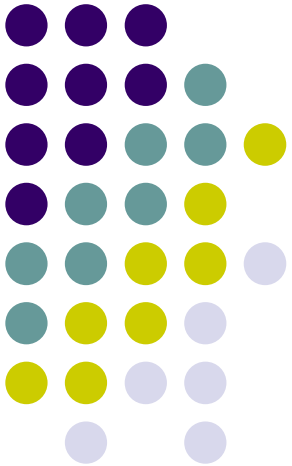
CSE 140 Discussion

Chengmo Yang

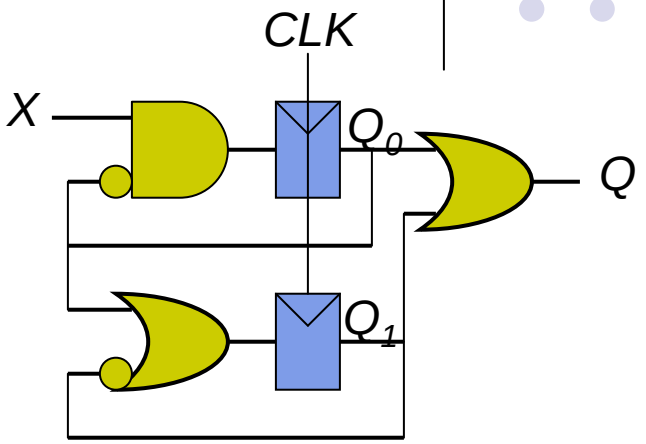
05/11/09



3.28

Step1: get next state function

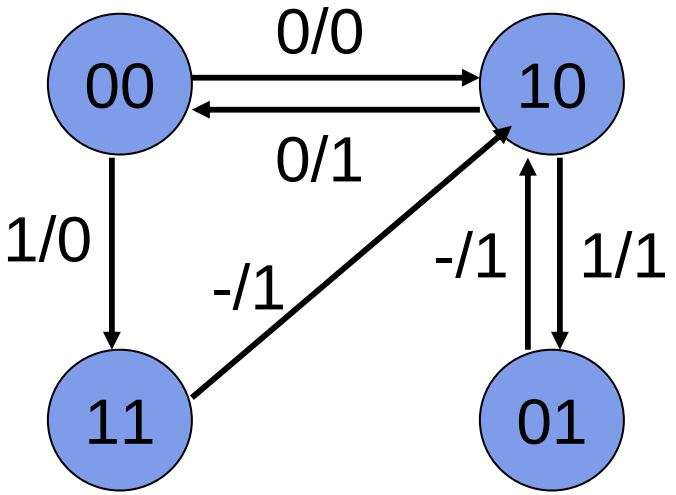
- $Q_0(t+1) = X\ Q_0(t)'$
- $Q_1(t+1) = Q_0(t)+Q_1(t)'$



