Chapter 1: Review of Logic Design Fundamentals

1.1

A	B	C	$A \oplus B$	$B' \oplus C$	$(A \oplus B) \cdot C$	$A' \cdot (B' \oplus C)$	F
0	0	0	0	1	0	1	1
0	0	1	0	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	1	1	1	1
1	0	0	1	1	0	0	0
1	0	1	1	0	1	0	1
1	1	0	0	0	0	0	0
1	1	1	0	1	0	0	0

1.2

X	Y	B_{in}	$Diff$	B_{out}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Sum of Products: $Diff = XY'B_{in}' + X'Y'B_{in} + XYB_{in} + X'YB_{in}'$

$B_{out} = X'B_{in} + X'Y + YB_{in}$

Product of Sums: $Diff = (X + Y + B_{in})(X + Y' + B_{in}')(X' + Y + B_{in}')(X' + Y' + B_{in})$

$B_{out} = (Y + B_{in})(X' + B_{in})(X' + Y)$

1.3

CD \ AB	00	01	11	10
00	0	X	S	0
01	X	E	E	X
11	1	0	F′	1
10	0	0	F′	0

S = G + H′

Set all map-entered variables to 0 to get MS_0=B'D. Set E, F', and S to 1 one at a time and all 1's to X's to get MS_1=C'D(E), MS_2=ABC(F'), MS_3=BC'D'(G + H')

$Z = B'D + C'DE + ABCF' + BC'D'G + BC'D'H'$

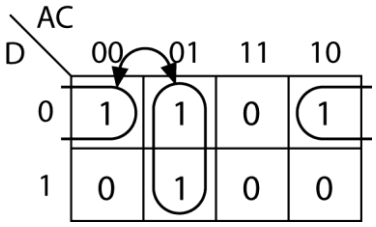
1.4 **(a)** $F = A'D' + AC'D + BCD' + A'B'C'E + BD'E$

(b) $Z = A'CD' + C'D + BC'E + B'DE + CD'F + A'C'G$

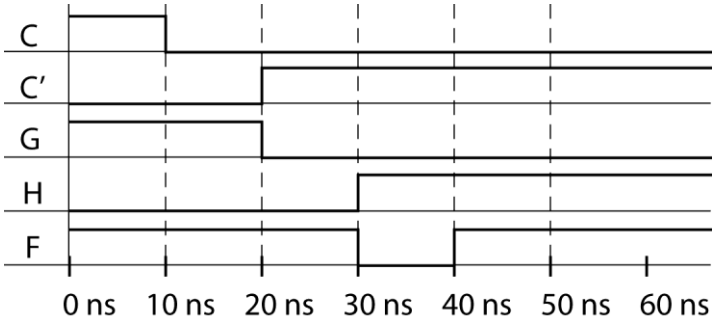
(c) $H = A'CD + A'B'CE + BCDF'$

(d) $G = C'E'F + DEF + AD'F'$

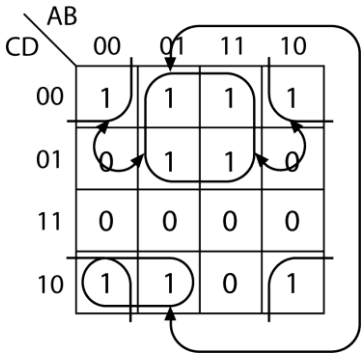
1.5



A static 1-hazard occurs when $A = 0$, $D = 0$, and C changes. When C changes from 1 to 0; $A'C$ also goes from 1 to 0. The hazard occurs because C' hasn't become 1 yet since it has to go through the inverter; therefore, F goes to 0 momentarily before going to 1. Gate delays are assumed to be 10ns in the timing diagram below.



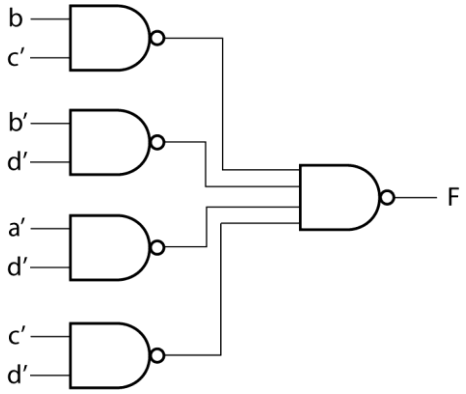
1.6



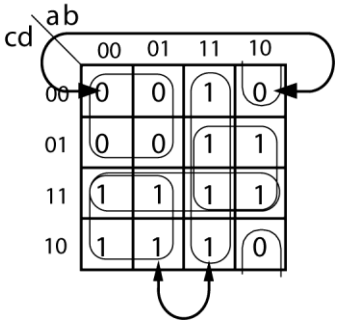
$F = bc' + b'd' + a'cd'$

3 hazards: A=0, C=0, D=0, B changes
A=1, C=0, D=0, B changes
A=0, B=1, D=0, C changes

To eliminate the hazards, add the term $c'd'$ (combining the four 1's in the top row) and replace $a'cd'$ with $a'd'$ (combining two 1's from the bottom left with two 1's from the top left.)



