

«به نام خدا»

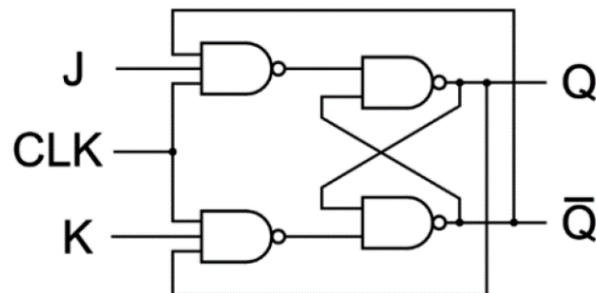
استاد: دکتر میثم عبداللهی

نام: فاطمه زهرا بخشنده

گزارش تمرین اول:

سوال اول:

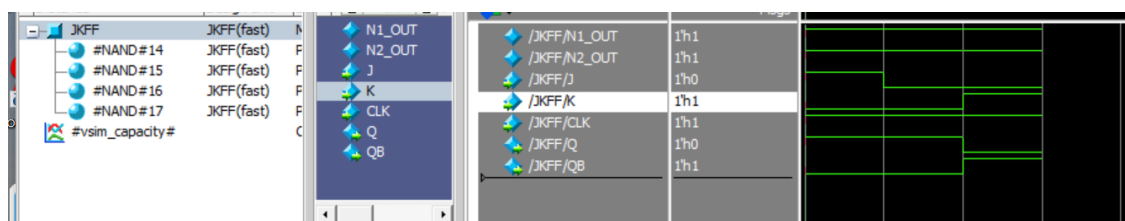
در این سوال باید برنامه ای برای یک jk flip flop بنویسیم.



همانطور که مشخص است سه ورودی و دو خروجی دارد و شامل 4 nand است. از روی همین structure برنامه آن را می نویسیم:

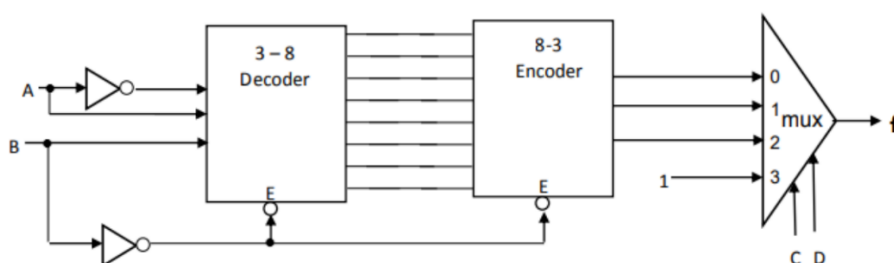
```
1 module JKFF(  
2     Q,  
3     QB,  
4     CLK,  
5     J,  
6     K  
7 );  
8  
9 wire N1_OUT;  
10 wire N2_OUT;  
11 input J, K, CLK;  
12 output Q, QB;  
13 |  
14 nand(N1_OUT, J, CLK, QB);  
15 nand(N2_OUT, K, CLK, Q);  
16 nand(Q, QB, N1_OUT);  
17 nand(QB, Q, N2_OUT);  
18  
19 endmodule
```

که شبیه سازی waveform آن به صورت زیر است:



سوال دوم:

در این سوال، باید مدار زیر را طراحی کنیم:



سه module تعریف میکنیم. 3x8 decoder، 8x3 encoder و 4x1 mux. پس از طراحی هرکدام، از آن ها استفاده کرده و برنامه مدار اصلی را می نویسیم:

```

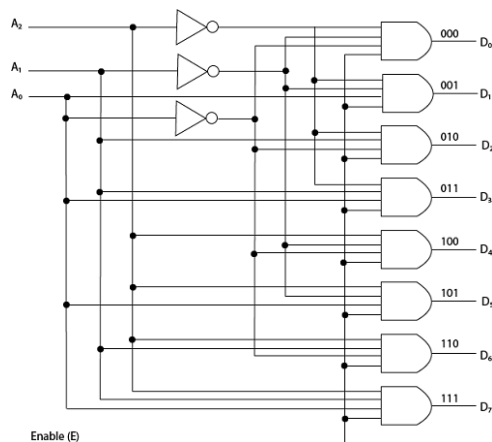
4 module Circuit(
5     F, A, B, C, D
6 );
7
8 wire A_NOT, B_NOT, Dout[7:0], Eout[2:0], one;
9 input A, B, C, D;
10 output F;
11
12 not(A_NOT, A);
13 not(B_NOT, B);
14 buf(one, 1);
15
16 Decoder DEC(
17     Dout[7], Dout[6], Dout[5],
18     Dout[4], Dout[3], Dout[2],
19     Dout[1], Dout[0], B_NOT,
20     A_NOT, A, B
21 );
22 Encoder ENC(
23     Eout[2], Eout[1], Eout[0],
24     Dout[7], Dout[6], Dout[5],
25     Dout[4], Dout[3], Dout[2],
26     Dout[1], Dout[0], B_NOT
27 );
28 MUX M(F, Eout[2], Eout[1], Eout[0], one, C, D);
29
30 endmodule

