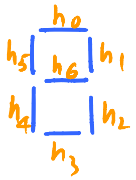
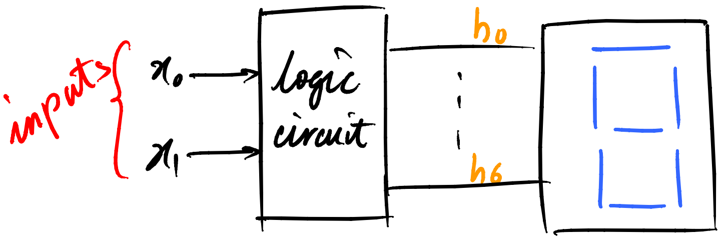
2.8 Hierarchical design

7-segment display



example

x_1	x_0	Display
0	0	0
0	1	1
1	0	2
1	1	3



x_1	x_0	h_0	h_1	h_2	h_3	h_4	h_5	h_6
0	0	1	1	1	1	1	1	0
0	1	0	1	1	0	0	0	0
1	0	1	1	0	1	1	0	1
1	1	1	1	1	1	0	0	1

$$h_0 = \overline{\bar{x}_1 x_0} = x_1 + \bar{x}_0$$

$$h_1 = 1$$

$$\vdots$$

$$h_6 = x_1$$

```
module seg7 (input x1, x0, output [0:6]H);
  assign H[0] = x1 | ~x0;
  assign H[1] = 1'b1;
  assign     ⋮
  assign
  assign
  assign
  assign H[6] = x1;
endmodule
```

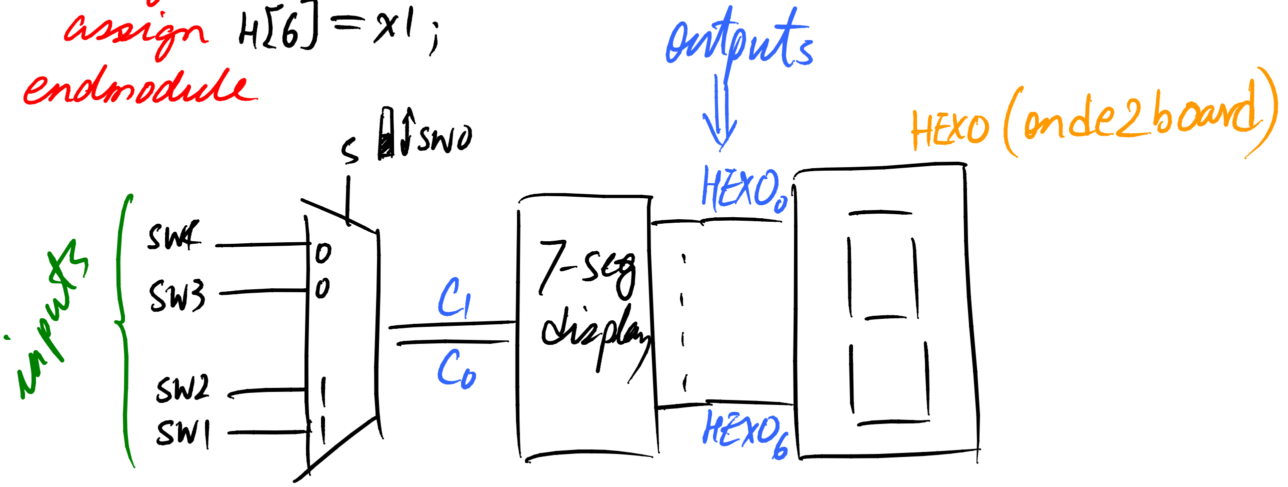
value "1"

binary

1 bit

// my comment

/* my long
comments */



```
module hier_ex (input [4:0] SW, output [0:6] HEX0);
   wire [1:0] C;
   wire [0:6] H;
   mux2bit_2to1 U1 (SW[4:3], SW[2:1], SW[0], C);
   seg7 U2 (C[1], C[0], H);
   assign HEX0 = ~H;   <-- DE2 lights up the segment
                           when the driving signal is "0"
endmodule

module mux2bit_2to1 (X, Y, s, M);
   input [1:0] X, Y;
   input s;
   output [1:0] M;
   mux2to1 U1 (X[0], Y[0], s, M[0]);
   mux2to1 U2 (X[1], Y[1], s, M[1]);
endmodule

module mux2to1 (x, y, s, m);
   input x, y, s;
   output m;
   assign m = (~s & x) | (s & y);
endmodule

module seg7 (input x1, x0, output [0:6] H);
   assign H[0] = x1 | ~x0;
   assign H[1] = 1'b1;
   assign H[2] = ~x1 | x0;
   assign H[3] = x1 | ~x0;
   assign H[4] = ~x0;
   assign H[5] = ~x1 & ~x0;
   assign H[6] = x1;
end module
```