Chapter 11

CLOCKED SEQUENTIAL CIRCUITS

Clocked Sequential Circuits

- Counters
  - Registers with predetermined sequence of states
  - The number of states are determined by the number of flip-flops used and the way in which these flip-flops are connected.
- Registers
  - A group of binary storage cells suitable for holding binary information
  - Consists of a group of flip-flops that may have combinational gates which perform certain data processing task

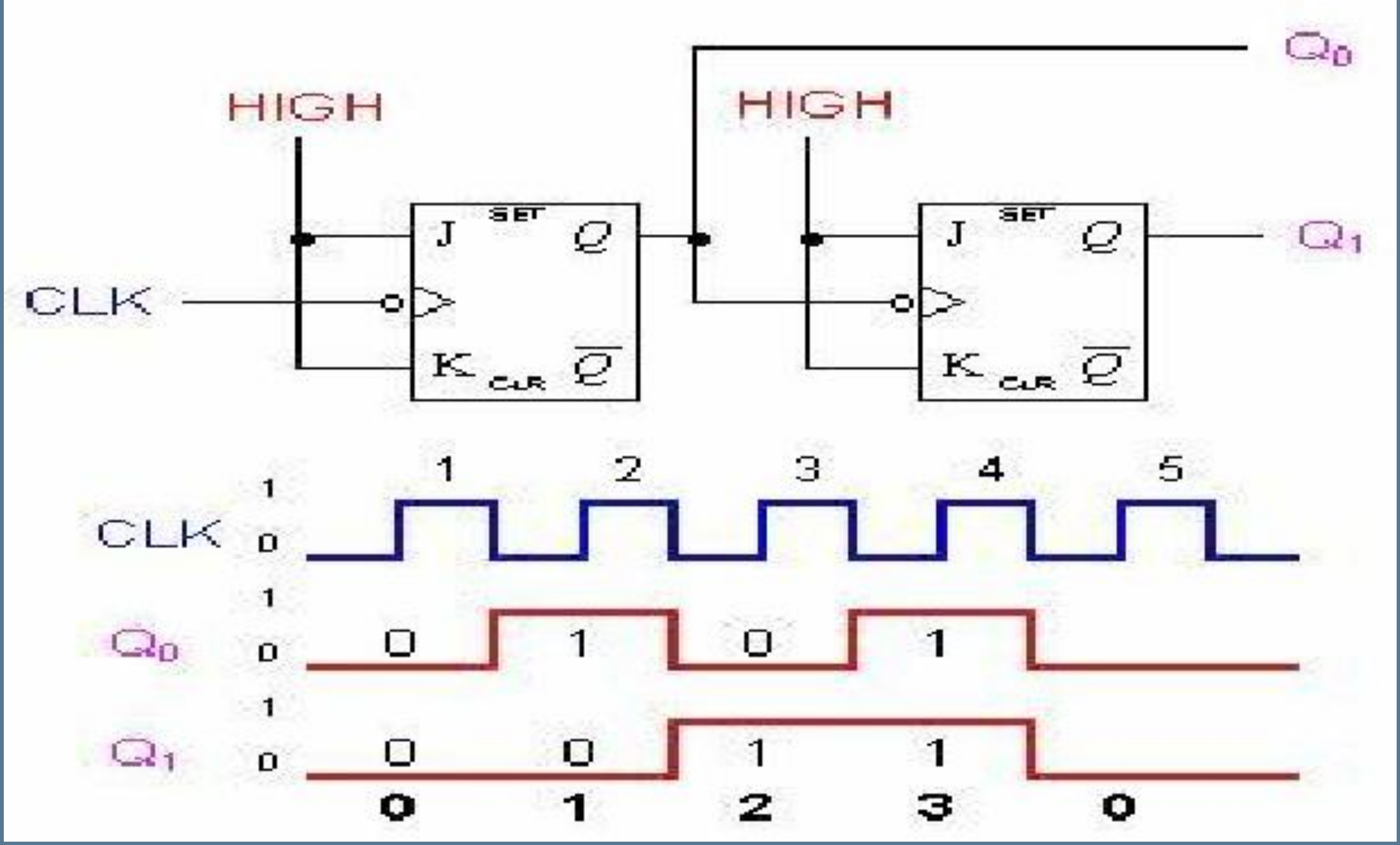
Counters

- In a digital circuit, counters are used to do three main functions:
  - Timing
  - Sequencing
  - Counting
- Counters are generally made up of flip-flops and logic gates.
- Main types of flip-flop used: JK FF and T FF

Classification of Counters

- Asynchronous
  - Counters in which the flip-flops within the counter do not change states at exactly the same time because they do not have a common clock pulse
  - Flip-flops are never simultaneously triggered
- Synchronous
  - All the flip-flops within the counter are clocked at the same time by a common clock pulse.

