

## -----monitor-----

```

30 //TO DO : Complete the run_phase method
31 task run_phase(uvm_phase phase);
32     forever begin
33         pkt_len=0;
34         //TO DO : Wait for rising edge of htax_rx_intf.rx_sot
35         wait(!htax_rx_intf.rx_sot);
36
37         //TO DO : On consecutive clock cycles append htax_rx_intf.rx_data to rx_mon_packet.data[] till htax_rx_intf.rx_eot pulse
38         while(htax_rx_intf.rx_eot == 0) begin //TO DO : Replace XXX with appropriate condition in while loop
39             @(posedge htax_rx_intf.clk);
40             rx_mon_packet.data = new[++pkt_len](rx_mon_packet.data);
41             rx_mon_packet.data[pkt_len - 1] = htax_rx_intf.rx_data;
42         end
43         //TO DO : Write the rx_mon_packet on analysis port
44         rx_collect_port.write(rx_mon_packet);
45     end
46 end
47 endtask : run_phase

```

## -----scoreboard-----

```

74 //TO DO : Complete the code for push_to_queue function -- preferably use the SV-Case statement
75 //Add incoming TX Monitor packet to corresponding queue from mon_pkt.dest_port
76 function void push_to_queue(htax_tx_mon_packet_c mon_pkt);
77     case (mon_pkt.dest_port)
78         0: port0_queue.push_back(mon_pkt);
79         1: port1_queue.push_back(mon_pkt);
80         2: port2_queue.push_back(mon_pkt);
81         3: port3_queue.push_back(mon_pkt);
82         default: `uvm_fatal("SCOREBOARD", $sformatf("Invalid dest_port value: %0d", mon_pkt.dest_port))
83     endcase
84 endfunction : push_to_queue

```

```

99 //TO DO : Write Method - RX[1] Monitor
100 function void write_rx1_export(htax_rx_mon_packet_c rx_mon_packet);
101     //TO DO : Create a new cmp_pkt[i]
102     // Pop the last element from queue i assign it to cmp_pkt[i]
103     // Compare the data field of cmp_pkt[i] and rx_mon_packet; Mismatch results into fatal error
104 `uvm_info("SCOREBOARD", $sformatf("Received Data Packet on Port1:"), UVM_NONE)
105     rx_mon_packet.print();
106     cmp_pkt[1] = new();
107     cmp_pkt[1] = port1_queue.pop_back();
108     if (cmp_pkt[1].data == rx_mon_packet.data)
109         `uvm_info("SCOREBOARD", "Data matches for received pkt on port 1", UVM_NONE)
110     else
111         `uvm_fatal("SCOREBOARD", "Data mismatch for received pkt on port 1")
112     `uvm_info("SCOREBOARD", "Dropping pkt from queue 1", UVM_NONE)
113 endfunction : write_rx1_export
114
115 //TO DO : Write Method - RX[2] Monitor
116 function void write_rx2_export(htax_rx_mon_packet_c rx_mon_packet);
117     //TO DO : Create a new cmp_pkt[i]
118     // Pop the last element from queue i assign it to cmp_pkt[i]
119     // Compare the data field of cmp_pkt[i] and rx_mon_packet; Mismatch results into fatal error
120 `uvm_info("SCOREBOARD", $sformatf("Received Data Packet on Port2:"), UVM_NONE)
121     rx_mon_packet.print();
122     cmp_pkt[2] = new();
123     cmp_pkt[2] = port2_queue.pop_back();
124     if (cmp_pkt[2].data == rx_mon_packet.data)
125         `uvm_info("SCOREBOARD", "Data matches for received pkt on port 2", UVM_NONE)
126     else
127         `uvm_fatal("SCOREBOARD", "Data mismatch for received pkt on port 2")
128     `uvm_info("SCOREBOARD", "Dropping pkt from queue 2", UVM_NONE)
129 endfunction : write_rx2_export

```

```

132 function void write_rx3_export(htax_rx_mon_packet_c rx_mon_packet);
133     //TO DO : Create a new cmp_pkt[i]
134     // Pop the last element from queue i assign it to cmp_pkt[i]
135     // Compare the data field of cmp_pkt[i] and rx_mon_packet; Mismatch results into fatal error
136 `uvm_info("SCOREBOARD", $sformatf("Received Data Packet on Port3:"), UVM_NONE)
137     rx_mon_packet.print();
138     cmp_pkt[3] = new();
139     cmp_pkt[3] = port3_queue.pop_back();
140     if (cmp_pkt[3].data == rx_mon_packet.data)
141         `uvm_info("SCOREBOARD", "Data matches for received pkt on port 3", UVM_NONE)
142     else
143         `uvm_fatal("SCOREBOARD", "Data mismatch for received pkt on port 3")
144     `uvm_info("SCOREBOARD", "Dropping pkt from queue 3", UVM_NONE)
145 endfunction : write_rx3_export