1.7 (a)



F = *((ab)'.(a+c)'+(a'+d)')'*
= *ab + ((a+c)'+(a'+d)')'*
= *ab* + (*a* + *c*)(*a*' + *d*)
= *ab* + *aa*' + *ad* + *a*'*c* + *cd*;

Circle all these terms on the K-map; arc shows nearby1's not in the same product term, indicating a 1 hazard.

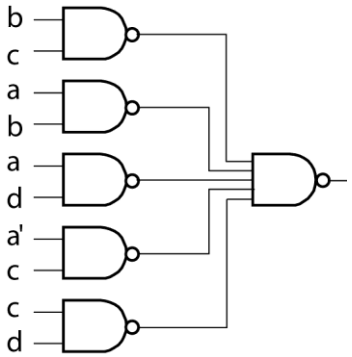
1-hazard *bcd* = 110, a changing 1 => 0 gates 1,5,3,4,5,
0 => 1 gates 3,4,5,1,5

F = *ab* + *aa*' + *ad* + *a*'*c* + *cd*
= *ab* + *a*(*a*' + *d*) + *c*(*a*' + *d*)
= *ab* + (*a*+*c*) (*a*' + *d*)
= (*ab* + *a* +*c*)(*ab* + *a*' +*d*) --see Table 1-1 for Boolean laws
= (*a* + *c*)(*a* + *a*' + *d*)(*a*' + *b* + *d*) -- a+ab=a; a'+d+ab=(a'+d+a)(a'+d+b)

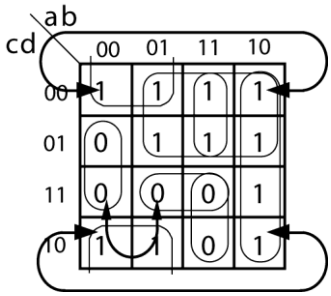
Circle all these terms of 0's in the K-map; arc shows 0's not in same term.

0-hazard *bcd* = 000, a changing 0 => 1 gates 3,4,5,3,5
1 => 0 gates 3,4,5,2,5

(b) We will design a 2-level sum of products circuit because a 2-level sum-of-products circuit has no 0-hazard as long as an input and its complement are not connected to the same AND gate. Avoid 1-hazard by adding product term bc. Use NAND gates as asked in the question.



1.8



(a) $Z = A'D' + (A + B)(B' + C')$
$= A'D' + AB' + AC' + BB' + BC'$

Static 1 hazard (see the arcs between nearby 1's not in the same product term.)
ABCD = 0000 to 1000
ABCD = 0010 to 1010

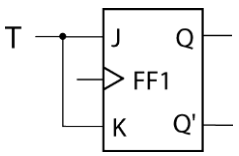
$Z = (A'D' + A + B)(A'D' + B' + C')$
$= (A' + A + B)(D' + A + B)(A' + B' + C')(D' + B' + C')$

Static 0 hazard (see the arcs between nearby 0's not in the same term)
ABCD = 0111 to 0011

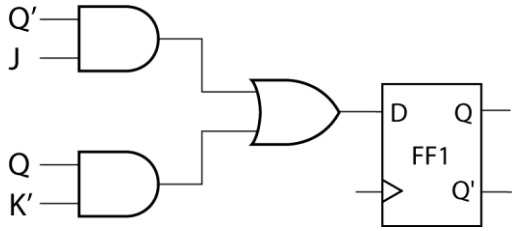
(b) One can design a hazard-free sum of products circuit as in the previous question. Or, one can design a product of sums (POS) circuit with no hazards. A properly designed 2-level POS circuit has no 1-hazards. Static 0-hazards can be avoided by including loops for all 0's that are adjacent. 4 terms here including the arc:

$Z = (A + B + D')(A' + B' + C')(D' + B' + C')(C' + D' + A)$
$= (D' + A + BC')(A' + B' + C')(D' + B' + C')$ combining 1st and 4th terms
$= (D' + A + BC')(B' + C' + A'D')$ combining 2nd and 3rd terms

