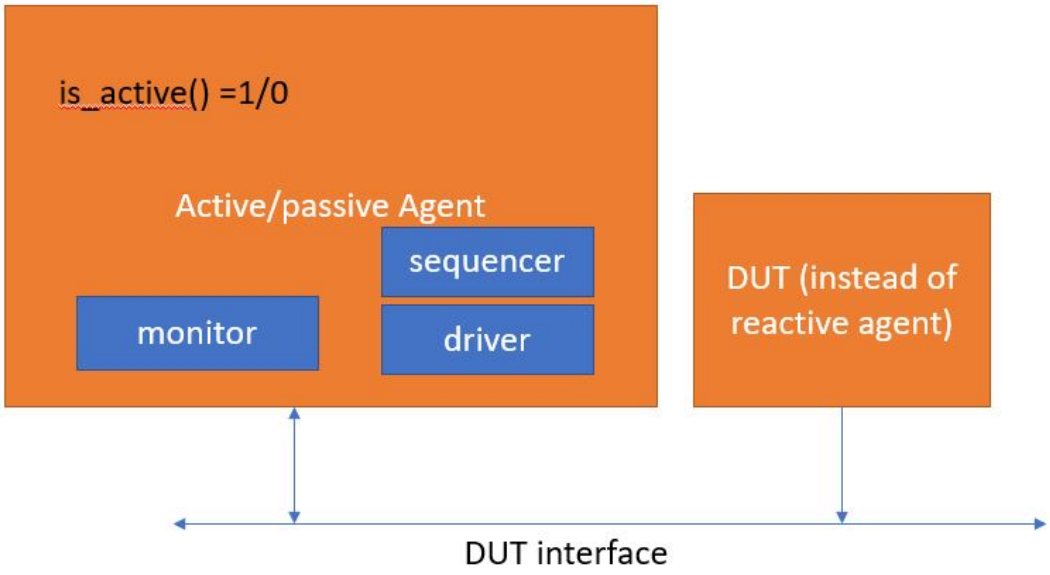
ASIC Verification Assignment Solution



Agent can be configured in env in build_phase by setting the "is_active" == 1.

Full testbench code running bringup test (dummy delay sequence)

https://www.edaplayground.com/x/CzGt

Test can be passed in run options by passing +UVM_TESTNAME= "any test name from test_lib.sv"

Assumptions made for the DUT-

1) DUT absorbs all the data and control signals that are given to it without raising/asserting any error signal.

2) DUT is always ready to accept given data.

Assumptions made for the TB-

1) **The DUT mimics the reactive agent.**
2) **The monitor and driver of active and passive agent has virtual interface handle of physical interface. So driver and monitor can drive and monitor subsequent data.**

TestPlan:

1) **Reset Scenario- While submitting random data, give reset and check whether all the control and data signals are resetted properly or not.**
2) **Non Reset Scenario-**
   i) Send Different combination of packets-
   -Send Message passthrough data with message type "MASS_QUOTE, HEARTBEAT, RESERVED" and random data from 128 to 1017 bytes.
   -Send Register write data with modes "NORMAL" and "BURST" and send random normal or burst mode data from 1 to 1016 bytes.
3) **Assert in_error = 1 in the incoming data packet and check whether in_ready of DUT is getting asserted or not.**

Knobs for controlling VIP -

1) For controlling byte_0 of packet- `uvm_do_with(… req. pkt_transition_type_e == MESSAGE_PASSTHROUGH/REGISTER UPDATE/RESERVED_TX)

2) For controlling transaction header message type- `uvm_do_with(… req.msg_type_e == MASS_QUOTE/HEARTBEAT/RESERVED_MX_TYPE)

3) For controlling register write data format mode - `uvm_do_with(… req.mode_ee == NORMAL/BURST)

4) For controlling burst length of message pass through data - `uvm_do_with(…req.length_for_msg_pass_thru inside [128:1017]

5)For controlling burst length of register write data – `uvm_do_with(…req.length_for_reg_wr inside [1:1016]

6) Activeness/passiveness of agent can be controlled like line 18 of env.sv or by creating a separate agent configuration object