14-Registers and Counters Part 2

Text: Unit 12

ECEGR/ISSC 201
Digital Operations and Computations
Winter 2011

Overview

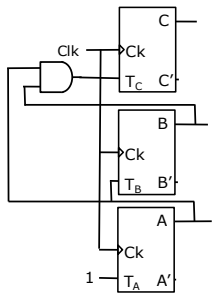
- Binary Counter Design
- Counters for other Sequences
- Counter Design using S-R and J-K Flip-Flops

Counters

- Output changes state in a prescribed sequence when input pulses are received
  - Made from two or more flip-flops
- Consider synchronous counters
  - Common clock signal

Design of Binary Counters

- Consider three T flip-flops
- The output is read: CBA
  - Example: 001 if C =0, B =0 and A = 1
- Assume initially:
  - A = B = C = 0



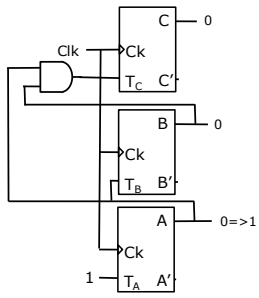
Design of Binary Counters

- Recall that for T flip-flop
  - $Q^+ = T'Q + TQ' = T \oplus Q$

T	Q	Q^+
0	0	0
0	1	1
1	0	1
1	1	0

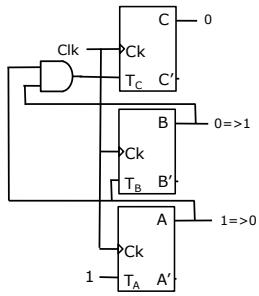
Design of Binary Counters

- Output: 000
- After the first active edge
  - Output: 001



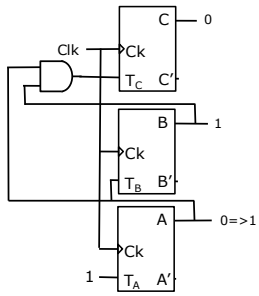
Design of Binary Counters

- Output: 001
- After the second active edge
  - Output: 010



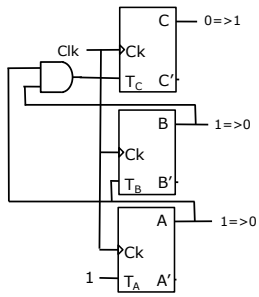
Design of Binary Counters

- Output: 010
- After the third active edge
  - Output: 011



Design of Binary Counters

- Output: 011
- After the fourth active edge
  - Output: 100

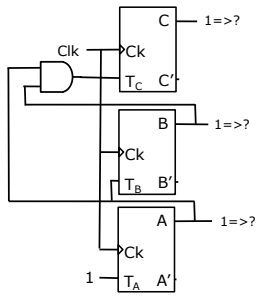


Design of Binary Counters

- Counting continues in binary until output is 111
- What happens next?

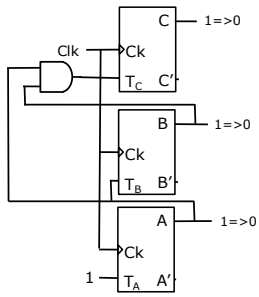
Design of Binary Counters

- Output: 111
- After the eighth active edge



Design of Binary Counters

- Output: 111
- After the eighth active edge
  - Output: 000
- Sequence repeats



Design of Binary Counters

- Any time Q does not equal Q^+, a 1 is applied to T

						Flip-flop inputs		
C	B	A	C^+	B^+	A^+	T_C	T_B	T_A
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	1	1	1			
1	1	1	0	0	0			

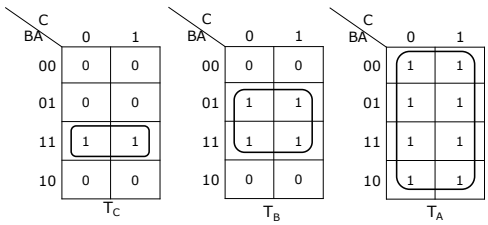
Design of Binary Counters

- Any time Q does not equal Q^+, a 1 is applied to T

						Flip-flop inputs		
C	B	A	C^+	B^+	A^+	T_C	T_B	T_A
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

