

## Homework #3

### Notice:

- ✓ Solve the problems 1~8. The problem must be solved by hand. (Use note or blank paper.)
- ✓ Scan and upload your result to KLAS assignment before due date. No late submission is allowed.
- ✓ You should solve the problems yourself. All answers should be written in Korean except any terminologies. No English sentence is allowed.
- ✓ All answers **must have enough solutions** to solve the problem.
- ✓ Deadline – 23:59 on May 25th, 2023, Thursday (UTC+9, Korean Standard Time)

1. Simplify the following Boolean equations using Boolean algebra.

A.  $AC + A'B'C$

$$AC + A'B'C$$

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

$$= C(AB + AB' + AB' + A'B')$$

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

$$= AC + B'C$$

B.  $A'B' + A'BC' + (A+C)'$

$$A'B' + A'BC' + (A+C)'$$

$$= A'B' + A'BC' + A'C$$

$$= A'(B' + BC' + C)$$

$$= A'(B' + BC' + B'C + BC)$$

$$= A'(B' + B'C + BC' + BC)$$

$$= A'(B' + B) = A'$$

C.  $ABCD' + A(BCD)' + (A+B+C+D)'$

$$ABCD' + A(BCD)' + (A+B+C+D)'$$

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

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

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

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

Hint:  $X + (X'Y) = X + Y$  (Redundancy law)

“꿈을 가지십시오. 그리고 정열적이고 명예롭게 이루십시오.”

2. Multiplier is a circuit that multiplies two binary number. For example, if we multiply  $1001_2$  ( $9_{10}$ ) and  $1000_2$  ( $4_{10}$ ), output should be  $0010\ 0100_2$  ( $36_{10}$ ). Design a 2-bit multiplier as shown in Figure 1.

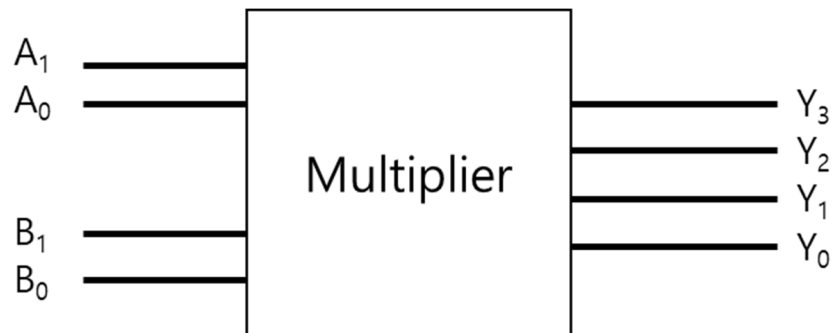


Figure 1 2-bit multiplier

A. Complete a truth table of 2-bit multiplier.

A <sub>1</sub>	A <sub>0</sub>	B <sub>1</sub>	B <sub>0</sub>	Y <sub>3</sub>	Y <sub>2</sub>	Y <sub>1</sub>	Y <sub>0</sub>
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	1	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	1
0	1	1	0	0	0	1	0
0	1	1	1	0	0	1	1
1	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	1	0	0
1	0	1	1	0	1	1	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	1	1	0
1	1	1	1	1	0	0	1

## B. Boolean equation of Y

Karnaugh map 혹은 Quine-McCluskey 방법 등을 이용하여 논리식을 구한다.

