

CSE140 L

Instructor: Thomas Y. P. Lee
February 8, 2006



Agenda

- ◆ *Asynchronous Counter*
- ◆ *Synchronous Counter*
- ◆ *Gray Counter*
- ◆ *Johnson Counter*
- ◆ *Pseudo Random Sequencer*
- ◆ *Binary Counter*
- ◆ *Converting Between FFs*
- ◆ *Minimum Clock Period*
- ◆ *Clock Skew*



Issues

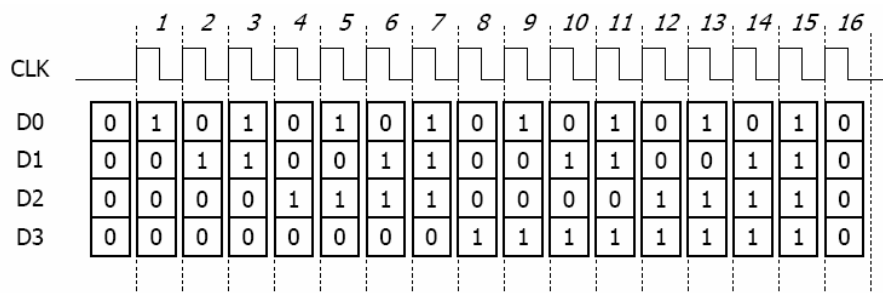


- ◆ *TA Office Hours and Lab Hours*
- ◆ *Why we want to build some digital circuits already exist in Altera Library?*
- ◆ *Lab3 – Vending Machine*
- ◆ *Lab4 – TA will have one to one meeting and oral presentation on lab report*

Counter



- A Counter counts the number of clock cycles
 - Up/down counting



Asynchronous and Synchronous Counter



Asynchronous counter

- ❖ No universal global clock signal
- ❖ Less area
 - You have more flexibility in designing FSMs
- ❖ Less power
 - Eliminates simultaneous D flip-flop switching during clock transitions
- ❖ High speed
 - In terms of throughput
- ❖ But ...hard to design and error prone
 - Complicate protocol for inter-module communication
 - Timing

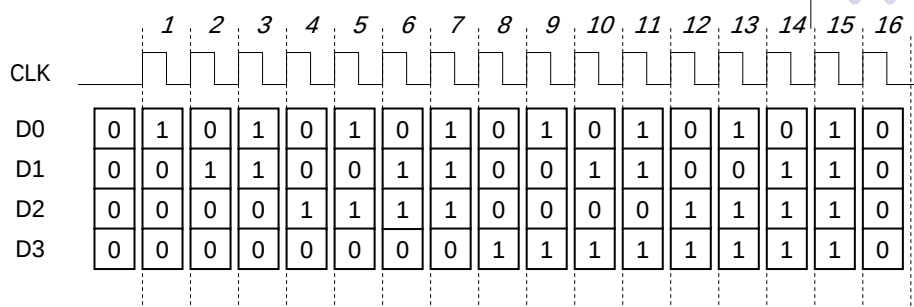
Synchronous counter

All flip-flops are governed by a universal global clock signal

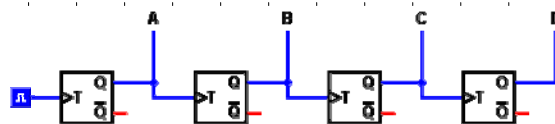
Asynchronous Counter



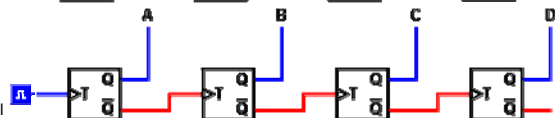
This is the result that we would like to see:



Up Counter

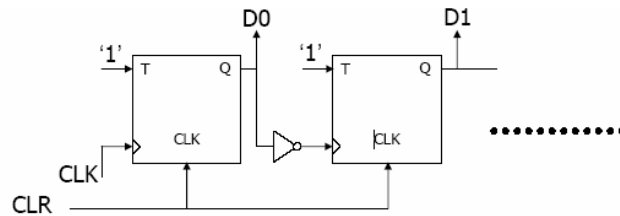


Down Counter

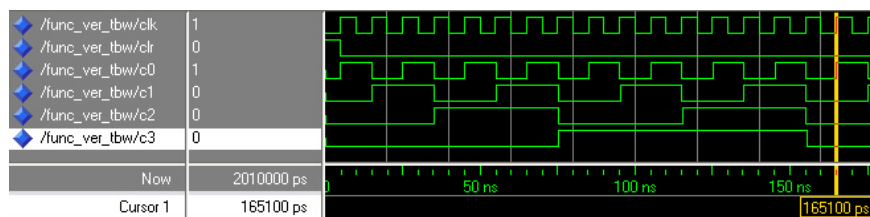


<http://www.play-hookey.com/digital>

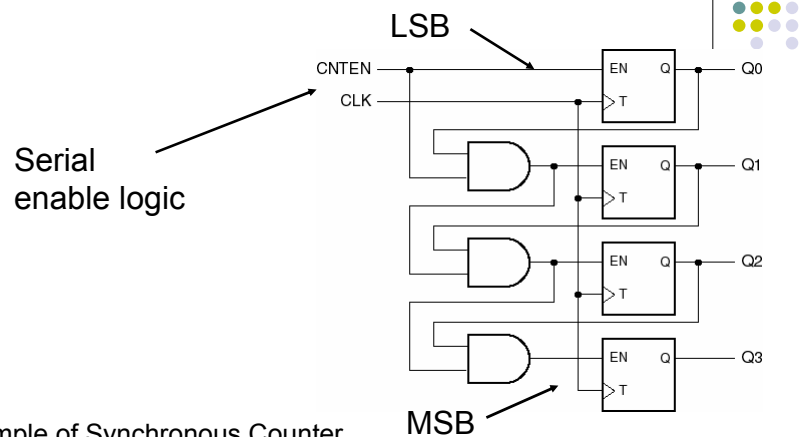
Asynchronous Counter (Cont.)



Verification



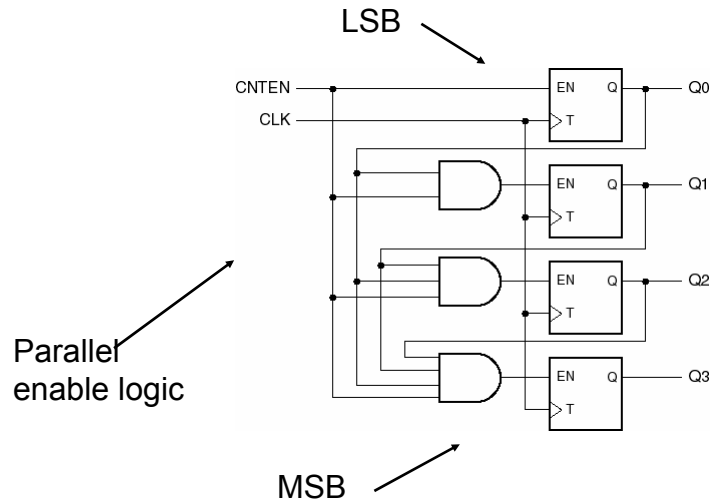
Synchronous Counter



Example of Synchronous Counter

- Gray Counter
- Johnson Counter
- Binary Counter (Up/Down)

Synchronous Counter (cont.)



Synchronous Counter

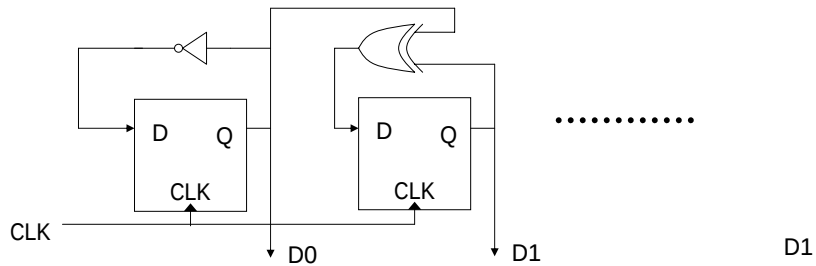
This is the result that we would like to see:

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
CLK																	
D0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
D1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
D2	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0
D3	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0

Synchronous Counter (Cont.)



- $D0_i = D0'_{i-1}$
- $D1_i = D0_{i-1} \cdot D1'_{i-1} + D0'_{i-1} \cdot D1_{i-1} = D0_{i-1} \oplus D1_{i-1}$
- $D2_i = ?$
- $D3_i = ?$



What is Gray Code?