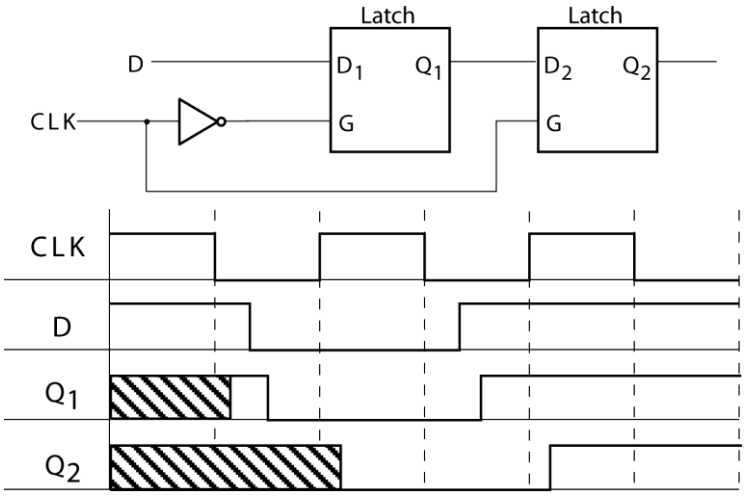
1.9 (a)



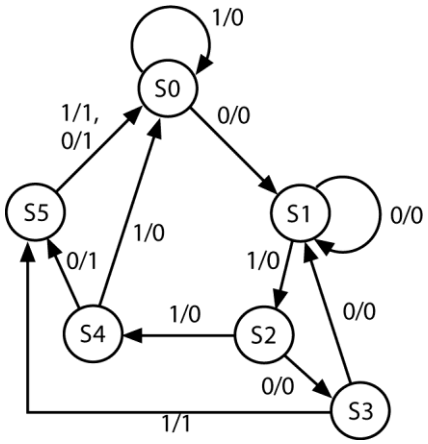
(b) From the characteristic equation for a J-K flip-flop ($Q^+ = JQ' + K'Q$):



1.10



1.11 (a)



Present State		Next State $X = 0$	1	Output $X = 0$	1
Reset	S0	S1	S0	0	0
0	S1	S1	S2	0	0
01	S2	S3	S4	0	0
010	S3	S1	S5	0	1
011	S4	S5	S0	1	0
0101 or 0110	S5	S0	S0	1	1

(b) Guidelines:

I. (0,1,3),(0,4,5)
II. (0,1),(1,2),(3,4),(1,5),(0,5)
III. (0,1,2,3),(4,5),(3,5)

For state assignment:

S0 = 000 S1 = 001 S2 = 010 S3 = 011 S4 = 100 S5 = 101

Equations for NAND gate network:

$J_1 = XQ_2$ $K_1 = X + Q_3$

$J_2 = XQ_1'Q_3$ $K_2 = X + Q_3$

$J_3 = X'$ $K_3 = XQ_2' + Q_1$

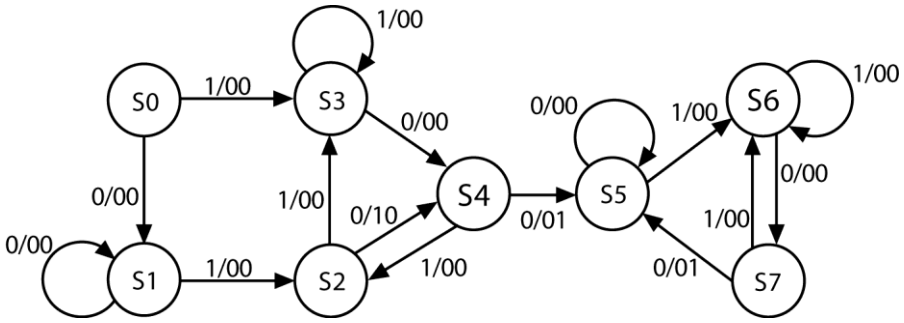
$Z = XQ_2Q_3 + X'Q_1 + Q_1Q_3$

For NOR gate network, use product of sums form:

$K_3 = (X + Q_1)(Q_2')$

$Z = (Q_1 + Q_2)(X' + Q_3)(X + Q_2')$

1.12 (a)



Present State	Next State $X = 0$	Next State 1	Output $X = 0$	Output 1
S0	S1	S3	00	00
S1	S1	S2	00	00
S2	S4	S3	10	00
S3	S4	S3	00	00
S4	S5	S2	01	00
S5	S5	S6	00	00
S6	S7	S6	00	00
S7	S5	S6	01	00

(b) Guidelines:

I. (0,1),(2,3),(4,5,7),(0,2,3),(1,4),(5,6,7)
II. (1,3),(1,2),2x(3,4),(2,5),2x(5,6),(6,7)
III. (0,1,3,5,6),(4,7)

For state assignment:

S0 = 000 S1 = 100 S2 = 001 S3 = 101
S4 = 111 S5 = 011 S6 = 010 S7 = 110

Equations for NAND gate network:

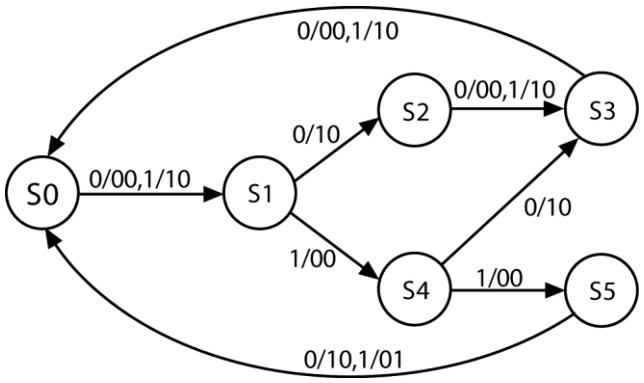
$J_1 = Q_2' + X'Q_3'$ $K_1 = XQ_3' + Q_2$

$J_2 = X'Q_3$ $K_2 = XQ_1Q_3$

$J_3 = XQ_2' + X'Q_1Q_2$ $\quad K_3 = XQ_1'Q_2$
$Z_1 = X'Q_1'Q_2'Q_3$
$Z_2 = X'Q_1Q_2$

For NOR gate network , use product of sums form:
$J_1 = (Q_2' + Q_3')(X' + Q_2')$ $\quad K_1 = (X + Q_2)(Q_2 + Q_3')$
$J_3 = (X + Q_2)(X' + Q_2')(X + Q_1)$

1.13 (a)



Present State	Next State X = 0	Next State 1	Output X = 0	Output 1
S0	S1	S1	00	10
S1	S2	S4	10	00
S2	S3	S3	00	10
S3	S0	S0	00	10
S4	S3	S5	10	00
S5	S0	S0	10	01

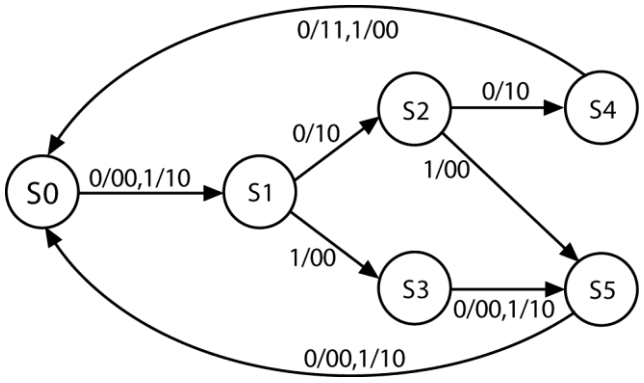
(b) Guidelines:
I. (2,4),2x(3,5)
II. (2,4),(3,5)
III. (0,2,3),(1,4,5)

For state assignment:
S0 = 000 S1 = 010 S2 = 001 S3 = 101 S4 = 011 S5 = 111

Equations for NAND gate network:
$D_1 = Q_1'Q_3$
$D_2 = Q_2'Q_3' + XQ_1'Q_2$
$D_3 = Q_1'Q_3 + Q_2Q_3'$ or $Q_1'Q_3 + Q_1'Q_2$
$S = XQ_2' + X'Q_2$
$V = XQ_1Q_2$

For NOR gate network , use product of sums form:
$S = (X + Q_2)(X' + Q_2')$
$D_2 = Q_1'(Q_2 + Q_3')(X + Q_2')$
$D_3 = Q_1'(Q_2 + Q_3)$

1.14 (a)



Present State	Next State X = 0	1	Output X = 0	1
S0	S1	S1	00	10
S1	S2	S3	10	00
S2	S4	S5	10	00
S3	S5	S5	00	10
S4	S0	S0	11	00
S5	S0	S0	00	10

(b) Guidelines:
I. (4,5),(2,3)
II. (2,3),(4,5)
III. (0,3,5),(1,2,4)

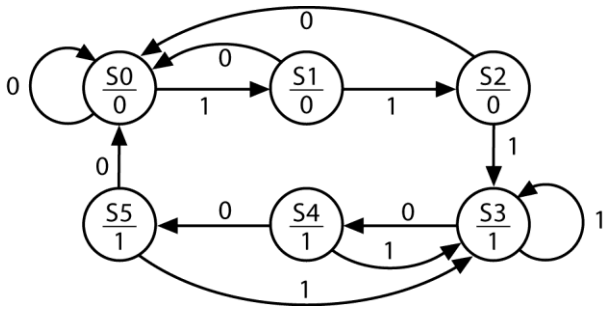
For state assignment:
S0 = 000 S1 = 100 S2 = 101 S3 = 001 S4 = 111 S5 = 011

Equations for NAND gate network:

$J_1 = Q_3'$ $K_1 = X + Q_2$
$J_2 = Q_3$ $K_2 = 1$
$J_3 = Q_1$ $K_3 = Q_2$
$D = X'Q_1 + XQ_1'Q_3$
$B = X'Q_1Q_2$

For NOR gate network, use product of sums form:
$D = (X' + Q_1')(X + Q_1)(Q_1 + Q_3)$

1.15



Present State	Next State X = 0	1	Output
S0	S0	S1	0
S1	S0	S2	0
S2	S0	S3	0
S3	S4	S3	1
S4	S5	S3	1
S5	S0	S3	1