$$Y_3 = A_1 A_0 B_1 B_0$$

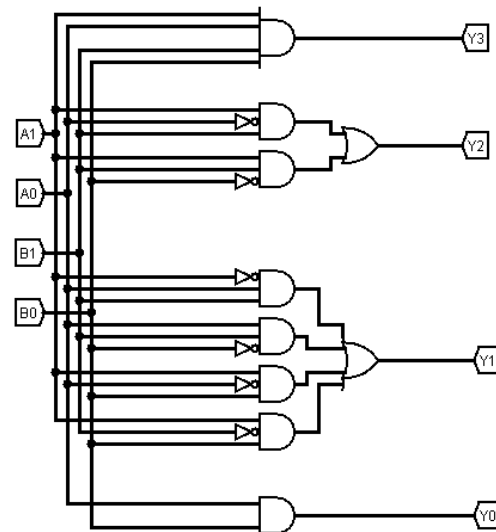
$$Y_2 = A_1 \overline{A_0} B_1 + A_1 B_1 \overline{B_0}$$

$$Y_1 = \overline{A_1} A_0 B_1 + A_0 B_1 \overline{B_0} + A_1 \overline{A_0} B_0 + A_1 \overline{B_1} B_0$$

$$Y_0 = A_0 B_0$$

### C. Schematic of 2-bit multiplier

예시)

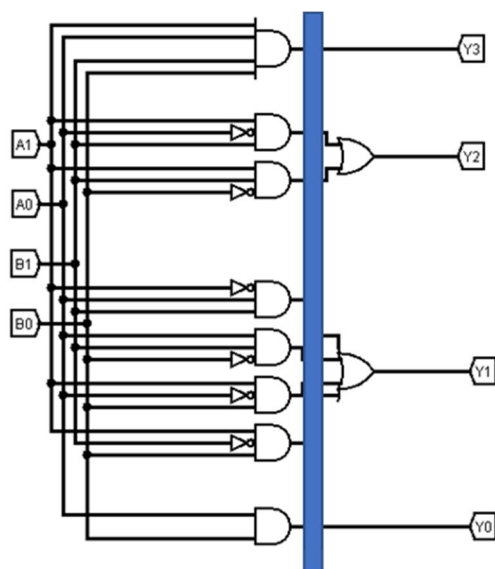


D. In your schematic, which path is critical path? You should explain why.

예시) Inverter+AND+OR 인 path가 critical. 제일 길어서.

E. Suppose your multiplier is between two flip-flops. You want to increase the clock frequency of your circuit; can you modify your circuit to be pipelined? (Assume you can add only one register) If so, where is the best place to have the register inserted?

예시) 아래 그림처럼 AND랑 OR 사이에 넣으면 균형도 맞고 동작주파수를 약 2배로 올릴 수 있음



3. Find a minimal Boolean equation for the function in table below using K-map and Quine-McCluskey method. Then compare the results of K-map and Quine-McCluskey method. Is it same or not? If not, describe why.

A	B	C	D	Y
0	0	0	0	0
0	0	0	1	X
0	0	1	0	1
0	0	1	1	0
0	1	0	0	X
0	1	0	1	0
0	1	1	0	X
0	1	1	1	X
1	0	0	0	X
1	0	0	1	1
1	0	1	0	X
1	0	1	1	0
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	X

K-map

AB/CD	00	01	11	10
00	0	X	0	1
01	X	0	X	X
11	0	1	X	1
10	X	1	0	X

Equation :  $AC'D + CD'$

(This page is left blank. Feel free to use.)

Column 1			Column 2			Column 3		
1	0001	V	(1,9)	-001	*	(2,6,10,14)	--10	*
2	0010	V	(2,6)	0-10	V	(4,6,12,14)	-1-0	*
4	0100	V	(2,10)	-010	V			
6	0110	V	(4,6)	01-0	V			
9	1001	V	(4,12)	-100	V			
10	1010	V	(6,7)	011-	*			
12	1100	V	(6,14)	-110	V			
7	0111	V	(9,11)	10-1	*			
11	1011	V	(9,13)	1-01	*			
13	1101	V	(10,11)	101-	*			
14	1110	V	(10,14)	1-10	V			
			(12,13)	110-	*			
			(12,14)	11-0	V			

