

1. 5.3

```
module alarm(panic, enable, exiting, window, door, garage, alarm);  
    input  logic panic, enable, exiting, window, door, garage;  
    output logic alarm;  
  
    logic secure;  
    assign secure = window & door & garage;  
    assign alarm = panic | (enable & ~exiting & ~secure);  
endmodule
```

Figure 1: alarm system

2. 5.23

```
module VrM35dec(N, M3, M5);  
    input  logic [5:0] N;  
    output logic M3, M5;  
  
    always_comb begin  
        if (N % 5 == 0) M5 = 1'b1;  
        else           M5 = 1'b0;  
        if (N % 3 == 0) M3 = 1'b1;  
        else           M3 = 1'b0;  
    end  
endmodule
```

Figure 2: VrM35dec

3. 5.24

```

`timescale 1ns/1ps
module VrM35dec_testbench();
  logic [5:0] N;
  logic M3, M5;

  VrM35dec dut (.N, .M3, .M5);

  initial begin
    integer i;
    for (i = 0; i < 64; i++) begin
      N = i; #5;
      if (((i % 3) == 0) != M3 || ((i % 5) == 0) != M5) begin
        $display("An error occurs! (i mode 3) (%d) != M3 (%d) || (i mdoe 5) (%d) != M5 (%d)", (i%3), (M3),
          (i%5), (M5));
      end
    end
    $stop;
  end
end

```

Figure 3: VrM35dec testbench

4. 5.25

```

module F(w, x, y, z, f);
  input logic w, x, y, z;
  output logic f;

  logic y1, y2;
  but b2 (.A1(w), .B1(y), .A2(x), .B2(z), .Y1(y1), .Y2(y2));

  assign f = y1 | y2;
endmodule

module but(A1, B1, A2, B2, Y1, Y2);
  input logic A1, B1, A2, B2;
  output logic Y1, Y2;

  always_comb begin
    if ( (A1 & B1) == 1'b1 && (A2 & B2) == 1'b0) Y1 = 1;
    else Y1 = 0;
    if ( (A2 & B2) == 1'b1 && (A1 & B1) == 1'b0) Y2 = 1;
    else Y2 = 0;
  end
endmodule

```

Figure 4: $F = \sum W, X, Y, Z$ module

```

module F_testbench();
    logic w, x, y, z, f;

    F dut (.w, .x, .y, .z, .f);

    initial begin
        integer i;
        for (i = 0; i < 16; i++) begin
            {w, x, y, z} = i; #5;
            if ((i == 5 || i == 7 || i == 10 || i == 11 || i == 13 || i == 14)) begin
                if (f != 1) $display("error! F = %d but output false", i);
            end else if (f == 1)
                $display("error! F = %d but output true", i);
            end
        end
    end
endmodule

```

Figure 5: $F = \sum W, X, Y, Z$ testbench