1.16

Present State ($Q_2Q_1Q_0$)	Next State ($Q_2^+Q_1^+Q_0^+$)
000	001
001	010
010	011
011	100
100	101
101	000

$J_2 = Q_1Q_0$ $K_2 = Q_0$

$J_1 = Q_2'Q_0$ $K_1 = Q_0$

$J_0 = 1$ $K_0 = 1$

1.17

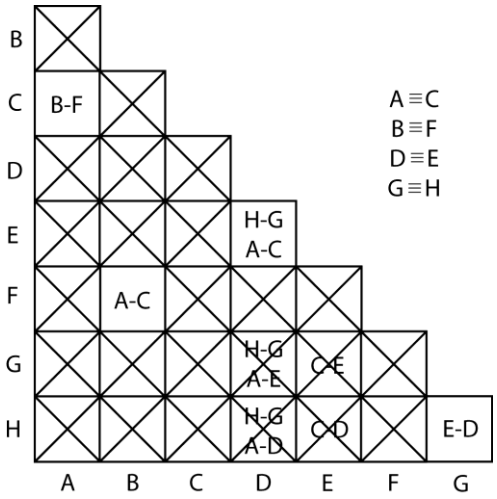
Present State ($Q_2Q_1Q_0$)	Next State ($Q_2^+Q_1^+Q_0^+$)
001	010
010	011
011	100
100	101
101	110
110	001

$D_2 = Q_1Q_0 + Q_2Q_1'$

$D_1 = Q_2'Q_0' + Q_1'Q_0$

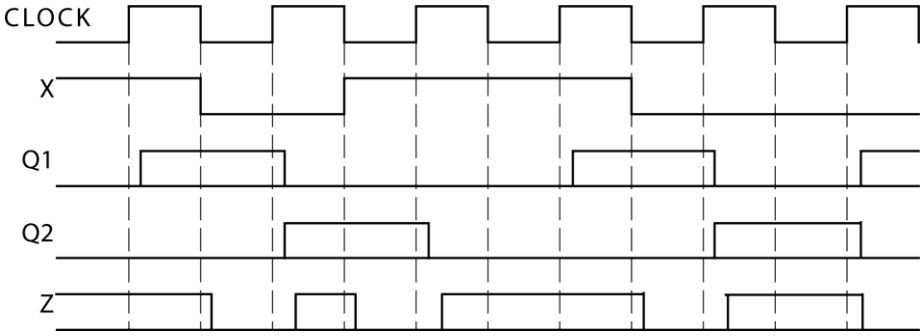
$D_0 = Q_0'$

1.18



present state	next state $X = 0$	next state $X = 1$	output $X = 0$	output $X = 1$
A	B	G	0	1
B	A	D	1	1
~~C~~	~~F~~	~~G~~	~~0~~	~~1~~
D	~~H~~ G	A	0	0
~~E~~	~~G~~	~~C~~	~~0~~	~~0~~
~~F~~	~~C~~	~~D~~	~~1~~	~~1~~
G	G	~~E~~ D	0	0
~~H~~	~~G~~	~~D~~	~~0~~	~~0~~

1.19 (a)



Z should be read just before the rising edge of the clock.

(b) Worst case $t_{xor} + t_p + t_{su} \leq t_{clk}$

$20ns + 10ns + 5ns \leq t_{clk}$

$t_{clk} \geq 35$ ns

Clock Rate = 28.6 MHz

However, the input X changes at the same time as the falling edge of the clock. Data is clocked into D flip-flop at the rising edge of the clock. Therefore, the time t between the falling edge and the rising edge of the clock should satisfy the gate delay of XOR and also the setup time

$t_{clk} \geq 20ns + 5ns$

$t_{clk} \geq 50ns$

Clock Rate = 20 MHz

For proper synchronous operation, both condition 1 and condition 2 should be satisfied.

$t_{clk} \geq$ 50ns is the limiting factor

Therefore, Clock Rate = 20 MHz

(c) Q1 should remain unchanged for 2ns (t_h) after D2 is clocked

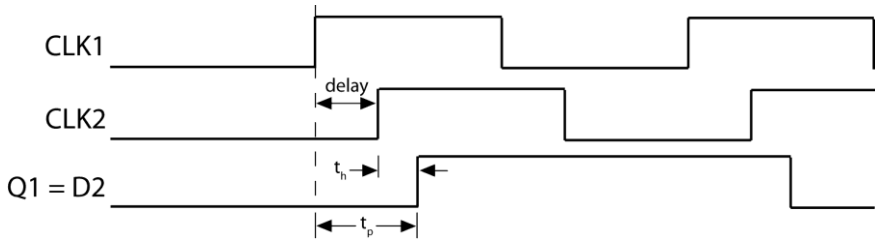
Q1 will be constant for at least 5ns (t_{pmin}) after the rising clock edge

