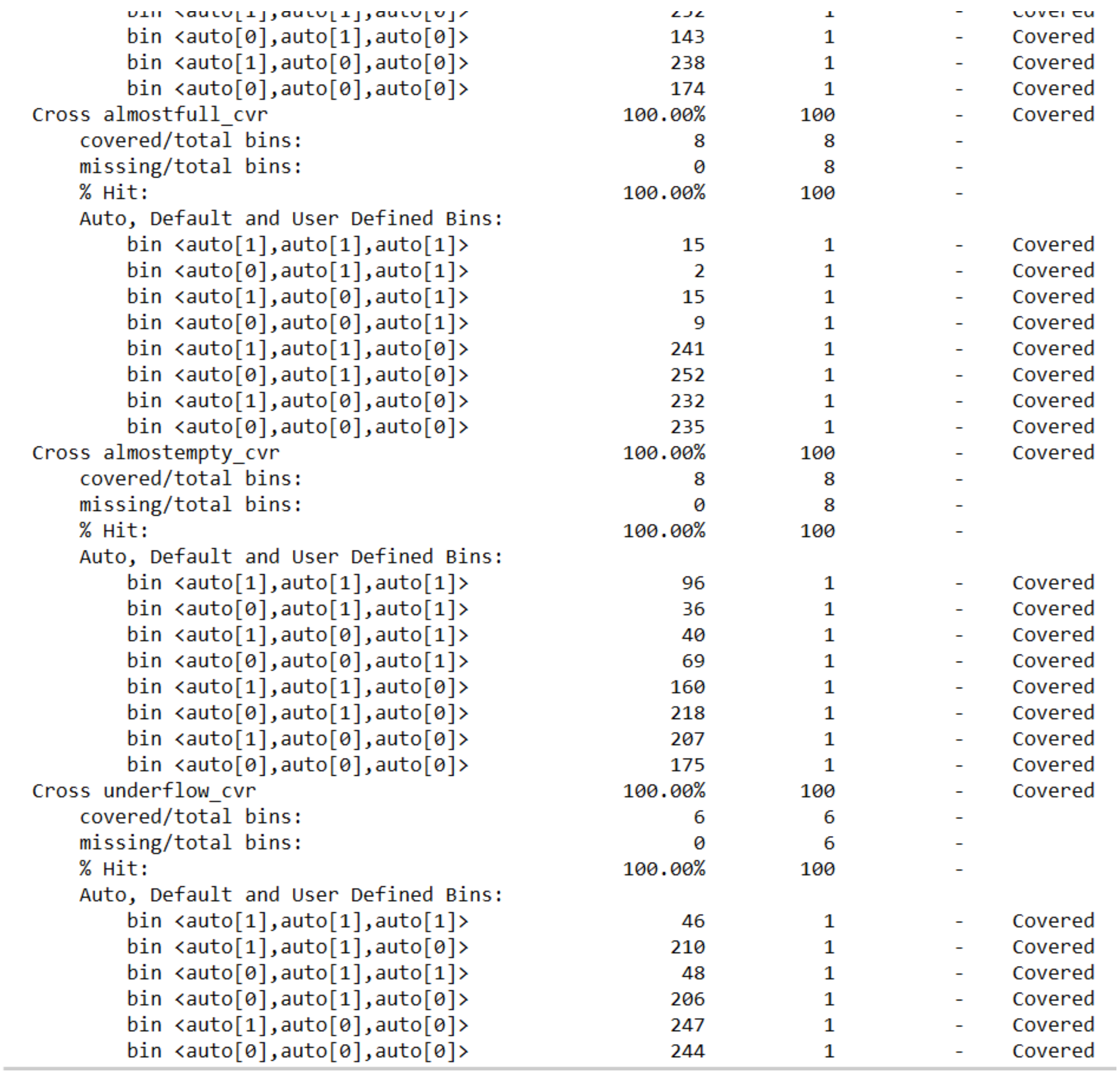


```
=== Instance: /FIFO_coverage_package
=== Design Unit: work.FIFO_coverage_package
=================================================================================

Covergroup Coverage:
    Covergroups                      1        na        na   100.00%
        Coverpoints/Crosses         16        na        na        na
            Covergroup Bins         66        66         0   100.00%
----------------------------------------------------------------------------------
Covergroup                                       Metric       Goal       Bins    Status

----------------------------------------------------------------------------------
 TYPE /FIFO_coverage_package/FIFO_coverage/cvr_gp   100.00%        100          -    Covered
    covered/total bins:                               66         66          -
    missing/total bins:                                0         66          -
    % Hit:                                       100.00%        100          -
    Coverpoint wr_en                             100.00%        100          -    Covered
        covered/total bins:                            2          2          -
        missing/total bins:                            0          2          -
        % Hit:                                   100.00%        100          -
        bin auto[0]                                  498          1          -    Covered
        bin auto[1]                                  503          1          -    Covered
    Coverpoint rd_en                             100.00%        100          -    Covered
        covered/total bins:                            2          2          -
        missing/total bins:                            0          2          -
        % Hit:                                   100.00%        100          -
        bin auto[0]                                  491          1          -    Covered
        bin auto[1]                                  510          1          -    Covered
    Coverpoint overflow                          100.00%        100          -    Covered
        covered/total bins:                            2          2          -
        missing/total bins:                            0          2          -
        % Hit:                                   100.00%        100          -
        bin auto[0]                                  990          1          -    Covered
        bin auto[1]                                   11          1          -    Covered