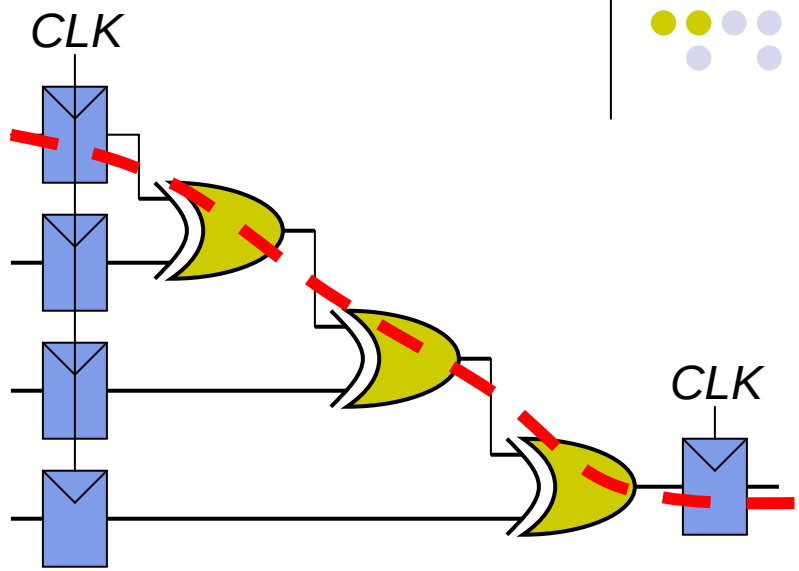
Step2: get state table

Q_1Q_0	Q	NS (X = 0)	NS (X = 1)
00	0	10	11
01	1	10	10
10	1	00	01
11	1	10	10

Step3: get state diagram



3.30 (a)



Requirements:

- $T_c \geqslant T_{pcq} + T_{pd} + T_{setup}$
- $T_{ccq} + T_{cd} \geqslant T_{hold}$

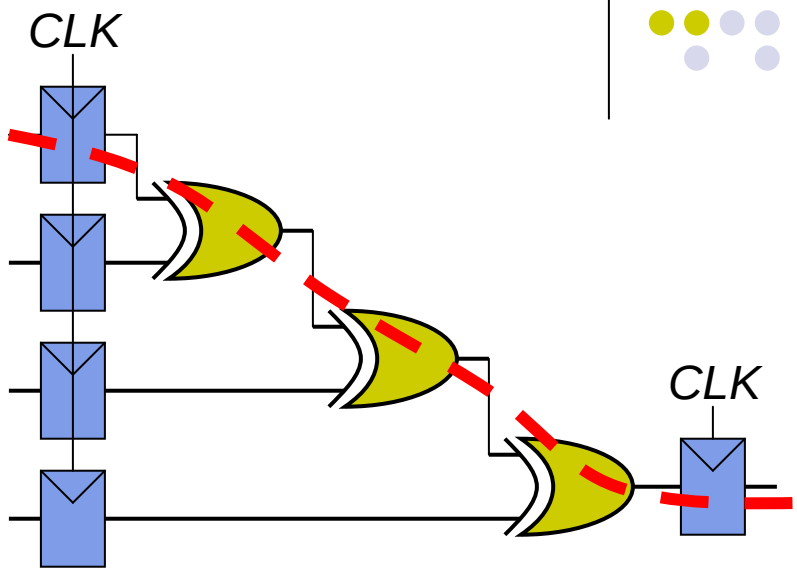
Longest path?

- $T_c \geqslant T_{pcq} + 3*T_{pd} + T_{setup}$
  $T_c \geqslant 70 + 3*100 + 60 = 430$ ps
- Max Frequency = $1/T_c = 2.33$ GHz

3.30 (b)

With clock skew:

- $T_c \geq T_{pcq} + T_{pd} + T_{setup} + T_{skew}$
- $T_{ccq} + T_{cd} \geq T_{hold} + T_{skew}$



- Max Frequency $\geq$ 2 GHz → $T_c \leq$ 500 ps
- $T_c \geq T_{pcq} + 3*T_{pd} + T_{setup} + T_{skew}$