```

```

Name                               Type                               Size  Value
-----
htax_tx_mon_packet_c               htax_tx_mon_packet_c              -      @5638
dest_port                         integral                          32      'h1
data                               da(integral)                      4      -
[0]                               integral                          64      'hec5ad14a77aa6b3e
[1]                               integral                          64      'h1798df7add7dde8a
[2]                               integral                          64      'hc8194eda20a15a07
[3]                               integral                          64      'h6bc701187cb9b52d

UVM_INFO ../tb/htax_tx_driver_c.sv(115) @ 13030000: uvm_test_top.tb.tx_port[1].tx_driver [htax_tx_driver_c] Ended Driving Data Packet to DUT
UVM_INFO ../tb/htax_seqs.sv(25) @ 13030000: uvm_test_top.tb.tx_port[1].sequencer@simple_random_seq [simple_random_seq] drop_objection
UVM_INFO ../tb/htax_scoreboard_c.sv(104) @ 13050000: uvm_test_top.tb.htax_sb [SCOREBOARD] Received Data Packet on Port1:

Name                               Type                               Size  Value
-----
htax_rx_mon_packet_c               htax_rx_mon_packet_c              -      @3722
data                               da(integral)                      4      -
[0]                               integral                          64      'hec5ad14a77aa6b3e
[1]                               integral                          64      'h1798df7add7dde8a
[2]                               integral                          64      'hc8194eda20a15a07
[3]                               integral                          64      'h6bc701187cb9b52d

UVM_INFO ../tb/htax_scoreboard_c.sv(109) @ 13050000: uvm_test_top.tb.htax_sb [SCOREBOARD] Data matches for received pkt on port 1
UVM_INFO ../tb/htax_seqs.sv(112) @ 13050000: uvm_test_top.tb.htax_sb [SCOREBOARD] Dropping pkt from queue 1
UVM_INFO /opt/coe/cadence/XCELIUM/tools/methodology/UVM/CDNS-1.1d/sv/src/base/uvm_objection.svh(1268) @ 63030000: reporter [TEST_DONE] 'run'
phase is ready to proceed to the 'extract' phase
UVM_INFO ../tb/htax_scoreboard_c.sv(148) @ 63030000: uvm_test_top.tb.htax_sb [SCOREBOARD] End of Simulation Checking
[The port0_queue.size is:] 0
[The port1_queue.size is:] 0
[The port2_queue.size is:] 0
[The port3_queue.size is:] 0
UVM_INFO ../tb/htax_scoreboard_c.sv(154) @ 63030000: uvm_test_top.tb.htax_sb [SCOREBOARD] Port 0 Queue is empty
UVM_INFO ../tb/htax_scoreboard_c.sv(158) @ 63030000: uvm_test_top.tb.htax_sb [SCOREBOARD] Port 1 Queue is empty
UVM_INFO ../tb/htax_scoreboard_c.sv(162) @ 63030000: uvm_test_top.tb.htax_sb [SCOREBOARD] Port 2 Queue is empty
UVM_INFO ../tb/htax_scoreboard_c.sv(166) @ 63030000: uvm_test_top.tb.htax_sb [SCOREBOARD] Port 3 Queue is empty

```

```

--- UVM Report catcher Summary ---

Number of demoted UVM_FATAL reports : 0
Number of demoted UVM_ERROR reports : 0
Number of demoted UVM_WARNING reports: 0
Number of caught UVM_FATAL reports : 0
Number of caught UVM_ERROR reports : 0
Number of caught UVM_WARNING reports : 0

--- UVM Report Summary ---

** Report counts by severity
UVM_INFO : 92
UVM_WARNING : 0
UVM_ERROR : 0
UVM_FATAL : 0
** Report counts by id
[RNTST] 1
[SCOREBOARD] 50
[TEST_DONE] 1
[TOP] 5
[UVMTOP] 1
[htax_tx_driver_c] 30
[simple_random_seq] 3
[simple_random_test] 1
Simulation complete via $finish(1) at time 63030 NS + 45
/opt/coe/cadence/XCELIUM/tools/methodology/UVM/CDNS-1.1d/sv/src/base/uvm_root.svh:457 $finish;
xcelium> exit

coverage setup:
workdir : ./cov_work
dutinst : top(top)
scope : scope
testname : test

coverage files:
model(design data) : ./cov_work/scope/icc_1cfd8b_4389b1cd.ucm (reused)

Simulation complete via $finish(1) at time 63030 NS + 45
/opt/coe/cadence/XCELIUM/tools/methodology/UVM/CDNS-1.1d/sv/src/base/uvm_root.svh:457 $finish;
xcelium> exit

coverage setup:
workdir : ./cov_work
dutinst : top(top)
scope : scope
testname : test

coverage files:
model(design data) : ./cov_work/scope/icc_1cfd8b_4389b1cd.ucm (reused)
data : ./cov_work/scope/test/icc_1cfd8b_4389b1cd.uct
TOOL: xrun 22.03-s012: Exiting on Nov 10, 2024 at 14:37:05 CST (total: 00:00:05)
[harrison900531@n04-hades sim]$

```