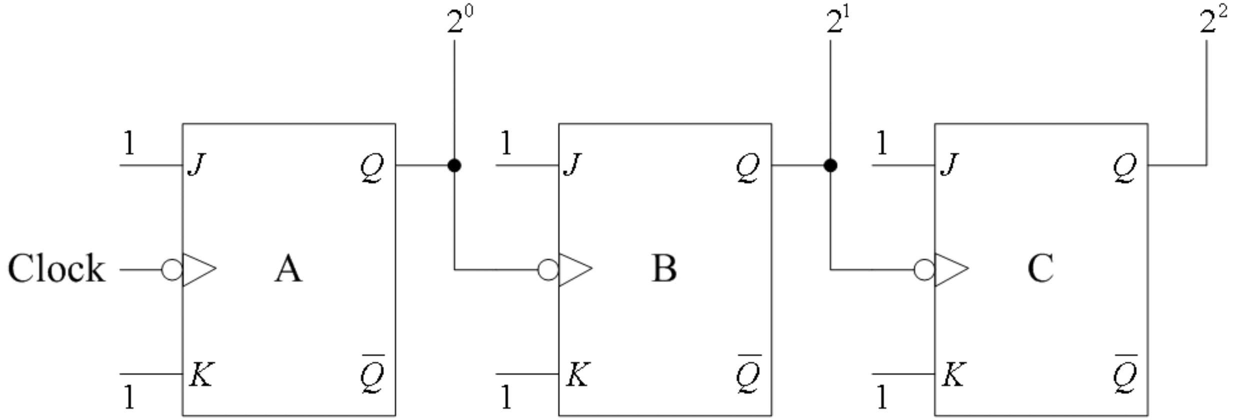
8 Counters

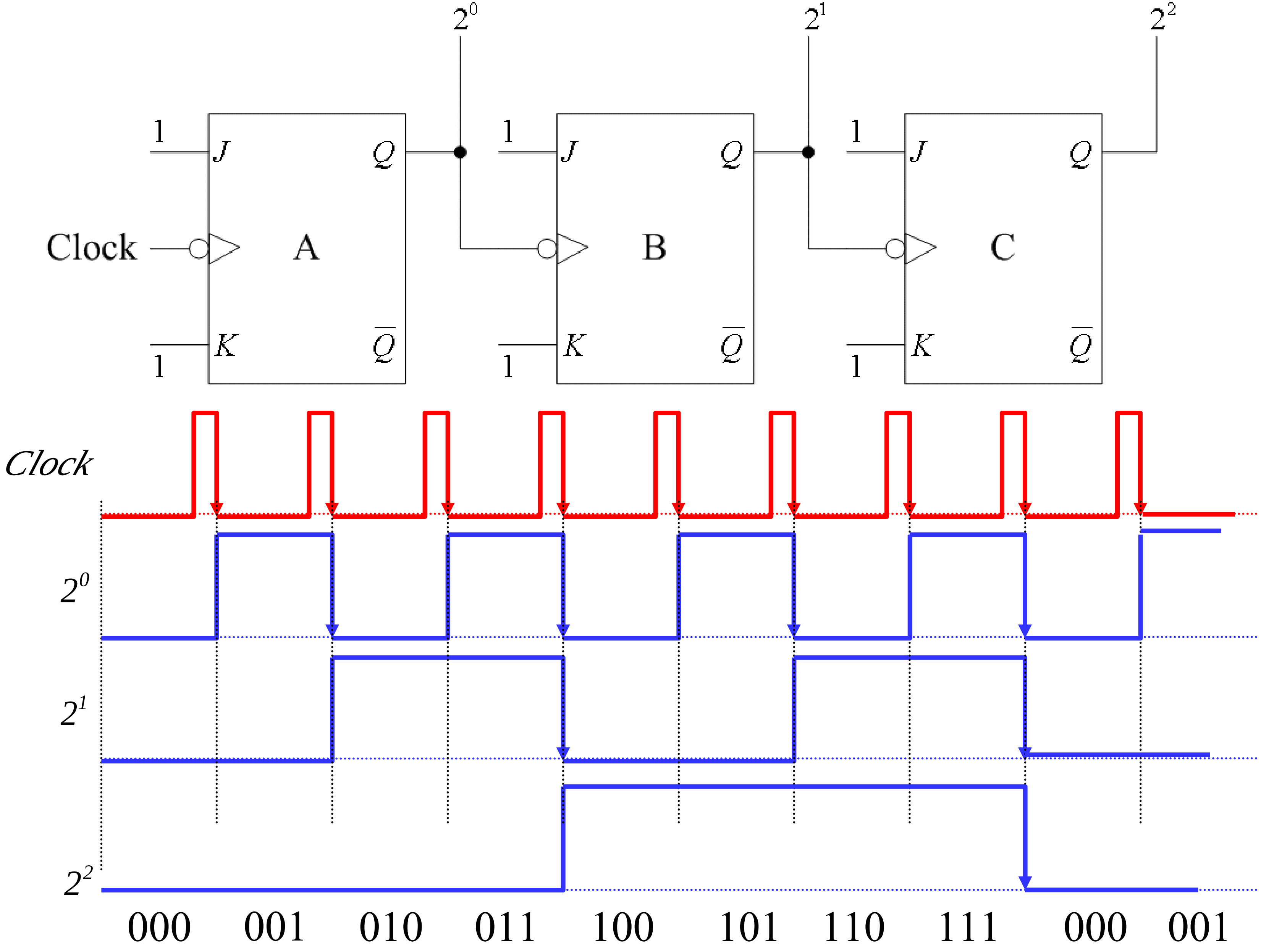
COUNTERS

- The counter is consisted of a number of *n* toggle flip-flops connected in cascade;
- The output of the Individual flip-flops will be toggled in sequence;
- And produces *n*-bit binary number, which increases/decreases by one;
- This forms a **Counter**;
- and be called the **Binary Counter**.



- For example,
- if we want to count 8 or 2^3 (3-bit binary numbers);
- We need to use 3 Toggle Flip-Flops connected in cascade;
- The circuit can generate 3-bit binary signals from 000 to 111;

Asynchronous Counter [4]



COUNTERS

- In the binary counting, the number of counting is called Modulus (Mod).
- For example, a 3-bit circuit (000 to 111) has 8 countable values, so Modulus = 8;
- This circuit is called the Mod 8 counter.

- Basically, the mod can be determined by the number of bits (or flip-flops) in counting.
- Mod = 2^{n} where *n* = number of bits used in counting (number of flip-flops);
- Start counting from 0;
- Maximum counting value = $2^{n} - 1$

- e.g. 3-bit Counter would be Mod 8,
- Start counting from 0;
- the maximum can be counted is 7

COUNTERS: Types

- Sometimes we need to count less than 2^n, i.e. 4, 8,16..
- e.g. count in a decimal digits.
- We can do it by designing a counter with Mod N, where $N \leq 2^n$;
- Where n is the number of bits (or Flip-flops) in counting;
- We can complete this counter by the way to activate the operation by the clock:
  - Asynchronous Counter

    - Clock triggering one by one,

    - Use *n* clocks.
  - Synchronous Counter

    - Clock triggering simultaneously,

    - Use only ONE clock.
- However, the types of counter can be divided by the direction of counting counter
  - Up Counter
  - Down Counter

Asynchronous Counter [1]

Asynchronous Counter or Ripple Counter

- Counter that has the Toggle Flip-Flops connecting in cascade;
- Every Flip-Flop, except the last one, will send its output signal as the clock signal to the next flip-flop;

