

```

54 //TO DO : COMPLETE THE DRIVER CODE BELOW
55 task htax_tx_driver_c::drive_thru_dut(htax_packet_c pkt);
56
57     `uvm_info (get_type_name(), $sformatf("Input Data Packet to DUT : \n%s", pkt.sprint()),UVM_NONE)
58
59     //Wait for pkt.delay clock cycles
60     repeat (pkt.delay) @(posedge htax_tx_intf.clk);
61
62     //On next clk-posedge
63     //TO DO : Set htax_tx_intf.tx_outport_req in one-hot fashion from pkt.dest_port
64     // Ex dest_port = 2 and PORTS = 4, tx_outport_req = 4'b0100
65     @(posedge htax_tx_intf.clk)
66     htax_tx_intf.tx_outport_req = (1 << pkt.dest_port) & ((1 << PORTS) - 1);
67
68     //TO DO : Assign htax_tx_intf.tx_vc_req from pkt.vc
69     htax_tx_intf.tx_vc_req = pkt.vc;
70
71     //TO DO : Wait till htax_tx_intf.tx_vc_gnt is received
72     wait (htax_tx_intf.tx_vc_gnt);
73
74     //On next clk-posedge
75     //TO DO : Set any one bit tx_sot[vc-1:0] to 1 from the ones granted by htax_tx_intf.tx_vc_gnt
76     // (refer to SPEC doc for more details)
77     @(posedge htax_tx_intf.clk)
78     htax_tx_intf.tx_sot = htax_tx_intf.tx_vc_gnt;
79
80     //TO DO : Drive pkt.data[0] on htax_tx_intf.tx_data
81     htax_tx_intf.tx_data = pkt.data[0];
82
83     //TO DO : Reset htax_tx_intf.tx_outport_req and htax_tx_intf.tx_vc_req (to zero)
84     htax_tx_intf.tx_outport_req = 'b0;
85     htax_tx_intf.tx_vc_req = 'b0;
86
87     //TO DO : On consecutive clk-posedges drive each of the packet's pkt.data on htax_tx_intf.tx_data
88     //TO DO : Assign htax_tx_intf.tx_sot to zero after first cycle
89     //TO DO : Assert htax_tx_intf.tx_release_gnt for one clock cycle when driving second last data packet
90     //TO DO : Assert htax_tx_intf.tx_eot for one clock cycle when driving last data packet
91     for(int i = 1; i < pkt.length; i++) begin //TO DO : Replace XXX and YYY with appropriate values for a packet
92         @(posedge htax_tx_intf.clk);
93         // Drive current data packet
94         htax_tx_intf.tx_data = pkt.data[i];
95         // Reset tx_sot after first cycle
96         if(i==1)
97             htax_tx_intf.tx_sot = 'b0;
98         // Assert tx_release_gnt on second last data cycle
99         if(i==pkt.length-2)
100             htax_tx_intf.tx_release_gnt = 'b1;
101         else
102             htax_tx_intf.tx_release_gnt = 'b0;
103         // Assert tx_eot on last data cycle
104         if(i==pkt.length-1)
105             htax_tx_intf.tx_eot = 'b1;
106         else
107             htax_tx_intf.tx_eot = 'b0;
108     end
109
110     //On next clk-posedge
111     //TO DO : Assign htax_tx_intf.tx_data to X and htax_tx_intf.tx_eot to zero
112     @(posedge htax_tx_intf.clk)
113     htax_tx_intf.tx_data = 'x;
114     htax_tx_intf.tx_eot = 'b0;
115
116     `uvm_info (get_type_name(), $sformatf("Ended Driving Data Packet to DUT"), UVM_NONE)
117 endtask : drive_thru_dut

```

```

35 //TO D0 : Complete the run_phase tasks with Hints below
36 task run_phase(uvm_phase phase);
37     forever begin
38         pkt_len=0;
39
40         @(posedge htax_tx_intf.tx_vc_gnt) begin
41             //TO D0 : Assign tx_mon_packet.dest_port from htax_tx_intf.tx_outport_req
42             tx_mon_packet.dest_port = $clog2(htax_tx_intf.tx_outport_req);
43
44         end
45
46         @(posedge htax_tx_intf.clk);
47         //TO D0 : On consecutive cycles append htax_tx_intf.tx_data to tx_mon_packet.data[] till htax_tx_intf.tx_eot pulse
48         while(!htax_tx_intf.tx_eot) begin // TO D0 : Replace XXX with appropriate condition
49             @(posedge htax_tx_intf.clk);
50             tx_mon_packet.data = new[pkt_len + 1](tx_mon_packet.data);
51             pkt_len++;
52             tx_mon_packet.data[pkt_len - 1] = htax_tx_intf.tx_data;
53         end
54         tx_mon_packet.print();
55     end
56 endtask : run_phase
57
58 endclass : htax_tx_monitor_c