Asynchronous Counters

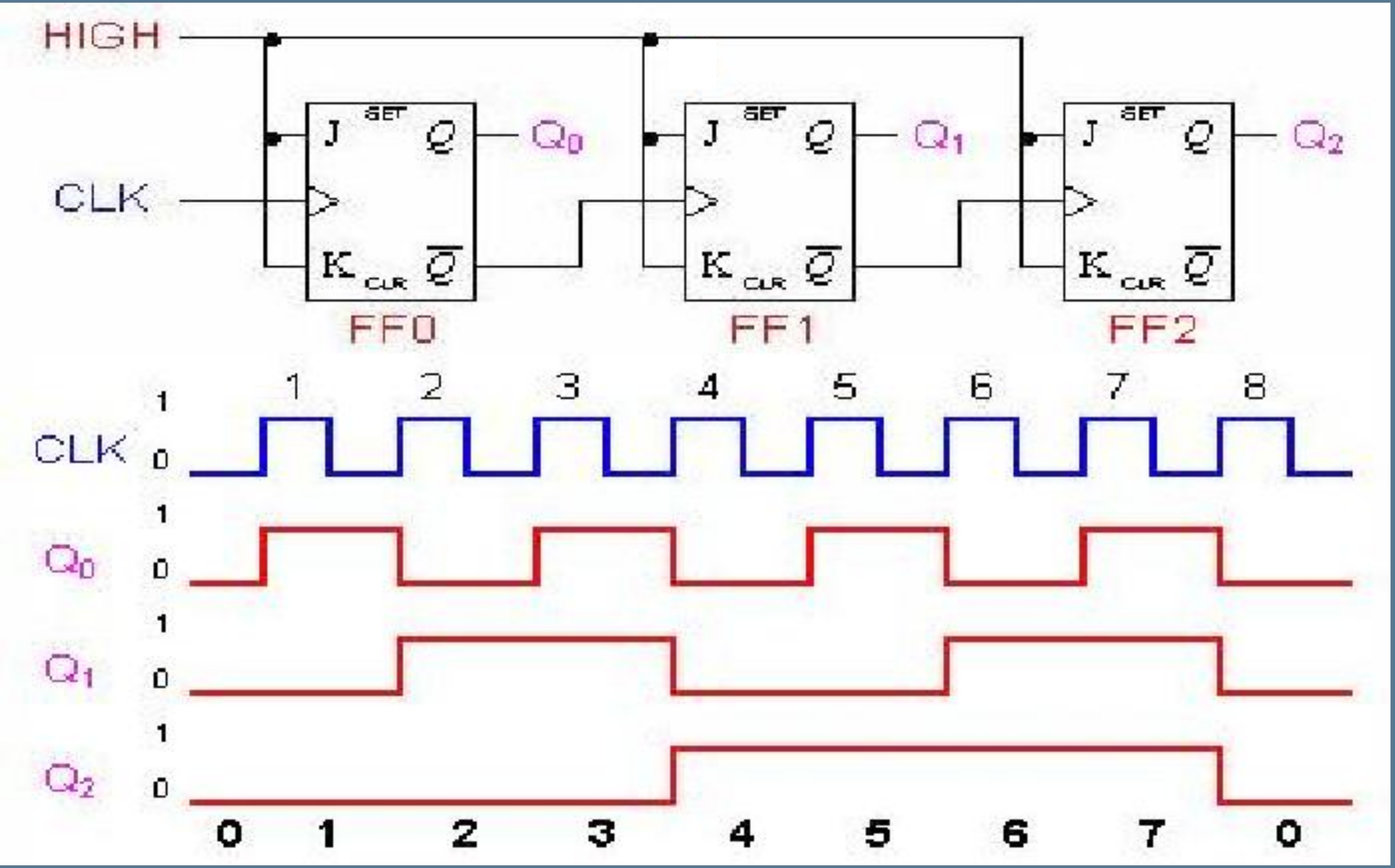


Example: 2-bit asynchronous counter

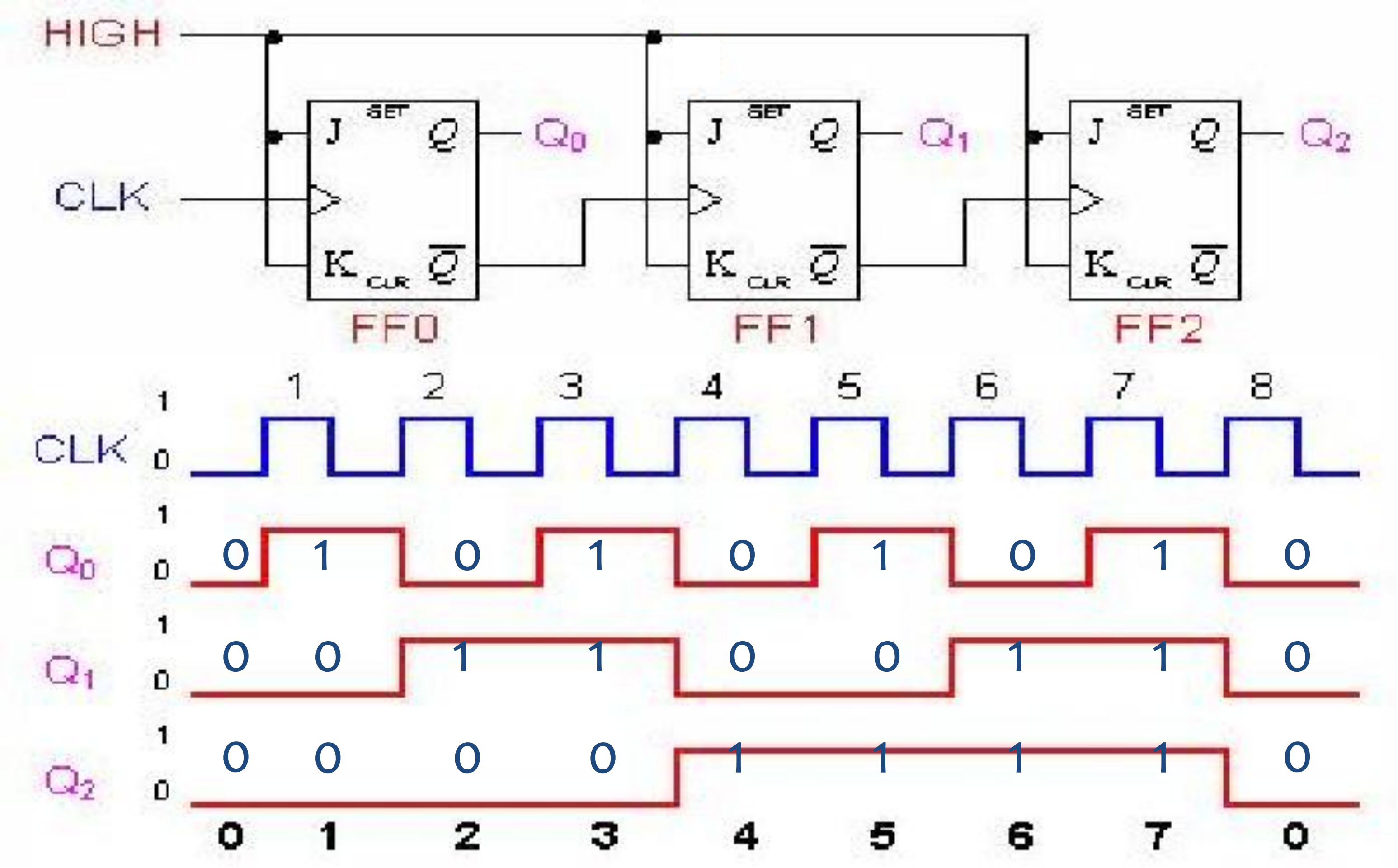
Ripple Counter

- Asynchronous counters are commonly referred to as ripple counters for the following reason:
  – The effect of the clock pulse is first "felt" by the first flip-flop.
  – This effect cannot get to the next flip-flop because of the propagation delay through the first FF.

Ripple Counter



Ripple Counter



Types of Synchronous Counters

- Binary counter – an n-bit binary counter that counts from 0 to (2^n-1) and back to 0 again.

