DSD LAB

Lab 5

Q1)

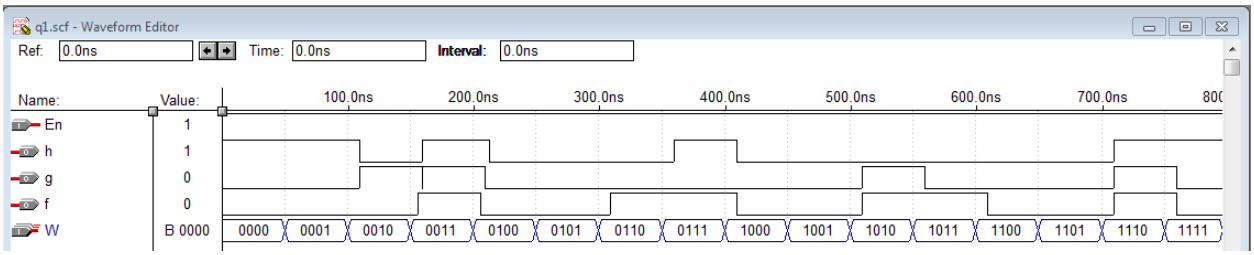
Source Code:

```
module decoder2to4(W,En,Y);
input[1:0]W;
input En;
output [0:3]Y;
reg [0:3]Y;
always@(W or En)
begin
if(En==1)
case(W)
0: Y=4'b1000;
1: Y=4'b0100;
2: Y=4'b0010;
3: Y=4'b0001;
endcase
else
Y=4'b0000;
end
endmodule

module decoder4to16(W,En,Y);
input [3:0]W;
input En;
output [0:15]Y;
wire[0:3]Z;

decoder2to4 d1(W[3:2],En,Z[0:3]);
decoder2to4 d2(W[1:0],Z[0],Y[0:3]);
decoder2to4 d3(W[1:0],Z[1],Y[4:7]);
decoder2to4 d4(W[1:0],Z[2],Y[8:11]);
decoder2to4 d5(W[1:0],Z[3],Y[12:15]);
endmodule

module q1(W, En, f, g, h);
input [3:0] W;
input En;
output f, g, h;

wire [0:15]Y;
```

decoder4to16 d1(W, En, Y);

or(f, Y[3], Y[6], Y[7], Y[10], Y[11], Y[14]);
or(g, Y[2], Y[3], Y[10], Y[14]);
or(h, Y[0], Y[1], Y[3], Y[7], Y[14], Y[15]);
endmodule

Output Waveform:



Q2)

Source Code:

module q2(a, b, cin, cout, sum);
input a, b, cin;
output cout, sum;

wire [0:3]dec0;
wire [0:7]decout;

decoder2to4 d1({a, b}, 1'b1, dec0);
decoder2to4 d2(dec0[0:1], cin, decout[0:3]);
decoder2to4 d3(dec0[2:3], cin, decout[4:7]);

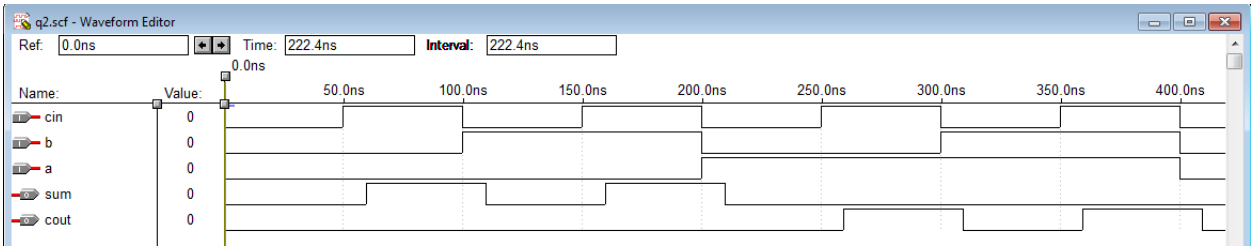
or(cout, decout[3], decout[5], decout[6], decout[7]);
or(sum, decout[1], decout[2], decout[4], decout[7]);

endmodule

module decoder2to4(W,En,Y);
input[1:0]W;
input En;
output [0:3]Y;
reg [0:3]Y;
always@(W or En)
begin
if(En==1)
case(W)
0: Y=4'b1000;

1: Y=4'b0100;
2: Y=4'b0010;
3: Y=4'b0001;
endcase
else
Y=4'b0000;
end
endmodule

Output Waveform:



Q3)

Source Code:

module q3(A, f, g, h);
input [3:0]A;
output f, g, h;
wire [15:0]Y;
decoder3to8 d1(A[2:0], A[3], Y[15:8]);
decoder3to8 d2(A[2:0], ~A[3], Y[7:0]);

or(f, Y[2], Y[4], Y[7], Y[9]);
or(g, Y[0], Y[3], Y[15]);
or(h, Y[0], Y[2], Y[10], Y[12]);
endmodule

module decoder3to8(W,En,Y);
input [2:0] W;
input En;
output [7:0] Y;
reg [7:0] Y;

```
 always @(W or En)
          begin
    if (En)
      begin
        Y=8'd0;
        case (W)
           3'b000: Y[0]=1'b1;
```

```
            3'b001: Y[1]=1'b1;
            3'b010: Y[2]=1'b1;
            3'b011: Y[3]=1'b1;
            3'b100: Y[4]=1'b1;
            3'b101: Y[5]=1'b1;
            3'b110: Y[6]=1'b1;
            3'b111: Y[7]=1'b1;
            default: Y=8'd0;
        endcase
    end
            else
            Y=8'd0;
            end
endmodule
```

Output: