Chap 6 (6)

Various Counter Designs; Counter w/ Unused States Ring Counter, Johnson Counter

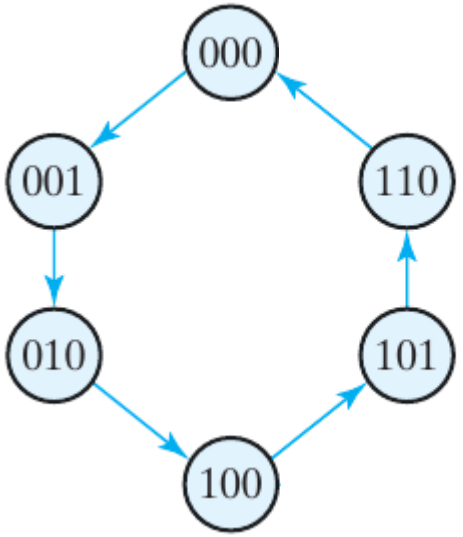
Hanwool Jeong

hwjeong@kw.ac.kr

Counter with Unused States

- There are occasions when a sequential circuit uses fewer than this maximum possible number of states.

Present State			Next State		
A	*B*	*C*	*A*	*B*	*C*
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	0	0	0



State Table

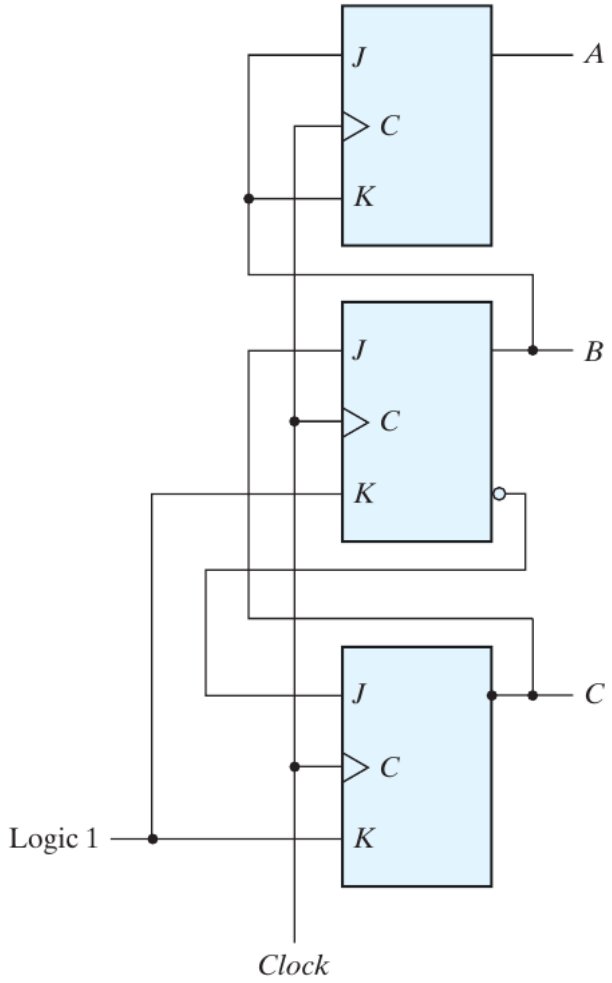
Present State			Next State			Flip-Flop Inputs					
A	B	C	A	B	C	J_A	K_A	J_B	K_B	J_C	K_C
0	0	0	0	0	1						
0	0	1	0	1	0						
0	1	0	1	0	0						
1	0	0	1	0	1						
1	0	1	1	1	0						
1	1	0	0	0	0						

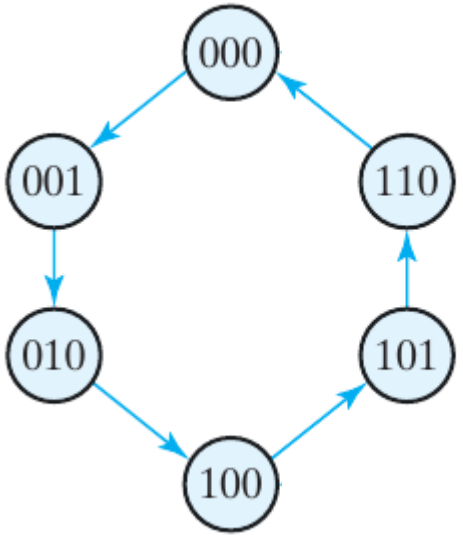
$$J_A = B \qquad K_A = B$$
$$J_B = C \qquad K_B = 1$$
$$J_C = B' \qquad K_C = 1$$

Logic Circuit Diagram & Check Unused States

- **Is this counter self correcting?**

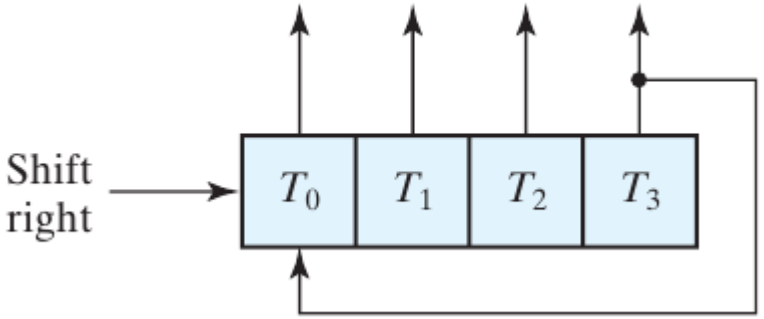
$$J_A = B \qquad K_A = B$$
$$J_B = C \qquad K_B = 1$$
$$J_C = B' \qquad K_C = 1$$

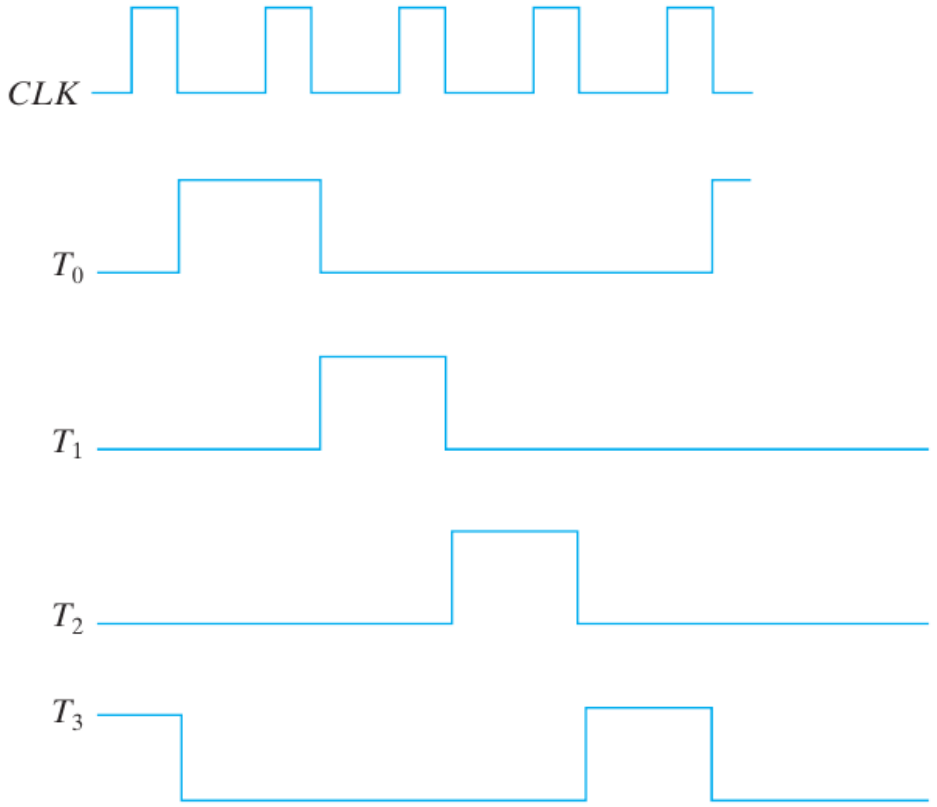




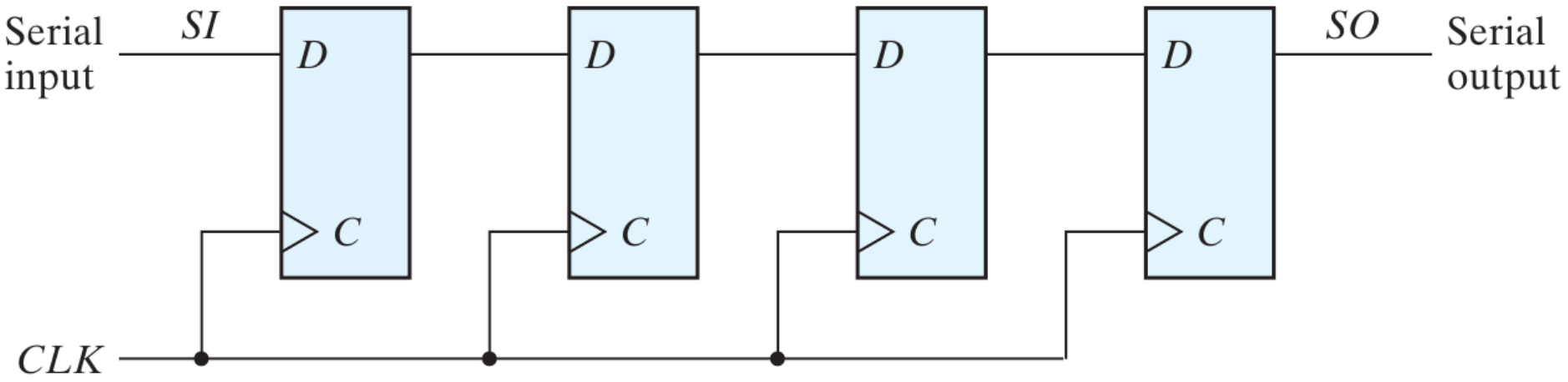