Example: 3-bit counter with Gray code sequence

- Define the problem:
  – Design a counter whose outputs progress in the sequence defined by the following table:

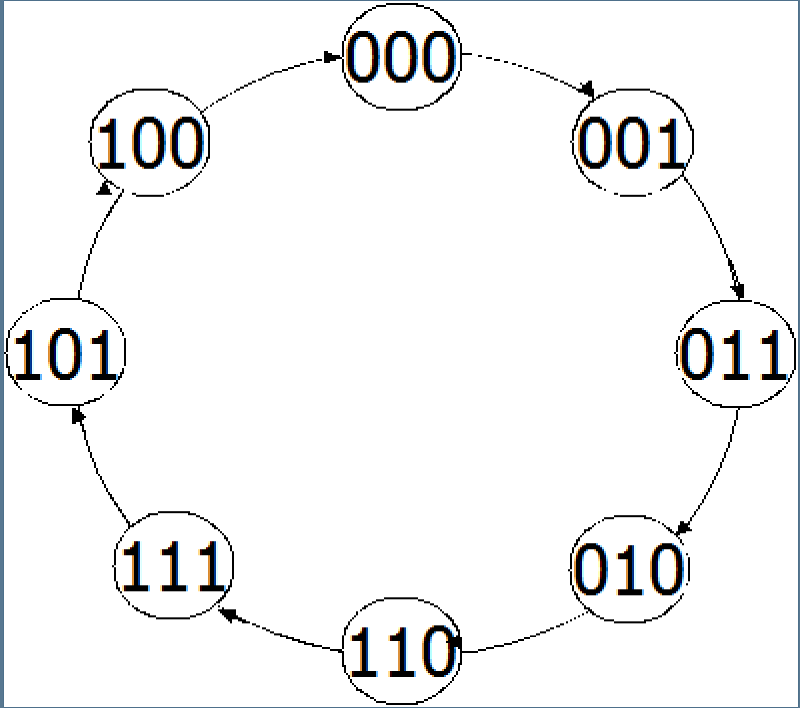
3-bit Gray code sequence

A	B	C
0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

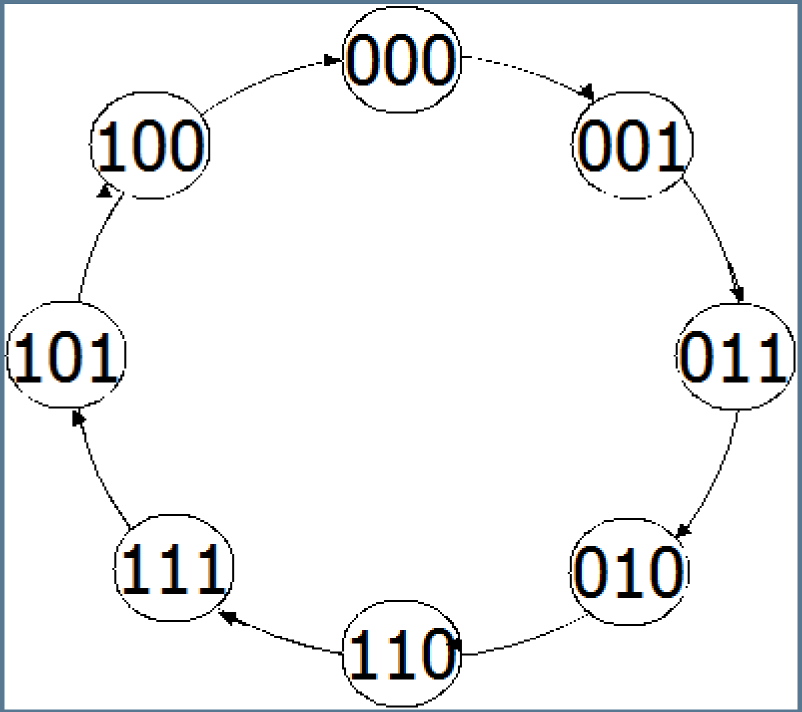
Example: State Diagram

3-bit Gray code sequence

A	B	C
0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

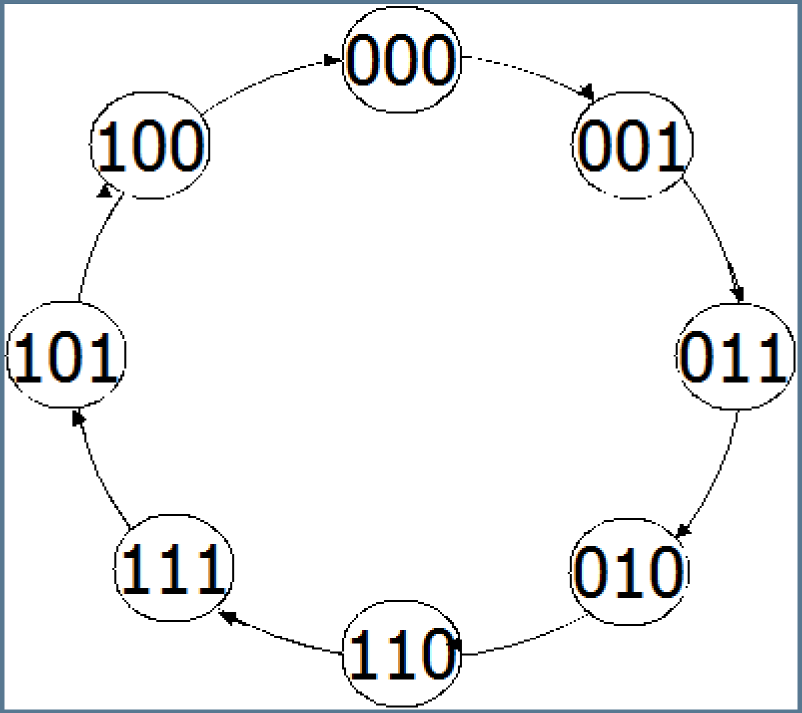


Example: State Table



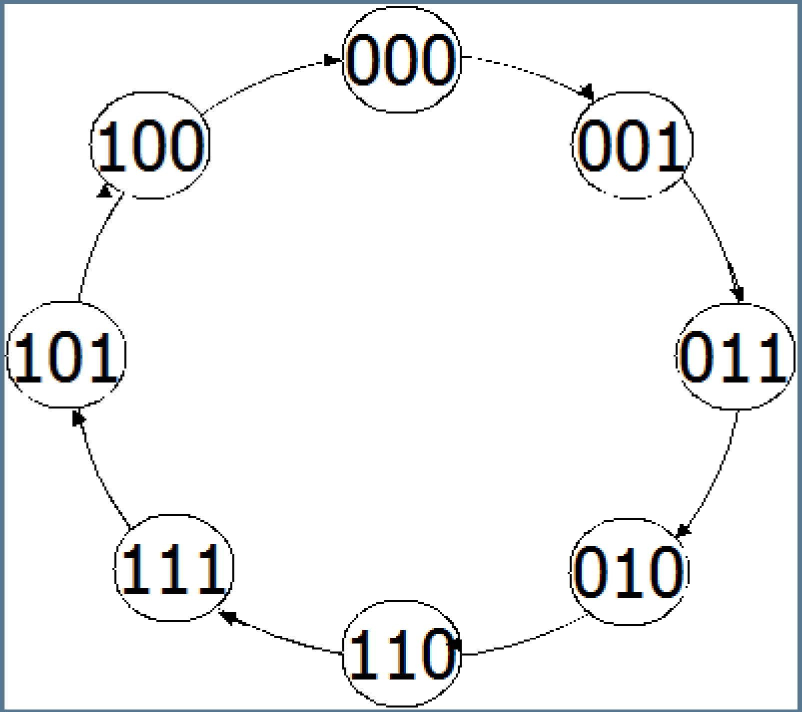
Present State			Next State		