```

در این سوال، اگر E فعال نباشد خروجی سیگنال های encoder و decoder مقدار صفر خواهند داشت. (active low است در نتیجه E باید صفر باشد تا فعال شوند)

طراحی هرکدام از سه مدار استفاده شده، به صورت زیر است:

1- 3x8 decoder:



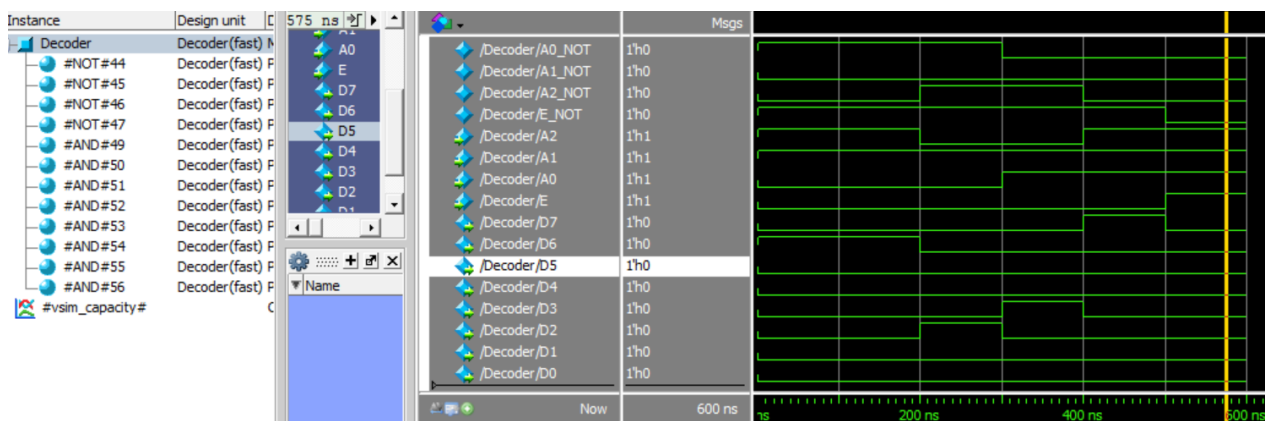
مدار بالا را طراحی میکنیم:

```

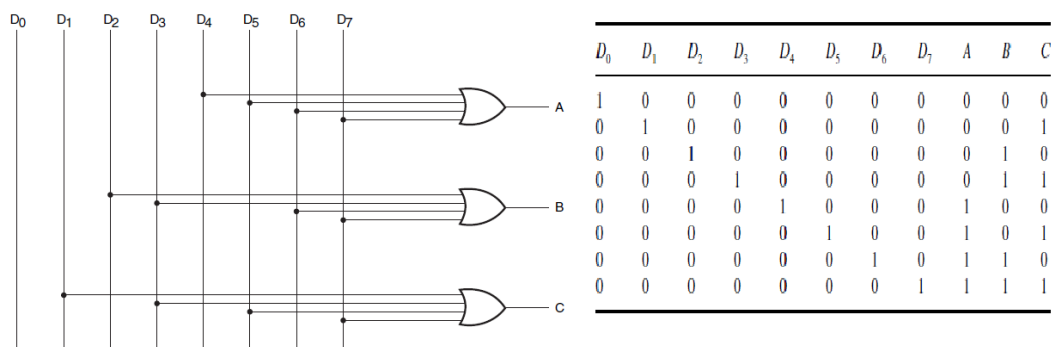
33
34 module Decoder(
35     D7, D6, D5,
36     D4, D3, D2,
37     D1, D0, E,
38     A2, A1, A0
39 );
40
41 wire A0_NOT, A1_NOT, A2_NOT, E_NOT;
42 input A2, A1, A0, E;
43 output D7, D6, D5, D4, D3, D2, D1, D0;
44
45 not(A0_NOT, A0);
46 not(A1_NOT, A1);
47 not(A2_NOT, A2);
48 not(E_NOT, E); //active-low
49
50 and(D0, A2_NOT, A1_NOT, A0_NOT, E_NOT); //000
51 and(D1, A2_NOT, A1_NOT, A0, E_NOT); //001
52 and(D2, A2_NOT, A1, A0_NOT, E_NOT); //010
53 and(D3, A2_NOT, A1, A0, E_NOT); //011
54 and(D4, A2, A1_NOT, A0_NOT, E_NOT); //100
55 and(D5, A2, A1_NOT, A0, E_NOT); //101
56 and(D6, A2, A1, A0_NOT, E_NOT); //110
57 and(D7, A2, A1, A0, E_NOT); //111
58
59 endmodule
60