    Coverpoint almostempty                       100.00%        100          -    Covered
        covered/total bins:                            2          2          -
        missing/total bins:                            0          2          -
        % Hit:                                   100.00%        100          -
        bin auto[0]                                  760          1          -    Covered
        bin auto[1]                                  241          1          -    Covered
    Coverpoint empty                             100.00%        100          -    Covered
```

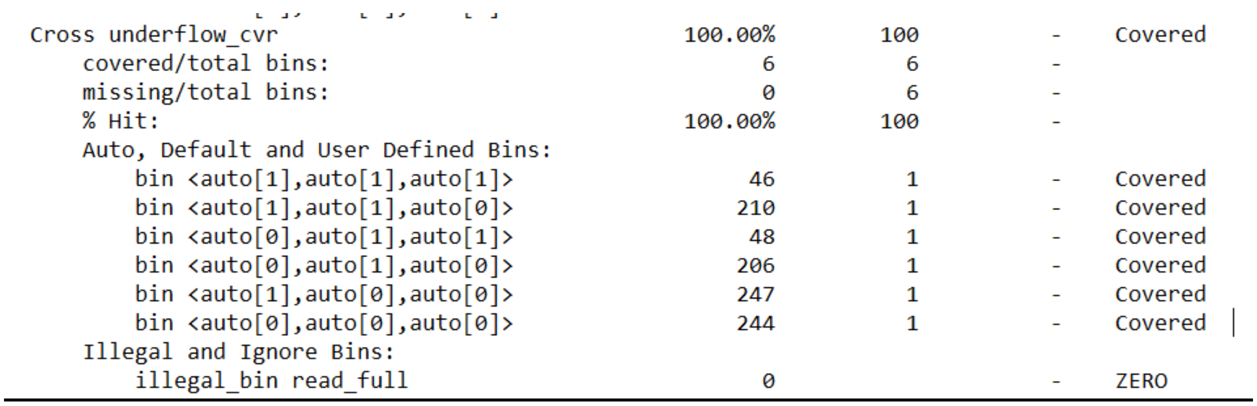


```
Coverpoint empty                                  100.00%        100          -    Covered
    covered/total bins:                                 2          2          -
    missing/total bins:                                 0          2          -
    % Hit:                                        100.00%        100          -
    bin auto[0]                                       807          1          -    Covered
    bin auto[1]                                       194          1          -    Covered
Coverpoint almostfull                             100.00%        100          -    Covered
    covered/total bins:                                 2          2          -
    missing/total bins:                                 0          2          -
    % Hit:                                        100.00%        100          -
    bin auto[0]                                       960          1          -    Covered
    bin auto[1]                                        41          1          -    Covered
Coverpoint underflow                              100.00%        100          -    Covered
    covered/total bins:                                 2          2          -
    missing/total bins:                                 0          2          -
    % Hit:                                        100.00%        100          -
    bin auto[0]                                       907          1          -    Covered
    bin auto[1]                                        94          1          -    Covered
Coverpoint full                                   100.00%        100          -    Covered
    covered/total bins:                                 2          2          -
    missing/total bins:                                 0          2          -