	-001	011-	10-1	1-01	101-	110-	--10	-1-0
2							X	
9	X		X	X				
10					X		X	
13				X		X		

Equation :  $AC'D + CD' \rightarrow \text{Same}$

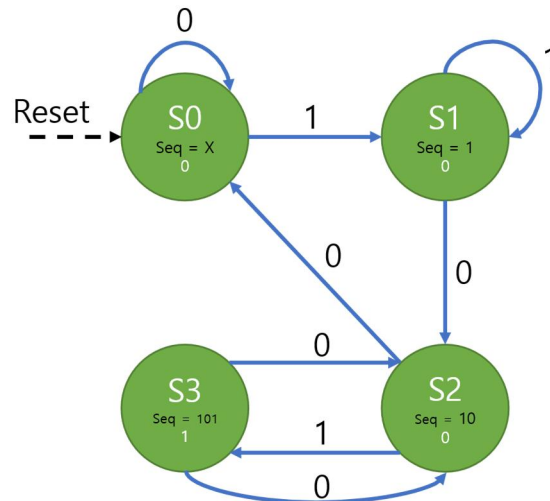
“꿈을 가지십시오. 그리고 정열적이고 명예롭게 이루십시오.”



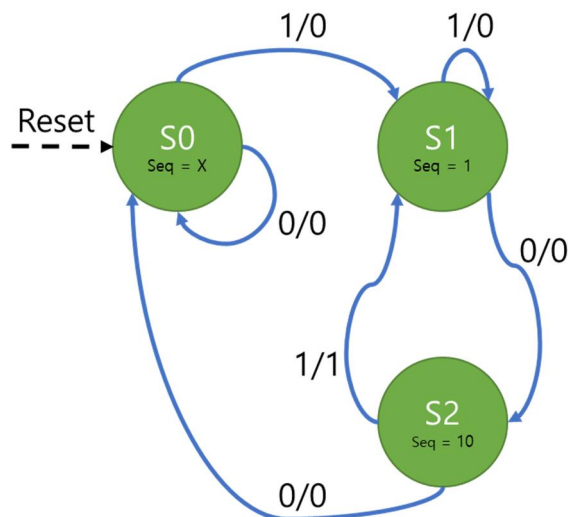
4. Design a sequence detector whose output changes from 0 to 1 when the sequence 101 of an input is detected. For example, if the sequence of 10101 is given as input, the sequence of output (except initial state) should be 00101 because there are two 101 sequences.

A. Draw the state transition diagram and find the minimum number of flip-flops in each FSM.

- i. In Moore machine (Hint: need 4 states)



- ii. In Mealy machine  
(Hint: need 3 states, Hint 2: You can convert from Moore machine to Mealy machine.)



B. Draw an encoded state transition table and derive the equation(s) of the flipflop input(s)

i. In Moore machine

Moore FSM example:

State	Encoding
S0 (X)	00
S1 (1)	01
S2 (10)	10
S3 (101)	11

Current State		Input	Next state	
Q <sub>1</sub>	Q <sub>0</sub>	A	D <sub>1</sub>	D <sub>0</sub>
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	0
1	0	1	1	1
1	1	0	1	0
1	1	1	0	1

$$D_0 = A$$

$$D_1 = Q_0A' + Q_1Q_0'A$$

B. Draw an encoded state transition table and derive the equation(s) of the flipflop input(s)  
(cont'd)

ii. In Mealy machine

Mealy FSM example:

State	Encoding
S0 (X)	00
S1 (0)	01
S2 (01)	10

Current State		Input	Next state	
Q <sub>1</sub>	Q <sub>0</sub>	A	D <sub>1</sub>	D <sub>0</sub>
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	0
1	0	1	0	1

$$D_0 = A$$

$$D_1 = Q_0 A'$$

C. Draw an output table and derive the equation of output signal Y

i. In Moore machine

Moore FSM example:

State	Output
$Q_1Q_0$	Y
00	0
01	0
10	0
11	1

$$Y = Q_0Q_1$$

ii. In Mealy machine

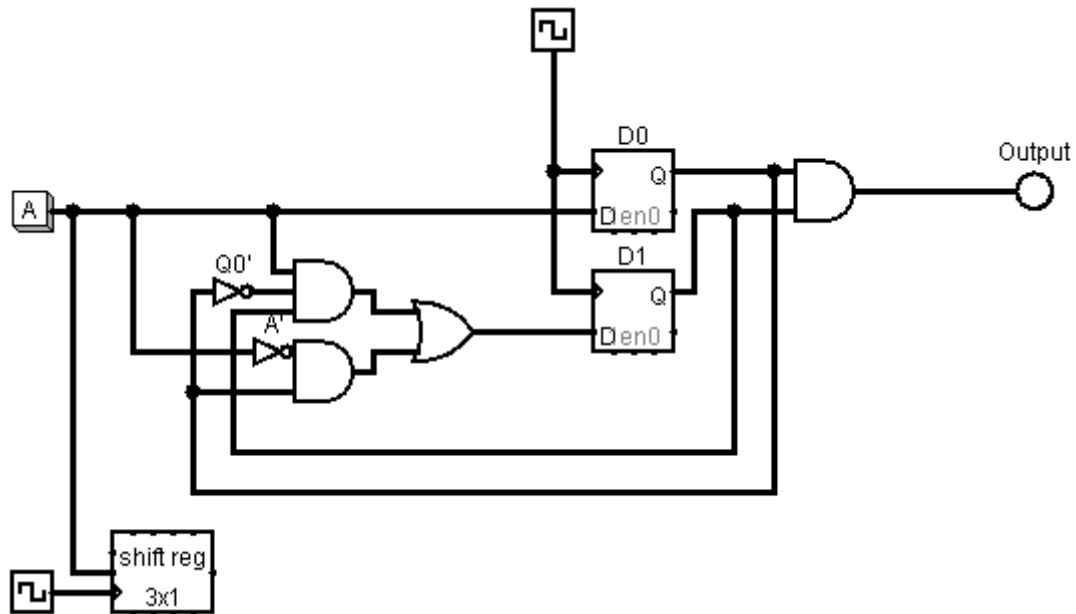
Mealy FSM example:

State	Input A	Output
$Q_1Q_0$	A	Y
00	0	0
00	1	0
01	0	0
01	1	0
10	0	0
10	1	1

$$Y = Q_1A$$

D. Draw a schematic of the sequence detector.

i. In Moore machine



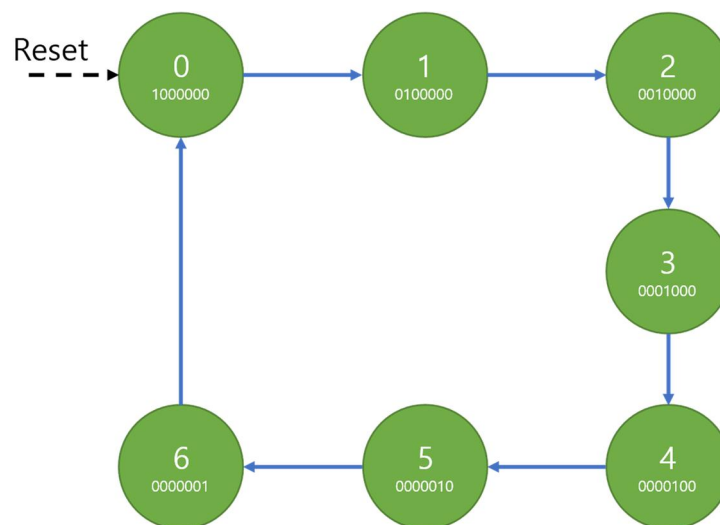
Shift register는 필요 없음



5. Ring counter, also known as one-hot counter, is a special kind of counter which has one-hot output. Unlike N-bit binary counter, which can count  $2^N$  numbers, N-bit ring counter can count N numbers. For example, outputs of 3-bit ring counter are 100, 010, 001 for 0, 1, 2, respectively. Design a 7-bit ring counter with no input, which starts from 1000000 and ends with 0000001. On each clock edge, the output should advance to next one-hot. After reaching 0000001, it goes back to 1000000.

