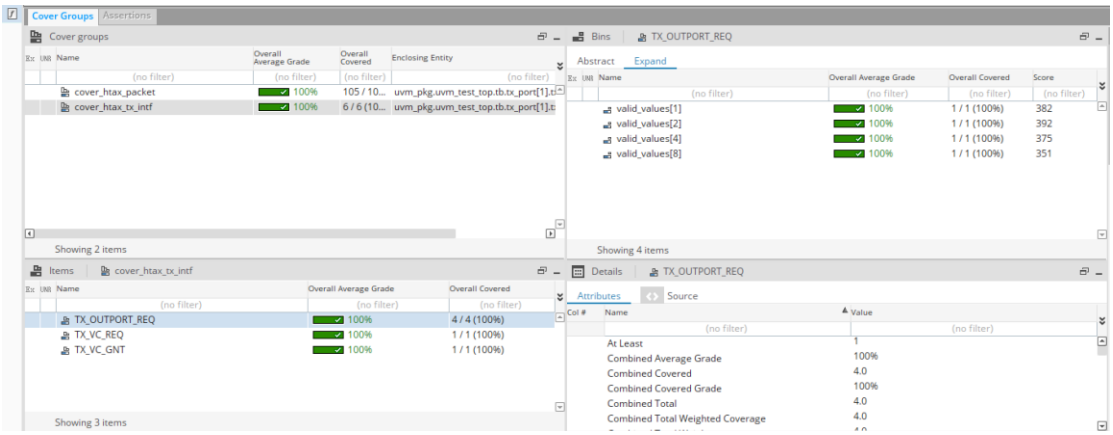
----------TX----------

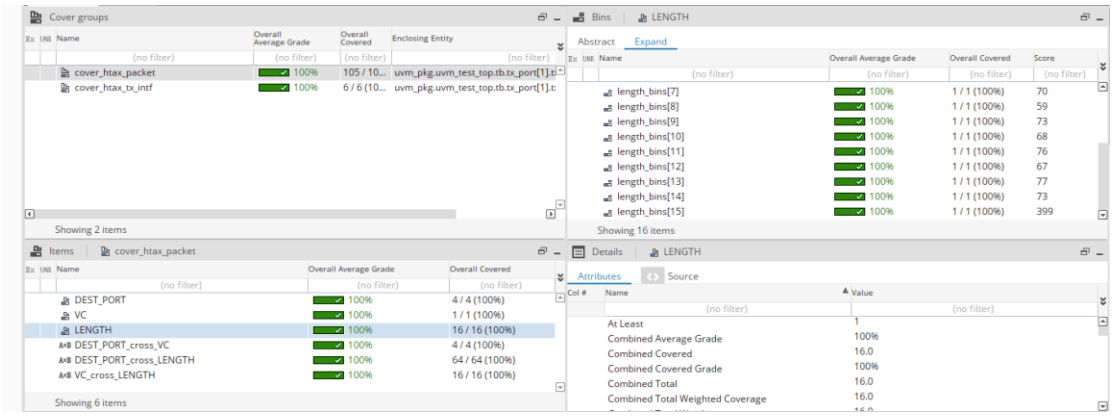
```
covergroup cover_htax_packet;
  option.per_instance = 1;
  option.name = "cover_htax_packet";


  // Coverpoint for htax packet field : destination port
  DEST_PORT : coverpoint tx_mon_packet.dest_port  {
      bins dest_port[] = {[0:3]};
  }

  // TO DO : Coverpoint for htax packet field : vc (include vc=0 in illegal bin)
      //VC :
  VC : coverpoint tx_mon_packet.vc {
      bins valid_vc[] = {[1:`VC-1]};  // 合法 VC 值
      illegal_bins illegal_vc = {0};  // VC=0 是非法值
  }


  // TO DO : Coverpoint for htax packet field : length (Divide range [3:63] into 16 bins)
      //LENGTH :
  LENGTH : coverpoint tx_mon_packet.length {
      bins length_bins[16] = {[3:7], [8:12], [13:17], [18:22],
                              [23:27], [28:32], [33:37], [38:42],
                              [43:47], [48:52], [53:57], [58:63]};
  }
```

```
  // Coverpoints for Crosses
  // TO DO : DEST_PORT cross VC
  DEST_PORT_cross_VC : cross DEST_PORT, VC;

  // TO DO : DEST_PORT cross LENGTH
  DEST_PORT_cross_LENGTH : cross DEST_PORT, LENGTH;

  // TO DO : VC cross LENGTH
  VC_cross_LENGTH : cross VC, LENGTH;

endgroup

  covergroup cover_htax_tx_intf;
  option.per_instance = 1;
  option.name = "cover_htax_tx_intf";

      // TO DO : Coverpoint for tx_outport_req: covered all the values 0001,0010,0100,1000
      TX_OUTPORT_REQ : coverpoint htax_tx_intf.tx_outport_req {
          bins valid_values[] = {4'b0001, 4'b0010, 4'b0100, 4'b1000};
      }
      // TO DO : Coverpoint for tx_vc_req: All the VCs are requested atleast once. Ignore what is not allowed, or put it as illegal
      TX_VC_REQ : coverpoint htax_tx_intf.tx_vc_req {
          bins valid_vc_req[] = {[1:`VC-1]};  // 合法 VC 值
          illegal_bins illegal_vc_req = {0}; // 非法 VC 请求
      }
      // TO DO : Coverpoint for tx_vc_gnt: All the virtual channels are granted atleast once.
      TX_VC_GNT : coverpoint htax_tx_intf.tx_vc_gnt {
          bins valid_vc_gnt[] = {[1:`VC-1]};  // 合法 VC 授予
          illegal_bins illegal_vc_gnt = {0}; // 非法 VC 授予
      }

  endgroup
```

----------RX----------

```
    // TO DO : Create covergroup for htax_rx_inf and add at least one coverpoint
    covergroup cover_htax_packet;
    option.per_instance = 1;
    option.name = "cover_htax_packet";

    data0 : coverpoint rx_mon_packet.data[0] {
        bins value[3] = {[0:127], [128:256], [256:$]}; // 具體 bins 根據需求定義
    }

    endgroup
```






```
--- 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 : 9017
UVM_WARNING :    0
UVM_ERROR :    0
UVM_FATAL :    0
** Report counts by id
[RNTST]     1
[SCOREBOARD]  6005
[TEST_DONE]     1
[TOP]     5
[UVMTOP]     1
[htax_tx_driver_c]  3000
[simple_random_seq]     3
[simple_random_test]     1
Simulation complete via $finish(1) at time 1475710 NS + 45
/opt/coe/cadence/XCELIUM/tools/methodology/UVM/CDNS-1.1d/sv/src/base/uvm_root.svh:457     $finish;
xcelium> exit
```