A	B	C	A	B	C
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Example: State Table



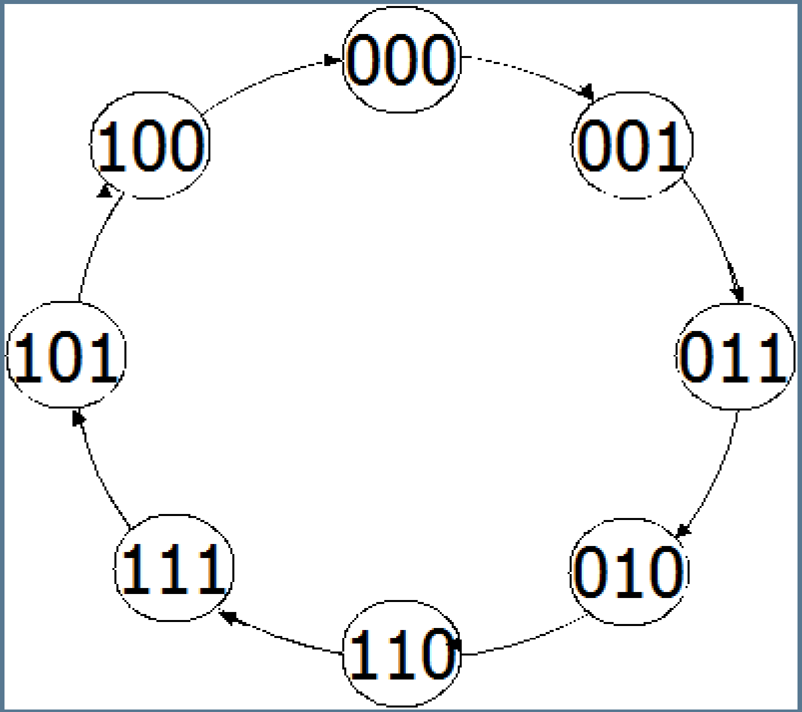
Present State			Next State		
A	B	C	A	B	C
0	0	0	0	0	1
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Example: State Table



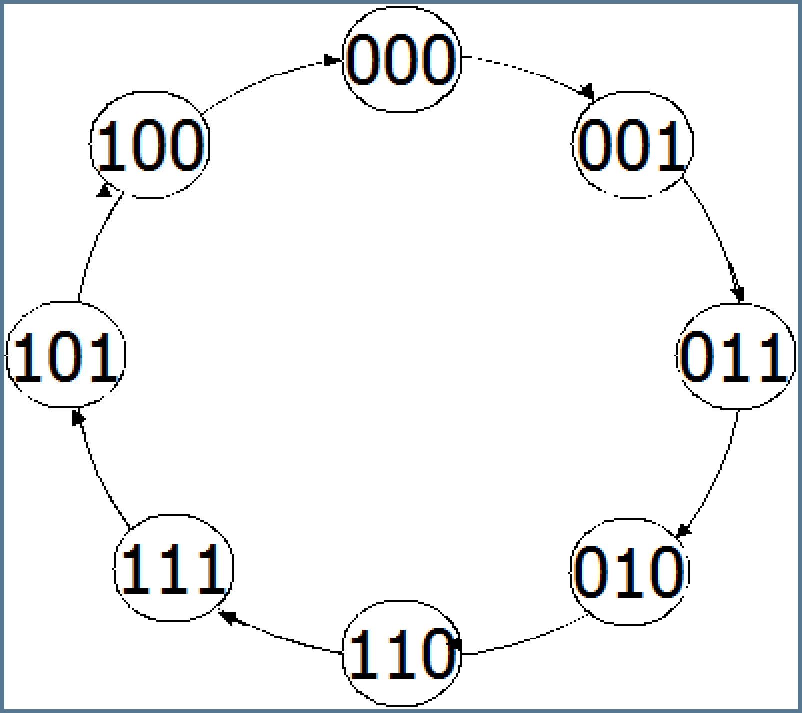
Present State			Next State		
A	B	C	A	B	C
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Example: State Table



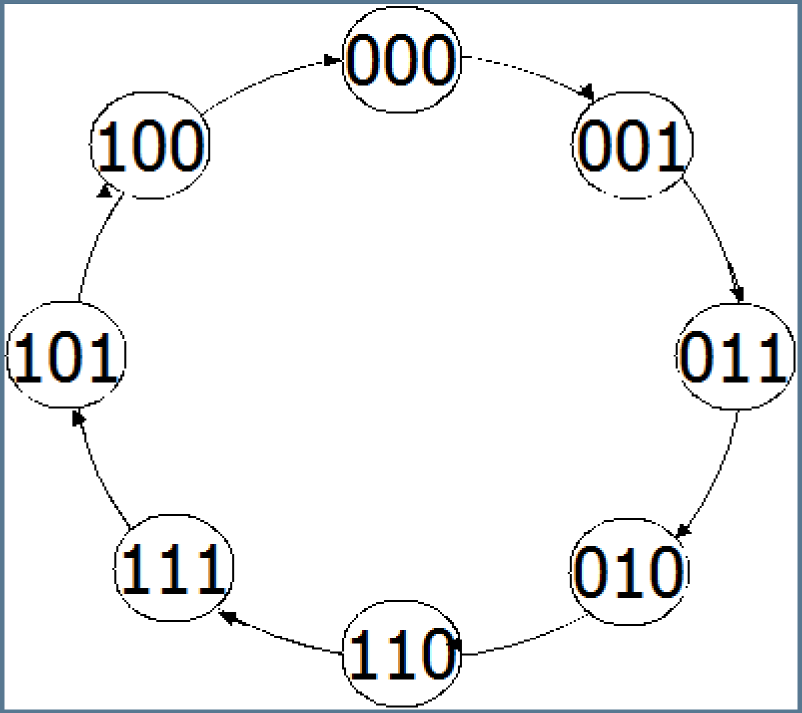
Present State			Next State		
A	B	C	A	B	C
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Example: State Table



