**Module: SV for Verification**

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

**Task 1 -** [**EDA Link**](https://www.edaplayground.com/x/LAyi)

Events

* **Code:**

The code has some syntactical errors, after resolving the issues, the final code will look like as follows:

// Author: Noman Rafiq

// Dated: Sep 09, 2024

program automatic events;

event e1, e2;

task trigger(event local\_event, input time wait\_time);

#wait\_time;

->local\_event;

endtask

initial begin

fork

trigger(e1, 10ns);

begin

wait(e1.triggered);

$display("%0t: e1 triggered", $time);

end

join

end

initial begin

fork

trigger(e2, 20ns);

begin

wait(e2.triggered);

$display("%0t: e2 triggered", $time);

end

join

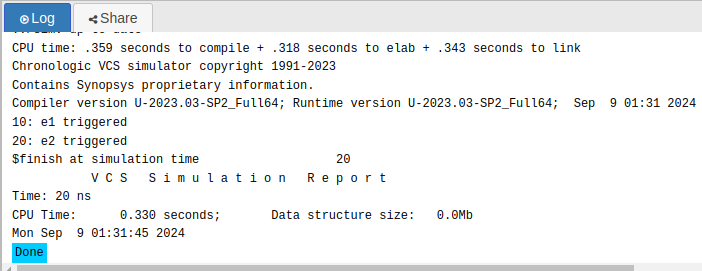
end

endprogram

* **Expected Output:**

This SystemVerilog code demonstrates the use of events and concurrent execution using the **fork-join** construct. Two events, **e1** and **e2**, are triggered after delays of 10ns and 20ns respectively, using the **trigger** task. The code waits for each event to be triggered before printing a message indicating the time at which the event occurred. The expected output would be that at 10ns, the message for **e1** is printed, and at 20ns, the message for **e2** is printed, showing concurrent event handling and synchronization.

* **Actual Output:**

****