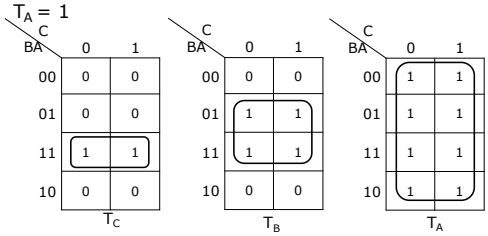
Design of Binary Counters

- Derivation of T_C, T_B, T_A from the K-map



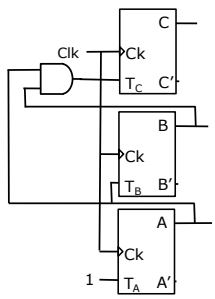
Design of Binary Counters

- $T_C = AB$
- $T_B = A$
- $T_A = 1$



Design of Binary Counters

- These equations match our circuit
  - $T_C = AB$
  - $T_B = A$
  - $T_A = 1$



Design of Binary Counters

- How else could we design this counter?
- Try using D flip-flops

						Flip-flop inputs		
C	B	A	C^+	B^+	A^+	D_C	D_B	D_A
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	1	1	1			
1	1	1	0	0	0			

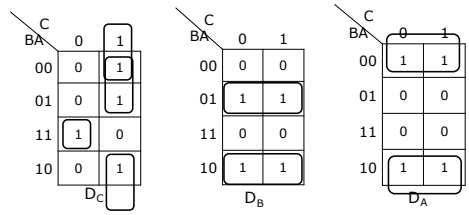
Design of Binary Counters

- Construct the K-maps for the flip-flop inputs

			Flip-flop inputs					
C	B	A	C^+	B^+	A^+	D_C	D_B	D_A
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	0
0	1	0	0	1	1	0	1	1
0	1	1	1	0	0	1	0	0
1	0	0	1	0	1	1	0	1
1	0	1	1	1	0	1	1	0
1	1	0	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0

Design of Binary Counters

- $D_C = C^+ = C'BA + CB' + CA'$
- $D_B = B^+ = B'A + BA'$
- $D_A = A^+ = A'$



Design of Binary Counters

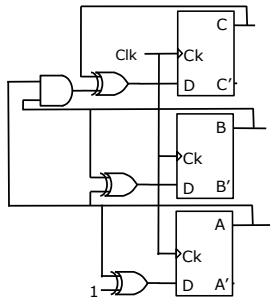
- We could build the circuit directly from these equations, but there is a more elegant solution
- $D_C = C^+ = C'BA + CB' + CA'$
- $D_B = B^+ = B'A + BA'$
- $D_A = A^+ = A'$

Design of Binary Counters

- We could build the circuit directly from these equations, but there is a more elegant solution
- $D_C = C^+ = C'BA + CB' + CA' = AB$ XOR C
- $D_B = B^+ = B'A + BA' = A$ XOR B
- $D_A = A^+ = A' = 1$ XOR A
- We can simply place XOR in front of the D Flip-flops and use the circuit from the T flip-flop

Design of Binary Counters

- Using D flip-flops

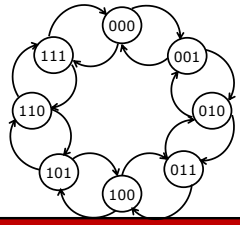


Up-Down Counter

- Binary Up-Down Counter
  - Inputs U, D
  - If U = 1, D =0
    - Sequence is 000, 001, 010, 011, 100 and so on (repeats after 111)
  - If D = 1, U =0
    - Sequence is 111, 110, 101, 100 and so on (repeats after 000)
  - If D = 0, U = 0
    - Output does not change
  - D = 1, U = 1 is not allowed