$t_{constnat} = t_h$ + delay = 2ns + delay

$t_{constnat} = t_{pmin}$ = 5ns

delay = $t_{pmin} - t_h$ = 3 ns

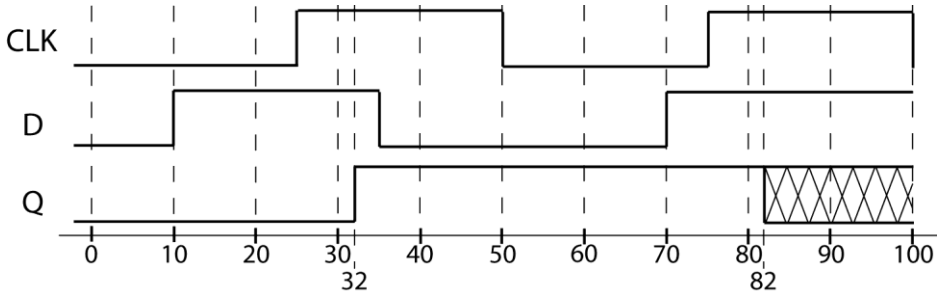
1.20 (a)



delay < $t_{pmin} - t_h$ = 10-2 = 8 ns

(b) $t_{clk} \geq t_{pmin} + t_{su}$ = 15+4 = 19 ns. (worst case occurs when delay = 0)

1.21 (a)



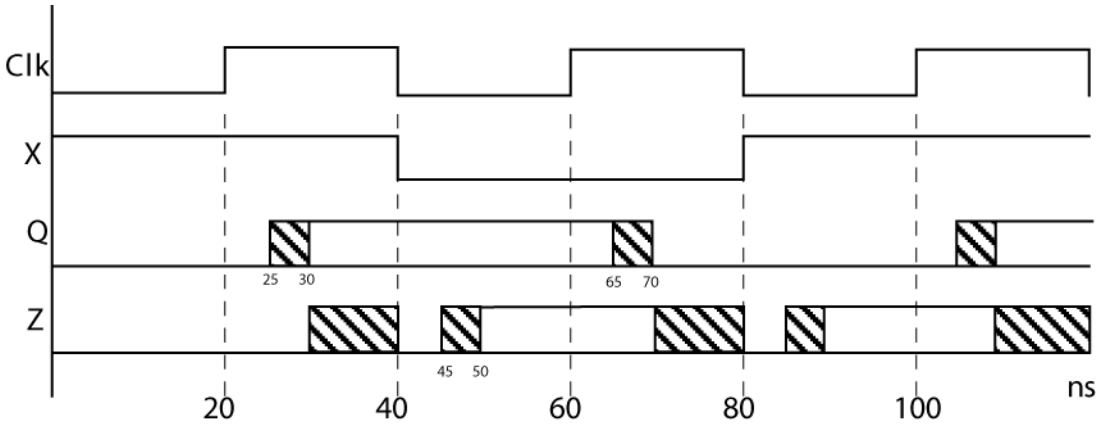
(b) By definition of set up and hold time, *D* should be constant 10 ns (t_{su}) before, and 5 ns (t_h) after the clock edge.

(c) External inputs should not change 18 ns before, and 1 ns after clock edge.

$t_y = t_{cmax} + t_{su}$ = (4+4) + 10 = 18

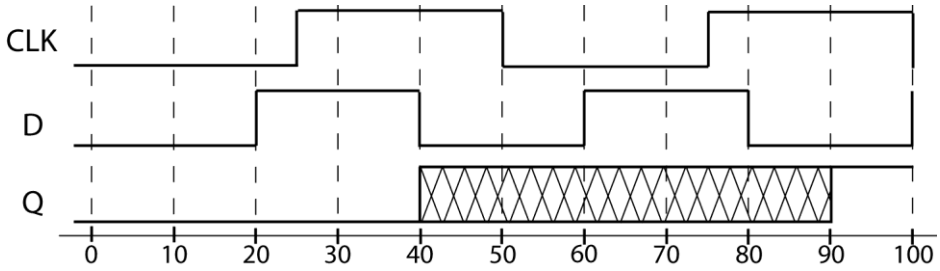
$t_x = t_h - t_{cmin}$ = 5 – (2+2) = 1

1.22 **(a)**



(b) Setup time = 20 ns -10 ns = 10 ns (due to change in *X*)
Hold time = $t_{pmin} + t_{pla}$ = 5 ns+ 5 ns= 10 ns (due to change in *Q*)

1.23



1.24 **(a)** $t_{clk} \geq t_{su} + t_{pmax} + t_{cmax}$
$t_{clk} \geq 5 + 12 + 4 = 21$ ns
$t_{clkmin} = 21$ ns

(b) $t_x = t_h - t_{cmin} = 3 - 1 = 2$ ns. X can change as early as 2 ns after the clock edge.

