TEST BENCH

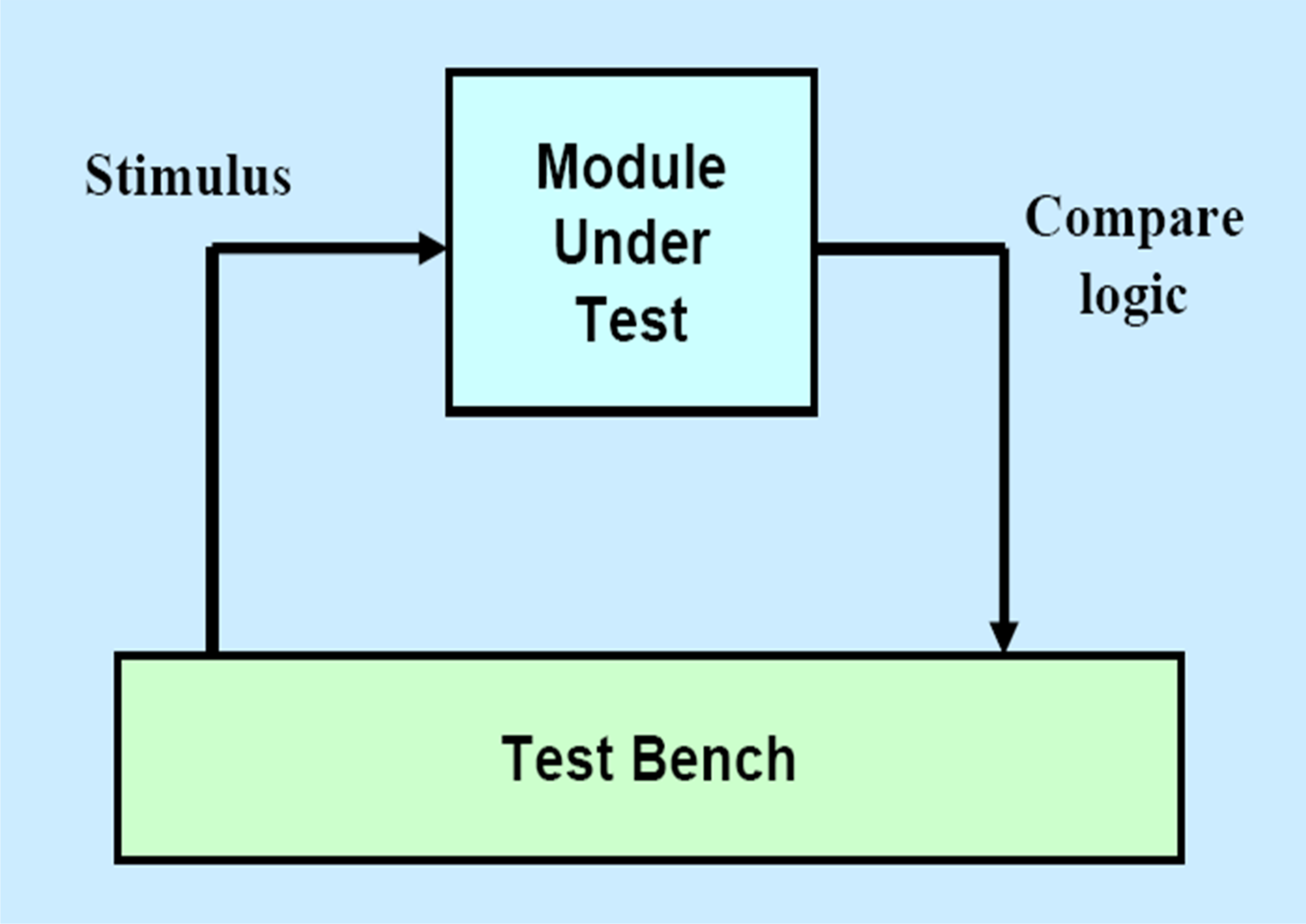
Verilog Test Bench

What is test bench?

– A Verilog procedural block which executes only once.

– Used for simulation.

– Test bench generates clock, reset, and the required test vectors


Stimulus
Module Under Test
Compare logic
Test Bench

How to Write Test bench?

- Create a dummy template

  -Declare inputs to the module-under-test (MUT) as "**reg**", and the outputs as "**wire**"

  – Instantiate the MUT.

- Initialization

  – Assign some known values to the MUT inputs.

- Clock generation logic

  – Various ways to do so.