Asynchronous

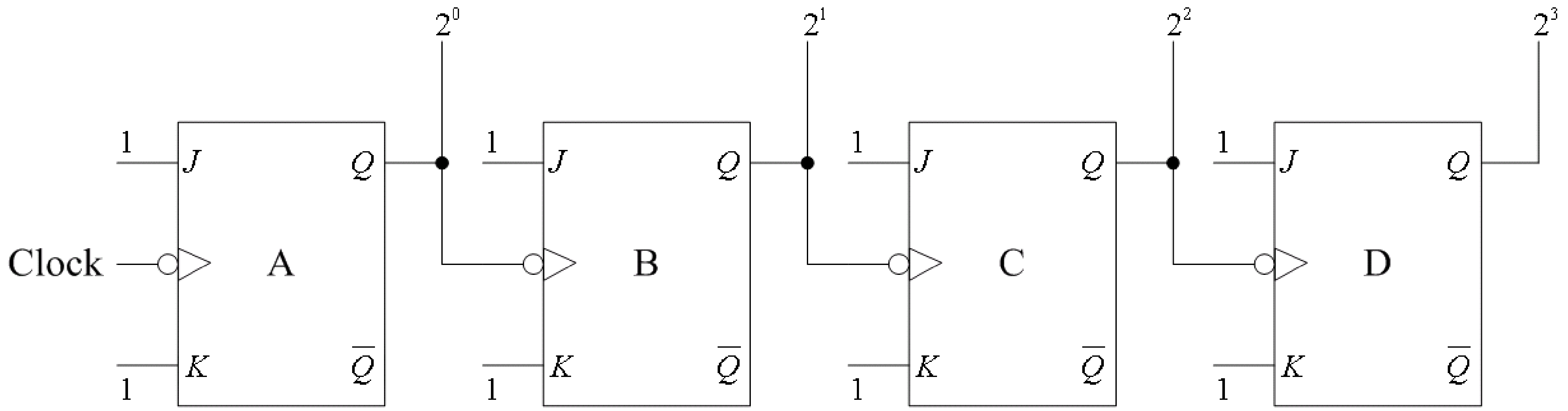
- which means each Flip-Flop in the circuit does not work at the same time;
- [Advantages] Easy to design. Just connecting FF in cascade.
- [Disadvantages] Error in Counting is easily occurred if using a high frequency clock.
- [Disadvantages] The working time of the counter is equal to the sum of the working time of all Flip-Flops.

➢ Counting of larger modulus has more delay time than that of less bits.

