Lab 03: Understanding UVM Phases

Objective: To understand and implement build_phase, connect_phase, and run_phase in UVM.

Step 1: Create the Driver Class

- Create a file named **driver.sv**.
- Extend the class from uvm_driver.
- Implement:
    - build_phase() (if needed).
    - connect_phase() to display a message.
    - run_phase() to print "Driver Run Phase Executed".

Step 2: Create the Monitor Class

- Create a file named **monitor.sv**.
- Extend the class from uvm_monitor.
- Implement:
    - build_phase() (if needed).
    - connect_phase() to display a message.
    - run_phase() to print "Monitor Run Phase Executed".

Step 3: Create the Environment Class

- Create a file named **env.sv**.
- Extend the class from uvm_env.
- Instantiate the driver and monitor components.
- Implement:
    - build_phase() to create instances using type_id create.
    - connect_phase() to connect components.

    - run_phase() to print "Env Run Phase Executed".

Step 4: Create the Test Class

- Create a file named **test.sv**.
- Extend the class from uvm_test.
- Instantiate the env component.
- Implement:
    - build_phase() to create the environment instance.
    - connect_phase() to print "Test Connect Phase Executed".
    - run_phase() to print "Test Run Phase Executed".

Step 5: Create the Top-Level Module

- Create a file named **tb.sv**.
- Include all created files.
- Use an initial block to call run_test("test").
- Run and observe the output.