```

که شبیه سازی waveform آن به صورت زیر است:



:8x3 encoder -2



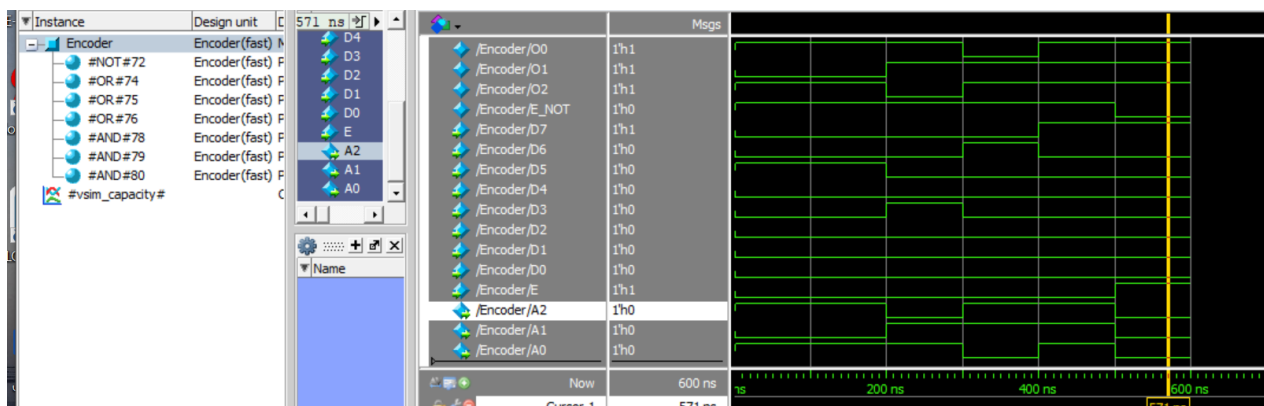
مدار بالا را طراحی میکنیم و در آخر خروجی هارا با not سیگنال E، and می کنیم:

```

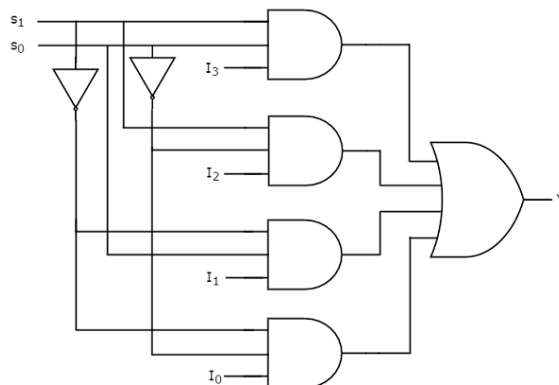
61
62 module Encoder(
63     A2, A1, A0,
64     D7, D6, D5,
65     D4, D3, D2,
66     D1, D0, E
67 );
68
69 wire O0, O1, O2, E_NOT;
70 input D7, D6, D5, D4, D3, D2, D1, D0, E;
71 output A2, A1, A0;
72
73 not(E_NOT, E); //active-low
74
75 or(O2, D4, D5, D6, D7);
76 or(O1, D2, D3, D6, D7);
77 or(O0, D1, D3, D5, D7);
78
79 and(A0, O0, E_NOT);
80 and(A1, O1, E_NOT);
81 and(A2, O2, E_NOT);
82
83 endmodule
84