Ring Counter

- A ring counter is a circular shift register with only one flip-flop being set at any particular time; all others are cleared.



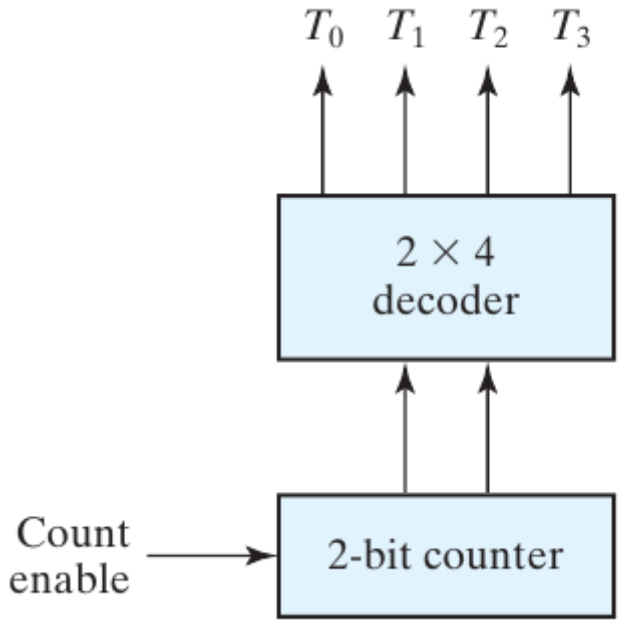


Revisit Shift Register



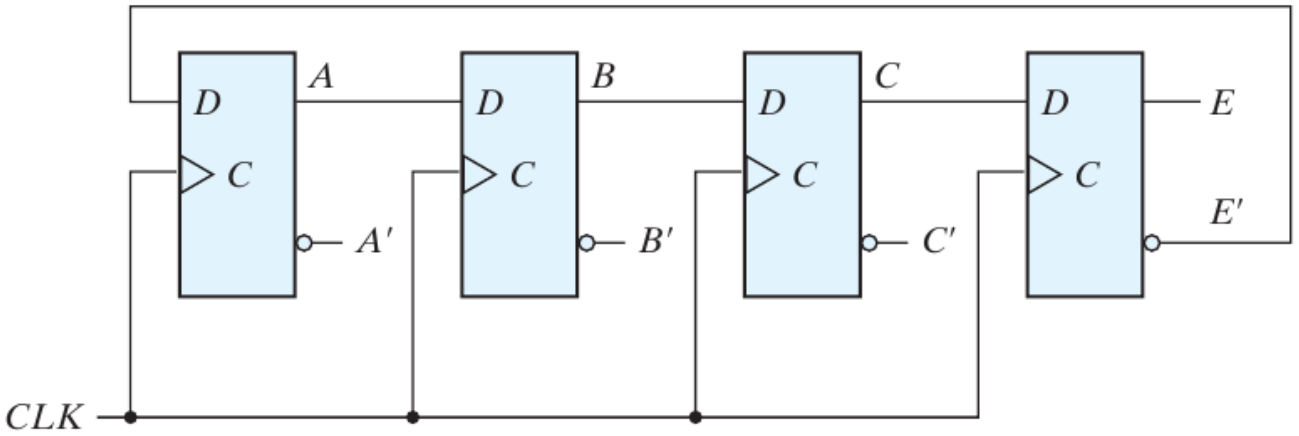
Ring Counter Implementation Using Decoder