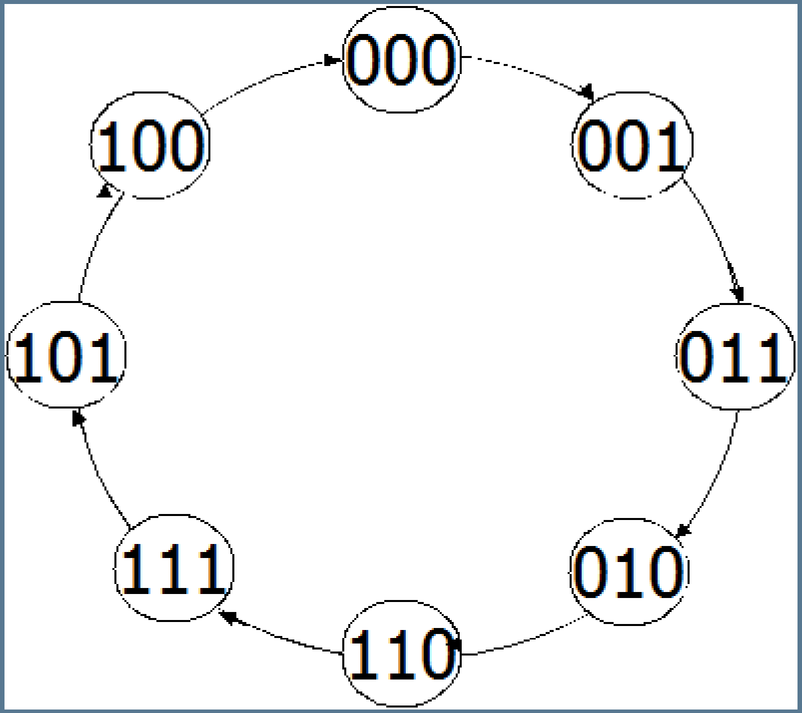
Present State			Next State		
A	B	C	A	B	C
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0			
1	0	1			
1	1	0			
1	1	1			

Example: State Table



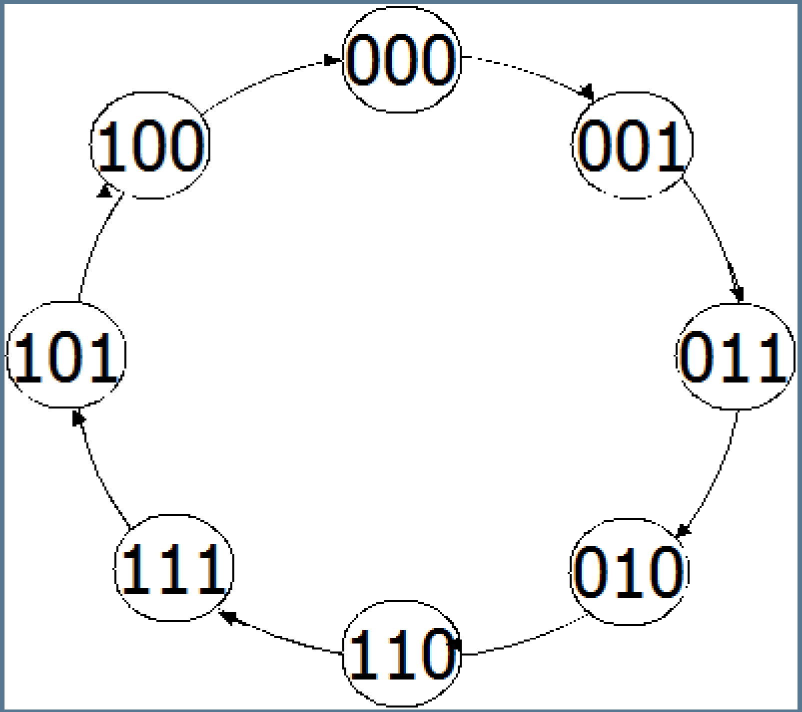
Present State			Next State		
A	B	C	A	B	C
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1			
1	1	0			
1	1	1			

Example: State Table



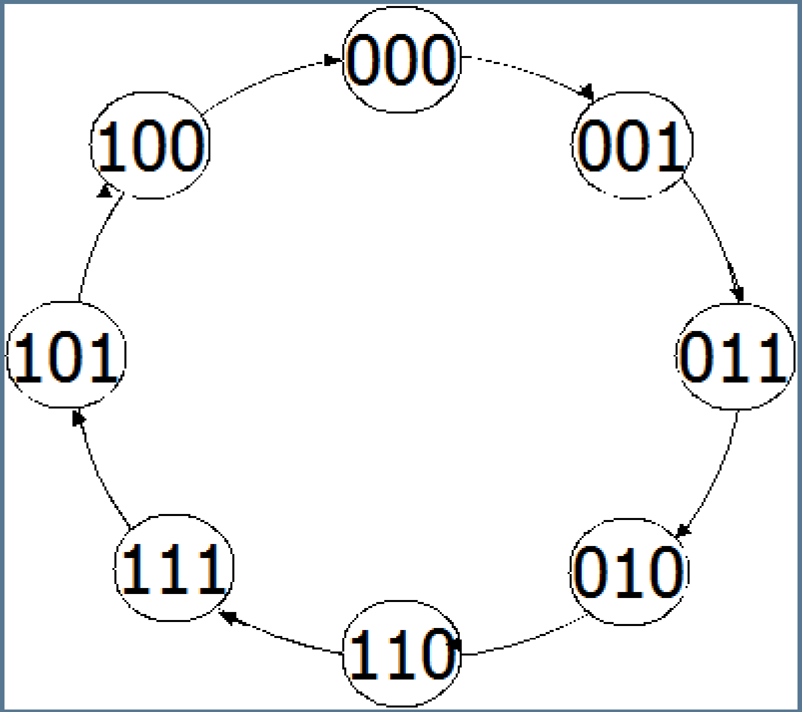
Present State			Next State		
A	B	C	A	B	C
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	0	0	0
1	0	1	1	0	0
1	1	0			
1	1	1			

Example: State Table



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

Example: State Table



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

Example

- Number of FFs: 3
- Type of FFs: JK

Q(t)	Q(t+1)	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

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

Example: FF Input Functions

Present State			Next State			Input functions					
A	B	C	A	B	C	JA	KA	JB	KB	JC	KC
0	0	0	0	0	1						
0	0	1	0	1	1						
0	1	0	1	1	0						
0	1	1	0	1	0						
1	0	0	0	0	0						
1	0	1	1	0	0						
1	1	0	1	1	1						
1	1	1	1	0	1						