Up-Down Counter

- State graph
  - U = (1 outer loop)
  - D = (1 inner loop)



Up-Down Counter

- Let's implement using D flip-flops

			Up			Down		
C	B	A	C^+	B^+	A^+	C^+	B^+	A^+
0	0	0	0	0	1	1	1	1
0	0	1	0	1	0	1	1	0
0	1	0	0	1	1	1	0	1
0	1	1	1	0	0	1	0	0
1	0	0	1	0	1	0	1	1
1	0	1	1	1	0	0	1	0
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	0	0	0

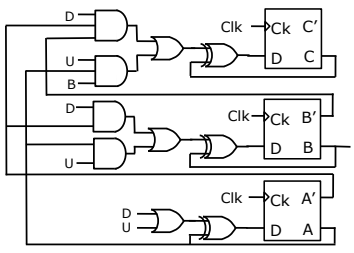
Up-Down Counter

- When U = 1 and D = 0
  - $D_A = A^+ = A'$
  - $D_B = B^+ = BA' + B'A = B \oplus A$
  - $D_C = C^+ = C'AB + CB'A' + CB'A + CBA'$
    $= C'AB + CB' + CA' = C'BA + C(BA)'$
    $= C \oplus (BA)$

Up-Down Counter

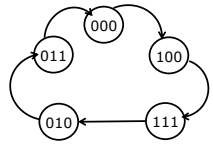
- When U = 0 and D = 1
  - $D_A = A^+ = A'$
  - $D_B = B^+ = B \oplus A'$
  - $D_C = C^+ = C \oplus B'A'$
- In general:
  - $D_A = A^+ = A \oplus (U+D)$
  - $D_B = B^+ = B \oplus (UA+DA')$
  - $D_C = C^+ = C \oplus (UBA+DB'A')$

Up-Down Counter



Counters for Other Sequences

- Some applications call for counters that are not in ascending or descending binary order
- Consider the following state graph
- Construct a counter using T flip-flops



Counters for Other Sequences

C	B	A	C^+	B^+	A^+
0	0	0	1	0	0
0	0	1	-	-	-
0	1	0	0	1	1
0	1	1	0	0	0
1	0	0	1	1	1
1	0	1	-	-	-
1	1	0	-	-	-
1	1	1	0	1	0

Counters for Other Sequences

- We could design a circuit using T flip-flops directly from the next state table as we have done previously
- Better approach is to examine the next-state K-maps

Counters for Other Sequences

- Resulting next-state K-maps

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

A^+

Counters for Other Sequences

- What must the T_C K-map look like to realize this?
- Begin by examining the C and C^+ relationship
- Whenever C changes state, a 1 to the T_C must be applied

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00		
01		
11		
10		

T_C

Counters for Other Sequences

- Consider C=B=A=0; the next state C^+ is 1
  - This is a change in state, so a 1 must be applied to T_c for the state to change

BA \ C	0	1
00	(1)	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	(1)	
01		
11		
10		

T_C

Counters for Other Sequences

- Consider C= 1, B=A=0; the next state C^+ is 1
  - No change in state so the input to T_C must be 0

BA \ C	0	1
00	1	(1)
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	1	(0)
01		
11		
10		

T_C

Counters for Other Sequences

- Consider C= 0, B=A=1; what should the input to T_c be?

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	1	0
01		
11		
10		

T_C

Counters for Other Sequences

- Consider C= 0, B=A=1; what should the input to T_c be?
  - No change, so T_c should be 0

BA \ C	0	1
00	1	1
01	X	X
11	(0)	0
10	0	X

C^+

BA \ C	0	1
00	1	0
01		
11	(0)	
10		

T_C

Counters for Other Sequences

- Consider C= B=A=1; what should the input to T_c be?

