DAY-90
#100DAYSRTL

"UVM: Introduction"

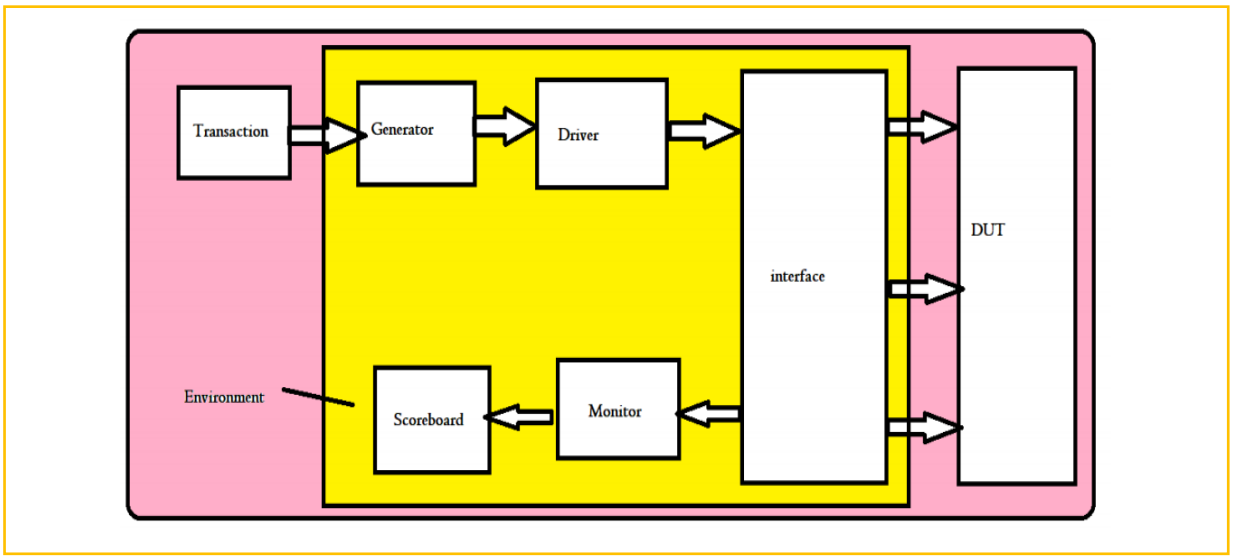
"Introduction":-

- The Accellera Universal Verification Methodology (UVM) standard verification methodology includes a set of class libraries for developing a verification environment.
- UVM is based on Open Verification Methodology (OVM) and Verification Methodology Manual (VVM).
- In System Verilog we build our verification code from Scratch but in UVM we do have some base classes so that we can extend them and build our verification environment
- The UVM API (Application Programming Interface) provides standardization for integration and creation of verification components.

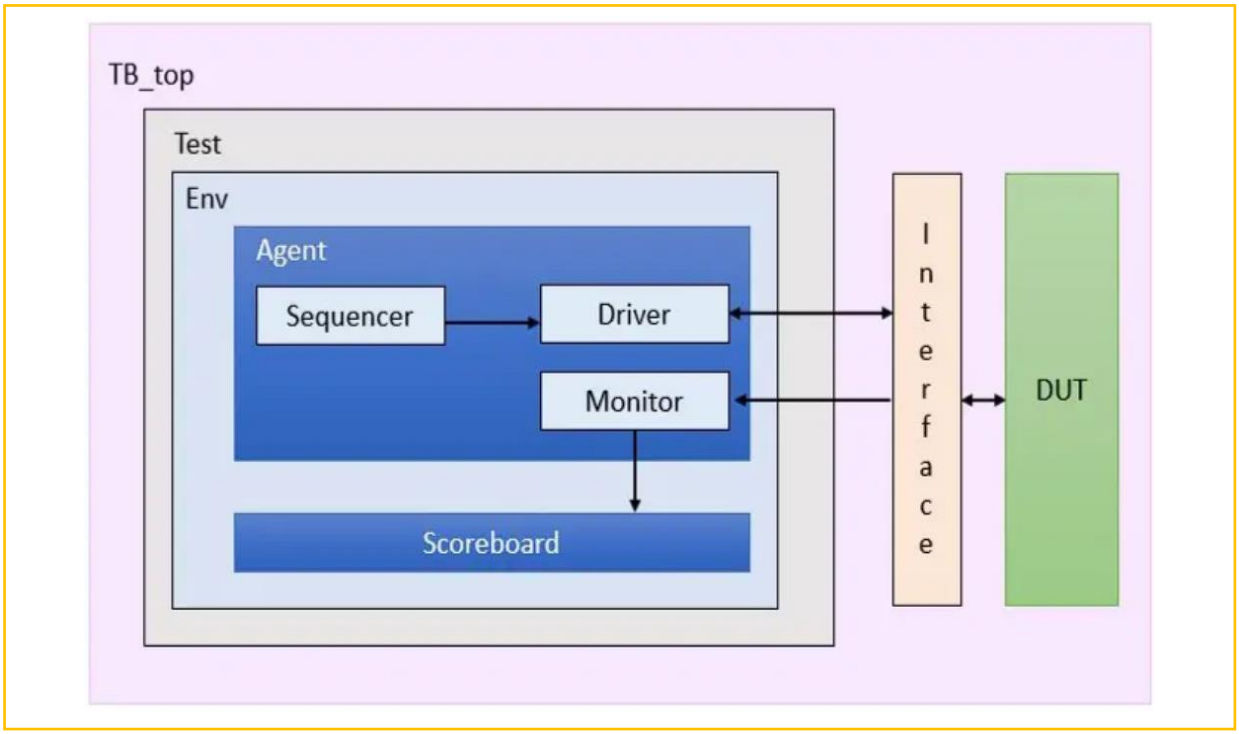
"Advantages of UVM":-

- UVM methodology provides scalable, reusable, and interoperable testbench development.
- To have uniformity in the testbench structure across the verification team, UVM provides guidelines for testbench development.
- UVM provides base class libraries so that users can inherit them to use inbuilt functionality.
- The driver-sequencer communication mechanism is an inbuilt mechanism in UVM that reduces verification efforts for the connection.
- UVM also provides verbosity to control message displays.

System Verilog TB:-



UVM TB:-



- These are two mandatory lines to start the uvm code in a console

```
`include "uvm_macros.svh"
import uvm_pkg::*;
```

"Hello World Program":-

```
`include "uvm_macros.svh"
import uvm_pkg::*;
module tb;
  initial begin
    `uvm_info("Chinnu","Hello World",UVM_MEDIUM);
  end
endmodule
```