```

که شبیه سازی waveform آن به صورت زیر می شود:



:4x1 mux -3



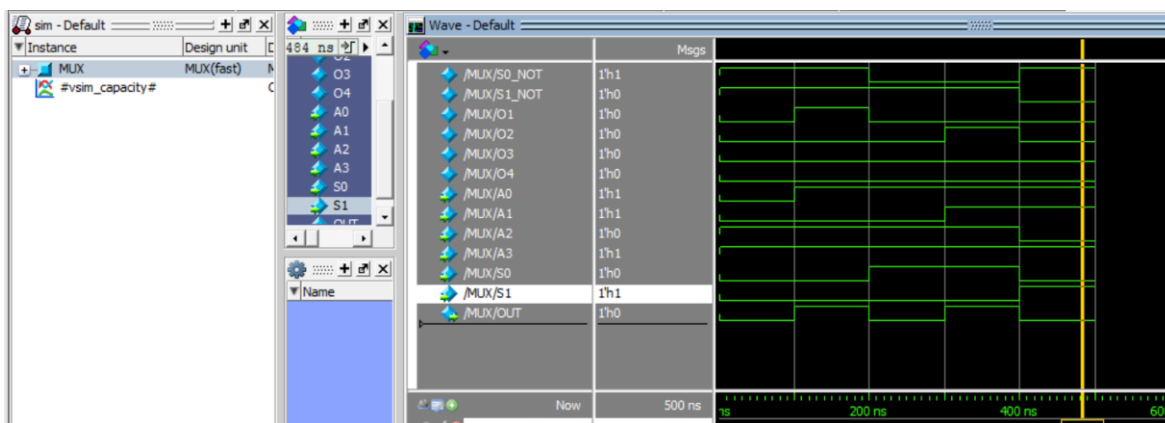
مدار بالا را طراحی میکنیم:

```

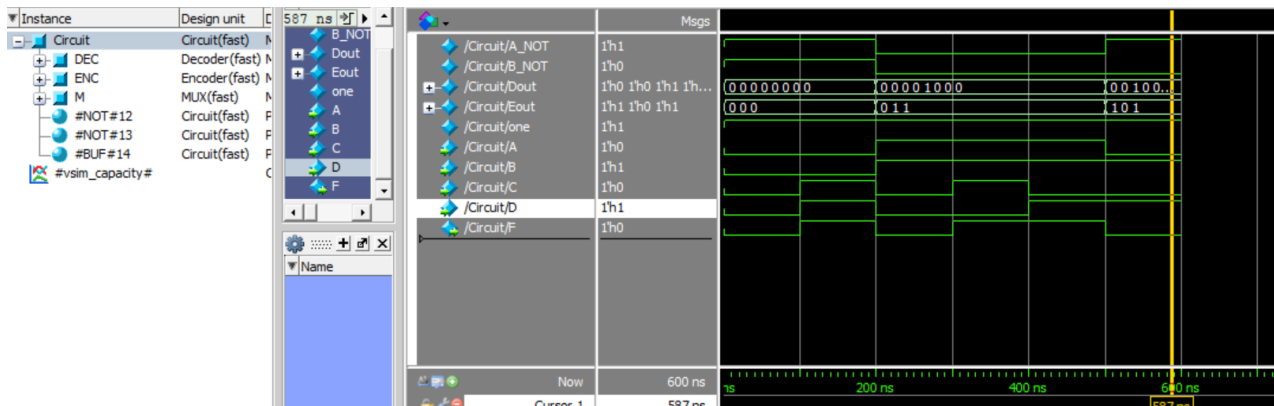
85
86 module MUX(
87     OUT,
88     A0, A1, A2, A3,
89     S1, S0
90 );
91
92 wire S0_NOT, S1_NOT, O1, O2, O3, O4;
93 input A0, A1, A2, A3, S0, S1;
94 output OUT;
95
96 not(S0_NOT, S0);
97 not(S1_NOT, S1);
98 and(O1, A0, S0_NOT, S1_NOT);
99 and(O2, A1, S1_NOT, S0);
100 and(O3, A2, S1, S0_NOT);
101 and(O4, A3, S0, S1);
102 or(OUT, O1, O2, O3, O4);
103
104 endmodule
105