    % Hit:                                        100.00%        100          -
    bin auto[0]                                       983          1          -    Covered
    bin auto[1]                                        18          1          -    Covered
Coverpoint wr_ack                                 100.00%        100          -    Covered
    covered/total bins:                                 2          2          -
    missing/total bins:                                 0          2          -
    % Hit:                                        100.00%        100          -
    bin auto[0]                                       522          1          -    Covered
    bin auto[1]                                       479          1          -    Covered
Cross wr_ack_cvr                                  100.00%        100          -    Covered
    covered/total bins:                                 6          6          -
    missing/total bins:                                 0          6          -
    % Hit:                                        100.00%        100          -
    Auto, Default and User Defined Bins:
        bin <auto[1],auto[1],auto[1]>                 244          1          -    Covered
        bin <auto[1],auto[0],auto[1]>                 235          1          -    Covered
        bin <auto[1],auto[1],auto[0]>                  12          1          -    Covered
        bin <auto[0],auto[1],auto[0]>                 254          1          -    Covered
        bin <auto[1],auto[0],auto[0]>                  12          1          -    Covered
        bin <auto[0],auto[0],auto[0]>                 244          1          -    Covered
```

```
            bin <auto[0],auto[0],auto[0]>             244          1          -    Covered
        Illegal and Ignore Bins:
            illegal_bin wr_wr_ack                       0                     -    ZERO
    Cross overflow_cvr                            100.00%        100          -    Covered
        covered/total bins:                             6          6          -
        missing/total bins:                             0          6          -
        % Hit:                                    100.00%        100          -
        Auto, Default and User Defined Bins:
            bin <auto[1],auto[1],auto[1]>               8          1          -    Covered
            bin <auto[1],auto[0],auto[1]>               3          1          -    Covered
            bin <auto[1],auto[1],auto[0]>             248          1          -    Covered