- Why might the below design be more efficient?
- Imagine we need 16bit ring counter.



Johnson Counter

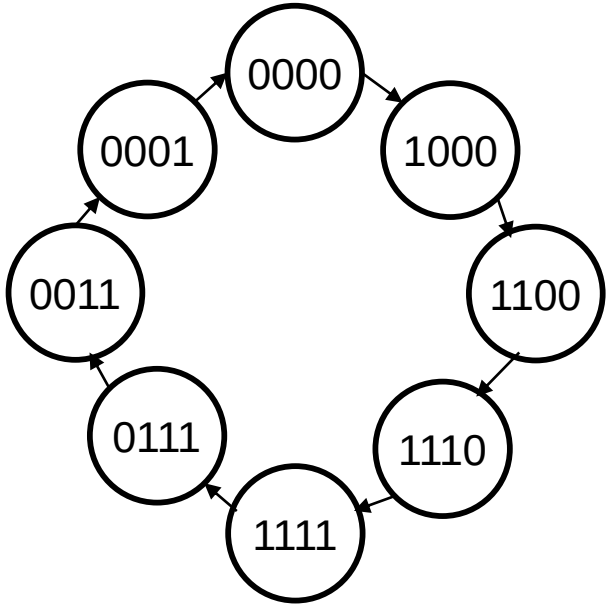
- The number of states can be doubled if the shift register is connected as a switch-tail ring counter.



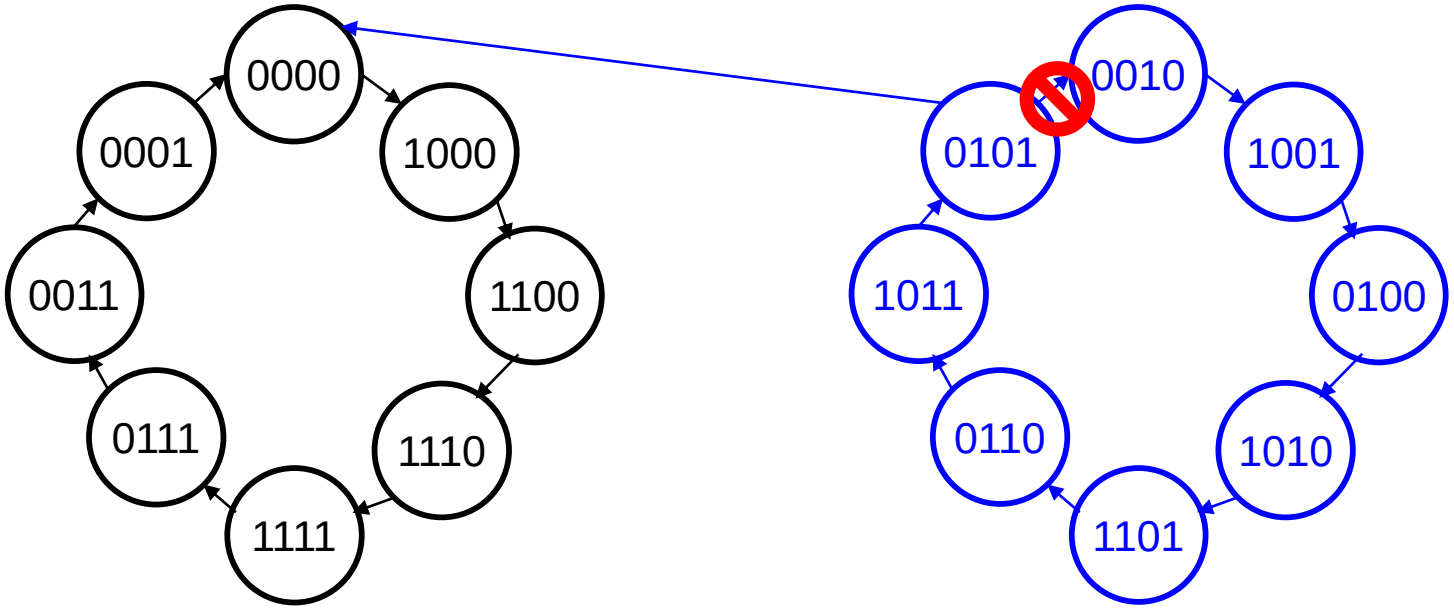
Sequence number	Flip-flop outputs				AND gate required for output
	A	*B*	*C*	*E*	
1	0	0	0	0	$A'E'$
2	1	0	0	0	AB'
3	1	1	0	0	BC'
4	1	1	1	0	CE'
5	1	1	1	1	AE
6	0	1	1	1	$A'B$
7	0	0	1	1	$B'C$
8	0	0	0	1	$C'E$