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	1	0
01		
11	0	
10		

T_C

Counters for Other Sequences

- Consider C= B=A=1; what should the input to T_c be?
  - $C^+ = 0$, so a change occurs,

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	1	0
01		
11	0	1
10		

T_C

Counters for Other Sequences

- Completing the map

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	1	0
01	X	X
11	0	1
10	0	X

T_C

Counters for Other Sequences

- Finding the characteristic equation
  - $T_C = C'B' + CB$

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

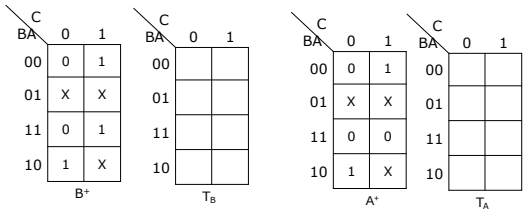
C^+

BA \ C	0	1
00	[1]	0
01	[X]	X
11	0	[1]
10	0	[X]

T_C

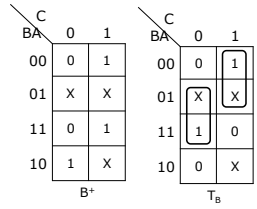
Counters for Other Sequences

- Find the simplified SoP equations for T_B and T_A



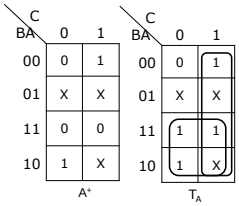
Counters for Other Sequences

- Find the simplified SoP equations for T_B and T_A
  - $T_B = C'A + CB'$



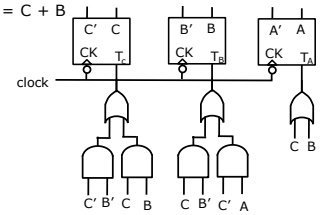
Counters for Other Sequences

- Find the simplified SoP equations for T_B and T_A
  - $T_c = C'B' + CB$
  - $T_B = C'A + CB'$
  - $T_A = C + B$



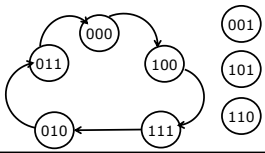
Counters for Other Sequences

- Circuit realization (feedback loops not shown)
  - $T_c = C'B' + CB$ (XOR can also be used)
  - $T_B = C'A + CB'$
  - $T_A = C + B$



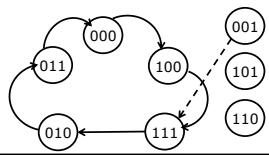
Counters for Other Sequences

- Some pitfalls
  - The state graph does not describe what happens for three states
  - When the circuit is turned on, the states are random
  - It should be checked that these states transition into the sequence, once there it should never leave



Counters for Other Sequences

- For example: 001
  - $T_c = C'B' + CB = 1$ => change $C^+ = 1$
  - $T_B = C'A + CB' = 1$ => change $B^+ = 1$
  - $T_A = C + B = 0$ => no change $A^+ = 1$



Counter Design Using D Flip-Flops

- Design using D flip-flops is simple
- Consider the same state graph previously considered
  - $D_C = C^+$

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00		
01		
11		
10		

D_C

Counter Design Using D Flip-Flops

- Design using D flip-flops is simple
- Consider the same state graph previously considered
  - $D_C = C^+$

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

D_C

Counter Design Using D Flip-Flops

- For the other D flip-flops
  - $D_B = B^+ = C + BA'$
  - $D_A = A^+ = CA' + BA' = A'(C+B)$

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

D_B

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

A^+

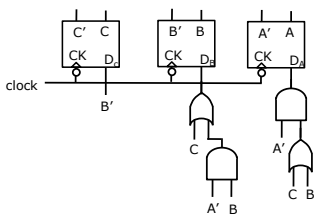
BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

D_A

Counter Design Using D Flip-Flops

- Draw the resulting circuit
  - $D_C = C^+ = B'$
  - $D_B = B^+ = C + BA'$
  - $D_A = A^+ = CA' + BA' = A'(C+B)$

