

```
design.sv +
1 module top
2 (
3     input clk,
4     input [3:0] a,b,
5     output reg [7:0] mul
6 );
7
8     always@(posedge clk)
9     begin
10         mul <= a * b;
11     end
12
13 endmodule
```

//Testbench.sv

//Create Transaction, Generator and Driver code for Synchronous 4-bit Multiplier

```
class transaction;
```

```
    randc bit [3:0]a,b;
```

```
    logic [7:0] mul;
```

```
    function void display();
```

```
        $display("a:%0d \t b:%0d \t mul:%0d",a,b,mul);
```

```
    endfunction
```

```
    function transaction copy();
```

```
        copy=new();
```

```
        copy.a=this.a;
```

```
        copy.b=this.b;
```

```
        copy.mul=this.mul;
```

```
    endfunction
```

```
endclass
```

```
class generator;
```

```
    transaction trans;
```

```
    mailbox #(transaction) mbx;
```

```
    event done;
```

```
    function new(mailbox#(transaction) mbx);
```

```

    this.mbx=mbx;

    trans=new();
endfunction

task run();
for (int i=0;i<10;i++) begin
    trans.randomize();
    $display("[Gen] generators' display");
    trans.display();
    mbx.put(trans.copy);

    #20;
end
->done;

endtask
endclass

interface mul_if ;
    logic [3:0] a,b;
    logic[7:0] mul;
    logic clk;

endinterface

class driver;
    mailbox#(transaction) mbx;
    transaction trans;
    virtual mul_if mif;

    function new(mailbox#(transaction) mbx);
        this.mbx=mbx;
    endfunction

    task run();
        forever begin

```

```

        mbx.get(trans);
        @(posedge mif.clk);
        mif.a<=trans.a;
        mif.b<=trans.b;
        $display("[drv] Driver's display");
        trans.display();
    end
endtask
endclass

module tb;

    generator gen;
    mailbox#(transaction) mbx;
    mul_if mif();
    driver drv;
    event done;
    top DUT(mif.clk,mif.a,mif.b,mif.mul);
    initial begin
        mif.clk<=0;
    end
    always#10 mif.clk<=~mif.clk;
    initial begin
        mbx=new();
        drv=new(mbx);
        gen=new(mbx);
        drv.mif=mif;
        done=gen.done;

    end
    initial begin
        fork
            gen.run();

```

```
    drv.run();  
    join_none //non-blocking join  
    wait(done.triggered);  
    $finish();  
end  
initial begin  
    $dumpfile("dump.vcd");  
    $dumpvars;  
end  
endmodule
```

//OUTPUT:-

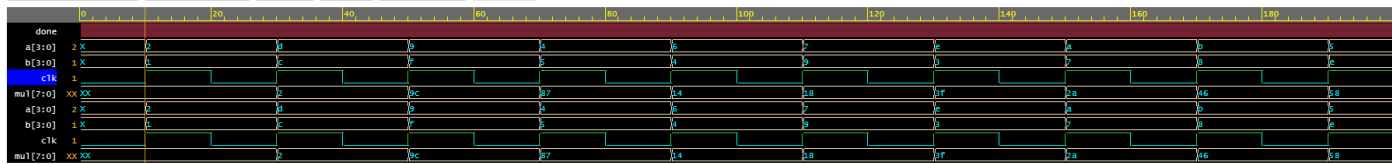
[Log](#)[Share](#)

```
# KERNEL: kernel process initialization done.
# Allocation: Simulator allocated 5501 kB (elbread=459 elab2=4891 ker
# KERNEL: ASDB file was created in location /home/runner/dataset.asdb
# KERNEL: [Gen] generators' display
# KERNEL: a:2    b:1    mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:2    b:1    mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:13   b:12   mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:13   b:12   mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:9    b:15   mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:9    b:15   mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:4    b:5    mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:4    b:5    mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:6    b:4    mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:6    b:4    mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:7    b:9    mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:7    b:9    mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:14   b:3    mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:14   b:3    mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:10   b:7    mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:10   b:7    mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:11   b:8    mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:11   b:8    mul:x
# KERNEL: [Gen] generators' display
# KERNEL: a:5    b:14   mul:x
# KERNEL: [drv] Driver's display
# KERNEL: a:5    b:14   mul:x
# RUNTIME: Info: RUNTIME_0068 testbench.sv (85): $finish called.
# KERNEL: Time: 200 ns Iteration: 0 Instance: /tb Process: @TM1
```

EPWave

From: 0ps To: 200ps

Get Signals Radix 100% 10ps tb/DUT/c1k



Note: To revert to EPWave opening in a new browser window, set that option on your user page.