Example: FF Input Functions

Present State			Next State			Input functions					
A	B	C	A	B	C	JA	KA	JB	KB	JC	KC
0	0	0	0	0	1						
0	0	1	0	1	1						
0	1	0	1	1	0						
0	1	1	0	1	0						
1	0	0	0	0	0						
1	0	1	1	0	0						
1	1	0	1	1	1						
1	1	1	1	0	1						

Example: FF Input Functions

Present State			Next State			Input functions					
A	B	C	A	B	C	JA	KA	JB	KB	JC	KC
0	0	0	0	0	1	0	X				
0	0	1	0	1	1	0	X				
0	1	0	1	1	0						
0	1	1	0	1	0						
1	0	0	0	0	0						
1	0	1	1	0	0						
1	1	0	1	1	1						
1	1	1	1	0	1						

Example: FF Input Functions

Present State			Next State			Input functions					
A	B	C	A	B	C	JA	KA	JB	KB	JC	KC
0	0	0	0	0	1	0	X				
0	0	1	0	1	1	0	X				
0	1	0	1	1	0	1	X				
0	1	1	0	1	0	0	X				
1	0	0	0	0	0						
1	0	1	1	0	0						
1	1	0	1	1	1						
1	1	1	1	0	1						

Example: FF Input Functions

Present State			Next State			Input functions					
A	B	C	A	B	C	JA	KA	JB	KB	JC	KC
0	0	0	0	0	1	0	X				
0	0	1	0	1	1	0	X				
0	1	0	1	1	0	1	X				
0	1	1	0	1	0	0	X				
1	0	0	0	0	0	X	1				
1	0	1	1	0	0	X	0				
1	1	0	1	1	1	X	0				
1	1	1	1	0	1	X	0				

Example: FF Input Functions

Present State			Next State			Input functions					
A	B	C	A	B	C	JA	KA	JB	KB	JC	KC
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	1	0	X	1	X	X	0
0	1	0	1	1	0	1	X	X	0	0	X
0	1	1	0	1	0	0	X	X	0	X	1
1	0	0	0	0	0	X	1	0	X	0	X
1	0	1	1	0	0	X	0	0	X	X	1
1	1	0	1	1	1	X	0	X	0	1	X
1	1	1	1	0	1	X	0	X	1	X	0

Example: FF Input Functions

- Simplify the expressions

A \ BC	00	01	11	10
0				1
1	X	X	X	X

JA = BC'

A \ BC	00	01	11	10
0		1	X	X
1			X	X

JB = A'C

A \ BC	00	01	11	10
0	1	X	X	
1		X	X	1

JC = A'B' + AB

Example: FF Input Functions

- Simplify the expressions

A \ BC	00	01	11	10
0	X	X	X	X
1	1			

KA = B'C'

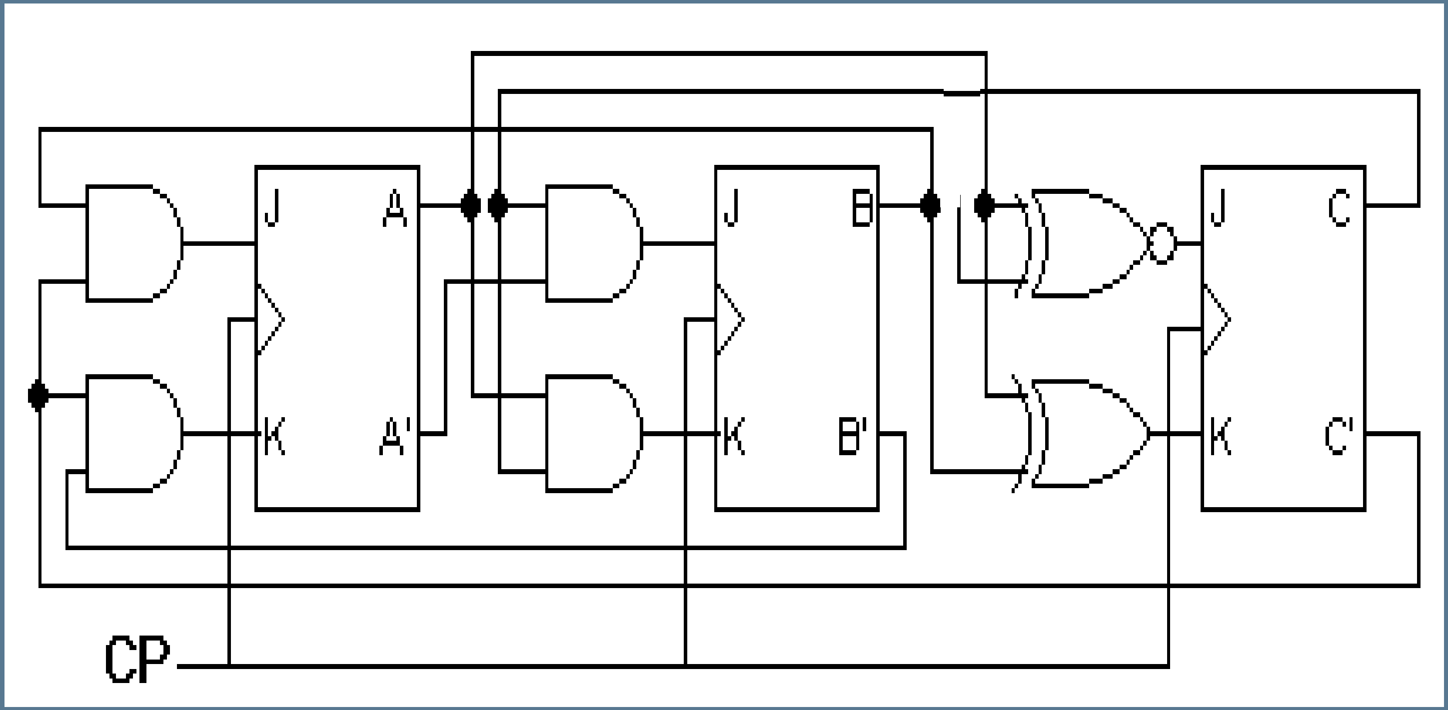
A \ BC	00	01	11	10
0	X	X		
1	X	X	1	

KB = AC

A \ BC	00	01	11	10
0	X		1	X
1	X	1	X	

KC = AB' + A'B

Example: Logic Diagram



JA = BC'

KA = B'C'