```

```

--- UVM Report Summary ---

** Report counts by severity
UVM_INFO : 42
UVM_WARNING : 0
UVM_ERROR : 0
UVM_FATAL : 0
** Report counts by id
[RNTST] 1
[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 13030 NS + 51
/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.ucd
TOOL: xrun 22.03-s012: Exiting on Nov 03, 2024 at 13:34:31 CST (total: 00:00:03)

```

Name	Type	Size	Value
req	htax_packet_c	-	@6393
delay	integral	32	'h9
dest_port	integral	32	'h0
vc	integral	2	'h2
length	integral	32	'h21
data	da(integral)	33	-
[0]	integral	64	'h77ccbfc7098f2f32
[1]	integral	64	'h14286bcac537e62b
[2]	integral	64	'he0423c62b3cd1e5d
[3]	integral	64	'hab2405b4d9c47248
[4]	integral	64	'h6769fbc443edfee5
...
[28]	integral	64	'h76a973390d7e8d48
[29]	integral	64	'h1a5f97156b32a728
[30]	integral	64	'h3ce323f6e8536e9
[31]	integral	64	'h133c872a769e1887
[32]	integral	64	'h8cf250dfc03281ba
begin_time	time	64	10000
depth	int	32	'd2
parent sequence (name)	string	17	simple_random_seq
parent sequence (full name)	string	54	uvm_test_top.tb.tx_port[1].sequencer.simple_random_seq
sequencer	string	36	uvm_test_top.tb.tx_port[1].sequencer

Name	Type	Size	Value
htax_tx_mon_packet_c	htax_tx_mon_packet_c	-	@4568
dest_port	integral	32	'h0
data	da(integral)	33	-
[0]	integral	64	'h77ccbf7098f2f32
[1]	integral	64	'h14286bcac537e62b
[2]	integral	64	'he0423c62b3cd1e5d
[3]	integral	64	'hab2405b4d9c47248
[4]	integral	64	'h6769fbc443edfee5
...
[28]	integral	64	'h76a973390d7e8d48
[29]	integral	64	'h1a5f97156b32a728
[30]	integral	64	'h3ce323f6e8536e9
[31]	integral	64	'h133c872a769e1887
[32]	integral	64	'h8cf250dfc03281ba

UVM_INFO ../tb/htax_tx_driver_c.sv(116) @ 930000: uvm_test_top.tb.tx_port[1].tx_driver [htax_tx_driver_c] Ended Driving Data Packet to DUT

UVM_INFO ../tb/htax_tx_driver_c.sv(57) @ 930000: uvm_test_top.tb.tx_port[1].tx_driver [htax_tx_driver_c] Input Data Packet to DUT :

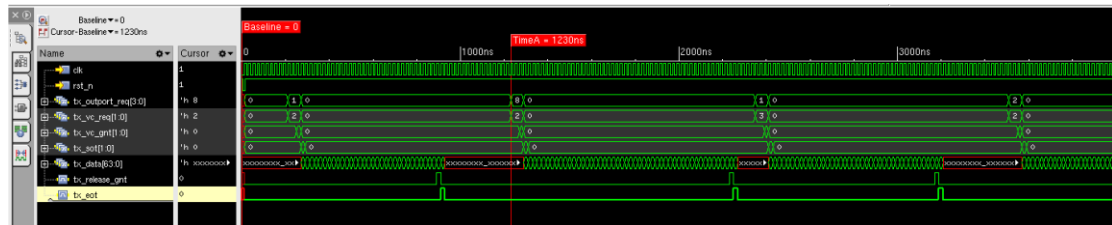
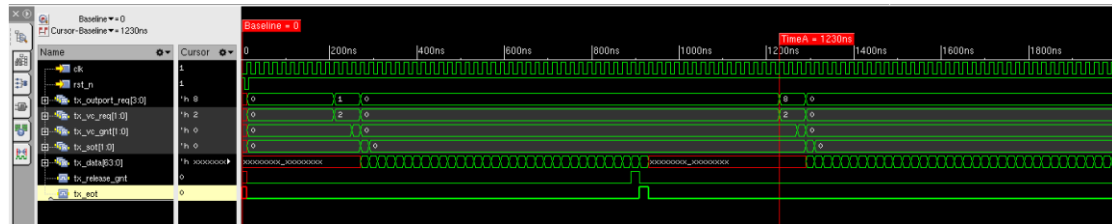
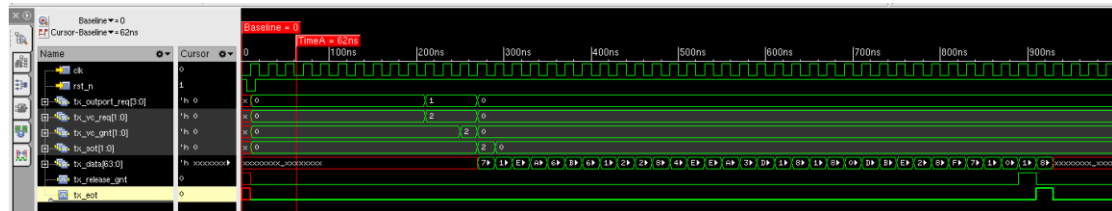
Name	Type	Size	Value
req	htax_packet_c	-	@6319
delay	integral	32	'he
dest_port	integral	32	'h3
ve	integral	2	'h2
length	integral	32	'h31
data	da(integral)	49	-
[0]	integral	64	'h814dd64c8ef0e04e
[1]	integral	64	'h14a07c540640547e
[2]	integral	64	'h28d0521fa44d94ae
[3]	integral	64	'hb5002dbf1502e46b
[4]	integral	64	'h566c637b166bbf2c
...
[44]	integral	64	'hb6b20ae4284ce242
[45]	integral	64	'h9d55ad57d2ad3c28
[46]	integral	64	'h8a5a24445d994829
[47]	integral	64	'hae0b7a116beeeaf3
[48]	integral	64	'hc17587c6f303de12
begin_time	time	64	930000
depth	int	32	'd2
parent sequence (name)	string	17	simple_random_seq
parent sequence (full name)	string	54	uvm_test_top.tb.tx_port[1].sequencer.simple_random_seq
sequencer	string	36	uvm_test_top.tb.tx_port[1].sequencer