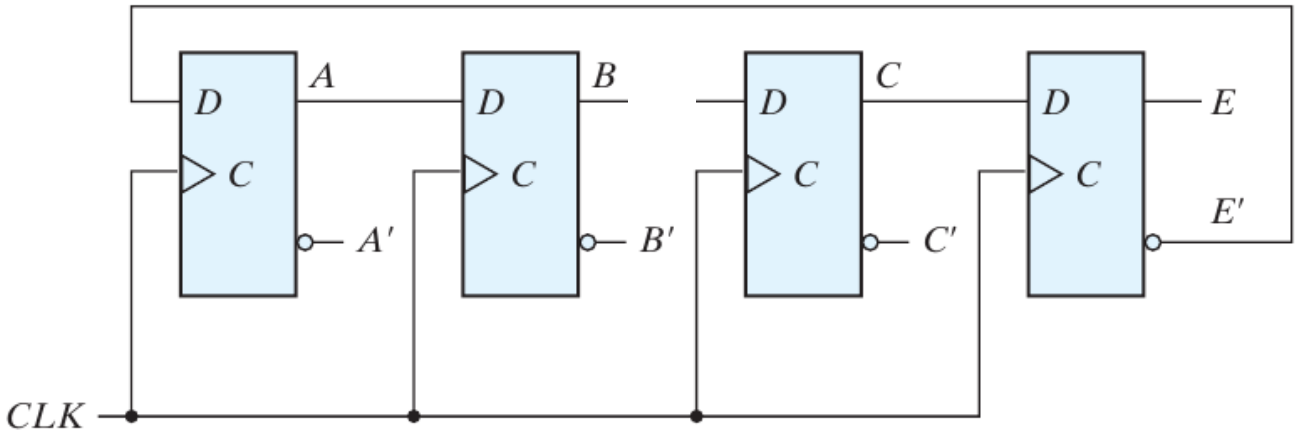
Check Unused States for Johnson Counter

Sequence number	Flip-flop outputs				AND gate required for output
	A	*B*	*C*	*E*	
1	0	0	0	0	$A'E'$
2	1	0	0	0	AB'
3	1	1	0	0	BC'
4	1	1	1	0	CE'
5	1	1	1	1	AE
6	0	1	1	1	$A'B$
7	0	0	1	1	$B'C$
8	0	0	0	1	$C'E$