            bin <auto[0],auto[1],auto[0]>             254          1          -    Covered
            bin <auto[1],auto[0],auto[0]>             244          1          -    Covered
            bin <auto[0],auto[0],auto[0]>             244          1          -    Covered
        Illegal and Ignore Bins:
            illegal_bin write_overflow                  0                     -    ZERO
    Cross full_cvr                                100.00%        100          -    Covered
        covered/total bins:                             6          6          -
        missing/total bins:                             0          6          -
        % Hit:                                    100.00%        100          -
        Auto, Default and User Defined Bins:
            bin <auto[1],auto[1],auto[0]>             256          1          -    Covered
            bin <auto[0],auto[1],auto[0]>             254          1          -    Covered
            bin <auto[1],auto[0],auto[1]>              13          1          -    Covered
            bin <auto[1],auto[0],auto[0]>             234          1          -    Covered
            bin <auto[0],auto[0],auto[1]>               5          1          -    Covered
            bin <auto[0],auto[0],auto[0]>             239          1          -    Covered
        Illegal and Ignore Bins:
            illegal_bin read_full                       0                     -    ZERO
    Cross empty_cvr                               100.00%        100          -    Covered
        covered/total bins:                             8          8          -
        missing/total bins:                             0          8          -
        % Hit:                                    100.00%        100          -
        Auto, Default and User Defined Bins:
            bin <auto[1],auto[1],auto[1]>               4          1          -    Covered
            bin <auto[0],auto[1],auto[1]>             111          1          -    Covered
            bin <auto[1],auto[0],auto[1]>               9          1          -    Covered
            bin <auto[0],auto[0],auto[1]>              70          1          -    Covered
            bin <auto[1],auto[1],auto[0]>             252          1          -    Covered
            bin <auto[0],auto[1],auto[0]>             143          1          -    Covered
```



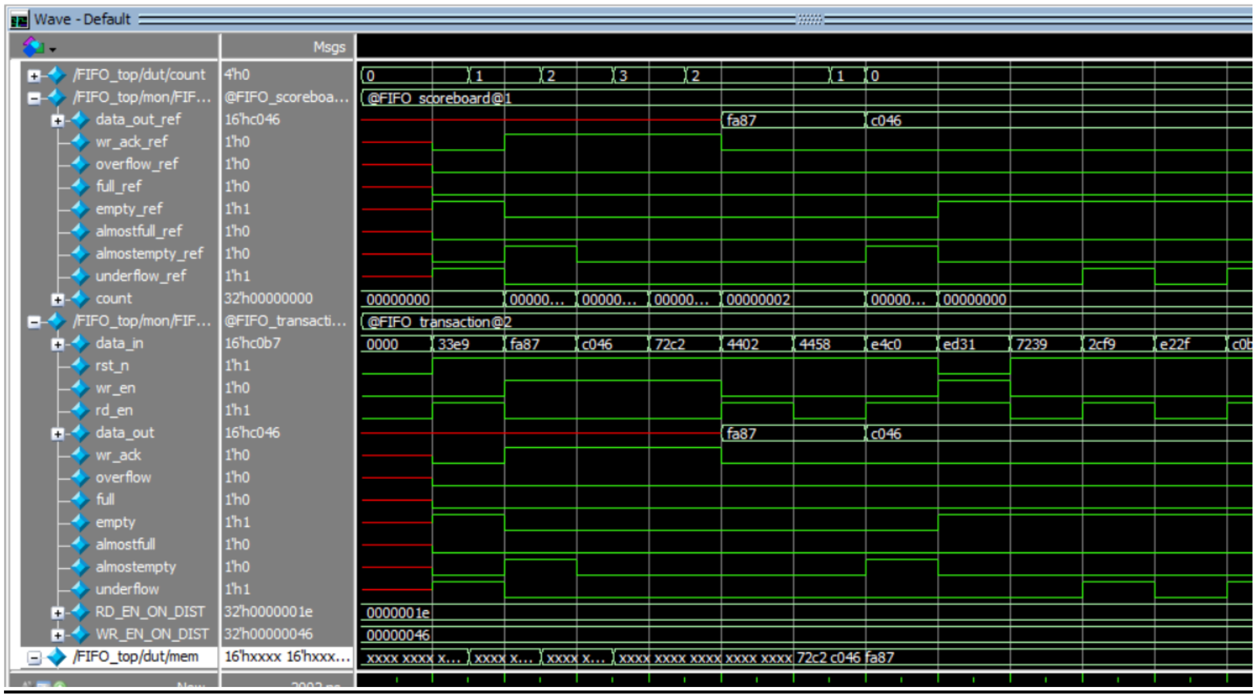
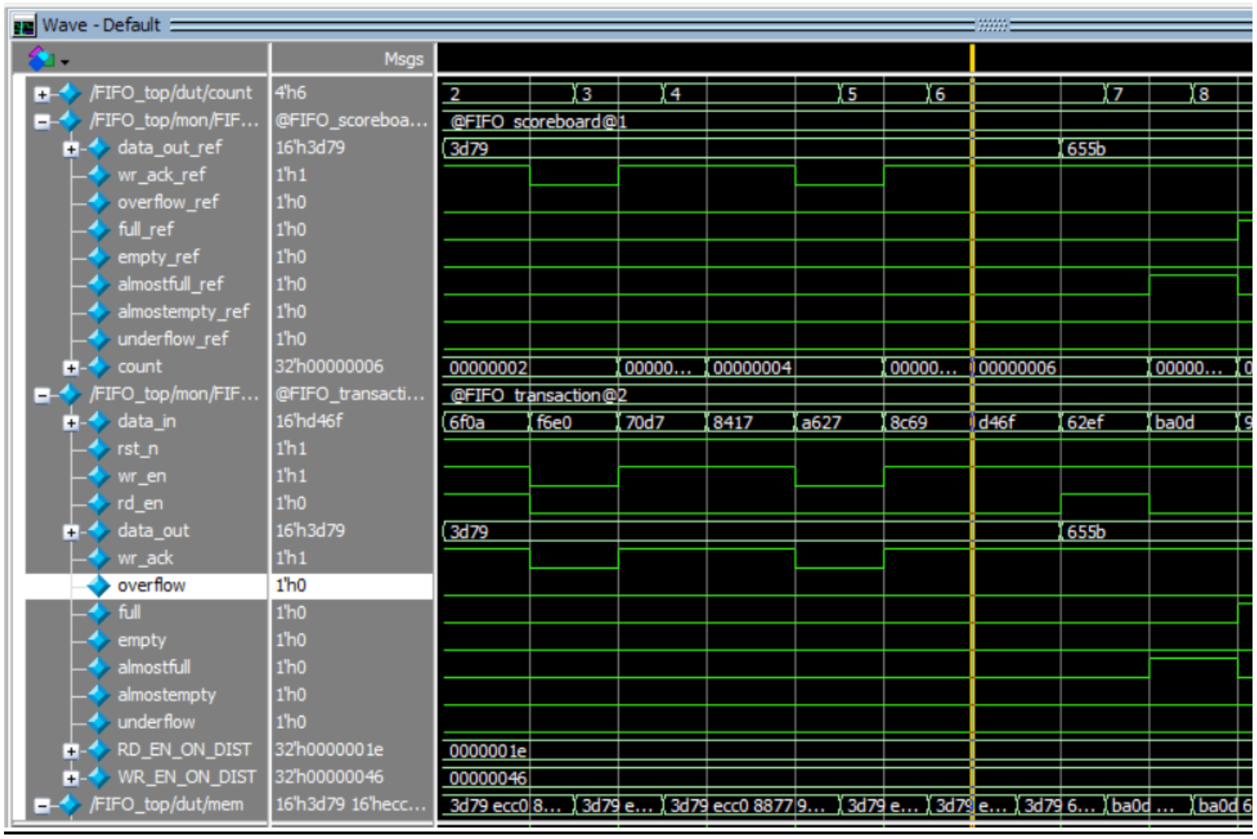
```
        bin <auto[0],auto[1],auto[0]>                  143           1              -    Covered
        bin <auto[1],auto[0],auto[0]>                  238           1              -    Covered
        bin <auto[0],auto[0],auto[0]>                  174           1              -    Covered
Cross almostfull_cvr                               100.00%         100              -    Covered
    covered/total bins:                                  8           8              -
    missing/total bins:                                  0           8              -
    % Hit:                                         100.00%         100              -
    Auto, Default and User Defined Bins:
        bin <auto[1],auto[1],auto[1]>                   15           1              -    Covered
        bin <auto[0],auto[1],auto[1]>                    2           1              -    Covered
        bin <auto[1],auto[0],auto[1]>                   15           1              -    Covered
        bin <auto[0],auto[0],auto[1]>                    9           1              -    Covered
        bin <auto[1],auto[1],auto[0]>                  241           1              -    Covered
        bin <auto[0],auto[1],auto[0]>                  252           1              -    Covered
        bin <auto[1],auto[0],auto[0]>                  232           1              -    Covered
        bin <auto[0],auto[0],auto[0]>                  235           1              -    Covered
Cross almostempty_cvr                              100.00%         100              -    Covered
    covered/total bins:                                  8           8              -
    missing/total bins:                                  0           8              -
    % Hit:                                         100.00%         100              -
    Auto, Default and User Defined Bins:
        bin <auto[1],auto[1],auto[1]>                   96           1              -    Covered
        bin <auto[0],auto[1],auto[1]>                   36           1              -    Covered
        bin <auto[1],auto[0],auto[1]>                   40           1              -    Covered
        bin <auto[0],auto[0],auto[1]>                   69           1              -    Covered
        bin <auto[1],auto[1],auto[0]>                  160           1              -    Covered
        bin <auto[0],auto[1],auto[0]>                  218           1              -    Covered
        bin <auto[1],auto[0],auto[0]>                  207           1              -    Covered