4-Bit Binary Counter

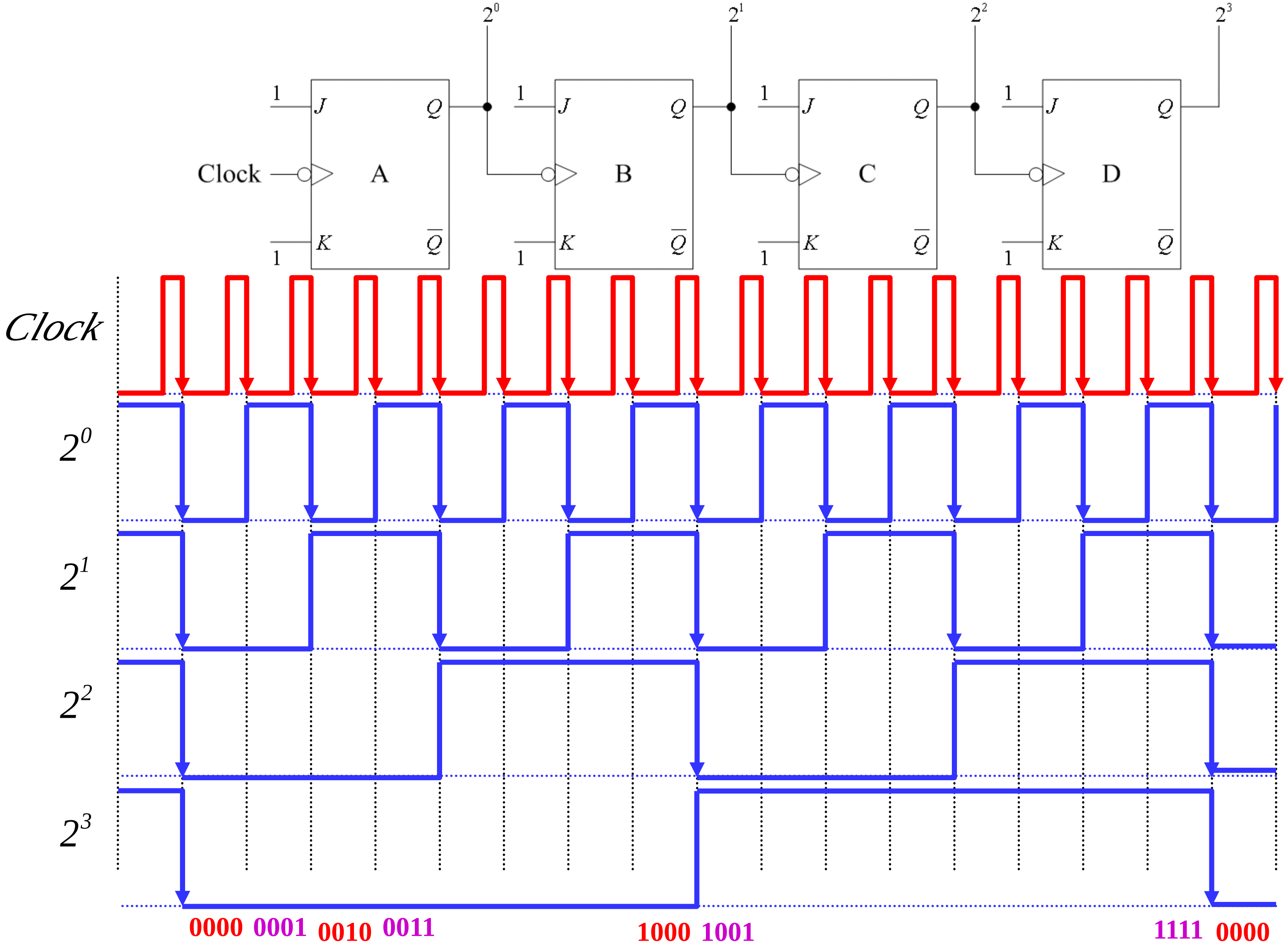
Asynchronous **Counter [2]**

This is an example of an asynchronous 4-bit counter.