1.25 **(a)** 16 ns + 24 ns + 8 ns = <u>48 ns.</u> (to satisfy setup time) (hold time is not a problem because it takes at least 12 + 2 = 14 ns from rising clock edge until D changes)

(b) earliest time: to satisfy hold time, $t_h - t_{cmin}$ = 4 ns – 2 ns = <u>2 ns</u>
latest time: to satisfy setup time, 8 ns + 16 ns = 24 ns before rising clock edge. 48 ns clock – 24 ns = <u>24 ns</u> after rising clock edge

1.26 **(a)** The maximum delay path of this circuit starts at flip-flop 2 and ends at flip-flop 1:

$$f_{max} = 1/(t_{pmax} + t_{cmax} + t_{su}) = 1/(24ns + 16ns + 8ns) \approx 20.83\ MHz$$

(b) $$f_{max} = 1/(t_{pmax} + t_{cmax} - t_{skew} + t_{su}) = 1/(24ns + 16ns - 5ns + 8ns) \approx 23.26\ MHz$$

(c) $$f_{max} = 1/(t_{pmax} + t_{cmax} + t_{skew} + t_{su}) = 1/(24ns + 16ns + 5ns + 8ns) \approx 18.87\ MHz$$

(d) $t_y \geq t_h - t_{cxmin} = 4ns - 2ns = 2ns$
$t_x \geq t_{cxmax} + t_{su} = 16ns + 8\ ns = 24\ ns$

X can change 2 ns after and 24 ns before the rising clock edge.

(e) From part (d), X can change 2 ns after and 24 ns before the rising clock edge of flip-flop 1. However, because the rising clock edge of flip-flop 2 is delayed 5 ns from the rising edge of flip-flop 1, then X can change -3 ns after and 29 ns before the rising clock edge of flip-flop 2. In other words, X cannot change between 29 and 3 ns before the rising clock edge of flip-flop 2.

(f) From part (d), X can change 2 ns after and 24 ns before the rising clock edge of flip-flop 1. However, because the rising clock edge of flip-flop 2 is advanced 5 ns from the rising edge of flip-flop 1, then X can change 7 ns after and 19 ns before the rising clock edge of flip-flop 2.

1.27 **(a)** Consider following delays:

input to first FF: $t_{su} = 20ns$
left FF to middle FF: $t_{pmax} + t_{c1max} + t_{skew1} + t_{su} = 10ns + 7ns + 0ns + 20ns = 37ns$
middle FF to right FF: $t_{pmax} + t_{c2max} - t_{skew2} + t_{su} = 10ns + 11ns - 0ns + 20ns = 41ns$
right FF to output: $t_{pmax} = 10ns$

The maximum of these is 41 ns. Therefore, the minimum clock period should be 41 ns.

(b) $t_{clk} \geq t_{pmax} + t_{c1max} + t_{su}$
$41\ ns \geq 10ns + 4\ ns + 20 = 34\ ns$

There is no setup time violation for the middle flip-flop. The setup time margin is 41 – 34 = 7 ns.

(c) $t_{pmin} + t_{c1min} \geq t_h$
$5ns + 1ns < 10ns$
There is a hold time violation for the middle flip-flop.

(d) For negative clock skew:
$t_{skewmin} = t_h - t_{pmin} - t_{cmin} = 10ns - 5ns - 1ns = 4ns$

To fix the hold time violation for the middle flip-flop, make $t_{skew1min}$ = 4 ns and keep $t_{skew2min}$ = 0 ns.

(e) The new worst-case delay of the path from the left flip-flop to the middle flip-flop is:
$t_{pmax} + t_{c1max} + t_{skew1} + t_{su} = 10ns + 4ns + 4ns + 20ns = 38ns$

However, this delay is still less than the 41 ns delay of the path from the middle flip-flop to the right flip-flop. Therefore, t_{clkmin} is still 41 ns.

1.28 **(a)** Consider following delays:

input to first FF: $t_{su} = 10ns$

left FF to middle FF: $t_{pmax} + t_{c1max} + t_{skew1} + t_{su} = 20ns + 7ns + 0ns + 10ns = 37ns$

middle FF to right FF: $t_{pmax} + t_{c2max} - t_{skew2} + t_{su} = 20ns + 11ns - 3ns + 10ns = 38ns$

right FF to output: $t_{pmax} = 20ns$

The maximum of these is 38 ns. Therefore, the minimum clock period should be 38 ns.

(b) $t_{clk} \geq t_{pmax} + t_{c1max} + + t_{skew1} + t_{su}$

$38\ ns \geq 20n + 4ns + 0ns + 10ns = 34\ ns$

There is no setup time violation for the middle flip-flop. The setup time margin is 38 – 34 = 4 ns.

