DAY-91
#100DAYSRTL

"UVM: Reporting Macros"

"Introduction":-

In Universal Verification Methodology (UVM), reporting macros are used to generate messages or log information during simulation. These macros are defined in the **"uvm_macros.svh"** file, and they provide a convenient way to report information at different levels of verbosity.

"Diffrenent UVM Macros":-

- **UVM_INFO:-**Informative Messaage
- **UVM_WARNING:-**Indicates a potential problem
- **UVM_ERROR:-**Indicates a real problem.simulation continues subject to the configured message action
- **UVM_FATAL:-**Indicates a critical problem from which the simulation cannot recover. Simulation exists via $finish after a #0 delay.

"Code Practising":-

```
`include "uvm_macros.svh"
import uvm_pkg::*;
module tb;
initial begin
   `uvm_info("TB_TOP", "This is Informative message", UVM_MEDIUM)
    #10;
   `uvm_warning("TB_TOP", "This is Warning")
    #10;
   `uvm_error("TB_TOP", "This is error")
    #10;
   `uvm_fatal("TB_TOP", "This is fatal error, stopping simulation")
end
endmodule
```

“Result”:-

```
UVM_INFO /home/runner/testbench.sv(5) @ 0: reporter [TB_TOP] This is Informative message
UVM_WARNING /home/runner/testbench.sv(7) @ 10: reporter [TB_TOP] This is Warning
UVM_ERROR /home/runner/testbench.sv(9) @ 20: reporter [TB_TOP] This is error
UVM_FATAL /home/runner/testbench.sv(11) @ 30: reporter [TB_TOP] This is fatal error, stopping simulation
UVM_INFO /home/build/vlib1/vlib/uvm-1.2/src/base/uvm_report_server.svh(869) @ 30: reporter [UVM/REPORT/SERVER]
--- UVM Report Summary ---

** Report counts by severity
UVM_INFO :    2
UVM_WARNING :    1
UVM_ERROR :    1
UVM_FATAL :    1
** Report counts by id
[TB_TOP]     4
[UVM/RELNOTES]     1
```

“display() Vs `uvm_info”:-

```
`include "uvm_macros.svh"
import uvm_pkg::*;
module tb;
initial begin
   `uvm_info("TB_TOP","uvm_info", UVM_MEDIUM)
    #10;
  $display("display");#10;
end
endmodule
```

- **UVM_Info :-**
  - It gives the type of the reporting macro
  - It gives the specific file that is sending
  - It gives the line where uvm_info mentioned
  - It gives the time at which it is triggered
  - It gives the path of the component in a hierarchy
  - It gives the Id and message
- **Display:-**
  - It gives the only message
- If you want to get all the things like in uvm_info Then you need to give them manually

“Working With Verbosity”:-

- Fundamentally the Verbosity level describes how verbose a Testbench can be.
- The default Verbosity is UVM_MEDIUM.
- Different Verbosity levels are being supported by UVM.
- These are UVM_NONE, UVM_LOW, UVM_MEDIUM (Default), UVM_HIGH, UVM_FULL, UVM_DEBUG

Verbosity Level	Value
UVM_NONE	0
UVM_LOW	100
UVM_MEDIUM	200
UVM_HIGH	300
UVM_FULL	400
UVM_DEBUG	500

- To get the verbosity level we should“uvm_top.get_report_ verbosity _level()”

“Code Practising”:-

```
`include "uvm_macros.svh"
import uvm_pkg::*;
module tb;
  int verbosity;
  initial begin
    verbosity = uvm_top.get_report_verbosity_level();
    $display("default Verbosity is %0d", verbosity);
    #10;
  end
endmodule;
```

“Result”:-

```
default Verbosity is 200
Simulation has finished. There are no more test vectors to simulate.
```

- The message with a verbosity level equal to or less than that threshold will be printed on the console otherwise they can't

"Code Practising":-

```
`include "uvm_macros.svh"
import uvm_pkg::*;
module tb;
  int verbosity;
  initial begin
    verbosity = uvm_top.get_report_verbosity_level();
    $display("default Verbosity is %0d", verbosity);
    #10;
    `uvm_info("Chinnu","Hello",UVM_LOW);#10;
  end
endmodule;
```

"Result":-

```
default Verbosity is 200
UVM_INFO /home/runner/testbench.sv(9) @ 10: reporter [Chinnu] Hello
Simulation has finished. There are no more test vectors to simulate.
```

- We cant see the message in the console Because the verbosity level is greater than the default verbosity value.

"Code Practising":-

```
`include "uvm_macros.svh"
import uvm_pkg::*;
module tb;
  int verbosity;
  initial begin
    verbosity = uvm_top.get_report_verbosity_level();
    $display("default Verbosity is %0d", verbosity);
    #10;
    `uvm_info("Chinnu","Hello",UVM_HIGH);#10;
  end
endmodule;
```

"Result":-

```
default Verbosity is 200
Simulation has finished. There are no more test vectors to simulate.
```

- To overwrite the default level we have a method "uvm_top.set_report_verbosity_level(UVM_HIGH);"

"Code Practising":-

```
`include "uvm_macros.svh"
import uvm_pkg::*;
module tb;
  int verbosity;
  initial begin
    uvm_top.set_report_verbosity_level(UVM_HIGH);
    #10;
    `uvm_info("Chinnu","Hello",UVM_HIGH);#10;
  end
endmodule;
```

"Result":-

```
UVM_INFO /home/runner/testbench.sv(8) @ 10: reporter [Chinnu] Hello
Simulation has finished. There are no more test vectors to simulate.
```

- How we are going to send the values of the variables to the console.
- We have three different methods to do this
  - ✓ $sformat(Whenever we have single variable)
  - ✓ core method(Whenever we have to send transaction to the console)
  - ✓ do hooks("")

"Code Practising":-

```
`include "uvm_macros.svh"
import uvm_pkg::*;
module tb;
  int data=0;
  initial begin
    `uvm_info("TB_TOP",$sformatf("The value of the data %0b",data),UVM_MEDIUM);
  end
endmodule
```

"Result":-

```
ASDB file was created in location /home/runner/dataset.asdb
UVM_INFO /home/runner/testbench.sv(6) @ 0: reporter [TB_TOP] The value of the data 00000000000b
Simulation has finished. There are no more test vectors to simulate.
```