```

که شبیه سازی waveform آن به صورت زیر می شود:

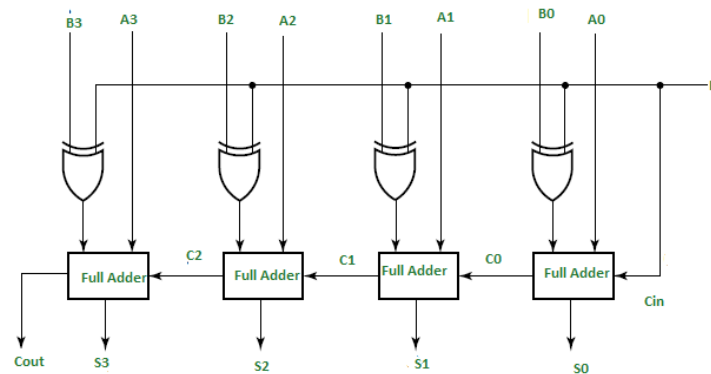


در آخر شبیه سازی waveform مدار اصلی به صورت زیر است:



سوال سوم:

مدار یک 4-bit Adder/Subtractor به صورت زیر است:



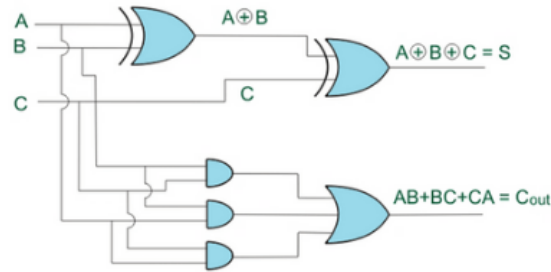
مدار بالا را طراحی میکنیم:

```

3 module Add_Sub(
4     OutP,
5     Cout,
6     A,
7     B,
8     select
9 );
10
11 wire B0, B1, B2, B3;
12 wire C0, C1, C2, C3;
13 input[3:0] A, B;
14 input select;
15 output[3:0] OutP;
16 output Cout;
17
18 xor(B0, B[0], select);
19 xor(B1, B[1], select);
20 xor(B2, B[2], select);
21 xor(B3, B[3], select);
22 xor(Cout, C3, select);
23
24 FA F0(OutP[0], C0, A[0], B0, select);
25 FA F1(OutP[1], C1, A[1], B1, C0);
26 FA F2(OutP[2], C2, A[2], B2, C1);
27 FA F3(OutP[3], C3, A[3], B3, C2);
28
29 endmodule
30

```

و یک full adder تعریف می کنیم تا از آن در مدار اصلی استفاده کنیم:



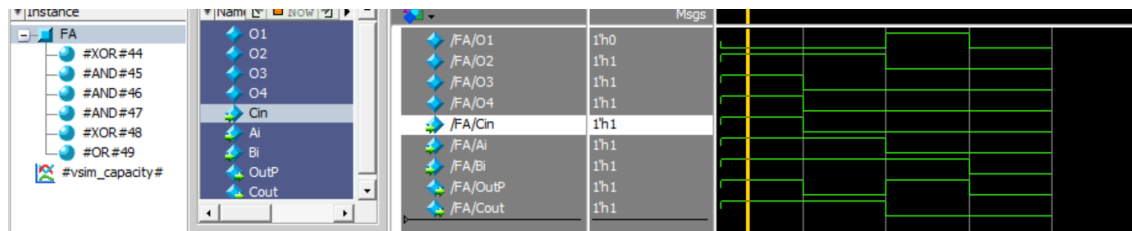
مدار بالا را برای full adder طراحی میکنیم:

```

31
32 module FA(
33     OutP,
34     Cout,
35     Cin,
36     Ai,
37     Bi
38 );
39
40 wire O1, O2, O3, O4;
41 input Cin, Ai, Bi;
42 output OutP, Cout;
43
44 xor(O1, Ai, Bi);
45 and(O2, Ai, Bi);
46 and(O3, Ai, Cin);
47 and(O4, Bi, Cin);
48 xor(OutP, O1, Cin);
49 or(Cout, O2, O3, O4);
50
51 endmodule

```

که شبیه سازی waveform آن به صورت زیر می شود:



در آخر شبیه سازی waveform مدار اصلی به صورت زیر است:

