

1) Two-level logic minimization

Suppose

$$G(A, B, C, D) = \sum m(1, 3, 4, 6, 9, 11, 14, 15)$$

and suppose

$$F(A, B, C, D) = \bar{G}$$

		C D			
		00	01	11	10
A B	00	1	0	0	1
	01	0	1	1	0
	11	1	1	0	0
	10	1	0	0	1

a) Identify all the prime and essential prime implicants **for the function F** (not G).

Primes (including Essentials)		Essential Primes Only
-0-0	$\bar{B}\bar{D}$	$\bar{B}\bar{D}$
01-1	$\bar{A}BD$	$\bar{A}BD$
110-	$AB\bar{C}$	
-101	$B\bar{C}D$	
1-00	$A\bar{C}\bar{D}$	

b) Find the minimum two-level logic implementation **for the function F** (again not G).

$$F = \bar{B}\bar{D} + \bar{A}BD + AB\bar{C}$$

(SHOULD JUST BE ONE SOLUTION)

2) Multi-level logic minimization

Given following logic equations, minimize the number of literals, e.g. by using common subexpressions, Boolean rules, etc. You can introduce new immediate equations for common sub-expressions if it helps to reduce the number of literals. Put a box around your final answer, and indicate the number of literals in your final answer. (Use only AND, OR, or INVERT operators in the Boolean expressions.)

$$\begin{aligned}X &= ABC \\Y &= DEF \\P &= ABCDEF + ABCGH + ABCG\bar{H} \\&= ABC(DEF + GH + G\bar{H}) \\&= ABC(DEF + G) \\&= X(Y + G) \\Q &= ABC\bar{D} + ABC\bar{E} + ABC\bar{F} + \bar{A}DEF + \bar{B}DEF + \bar{C}DEF \\&= ABC(\bar{D} + \bar{E} + \bar{F}) + DEF(\bar{A} + \bar{B} + \bar{C}) \\&= X\bar{Y} + \bar{X}Y \\R &= AEFG + \bar{E} + \bar{F} \\&= AG + \bar{E} + \bar{F}\end{aligned}$$

17 literals

3) Number representation

a) Represent these numbers in 7-bit signed 2's complement form or indicate that it is not possible.

-14	1110010
-64	1000000
-39	1011001
-32	1100000
57	0111001
65	not possible

b) For each binary vector below, what does it represent in Octal and Hexadecimal?

Binary	Octal	Hexadecimal
111111	77	3F
101010	52	2A
100110	46	26
101111	57	2F
000110	06	06

4) State minimization

Minimize the following state machine (show work on an implication chart).

State	Next State					Output				
	XY=	00	01	10	11	XY=	00	01	10	11
A		D	B	G	A		0	0	0	0
B		C	E	A	D		1	1	1	1
C		B	G	F	F		1	1	1	1
D		A	C	E	F		0	0	0	0
E		D	D	A	A		1	0	1	0
F		F	C	E	D		0	0	0	0
G		A	A	D	D		1	0	1	0

$$A = D = F$$

$$B = C$$

$$E = G$$

State	Next State					Output				
	XY=	00	01	10	11	XY=	00	01	10	11
A		A	B	E	A		0	0	0	0
B		B	E	A	A		1	1	1	1
E		A	A	A	A		1	0	1	0

5) Sequential logic implementation

Consider the following incompletely-specified Mealy machine (“*” indicates don’t cares).

State	Next State		Output	
	I=0	1	I=0	1
A	D	B	0	0
B	C	E	1	1
C	B	*	1	*
D	*	C	*	0
E	D	D	1	0

Suppose we use the following state encoding. (Assume the name the input variable is I .)

State	$S_2S_1S_0$
A	000
B	001
C	010
D	011
E	100

Derive the **minimum** two-level logic for each of the following (you should minimize each logic function separately, no need to worry about sharing logic between logic functions).

$$\begin{aligned}
 F &= \bar{I}S_2 + \bar{S}_1S_0 + S_1\bar{S}_0 \text{ OR} \\
 &= \bar{I}S_2 + \bar{S}_1S_0 + \bar{I}S_1 \\
 S_2^+ &= I\bar{S}_1S_0 \\
 S_1^+ &= \bar{I}\bar{S}_1 + S_2 + S_1S_0 \text{ OR} \\
 &= \bar{I}\bar{S}_1 + S_2 + IS_1 \\
 S_0^+ &= \bar{S}_0
 \end{aligned}$$

		F			
I	S2	S1	S0		
		00	01	11	10
00	00	0	1	x	1
01	01	1	x	x	x
11	11	0	x	x	x
10	10	0	1	0	x

		S2+			
I	S2	S1	S0		
		00	01	11	10
00	00	0	0	x	0
01	01	0	x	x	x
11	11	0	x	x	x
10	10	0	1	0	x

		S1+			
I	S2	S1	S0		
		00	01	11	10
00	00	1	1	x	0
01	01	1	x	x	x
11	11	1	x	x	x
10	10	0	0	1	x

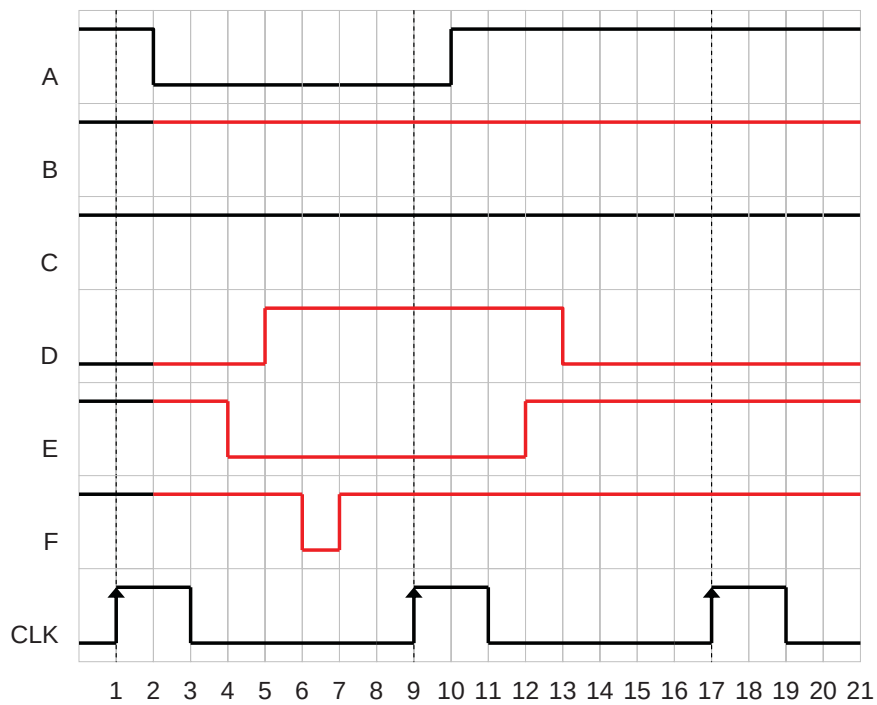
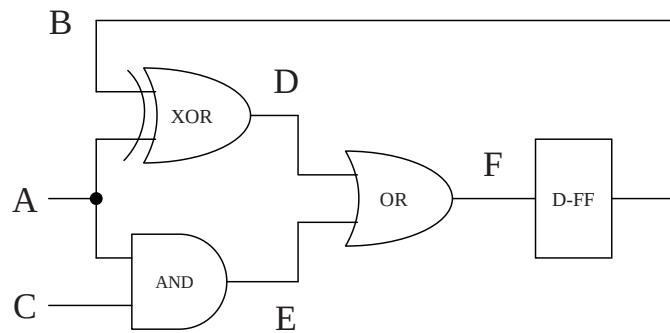
		S0+			
I	S2	S1	S0		
		00	01	11	10
00	00	1	0	x	1
01	01	1	x	x	x
11	11	1	x	x	x
10	10	1	0	0	x

6) Timing analysis

Consider the following sequential logic diagram.

Assume the gate delays are **2 ns** for both 2-AND and 2-OR, **3 ns** for both 2-XOR. Assume the positive edge-triggered flip-flop delay is $T_{delay_{FF}} = 1 \text{ ns}$, and assume the positive edge-triggered flip-flop setup time is also $T_{setup_{FF}} = 1 \text{ ns}$.

Fill out the rest of the timing diagram for A , B , C , D , E , and F .



7) Clock period analysis

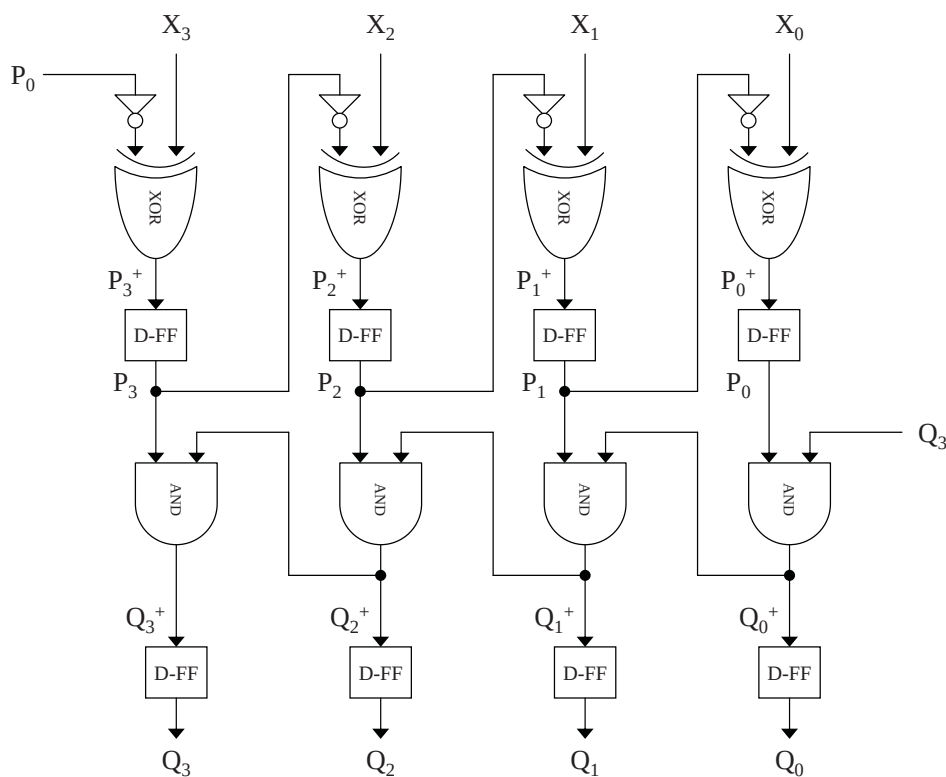
Consider the following sequential logic diagram where X_3 , X_2 , X_1 , and X_0 are **external inputs**, which we assume to be stable (available) at the beginning of the clock period.

Assume the gate delay for 2-XOR is **3 ns**, 2-AND is **2 ns**, and INV is **1 ns**. Assume the positive edge-triggered flip-flop delay is $T_{delayFF} = \mathbf{1\ ns}$, and assume the positive edge-triggered flip-flop setup time is also $T_{setupFF} = \mathbf{1\ ns}$.

What is the **minimum clock period** that would satisfy these delays? Provide the critical path (if there are multiple critical path, just specify one of them).

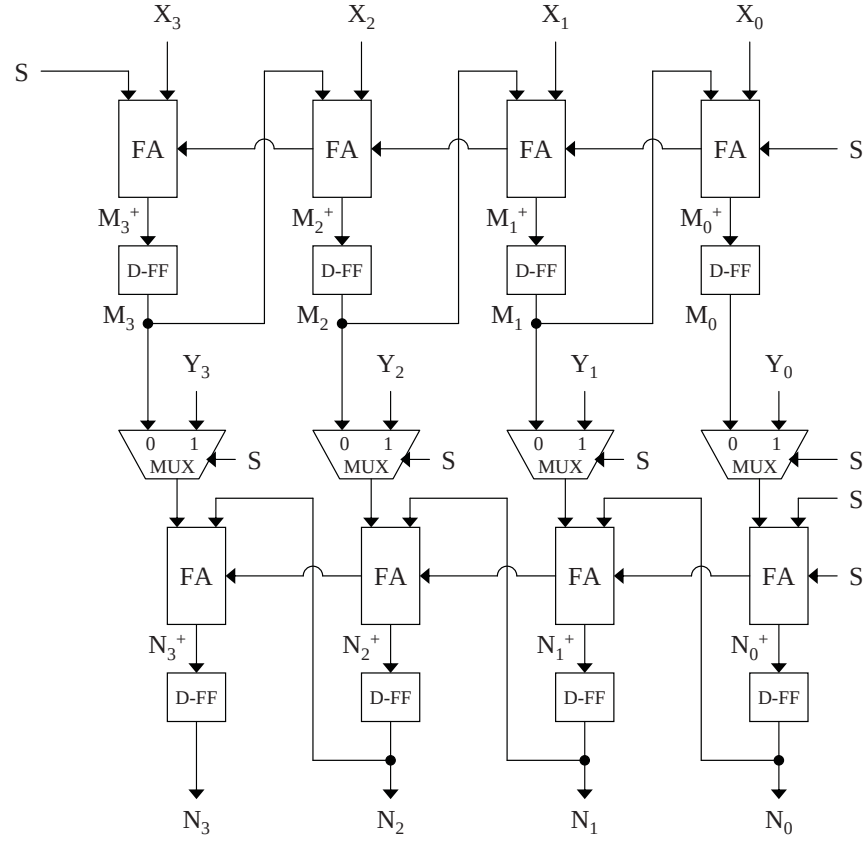
$$\text{Clock period} = T_{delayFF} + \text{"four 2-AND gates"} + T_{setupFF} = 10\ \text{ns}$$

$$\text{Critical path} = Q_3 \rightarrow Q_O^+ \rightarrow Q_1^+ \rightarrow Q_2^+ \rightarrow Q_3^+$$



8) Datapath/registers

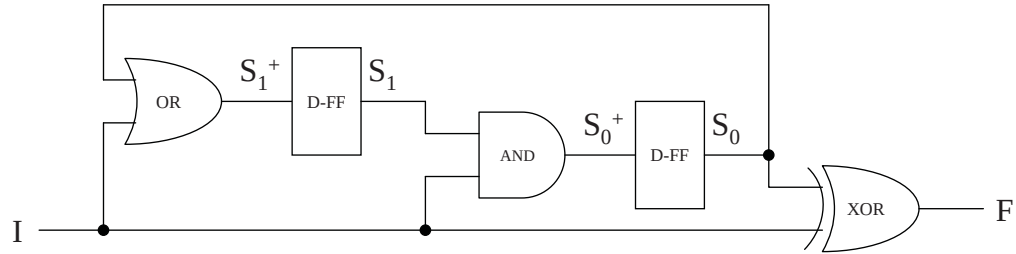
Consider the following diagram.



Given the values for control input S , the data inputs $X_3X_2X_1X_0$ and $Y_3Y_2Y_1Y_0$, and the flip-flop values $M_3M_2M_1M_0$ and $N_3N_2N_1N_0$, what will be the new flip flop values $M_3^+M_2^+M_1^+M_0^+$ and $N_3^+N_2^+N_1^+N_0^+$?

S	$X_3X_2X_1X_0$	$Y_3Y_2Y_1Y_0$	$M_3M_2M_1M_0$	$N_3N_2N_1N_0$	$M_3^+M_2^+M_1^+M_0^+$	$N_3^+N_2^+N_1^+N_0^+$
0	0011	1111	0001	1001	0011	0011
0	0100	1011	0000	0111	0100	1110
1	0001	0100	0000	1000	1010	0110
1	0011	0011	0001	0011	1100	1011

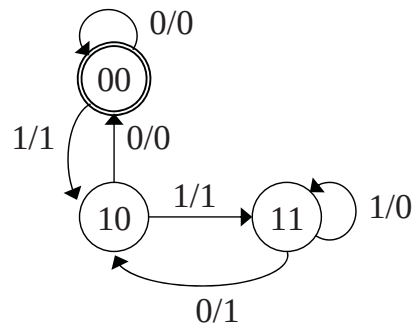
9) Reverse engineer sequential circuit to finite state machine



a) Fill out the following functions (OK to use $\oplus$ in functions).

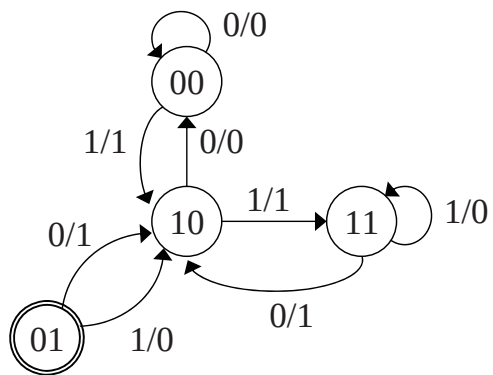
$$\begin{aligned} F &= I \oplus S_0 \\ S_1+ &= I + S_0 \\ S_0+ &= IS_1 \end{aligned}$$

b) Assume initial state is $S_1S_0 = 00$, draw the state graph (**only include reachable states**).



(state “01” is unreachable)

c) Assume initial state is $S_1S_0 = 01$, draw the state graph (**only include reachable states**).



(all states reachable)

10) Miscellaneous

Suppose you want to implement a 32-input **AND** function – i.e., $F = X_{31}X_{30} \dots X_1X_0$, and you can only use 2-input AND gates. Suppose a 2-AND has a delay of **2 ns**.

What is the lowest delay implementation in terms of:

Delay = “5 levels of 2-AND gates” = 10 ns

Number of 2-AND gates = 31