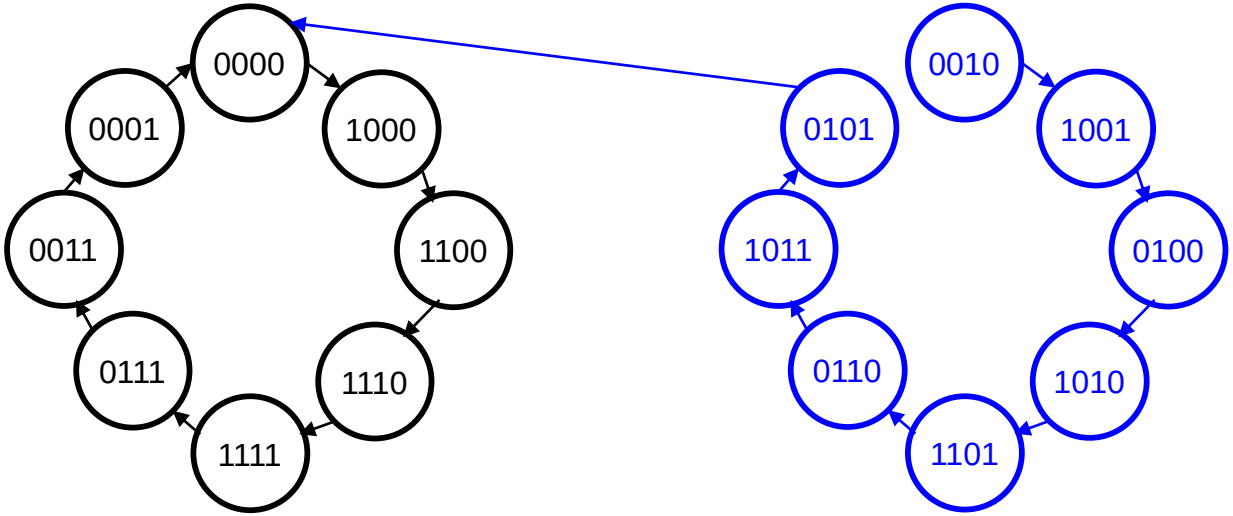
Let's Correct State!

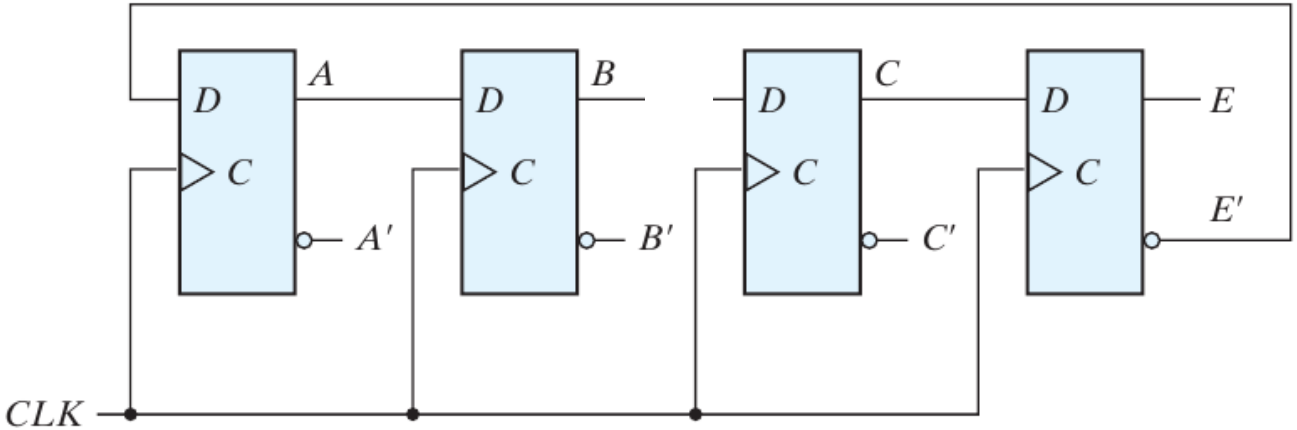




Correction Circuit Design

- When 0101, D_C should be 0, not B = 1





A	B	C	E	D_C
0	0	0	0	0
0	0	0	1	0
0	**0**	**1**	**0**	X
0	0	1	1	0
0	**1**	**0**	**0**	X
0	**1**	**0**	**1**	**0**
0	**1**	**1**	**0**	X
0	1	1	1	1
1	0	0	0	0
1	**0**	**0**	**1**	X
1	**0**	**1**	**0**	X
1	**0**	**1**	**1**	X
1	1	0	0	1
1	**1**	**0**	**1**	X
1	1	1	0	1
1	1	1	1	1

Checkpoints

- ✓Counter w/ unused states
- ✓Ring counter & Implementation w/ decoder
- ✓Johnson counter & handling unused states
- ✓Coming up next : Chap 7. Random access memory