        bin <auto[0],auto[0],auto[0]>                  175           1              -    Covered
Cross underflow_cvr                                100.00%         100              -    Covered
    covered/total bins:                                  6           6              -
    missing/total bins:                                  0           6              -
    % Hit:                                         100.00%         100              -
    Auto, Default and User Defined Bins:
        bin <auto[1],auto[1],auto[1]>                   46           1              -    Covered
        bin <auto[1],auto[1],auto[0]>                  210           1              -    Covered
        bin <auto[0],auto[1],auto[1]>                   48           1              -    Covered
        bin <auto[0],auto[1],auto[0]>                  206           1              -    Covered
        bin <auto[1],auto[0],auto[0]>                  247           1              -    Covered
        bin <auto[0],auto[0],auto[0]>                  244           1              -    Covered
```

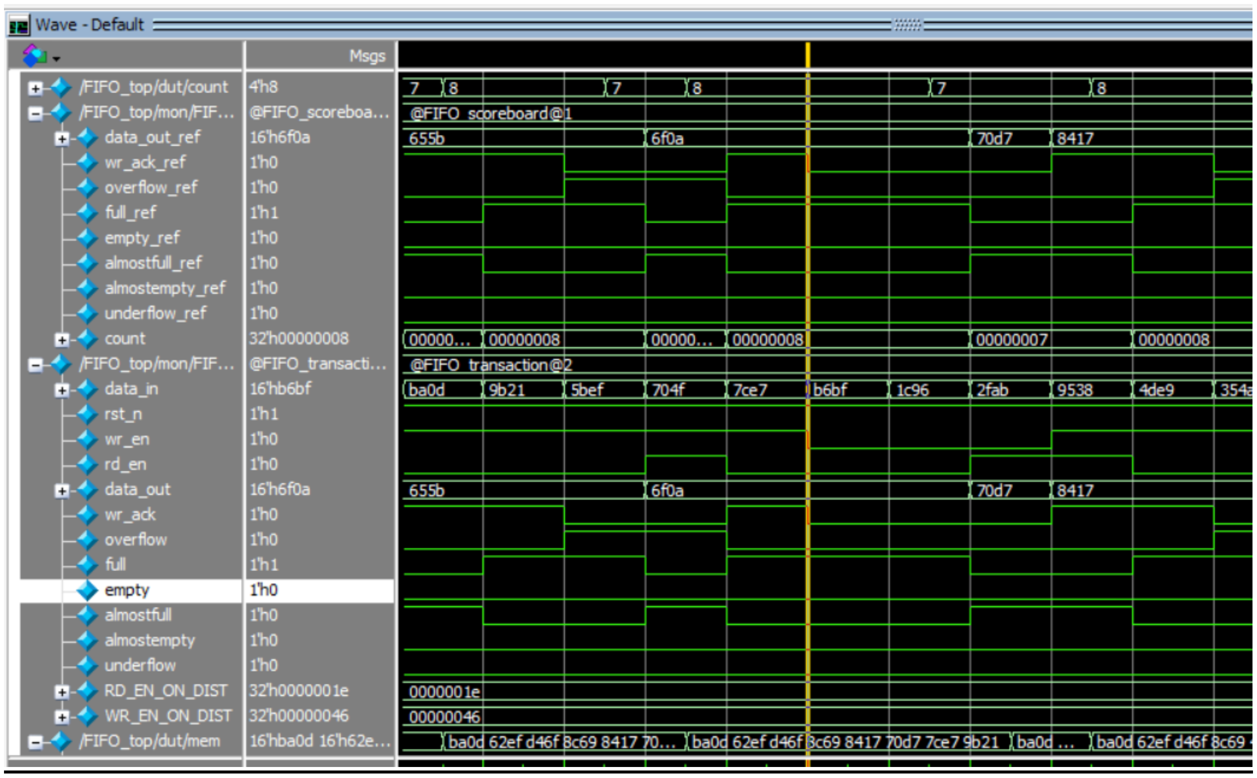


```
Cross underflow_cvr                                100.00%         100              -    Covered
    covered/total bins:                                  6           6              -
    missing/total bins:                                  0           6              -
    % Hit:                                         100.00%         100              -
    Auto, Default and User Defined Bins:
        bin <auto[1],auto[1],auto[1]>                   46           1              -    Covered
        bin <auto[1],auto[1],auto[0]>                  210           1              -    Covered
        bin <auto[0],auto[1],auto[1]>                   48           1              -    Covered
        bin <auto[0],auto[1],auto[0]>                  206           1              -    Covered
        bin <auto[1],auto[0],auto[0]>                  247           1              -    Covered
        bin <auto[0],auto[0],auto[0]>                  244           1              -    Covered
```

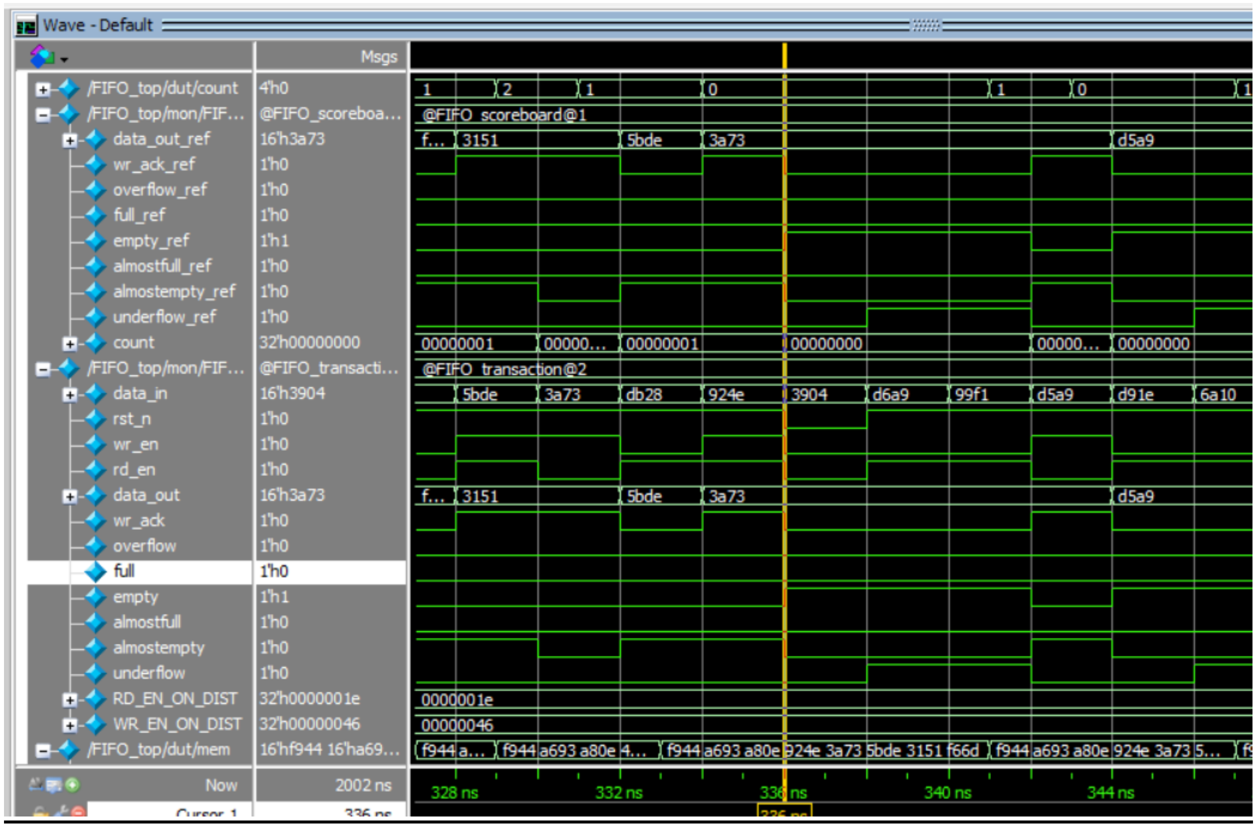
```
    Illegal and Ignore Bins:
        illegal_bin read_full                            0                          -    ZERO
```

Questasim Snippets:

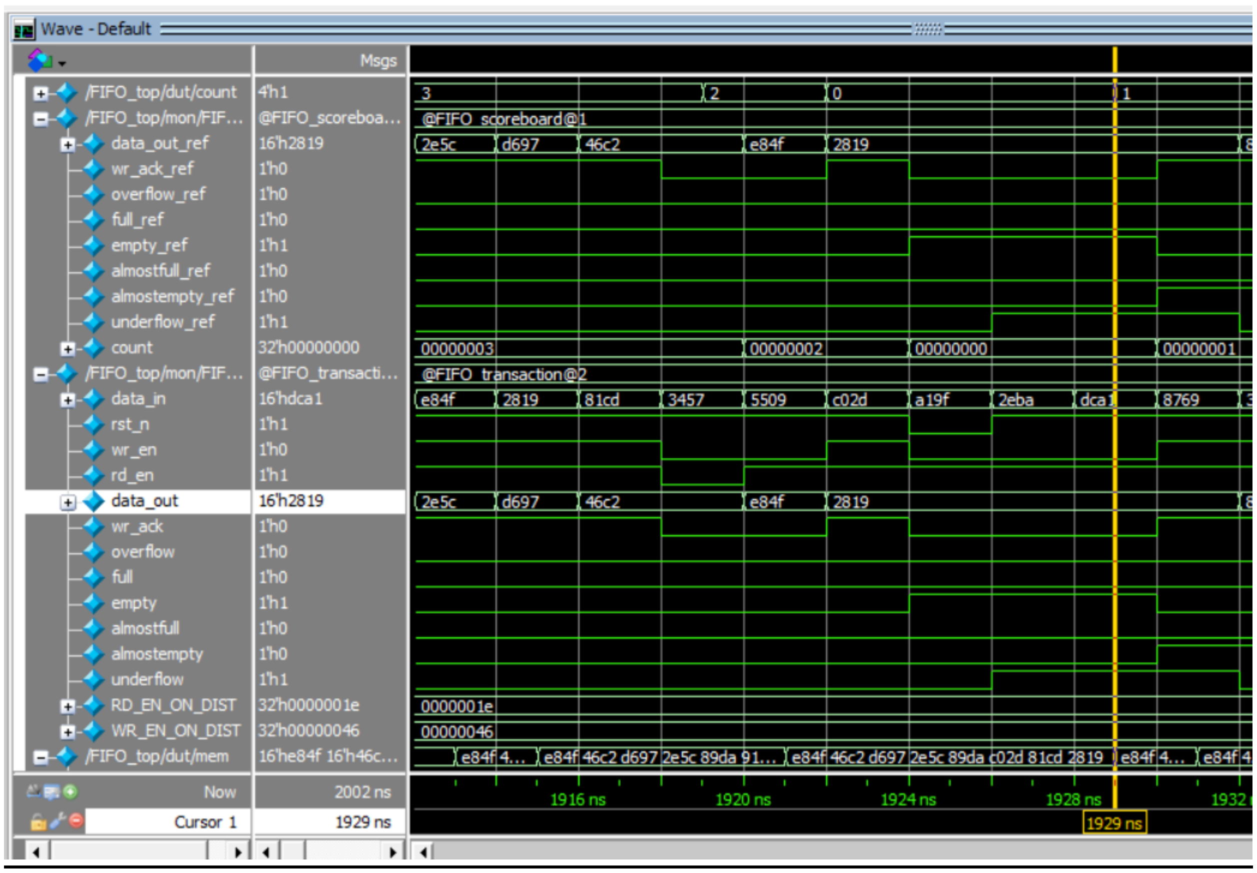

```
Wave - Default
                                Msgs
/FIFO_top/dut/count             4'h0
/FIFO_top/mon/FIF...            @FIFO_scoreboa...