Number	One-hot						
0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0
2	0	0	1	0	0	0	0
3	0	0	0	1	0	0	0
4	0	0	0	0	1	0	0
5	0	0	0	0	0	1	0
6	0	0	0	0	0	0	1

- A. Finite state machine with output in each state transition diagram of ring counter with a minimum number of states.



\*\*\*\* Moore machine의 경우 input과 output이 꼭 있어야 함

“꿈을 가지십시오. 그리고 정열적이고 명예롭게 이루십시오.”

- B. Design a ring counter in finite state machine using states you defined above. You should provide state encoding table, state transition table, output table and Boolean equation of output. Also, you should use **minimum number of flip-flops** and simplify your equation as you can. (No schematic is required)

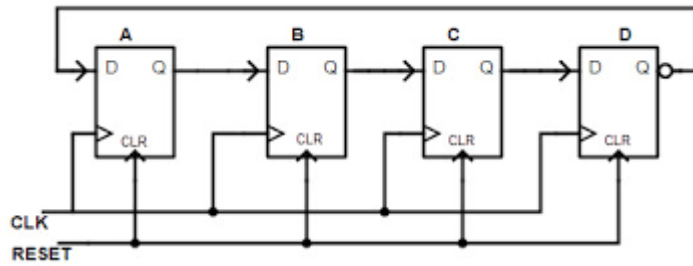
State	Encoding
0	000
1	001
2	010
3	011
4	100
5	101
6	110

Current state			Next state		
Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	0	0	0

Current state			Output
Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>0</sub>	Y[4:0]
0	0	0	10000
0	0	1	01000
0	1	0	00100
0	1	1	00010
1	0	0	00001



C. Is there any other easy way to design a 7-bit ring counter?



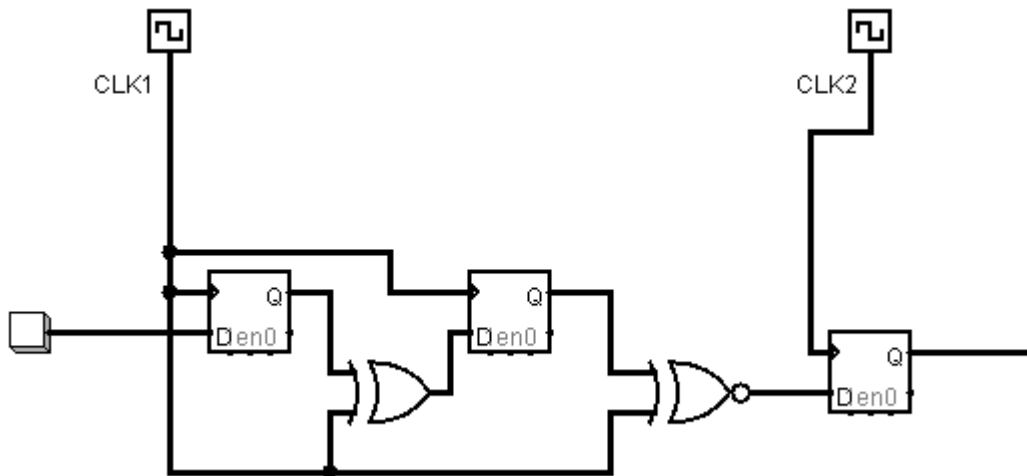
Flip-flop의 직렬 연결로 쉽게 구현 가능

위와 같은 회로를 구성하고 첫번째 Flip-Flop의 (Pre)set, 나머지 Flip-Flop의 Reset을 묶어 1000....0으로 초기화가 가능하고 한번 초기화 되고 난 이후에는 Clock만 주면 Ring counter로 동작함