Counter Design Using D Flip-Flops



Counter Design Using S-R and J-K Flip-Flops

- We next examine counter design using S-R and J-K flip-flops
- Recall for an S-R flip-flop

S	R	Q	Q^+
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	-
1	1	1	-

Counter Design Using S-R and J-K Flip-Flops

- Design approach is similar to T and D flip-flops but now there are two inputs S and R

Q	Q^+	S	R
0	0	0	0
		0	1
0	1	1	0
1	0	0	1
1	1	0	0
		1	0

=>

Q	Q^+	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

Counter Design Using S-R and J-K Flip-Flops

- Consider C^+

C	B	A	C^+	B^+	A^+	S_C	R_C
0	0	0	1	0	0		
0	0	1	-	-	-		
0	1	0	0	1	1		
0	1	1	0	0	0		
1	0	0	1	1	1		
1	0	1	-	-	-		
1	1	0	-	-	-		
1	1	1	0	1	0		

Counter Design Using S-R and J-K Flip-Flops

- Consider C^+

C	B	A	C^+	B^+	A^+	S_C	R_C
0	0	0	1	0	0	1	0
0	0	1	-	-	-	X	X
0	1	0	0	1	1	0	X
0	1	1	0	0	0	0	X
1	0	0	1	1	1	X	0
1	0	1	-	-	-	X	X
1	1	0	-	-	-	X	X
1	1	1	0	1	0	0	1

Counter Design Using S-R and J-K Flip-Flops

- Next consider B^+

C	B	A	C^+	B^+	A^+	S_C	R_C	S_B	R_B
0	0	0	1	0	0	1	0	0	X
0	0	1	-	-	-	X	X	X	X
0	1	0	0	1	1	0	X	X	0
0	1	1	0	0	0	0	X	0	1
1	0	0	1	1	1	X	0	1	0
1	0	1	-	-	-	X	X	X	X
1	1	0	-	-	-	X	X	X	X
1	1	1	0	1	0	0	1	X	0

Counter Design Using S-R and J-K Flip-Flops

- Finally, consider A^+

C	B	A	C^+	B^+	A^+	S_C	R_C	S_B	R_B	S_A	R_A
0	0	0	1	0	0	1	0	0	X	0	X
0	0	1	-	-	-	X	X	X	X	X	X
0	1	0	0	1	1	0	X	X	0	1	0
0	1	1	0	0	0	0	X	0	1	0	1
1	0	0	1	1	1	X	0	1	0	1	0
1	0	1	-	-	-	X	X	X	X	X	X
1	1	0	-	-	-	X	X	X	X	X	X
1	1	1	0	1	0	0	1	X	0	0	1

Counter Design Using S-R and J-K Flip-Flops

- We could build directly from the next state table, but it is easier to use next state K-maps

Counter Design Using S-R and J-K Flip-Flops

- Need an S and R for each next state
- Do not let 1s (or Don't Cares taken as 1s) to overlap between S and R
  - S = B'; R = A

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	1	X
01	X	X
11	0	0
10	0	X

S_C

BA \ C	0	1
00	0	0
01	X	X
11	X	1
10	X	X

R_C

Counter Design Using S-R and J-K Flip-Flops

- Complete for B^+ and A^+

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

A^+

Counter Design Using S-R and J-K Flip-Flops

- Complete for B^+ and A^+
  - $S_B = C$
  - $R_B = C'A$

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	0	X
10	X	X

S_B

BA \ C	0	1
00	X	0
01	X	X
11	1	0
10	0	X

R_B

Counter Design Using S-R and J-K Flip-Flops

- Complete for B^+ and A^+
  - $S_B = CA' + BA' = A'(C+B)$
  - $R_B = A$

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

A^+

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

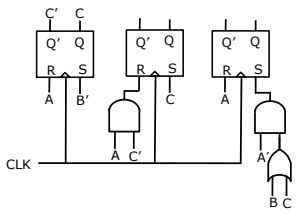
S_A

BA \ C	0	1
00	X	0
01	X	X
11	1	1
10	0	X

R_A

Counter Design Using S-R and J-K Flip-Flops

- Circuit realization



Counter Design Using S-R and J-K Flip-Flops

- Now using J-K flip-flops

J	K	Q	Q^+
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

=>

Q	Q^+	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Counter Design Using S-R and J-K Flip-Flops