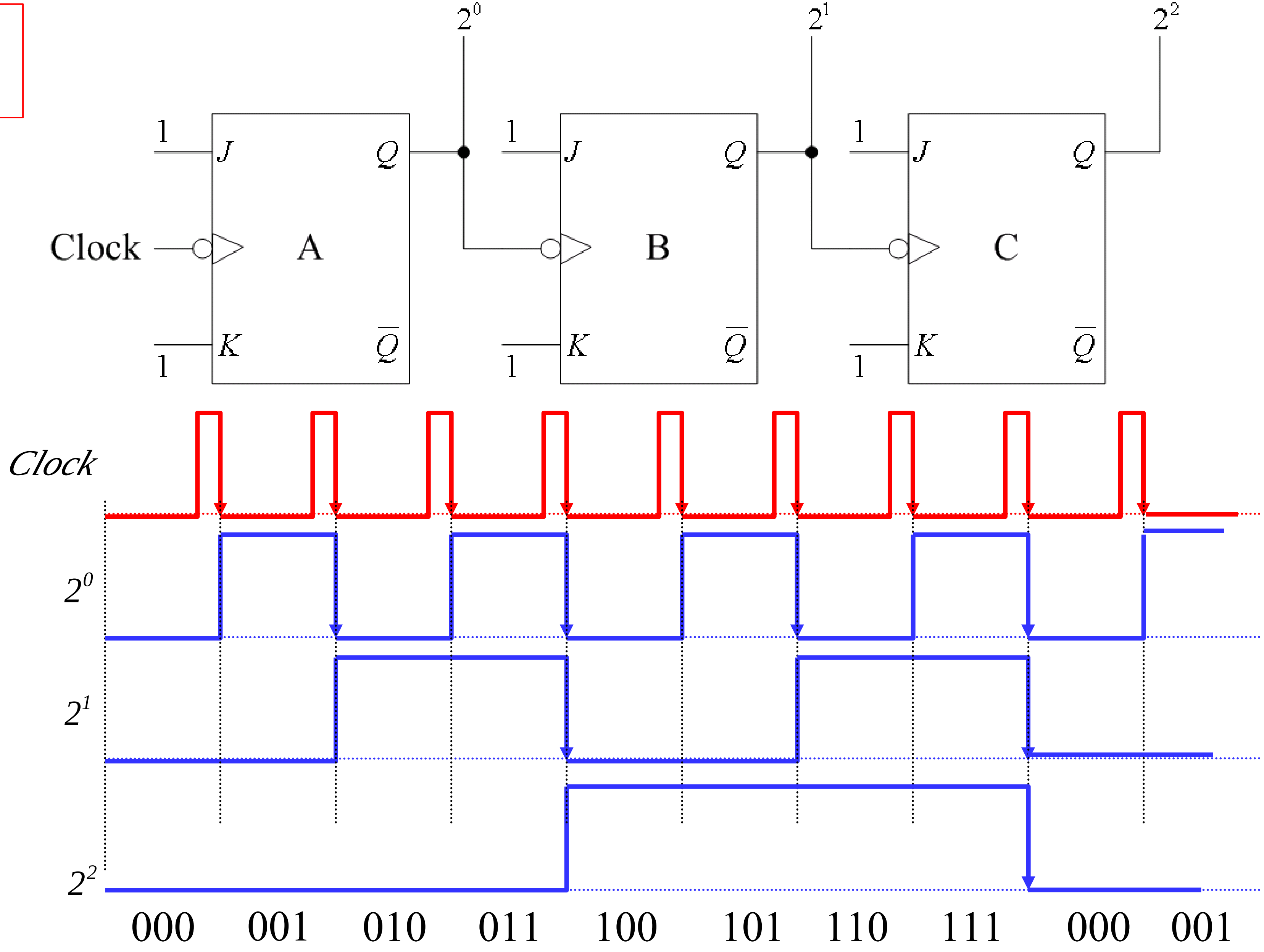
- Use 4 T Flip-flops connecting in cascade;
- Every flip-flop would send its output signal as the clock signal to the next flip-flop.

Asynchronous Counter [3]



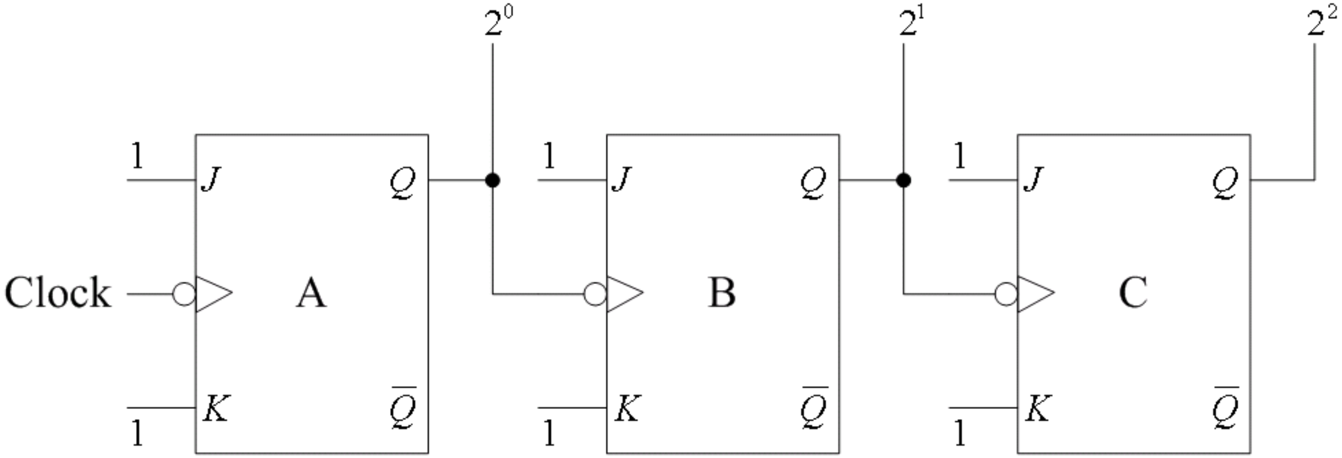
Asynchronous Counter [4]

