Tutorial - 1

DIGITAL SYSTEMS ES 204

GUNTAS SINGH SARAN (22110089)

VERILOG CODE

Q1 - 3

A. VERILOG CODE

$$f(x, y, z, w) = \sum m(0,2,3,4,7,8,10,11,15)$$

VERILOG CODE FOR K-MAPS

```
`timescale 1ns / 1ps

module mod1(A, OUT);

input [3:0] A;
output OUT;

assign OUT = (~A[2] & ~A[0]) | (A[1] & A[0]) | (~A[3] & ~A[1] &
~A[0]);

endmodule
```

VERILOG CODE FOR 4:16 DECODER

```
`timescale 1ns / 1ps

module decoder(input [3:0] A, output reg [15:0] OUT);

always @(*) begin
    case(A)
        4'd0  : OUT = 16'b0000000000000001;
        4'd1  : OUT = 16'b0000000000000010;
        4'd2  : OUT = 16'b0000000000000100;
        4'd3  : OUT = 16'b0000000000001000;
        4'd4  : OUT = 16'b0000000000010000;
        4'd5  : OUT = 16'b0000000000100000;
        4'd6  : OUT = 16'b0000000001000000;
        4'd7  : OUT = 16'b0000000010000000;
        4'd8  : OUT = 16'b0000000100000000;
        4'd9  : OUT = 16'b0000001000000000;
        4'd10 : OUT = 16'b0000010000000000;
        4'd11 : OUT = 16'b0000100000000000;
        4'd12 : OUT = 16'b0001000000000000;
        4'd13 : OUT = 16'b0010000000000000;
        4'd14 : OUT = 16'b0100000000000000;
        4'd15 : OUT = 16'b1000000000000000;
        default : OUT = 16'b0000000000000000;
    endcase
end
endmodule



module logc(A, OUT);

input [3:0] A;
output OUT;

wire [15:0] decode;
decoder dec(.A(A), .OUT(decode));
```

```
assign OUT = decode[0] | decode[2] | decode[3] | decode[4] |
decode[7] | decode[8] | decode[10] | decode[11] | decode[15];

endmodule
```

TESTBENCH

```
module testbench;

reg [3:0] A;
wire OUT;

logc logi(.A(A), .OUT(OUT));

initial begin
    $monitor($time, " A = %b, OUT = %b", A, OUT);
    A = 4'b0000;
    #1 A = 4'b0001;
    #1 A = 4'b0010;
    #1 A = 4'b0011;
    #1 A = 4'b0100;
    #1 A = 4'b0101;
    #1 A = 4'b0110;
    #1 A = 4'b0111;
    #1 A = 4'b1000;
    #1 A = 4'b1001;
    #1 A = 4'b1010;
    #1 A = 4'b1011;
    #1 A = 4'b1100;
    #1 A = 4'b1101;
    #1 A = 4'b1110;
    #1 A = 4'b1111;
    #1 $finish;
end

endmodule
```

B. SIMULATION