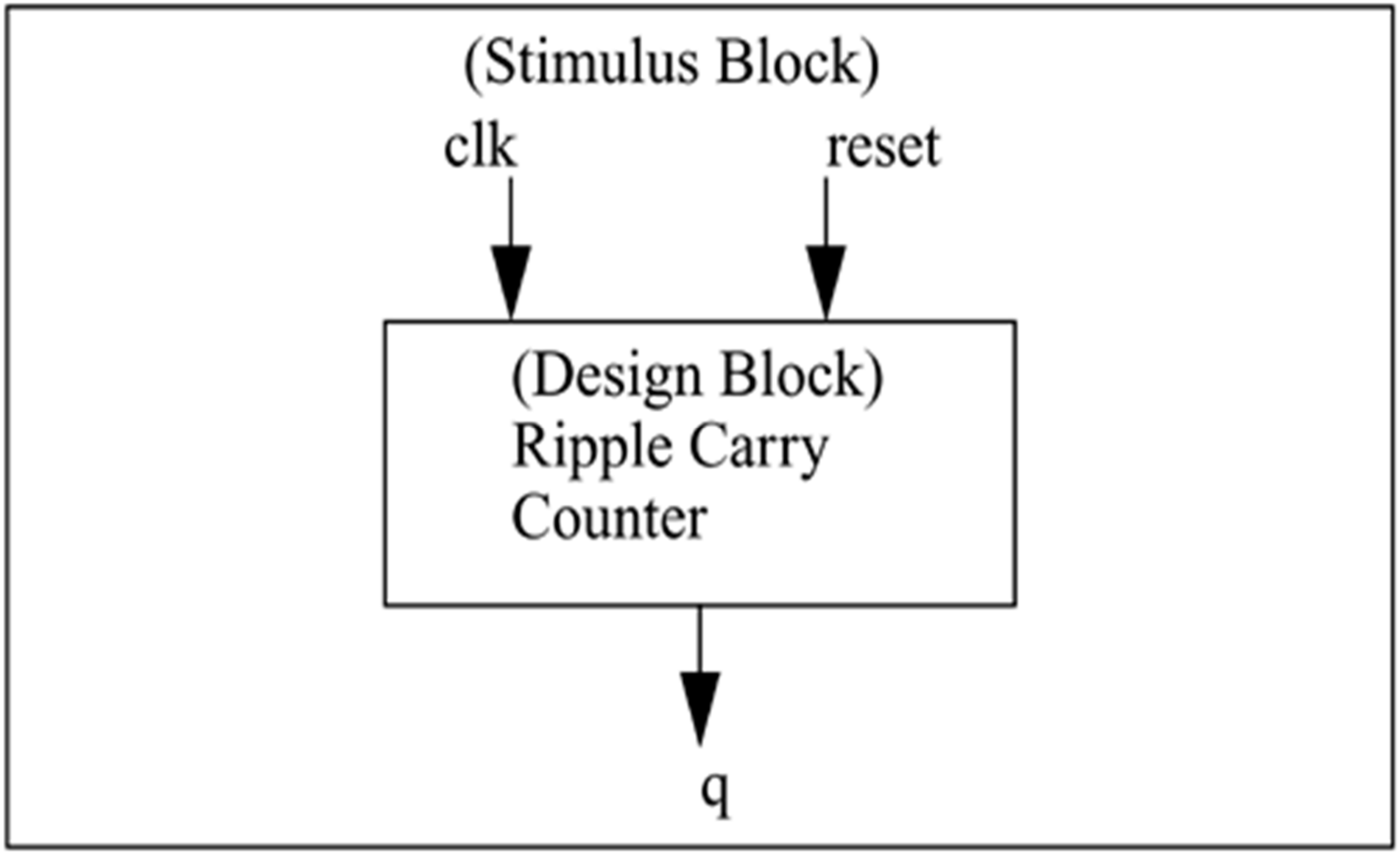
- May include several simulator directives