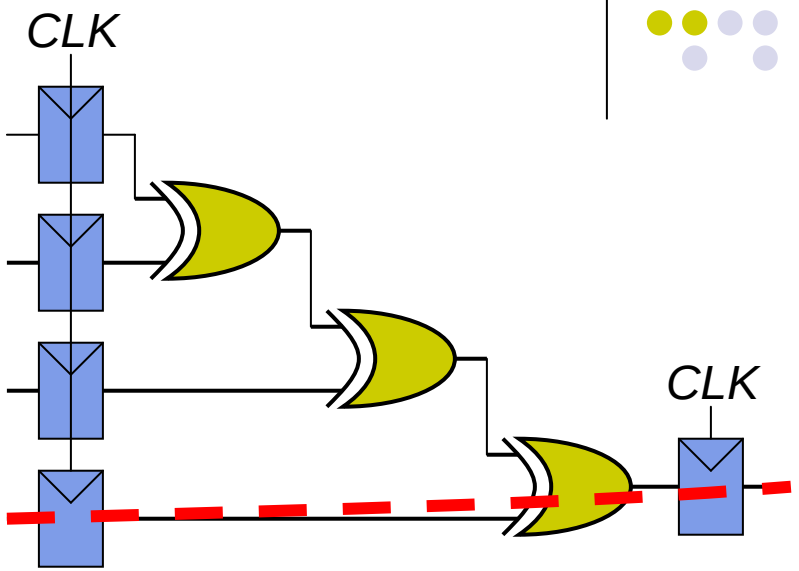
  $500 \geq 70 + 3*100 + 60 + T_{skew}$

  $T_{skew} \leq 70$ ps

3.30 (c)

With clock skew:

- $T_c \geq T_{pcq} + T_{pd} + T_{setup} + T_{skew}$
- $T_{ccq} + T_{cd} \geq T_{hold} + T_{skew}$



Shortest Path?

- $T_{ccq} + T_{cd} \geq T_{hold} + T_{skew}$

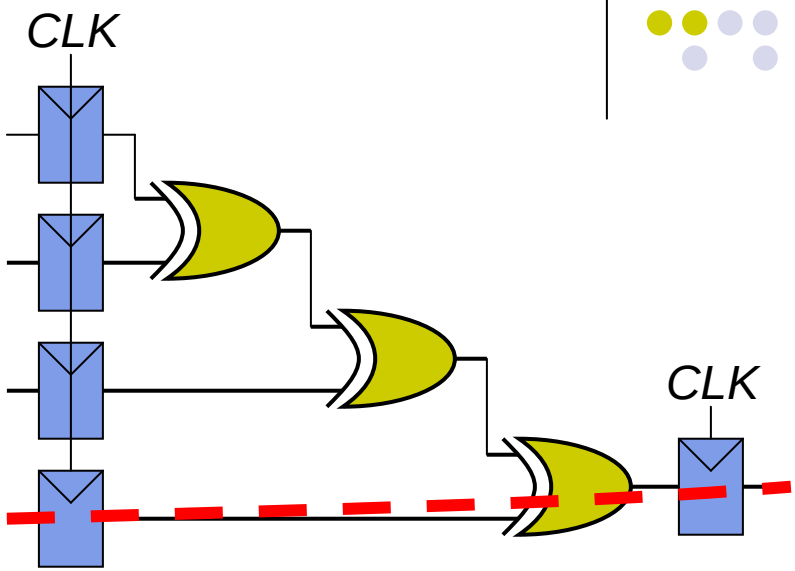
  $50 + 55 \geq 20 + T_{skew}$

  $T_{skew} \leq 85$ ps

3.30 (d)

With clock skew:

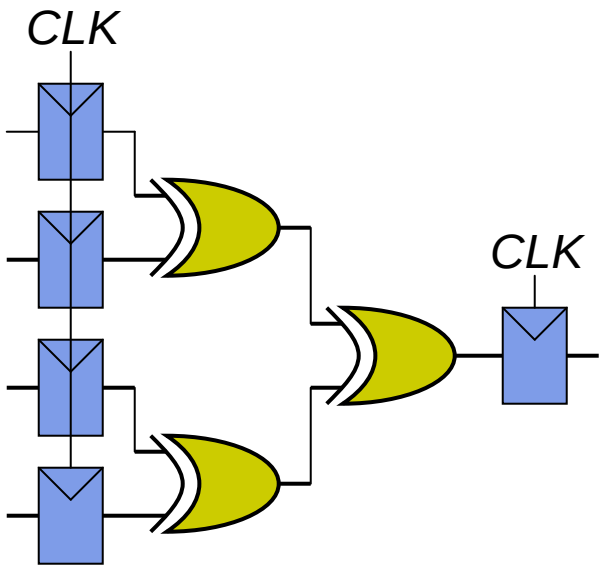
- $T_c \geq T_{pcq} + T_{pd} + T_{setup} + T_{skew}$
- $T_{ccq} + T_{cd} \geq T_{hold} + T_{skew}$



Redesign the circuit?