- What is Gray Code?
Gray code is a number representation, Invented by Emile Baudot (1845-1903)
- Gray Code Algorithm
 1. For G0, write one 0s followed by two 1's, two 0's, and so on
 2. For G1, write two 0s followed by alternating groups of four 1s and four 0s
 3. For Gn, start by writing 2^n 0s, then write alternating groups of 2^{n+1} 1s and 0s

Decimal Number	Binary Number	Gray Code Number
	B3 B2 B1 B0	G3 G2 G1 G0
0	0 0 0 0	0 0 0 0
1	0 0 0 1	0 0 0 1
2	0 0 1 0	0 0 1 1
3	0 0 1 1	0 0 1 0
4	0 1 0 0	0 1 1 0
5	0 1 0 1	0 1 1 1
6	0 1 1 0	0 1 0 1
7	0 1 1 1	0 1 0 0
8	1 0 0 0	1 1 0 0
9	1 0 0 1	1 1 0 1
10	1 0 1 0	1 1 1 1
11	1 0 1 1	1 1 1 0
12	1 1 0 0	1 0 1 0
13	1 1 0 1	1 0 1 1
14	1 1 1 0	1 0 0 1
15	1 1 1 1	1 0 0 0

What is Gray Code? (Cont.)



a	b	c	d	e
0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0
0 0 0 1	0 0 0 1	0 0 0 1	0 0 0 1	0 0 0 1
	0 0 1 1	0 0 1 1	0 0 1 1	0 0 1 1
		0 0 1 0	0 0 1 0	0 0 1 0
			0 1 1 0	0 1 1 0
				0 1 1 1

Start with 0, change the least significant bit that forms a new code word

Gray Counter



- Every two neighboring states differ in one bit
- May reduce hazards in digital circuit

The following is the result that we would like to see:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
CLK																
D0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
D1	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
D2	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
D3	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0

Gray Counter (Cont.)



State Transition Table from Gray Code Counting

Corresponding Truth Table of States

Present State of D0, D1, D2, D3				Next State of D0, D1, D2, D3			
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	1
0	0	1	1	0	0	1	0
0	0	1	0	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	0	1	0	1
0	1	0	1	0	1	0	0
0	1	0	0	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	1
1	1	1	1	1	1	1	0
1	1	1	0	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	0	0	1
1	0	0	1	1	0	0	0
1	0	0	0	0	0	0	0

Inputs (Present) of D3, D2, D1, D0				Outputs (Next) of D3, D2, D1, D0			
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	1
0	0	1	0	0	1	1	0
0	0	1	1	0	0	1	0
0	1	0	0	1	1	0	0
0	1	0	1	0	1	0	0
0	1	1	0	0	1	1	1
0	1	1	1	0	1	0	1
1	0	0	0	0	0	0	0
1	0	0	1	1	0	0	0
1	0	1	0	1	0	1	1
1	0	1	1	1	0	0	1
1	1	0	0	1	1	1	1
1	1	0	1	1	0	1	0
1	1	1	1	1	1	1	0

Gray Counter (Cont.)



Implement with D Flip Flops

	00	01	11	10	D1 D0
00	0	0	0	X	
01	X	X	1	0	
11	1	X	0	1	
10	1	1	X	X	
D3 D2					

	00	01	11	10	D1 D0
00	0	0	1	X	
01	X	X	0	1	
11	1	X	0	1	
10	0	1	X	X	
D3 D2					

$$(D3)_{\text{next}} = D3.D1' + D3.D0' + D3'.D2.D0 \quad (D2)_{\text{next}} = D2'.D1 + D2.D0' + D3.D2'.D0$$

Gray Counter (Cont.)



Implement with D Flip Flops

	00	01	11	10	D1 D0
00	0	1	1	X	
01	X	X	0	1	
11	1	X	0	1	
10	0	0	X	X	
D3 D2					

	00	01	11	10	D1 D0
00	1	1	0	X	
01	X	X	0	1	
11	0	X	0	1	
10	1	0	X	X	
D3 D2					

$$(D1)_{\text{next}} = D2.D0' + D3'.D2'.D0$$

$$(D0)_{\text{next}} = D3'.D1' + D1.D0' + D3.D2'.D0'$$

Gray Counter (Cont.)