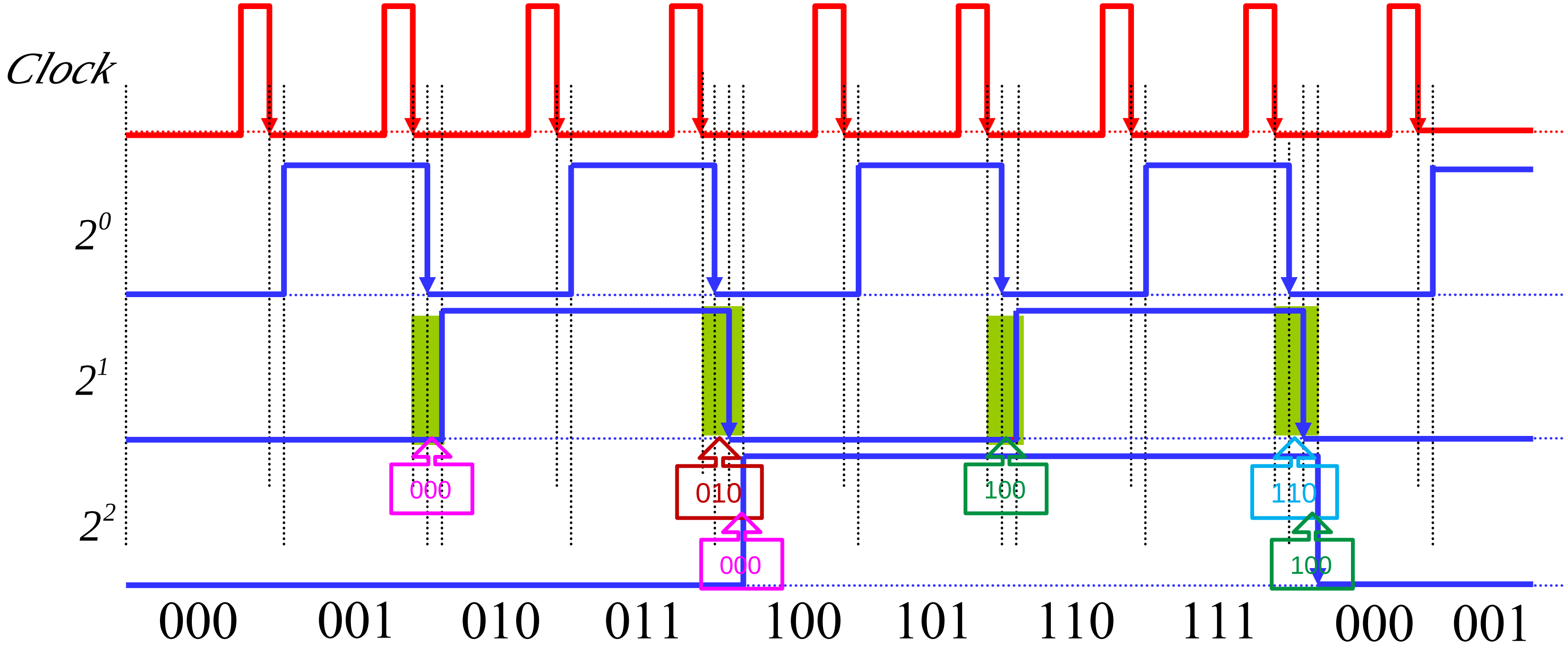
The ideal case:
No Problem.