Idea: increase shortest path, reduce longest path!

- $T_c \geq T_{pcq} + 2*T_{pd} + T_{setup} + T_{skew}$
  $T_c \geq 330 + T_{skew}$
- $T_{ccq} + 2T_{cd} \geq T_{hold} + T_{skew}$
  $T_{skew} \leq 140$ ps



2.29

- First need to get *X*

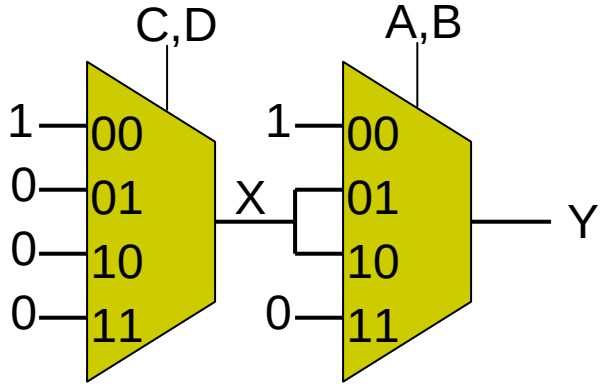
  $X = C'D'$

- Then we can get *Y*

  $Y = A'B' + A'BX + AB'X$

  $Y = A'B' + A'BC'D' + AB'C'D'$

  $Y = A'B' + A'C'D' + B'C'D'$



- **MUX is universal**
  - **Can be used to implement all Boolean functions**

2.30

Implement $Y = A'B'C' + ABC$ using

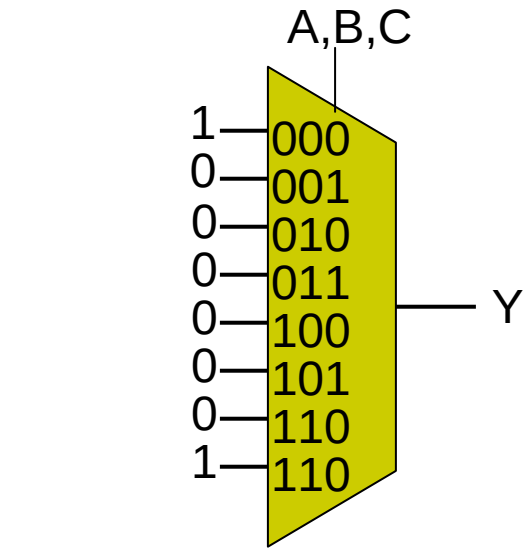
- An 8:1 MUX

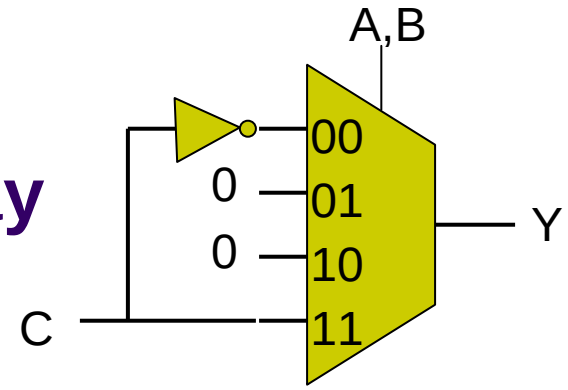
  Trivial

- A 4:1 MUX and a inverter

  Use A, B to select C and C'

- **All 3-input boolean functions can be implemented in this way**





2.30

Implement *Y = A'B'C' + ABC* using

- A 2:1 MUX, and two other gates
  - Use C to select A'B' and AB
  - A'B' = (A+B)'