Hint: What if we use T Flip-Flop

T0	00	01	11	10	D0D1
00	1		1		
01		1		1	
11	1		1		
10		1		1	

D2D3

$$T0_{\text{next}} = (D0 \oplus D1 \oplus D2 \oplus D3)'$$

$$T1 = ?$$

T2	00	01	11	10	D0D1
00		1			
01					
11		1			
10					

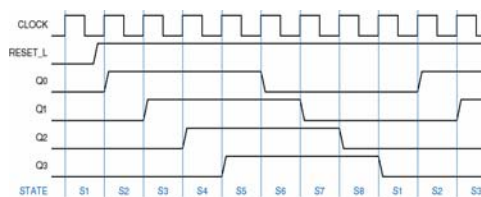
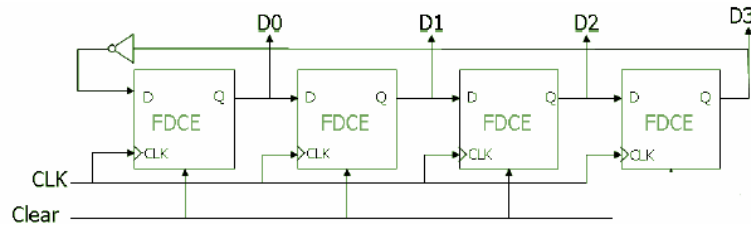
D2D3

$$T2_{\text{next}} = D0'.D1'.(D2 \oplus D3)'$$

$$T3 = ?$$

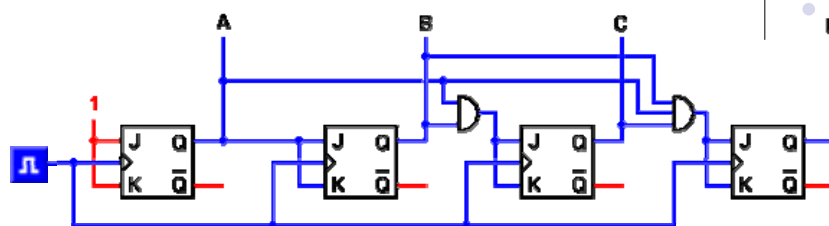
Johnson Counter

- Johnson Counter - "Twisted ring" counter or Mobius Counter
- Last output inverted and feed back to the first input



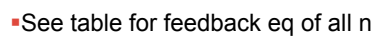
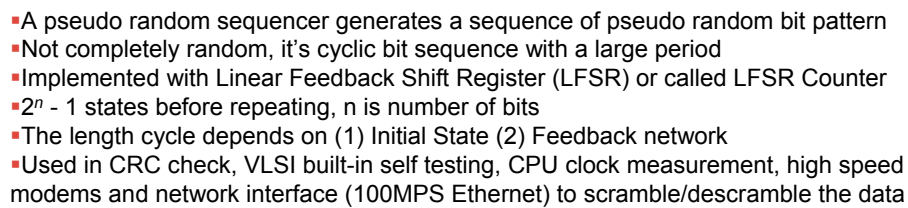
	1	2	3	4	5	6	7	8	9
CLK									
D0	0	1	1	1	1	0	0	0	1
D1	0	0	1	1	1	1	0	0	0
D2	0	0	0	1	1	1	1	0	0
D3	0	0	0	0	1	1	1	1	0

Binary Counter



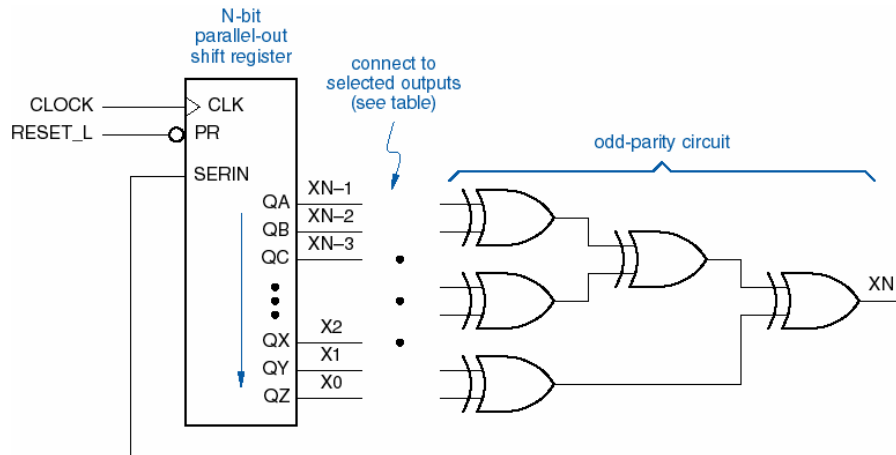
- What is the worst case time delay of this counter?
Assume J-K FF propagation delay = 8ns
AND gate delay = 5 ns
- Think about Divide-by-3, Divide-by-5 Counter Design?