6. What is the problem of circuit below? Is there any possible way to solve the problem?

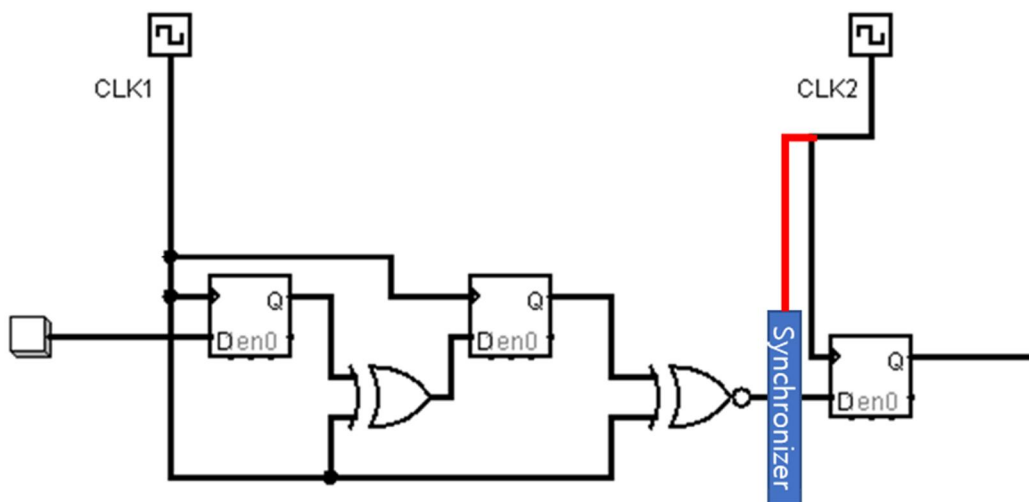
출제오류로 후속 처리 예정...

원래 의도)



두 Clock domain이 다르므로 synchronous circuit이 될 수 없음

이를 해결하기 위해서는 2FF synchronizer를 달아줄 필요가 있을 것



그러나 실제로는 Clock 이 input 에 연결되어 있으므로 성립할 수 없는 회로입니다.

비록 문제는 잘못됐지만 여러분은 synchronizer 가 무엇인지 다시 확인해 보시기 바랍니다.

“꿈을 가지십시오. 그리고 정열적이고 명예롭게 이루십시오.”

7. A field programmable gate array (FPGA) uses look-up tables (LUTs) rather than logic gates to implement combinational logic. The ESAL NoGwaje 3 FPGA has propagation and contamination delays of 0.49 and 0.24 ns, respectively, for each LUT. It also contains flip-flops with propagation and contamination delays of 0.71 and 0.31 ns, and setup and hold times of 0.51 and 0 ns, respectively.
- A. If you are building a logic circuit that needs to run at 110 MHz, how many consecutive LUTs can you use between two flip-flops? Assume there is no clock skew and no delay through wires between LUTs.