(c) $t_{pmin} + t_{c1min} \geq t_h - t_{skew1}$

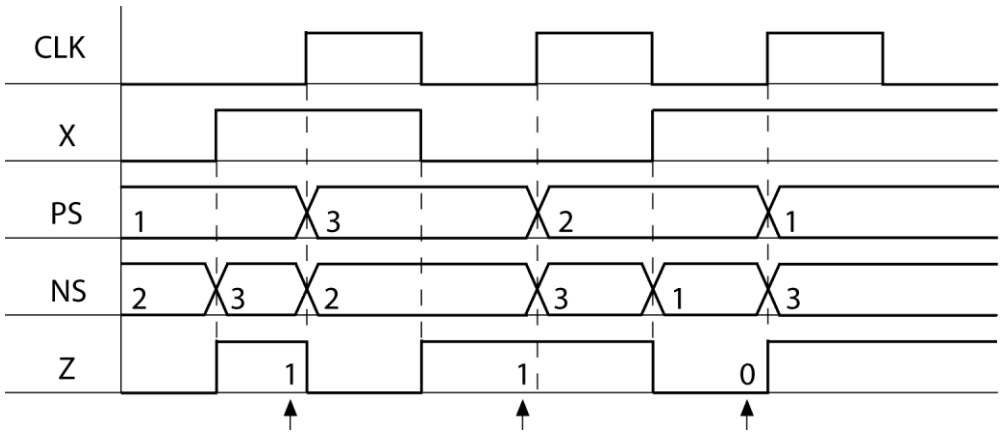
$12ns + 1ns \geq 2ns - 0ns$

There is no hold time violation for the middle flip-flop. The hold time margin is 13 – 2 = 11 ns.

(d) Because both setup and hold time requirements are met for the middle flip-flop, the clock skew delays in place are valid.

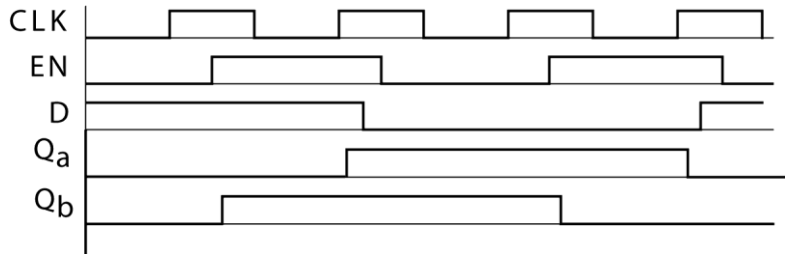
(e) The worst-delay path is still from the middle flip-flop to the right flip-flop. Therefore, t_{clkmin} is still 38 ns.

1.29

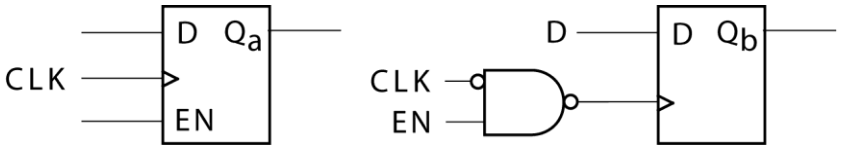


1.30 **(a)** No

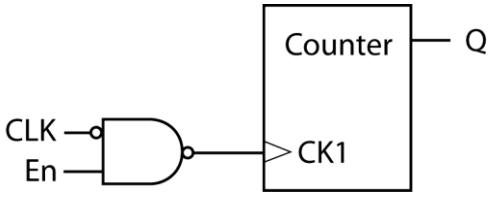
(b)



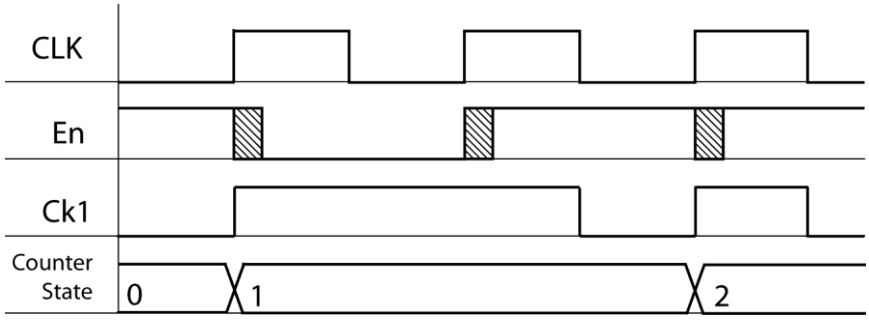
(c)



1.31 **(a)**



(b)



The clock input to the NAND gate in (a) is inverted. After the rising edge of the clock, this input to the NAND gate is a '0', so CK1 will remain a constant '1', regardless of any changes that may occur in the EN input of the gate due to transients.

1.32 *Eni* = 0 *Ena*= 0 *Enb* = 0 *Enc* = 1 *Lda* = 1 *Ldb*= 1 *Ldc*= 0