  data_out_ref                  16'hc046
  wr_ack_ref                    1'h0
  overflow_ref                  1'h0
  full_ref                      1'h0
  empty_ref                     1'h1
  almostfull_ref                1'h0
  almostempty_ref               1'h0
  underflow_ref                 1'h1
  count                         32'h00000000
/FIFO_top/mon/FIF...            @FIFO_transacti...
  data_in                       16'hc0b7
  rst_n                         1'h1
  wr_en                         1'h0
  rd_en                         1'h1
  data_out                      16'hc046
  wr_ack                        1'h0
  overflow                      1'h0
  full                          1'h0
  empty                         1'h1
  almostfull                    1'h0
  almostempty                   1'h0
  underflow                     1'h1
  RD_EN_ON_DIST                 32'h0000001e
  WR_EN_ON_DIST                 32'h00000046
/FIFO_top/dut/mem               16'hxxxx 16'hxxx...
```

```
Wave - Default
                                Msgs
/FIFO_top/dut/count             4'h6
/FIFO_top/mon/FIF...            @FIFO_scoreboa...
  data_out_ref                  16'h3d79
  wr_ack_ref                    1'h1
  overflow_ref                  1'h0
  full_ref                      1'h0
  empty_ref                     1'h0
  almostfull_ref                1'h0
  almostempty_ref               1'h0
  underflow_ref                 1'h0
  count                         32'h00000006
/FIFO_top/mon/FIF...            @FIFO_transacti...
  data_in                       16'hd46f
  rst_n                         1'h1
  wr_en                         1'h1
  rd_en                         1'h0
  data_out                      16'h3d79
  wr_ack                        1'h1
  overflow                      1'h0
  full                          1'h0
  empty                         1'h0
  almostfull                    1'h0
  almostempty                   1'h0
  underflow                     1'h0
  RD_EN_ON_DIST                 32'h0000001e
  WR_EN_ON_DIST                 32'h00000046
/FIFO_top/dut/mem               16'h3d79 16'hecc...
```

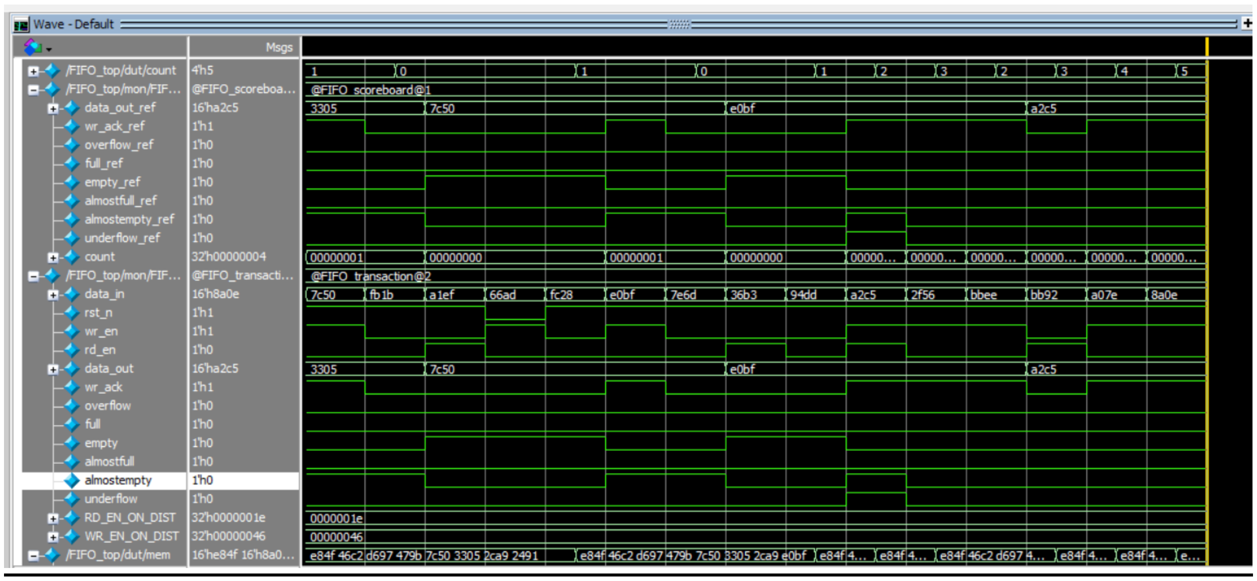


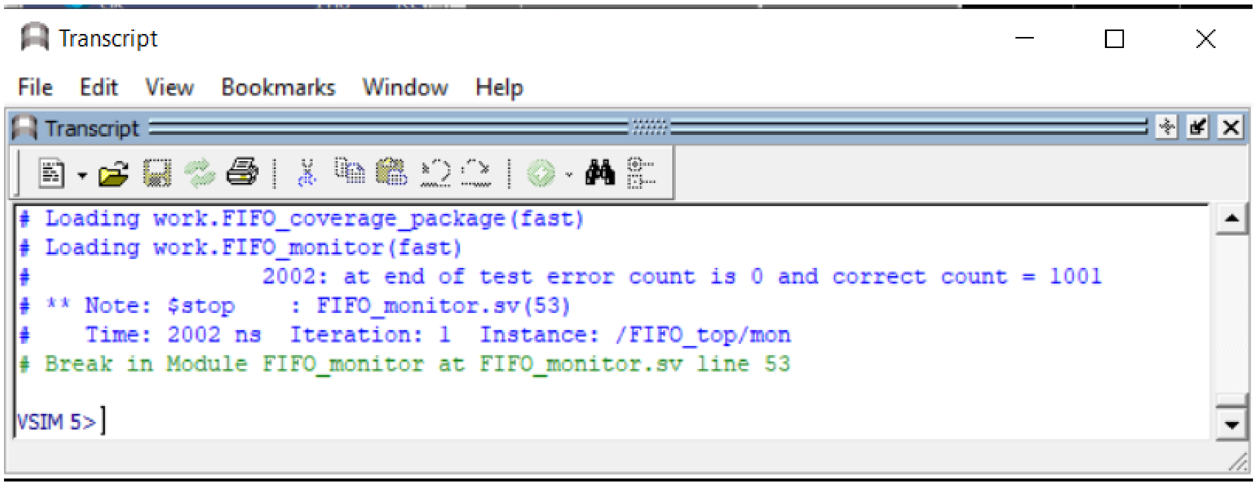
	Msgs
/FIFO_top/dut/count	4'h8
/FIFO_top/mon/FIF...	@FIFO_scoreboa...