$$T_c = 1 / 110\text{Mhz} = 9.1 \text{ ns}$$

$$T_c \geq t_{pcq} + Nt_{LUT} + t_{setup}$$

$$9.1 \geq 0.71 + N(0.49) + 0.51$$

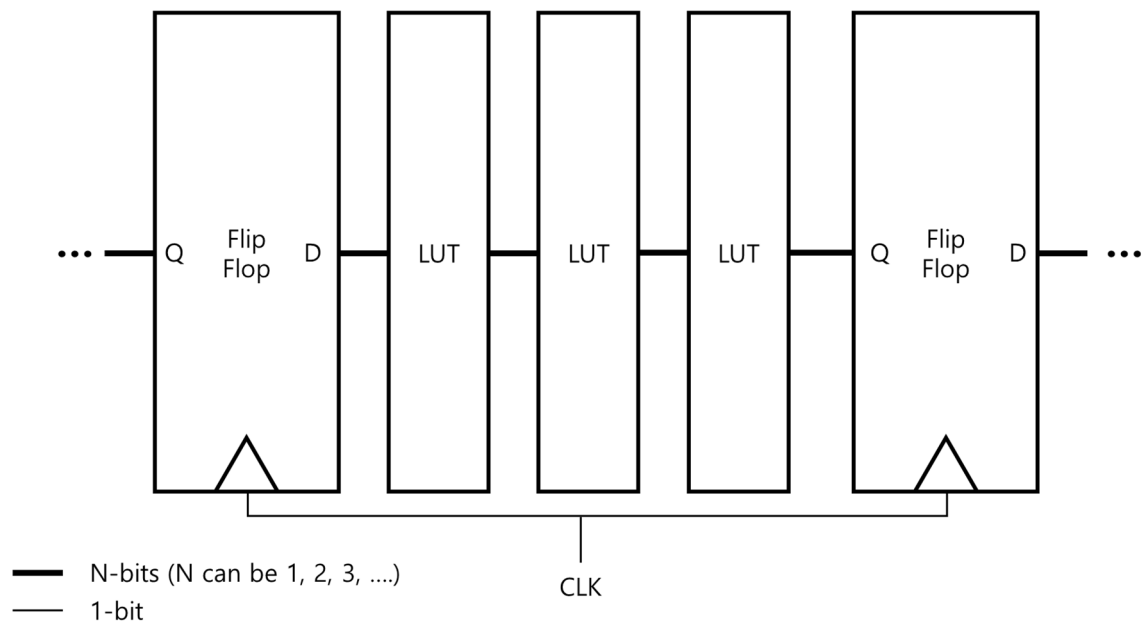
$$16.08 \geq N \rightarrow N = 16$$

- B. Suppose that all paths between flip-flops pass through at least one LUT. How much clock skew can the FPGA have without violating the hold time?

$$t_{skew} < (t_{ccq} + t_{cd}) - t_{hold}$$

$$t_{skew} < 0.71 + 0.24 = 0.95\text{ns}$$

C. Assume a circuit which has three LUTs between two flip-flops. What is the maximum operating **frequency** of FPGA?



$$T_c \geq t_{pcq} + Nt_{LUT} + t_{setup}$$

$$T_c \geq 0.71 + 3 \cdot (0.49) + 0.51 = 2.69 \text{ ns}$$

Max. Frequency =  $1 / 2.69 \text{ ns} = 372 \text{ Mhz}$

8. Compute the MTBF of a two-flipflop synchronizer and a three-flipflop synchronizer.

$t_{setup} = t_{hold} = t_{dCQ} = t = 200\text{ps}$   
 $t_{ck} = 2\text{ns}$   
 must sample a  $f_{input} = 1\text{MHz}$  asynchronous signal  
 $\tau = 100\text{ps}$  &  $G = 3$

### 2FF Synchronizer

$$\begin{aligned}
 P_E &= \frac{0.2\text{ns} + 0.2\text{ns}}{2\text{ns}} = 0.2 \\
 P_U &= \frac{e^{-\frac{1.6\text{ns}}{0.1\text{ns}}}}{3} = \frac{e^{-16}}{3} = 3.75 \times 10^{-8} \\
 P_F &= P_E P_U = 7.50 \times 10^{-9} \\
 R_F &= f_{input} P_F = 7.50 \times 10^{-3} [s^{-1}] \\
 MTBF &= \frac{1}{R_F} = 133 [sec]
 \end{aligned}$$

### 3FF Synchronizer

$$\begin{aligned}
 P_E &= \frac{0.2\text{ns} + 0.2\text{ns}}{2\text{ns}} = 0.2 \\
 P_U &= \frac{e^{-\frac{3.2\text{ns}}{0.1\text{ns}}}}{3} = \frac{e^{-32}}{3} = 4.22 \times 10^{-15} \\
 P_F &= P_E P_U = 8.44 \times 10^{-16} \\
 R_F &= f_{input} P_F = 8.44 \times 10^{-10} [s^{-1}] \\
 MTBF &= \frac{1}{R_F} = 1.18 \times 10^9 [sec] = 37 [year]
 \end{aligned}$$