JB = A'C

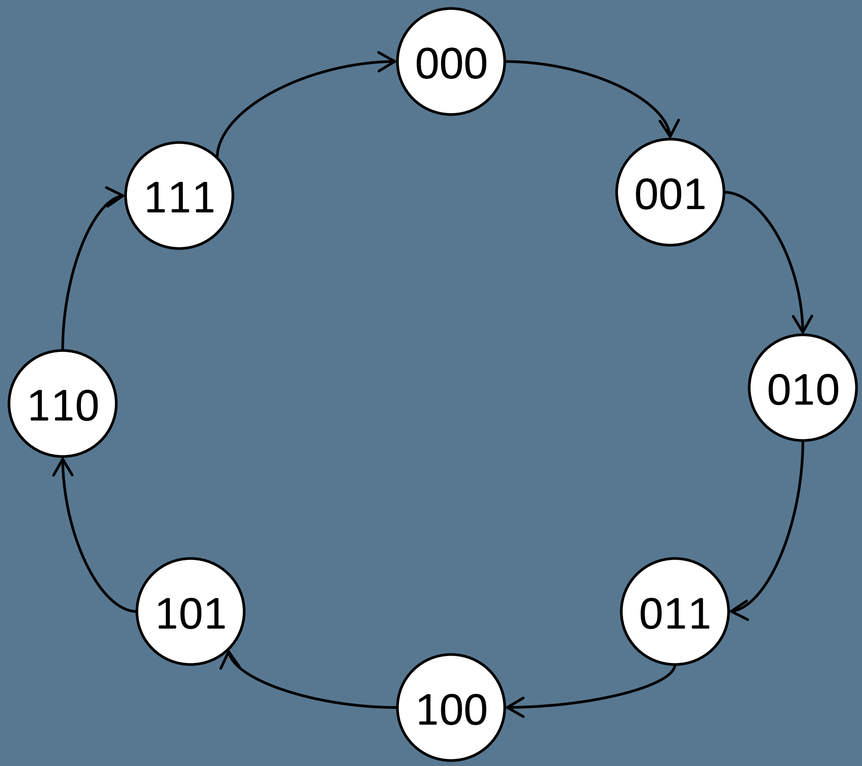
KB = AC

JC = A ⊙ B

KC = A ⊕ B

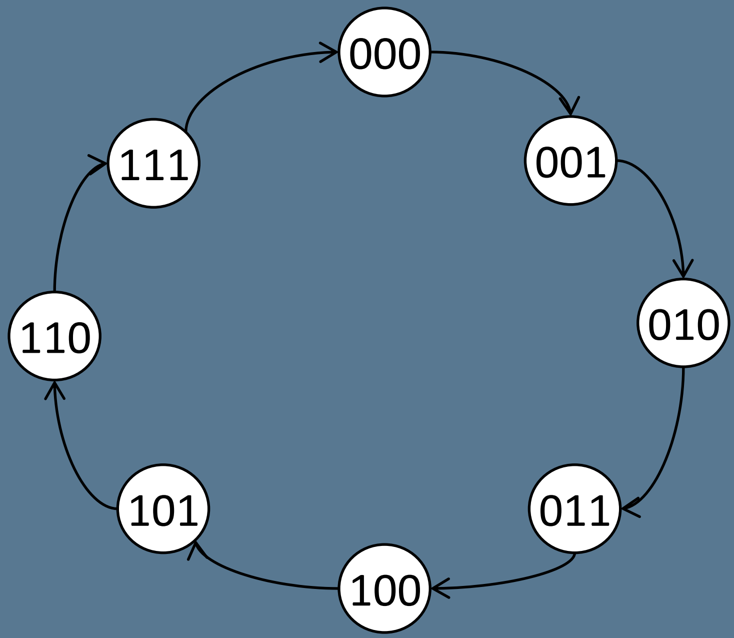
Example: 3-bit counter

State diagram



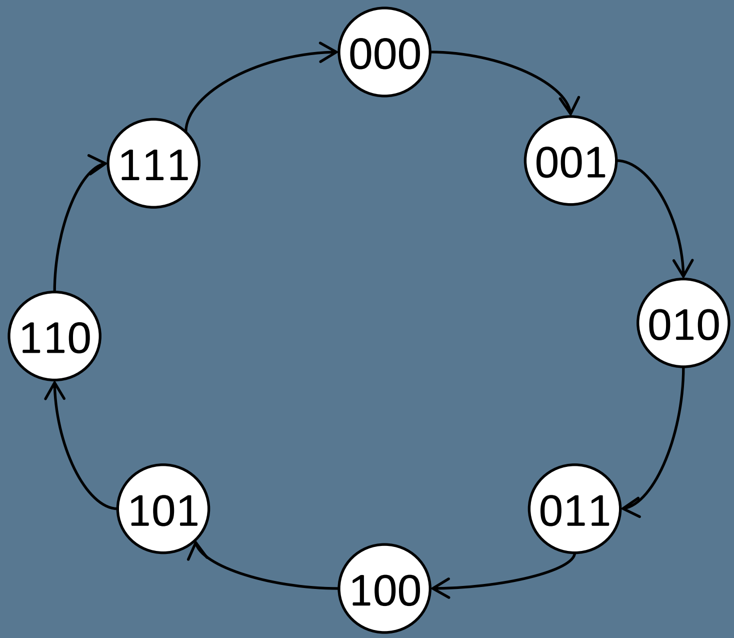
A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Example: State Table



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

Example: State Table



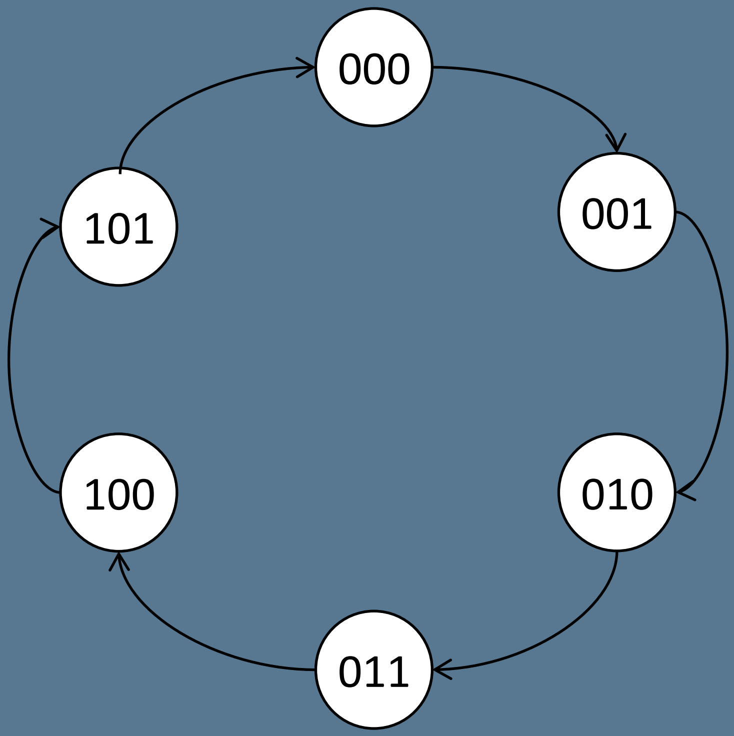
Present State			Next State		
A	B	C	A	B	C
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