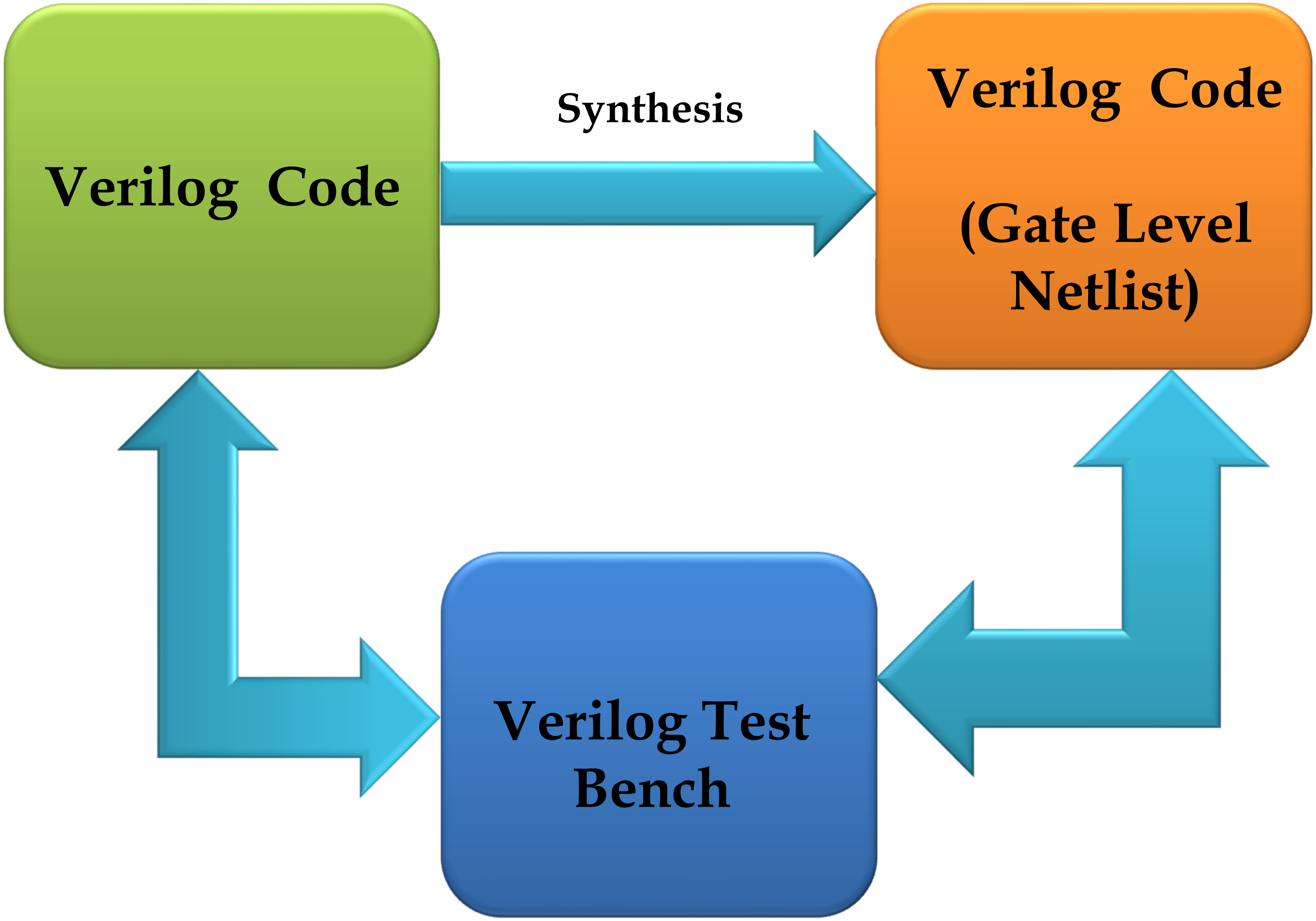
  – Like **$display, $monitor**, **$dumpfile**, **$dumpvars, $finish.**

Components of Simulation

- Once a design block is completed, it must be tested
- A design block needs to be tested for its functionality by applying stimulus and checking results.
- For testing the design block, a stimulus block or test bench can be created in Verilog itself.
- In the test bench we provide some stimulus to the corresponding inputs of the design module.

Stimulus Block Instantiates Design Block




Verilog Code
Synthesis
Verilog Code
(Gate Level Netlist)
Verilog Test Bench

Example

```
module and_gate (a,b,y);
input a,b;
output y;
assign y= a & b;
endmodule
```

```
module and_test ();
reg a,b;
wire y;
and_gate a1(a,b,y);
initial
begin
a=1’b0;b=1’b0;
#100;
a=1’b0; b=1’b1;
#100;
a=1’b1; b=1’b0;
#100;
a=1’b1; b=1’b1;
end
endmodule
```

Example

```
module CC  (a,b,c,d,y);
  input a,b,c,d;
  output y;

assign  y = (d==0) ? (a & b & c) :
                   (a ^ b ^ c);
endmodule
```

```
module CombinationalCircuit_TB;
              reg a,b,d,c;
              wire y;
CC  dut (.a(a),.b(b), .c(c),.d(d),.y(y));

integer k;
initial
 begin
{a,b,c,d} = 4’b0;
for (k=0; k<=16; k=k+1)
      #5 {a,b,c,d} = k;
#20          $stop;
End
endmodule
```