data_out_ref	16'h6f0a
wr_ack_ref	1'h0
overflow_ref	1'h0
full_ref	1'h1
empty_ref	1'h0
almostfull_ref	1'h0
almostempty_ref	1'h0
underflow_ref	1'h0
count	32'h00000008
/FIFO_top/mon/FIF...	@FIFO_transacti...
data_in	16'hb6bf
rst_n	1'h1
wr_en	1'h0
rd_en	1'h0
data_out	16'h6f0a
wr_ack	1'h0
overflow	1'h0
full	1'h1
empty	1'h0
almostfull	1'h0
almostempty	1'h0
underflow	1'h0
RD_EN_ON_DIST	32'h0000001e
WR_EN_ON_DIST	32'h00000046
/FIFO_top/dut/mem	16'hba0d 16'h62e...



	Msgs
/FIFO_top/dut/count	4'h0
/FIFO_top/mon/FIF...	@FIFO_scoreboa...
data_out_ref	16'h3a73
wr_ack_ref	1'h0
overflow_ref	1'h0
full_ref	1'h0
empty_ref	1'h1
almostfull_ref	1'h0
almostempty_ref	1'h0
underflow_ref	1'h0
count	32'h00000000
/FIFO_top/mon/FIF...	@FIFO_transacti...
data_in	16'h3904
rst_n	1'h1
wr_en	1'h0
rd_en	1'h0
data_out	16'h3a73
wr_ack	1'h0
overflow	1'h0
full	1'h0
empty	1'h1
almostfull	1'h0
almostempty	1'h0
underflow	1'h0
RD_EN_ON_DIST	32'h0000001e
WR_EN_ON_DIST	32'h00000046
/FIFO_top/dut/mem	16'hf944 16'ha69...
Now	2002 ns
Cursor 1	336 ns


Wave - Default
Msgs
/FIFO_top/dut/count 4'h1
/FIFO_top/mon/FIF... @FIFO_scoreboa...
data_out_ref 16'h2819
wr_ack_ref 1'h0
overflow_ref 1'h0
full_ref 1'h0
empty_ref 1'h1
almostfull_ref 1'h0
almostempty_ref 1'h0
underflow_ref 1'h1
count 32'h00000000
/FIFO_top/mon/FIF... @FIFO_transacti...
data_in 16'hdca1
rst_n 1'h1
wr_en 1'h0
rd_en 1'h1
data_out 16'h2819
wr_ack 1'h0
overflow 1'h0
full 1'h0
empty 1'h1
almostfull 1'h0
almostempty 1'h0
underflow 1'h1
RD_EN_ON_DIST 32'h0000001e
WR_EN_ON_DIST 32'h00000046
/FIFO_top/dut/mem 16'he84f 16'h46c...
Now 2002 ns
Cursor 1 1929 ns
1916 ns
1920 ns
1924 ns
1928 ns
1929 ns


Wave - Default
Msgs
/FIFO_top/dut/count 4'h5
/FIFO_top/mon/FIF... @FIFO_scoreboa...
data_out_ref 16'ha2c5
wr_ack_ref 1'h1
overflow_ref 1'h0
full_ref 1'h0
empty_ref 1'h0
almostfull_ref 1'h0
almostempty_ref 1'h0
underflow_ref 1'h0
count 32'h00000004
/FIFO_top/mon/FIF... @FIFO_transacti...
data_in 16'h8a0e
rst_n 1'h1
wr_en 1'h1
rd_en 1'h0
data_out 16'ha2c5
wr_ack 1'h1
overflow 1'h0
full 1'h0
empty 1'h0
almostfull 1'h0
almostempty 1'h0
underflow 1'h0
RD_EN_ON_DIST 32'h0000001e
WR_EN_ON_DIST 32'h00000046
/FIFO_top/dut/mem 16'he84f 16'h8a0...



```
# Loading work.FIFO_coverage_package(fast)
# Loading work.FIFO_monitor(fast)
#                  2002: at end of test error count is 0 and correct count = 1001
# ** Note: $stop    : FIFO_monitor.sv(53)
#    Time: 2002 ns  Iteration: 1  Instance: /FIFO_top/mon
# Break in Module FIFO_monitor at FIFO_monitor.sv line 53

VSIM 5>
```