- Consider C^+

C	B	A	C^+	B^+	A^+	J_C	K_C
0	0	0	1	0	0		
0	0	1	-	-	-		
0	1	0	0	1	1		
0	1	1	0	0	0		
1	0	0	1	1	1		
1	0	1	-	-	-		
1	1	0	-	-	-		
1	1	1	0	1	0		

Counter Design Using S-R and J-K Flip-Flops

- Consider C^+

C	B	A	C^+	B^+	A^+	J_C	K_C
0	0	0	1	0	0	1	X
0	0	1	-	-	-	X	X
0	1	0	0	1	1	0	X
0	1	1	0	0	0	0	X
1	0	0	1	1	1	X	0
1	0	1	-	-	-	X	X
1	1	0	-	-	-	X	X
1	1	1	0	1	0	X	1

Counter Design Using S-R and J-K Flip-Flops

- Next consider B^+

C	B	A	C^+	B^+	A^+	J_C	K_C	J_B	K_B
0	0	0	1	0	0	1	X	0	X
0	0	1	-	-	-	X	X	X	X
0	1	0	0	1	1	0	X	X	0
0	1	1	0	0	0	0	X	X	1
1	0	0	1	1	1	X	0	1	X
1	0	1	-	-	-	X	X	X	X
1	1	0	-	-	-	X	X	X	X
1	1	1	0	1	0	X	1	X	0

Counter Design Using S-R and J-K Flip-Flops

- Finally, consider A^+

C	B	A	C^+	B^+	A^+	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	1	0	0	1	X	0	X	0	X
0	0	1	-	-	-	X	X	X	X	X	X
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	0	0	0	0	X	X	1	X	1
1	0	0	1	1	1	X	0	1	X	1	X
1	0	1	-	-	-	X	X	X	X	X	X
1	1	0	-	-	-	X	X	X	X	X	X
1	1	1	0	1	0	X	1	X	0	X	1

Counter Design Using S-R and J-K Flip-Flops

- Need a J and K for each next state
  - $J_C = B'$
  - $K_C = A$

BA \ C	0	1
00	1	1
01	X	X
11	0	0
10	0	X

C^+

BA \ C	0	1
00	1	X
01	X	X
11	0	X
10	0	X

J_C

BA \ C	0	1
00	X	0
01	X	X
11	X	1
10	X	X

K_C

Counter Design Using S-R and J-K Flip-Flops

- Complete for B^+ and A^+

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

A^+

Counter Design Using S-R and J-K Flip-Flops

- Complete for B^+ and A^+
  - $J_B = C$
  - $K_B = C'A$

BA \ C	0	1
00	0	1
01	X	X
11	0	1
10	1	X

B^+

BA \ C	0	1
00	0	1
01	X	X
11	X	X
10	X	X

J_B

BA \ C	0	1
00	X	0
01	X	X
11	1	0
10	0	X

K_B

Counter Design Using S-R and J-K Flip-Flops

- Complete for B^+ and A^+
  - $J_A = C$
  - $K_A = 1$

BA \ C	0	1
00	0	1
01	X	X
11	0	0
10	1	X

A^+

BA \ C	0	1
00	0	1
01	X	X
11	X	X
10	1	X

J_A

BA \ C	0	1
00	X	X
01	X	X
11	1	1
10	X	X

K_A

Counter Design Using S-R and J-K Flip-Flops

- Circuit realization