Name	Type	Size	Value
htax_tx_mon_packet_c	htax_tx_mon_packet_c	-	@4568
dest_port	integral	32	'h3
data	da(integral)	49	-
[0]	integral	64	'h814dd64c8ef0e04e
[1]	integral	64	'h14a07c540640547e
[2]	integral	64	'h28d0521fa44d94ae
[3]	integral	64	'hb5002dbf1502e46b
[4]	integral	64	'h566c637b166bbf2c
...
[44]	integral	64	'hb6b20ae4284ce242
[45]	integral	64	'h9d55ad57d2ad3c28
[46]	integral	64	'h8a5a24445d994829
[47]	integral	64	'hae0b7a116beeeaf3
[48]	integral	64	'hc17587c6f303de12

Name	Type	Size	Value
req	htax_packet_c	-	@6256
delay	integral	32	'h11
dest_port	integral	32	'h1
vc	integral	2	'h3
length	integral	32	'h4
data	da(integral)	4	-
[0]	integral	64	'hec5ad14a77aa6b3e
[1]	integral	64	'h1798df7add7dde8a
[2]	integral	64	'hc8194eda20a15a07
[3]	integral	64	'h6bc701187cb9b52d
begin_time	time	64	12530000
depth	int	32	'd2
parent sequence (name)	string	17	simple_random_seq
parent sequence (full name)	string	54	uvm_test_top.tb.tx_port[1].sequencer.simple_random_seq
sequencer	string	36	uvm_test_top.tb.tx_port[1].sequencer

Name	Type	Size	Value
htax_tx_mon_packet_c	htax_tx_mon_packet_c	-	@4568
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(116) @ 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 /opt/cadence/XCELIUM/tools/methodology/UVM/CDNS-1.1d/sv/src/base/uvm_objection.svh(1268) @ 13030000: reporter [TEST_DONE] 'run'
phase is ready to proceed to the 'extract' phase



Name	Type	Size	Value
uvm_test_top	simple_random_test	-	@2723
tb	htax_env	-	@2793
tx_port[0]	htax_tx_agent_c	-	@2854
sequencer	htax_sequencer_c	-	@3114
rsp_export	uvm_analysis_export	-	@3173
seq_item_export	uvm_seq_item_pull_imp	-	@3721
arbitration_queue	array	0	-
lock_queue	array	0	-
num_last_reqs	integral	32	'd1
num_last_rsp	integral	32	'd1
tx_driver	htax_tx_driver_c	-	@2979
rsp_port	uvm_analysis_port	-	@3083
seq_item_port	uvm_seq_item_pull_port	-	@3032
tx_monitor	htax_tx_monitor_c	-	@3780
is_active	uvm_active_passive_enum	1	UVM ACTIVE
tx_port[1]	htax_tx_agent_c	-	@2884
sequencer	htax_sequencer_c	-	@3944
rsp_export	uvm_analysis_export	-	@3999
seq_item_export	uvm_seq_item_pull_imp	-	@4539
arbitration_queue	array	0	-
lock_queue	array	0	-
num_last_reqs	integral	32	'd1
num_last_rsp	integral	32	'd1
tx_driver	htax_tx_driver_c	-	@3818
rsp_port	uvm_analysis_port	-	@3914
seq_item_port	uvm_seq_item_pull_port	-	@3866
tx_monitor	htax_tx_monitor_c	-	@4569
is_active	uvm_active_passive_enum	1	UVM ACTIVE
tx_port[2]	htax_tx_agent_c	-	@2914
sequencer	htax_sequencer_c	-	@4753
rsp_export	uvm_analysis_export	-	@4808
seq_item_export	uvm_seq_item_pull_imp	-	@5348
arbitration_queue	array	0	-
lock_queue	array	0	-
num_last_reqs	integral	32	'd1
num_last_rsp	integral	32	'd1
tx_driver	htax_tx_driver_c	-	@4627
rsp_port	uvm_analysis_port	-	@4723