<http://www.play-hookey.com/digital>



n	Feedback Equation
2	$X_2 = X_1 \oplus X_0$
3	$X_3 = X_1 \oplus X_0$
4	$X_4 = X_1 \oplus X_0$
5	$X_5 = X_2 \oplus X_0$
6	$X_6 = X_1 \oplus X_0$
7	$X_7 = X_3 \oplus X_0$
8	$X_8 = X_4 \oplus X_3 \oplus X_2 \oplus X_0$
12	$X_{12} = X_6 \oplus X_4 \oplus X_1 \oplus X_0$
16	$X_{16} = X_5 \oplus X_4 \oplus X_3 \oplus X_0$
20	$X_{20} = X_3 \oplus X_0$
24	$X_{24} = X_7 \oplus X_2 \oplus X_1 \oplus X_0$
28	$X_{28} = X_3 \oplus X_0$
32	$X_{32} = X_{22} \oplus X_2 \oplus X_1 \oplus X_0$

Pseudo Random Sequencer (Cont.)



Converting Between FFs

- Q^+ means “the next value of Q .” “Excitation” is the input that determines the next state.
- “Characteristic equation” specifies the next state of a device as a function of its excitation.
- S-R latch: $Q^+ = S + R' \cdot Q$

Characteristic Equations

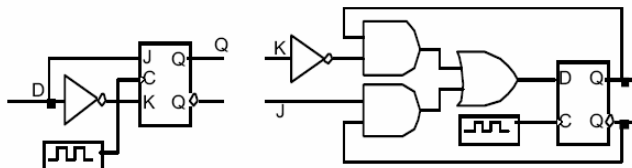
D: $Q^+ = D$

J-K: $Q^+ = J \bar{Q} + \bar{K} Q$

T: $Q^+ = T \bar{Q} + \bar{T} Q$

E.g., $J=K=0$, then $Q^+ = Q$
 $J=1, K=0$, then $Q^+ = 1$
 $J=0, K=1$, then $Q^+ = 0$
 $J=1, K=1$, then $Q^+ = \bar{Q}$

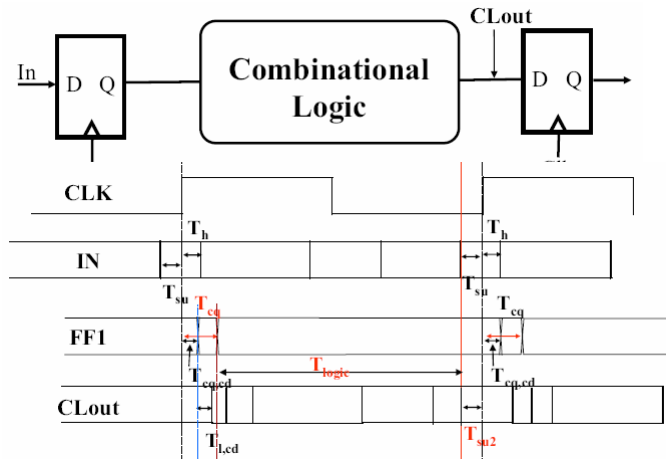
Implementing One FF in Terms of Another



D implemented with J-K

J-K implemented with D

Timing Parameter – Minimum Clock Period

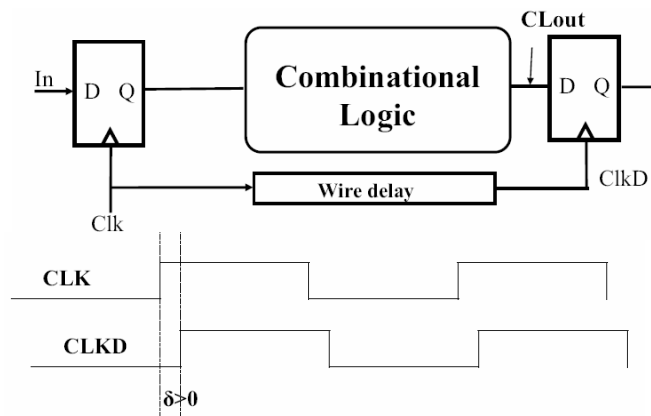


We must have
 $T > T_{cq} + T_{logic} + T_{su}$

T_{logic} : contamination or minimum delay
 through logic network
 T_{su} : setup time of FF
 T_{hold} : hold time of FF

Clock Skew

δ : The differences between arrival times of the clock at different devices



How to overcome Clock Skew?

$$T_{cq,cd} + T_{logic,cd} > T_{hold} + \delta$$

$T_{cq,cd}$: contamination or minimum delay from
 Clock to q
 $T_{logic,cd}$: contamination or minimum delay
 through logic network
 T_{su} : setup time of FF
 T_{hold} : hold time of FF