**Module: SV for Verification**

**Section:** Threads & Interprocess Communication **Task:** Mailboxes

**Task**

Mailboxes

* **Code:**

task generator\_bad (input int n, input mailbox #(Transaction) mbx);

Transaction tr;

tr = new();

repeat ( n ) begin

SV\_RAND\_CHECK(tr.randomize());

$display("GEN: Sending addrOsh", tr.addr); mbx.put(tr);

end

endtask

* **Issues:**

The issue with this code lies in the object **tr**, which is created **once** before the **repeat** loop and reused inside the loop. As a result, the same **Transaction** object is randomized multiple times, and only the final value of **tr** will be in the mailbox multiple times. This means that multiple references to the same **Transaction** object are put into the mailbox, which can cause undesired behavior when retrieved.

**Fix:**

To fix this, you need to create a **new object** inside the loop for each iteration, ensuring each randomized transaction is independent.

* **Fixed Code:**

task generator\_good (input int n, input mailbox #(Transaction) mbx);

Transaction tr;

repeat (n) begin

tr = new(); // Create a new Transaction object inside the loop

SV\_RAND\_CHECK(tr.randomize());

$display("GEN: Sending addr = %0h", tr.addr);

mbx.put(tr); // Place the new object into the mailbox

end

endtask