Types of Synchronous Counters

- Binary counter – an n-bit binary counter that counts from 0 to (2^n-1) and back to 0 again.
- Modulo-N counter – counter that goes through a repeated sequence of N states. $(N <= 2^n)$
  - The maximum possible number of states (maximum modulus) of a counter is 2^n, where n is the number of flip-flops in the counter.

Example: modulo-6 counter

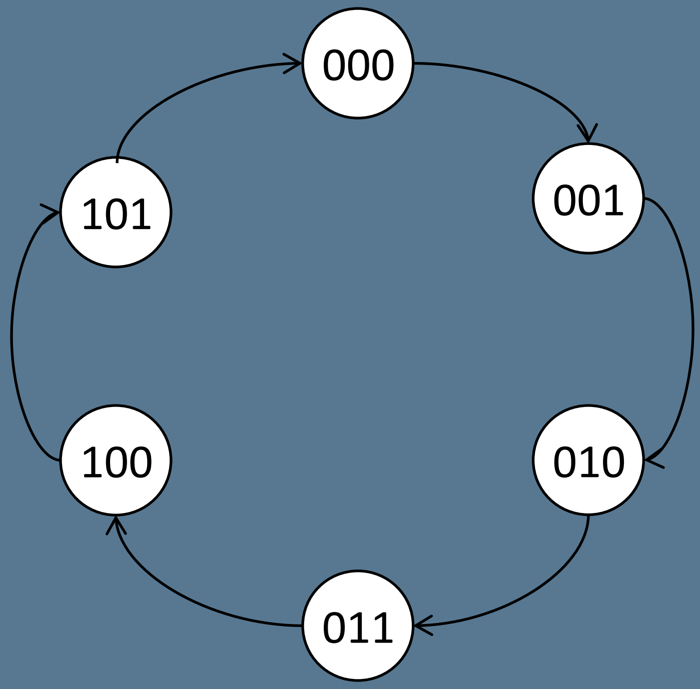
State diagram



A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1

Example: modulo-6 counter

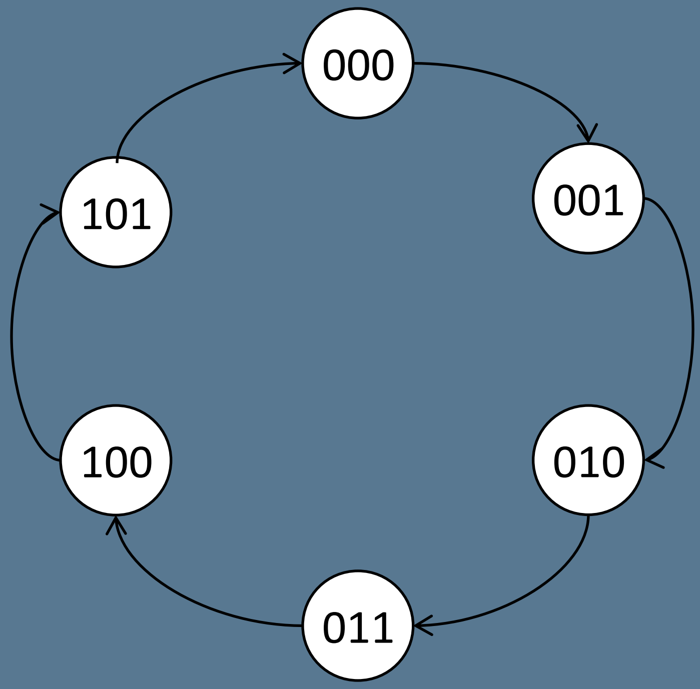
State table



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

Example: modulo-6 counter

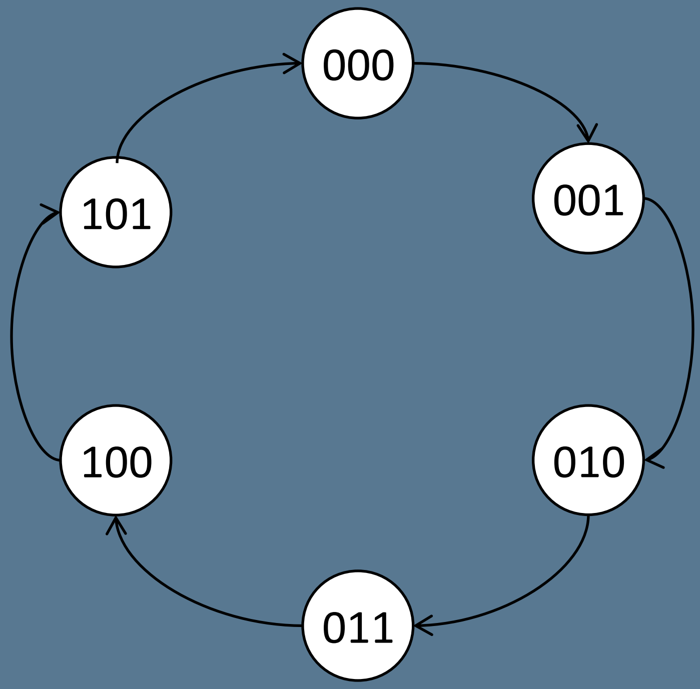
State table



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	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	0	0	0
1	1	0			
1	1	1			

Example: modulo-6 counter

State table



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	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	0	0	0
1	1	0	X	X	X
1	1	1	X	X	X

Decade Counter

- a counter with ten states in their sequence
- *recycle* – transition of the counter from its final state back to its original state

Clock Pulse	Q3	Q2	Q1	Q0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Types of Synchronous Counters

- Binary counter – an n-bit binary counter that counts from 0 to (2^n-1) and back to 0 again.
- Modulo-N counter – counter that goes through a repeated sequence of N states. ($N <= 2^n$)
  - The maximum possible number of states (maximum modulus) of a counter is 2^n, where n is the number of flip-flops in the counter.
- Up/down counter – The counter counts up or down according to the value of an additional input.