Asynchronous Counter [5]

The real case:
Problem occurs



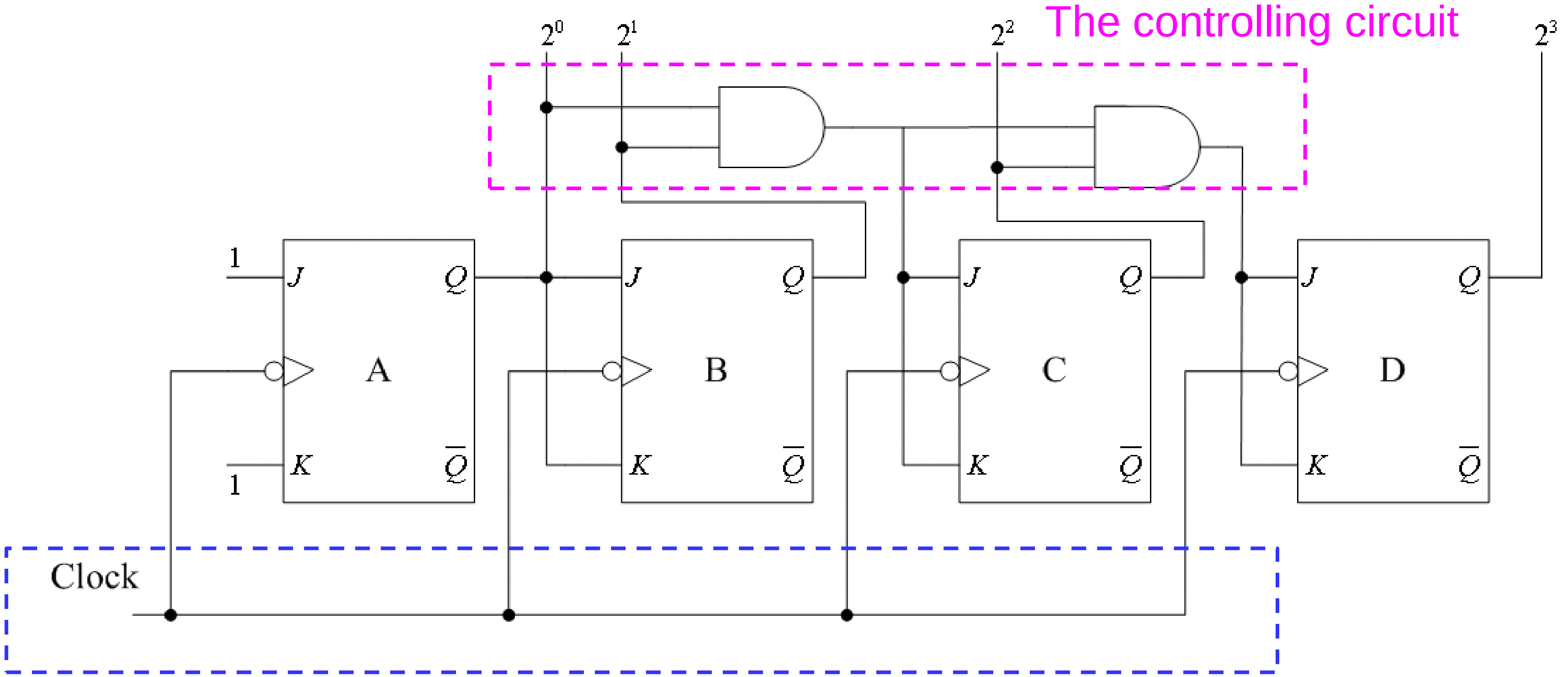


Synchronous Counter [1]

- To control ALL Flip-Flops to operation simultaneously;
- Using ONE clock signal (the clock signal is applied to every Flip-flop).
- In conjunction with the control signal circuit (has to be designed) for the desired operation;
- The counting output has the delay time that caused from only ONE Flip-flop;
- **Advantages;**
  - ➢The operation time of the circuit is equal to the single Flip-Flop delay time.
  - ➢Less error occurs
  - ➢Able to operate in a higher or faster frequency.
- **Disadvantages:**
  - ➢Require to design of the control circuits (complicated).

Synchronous Counter [2]

Example of 4-Bit Binary Synchronous Counter Circuit



The controlling circuit

The clock signal triggers ALL flip-flops providing operation at the same time.

Synchronous Counter: Control circuit

- The operation of the circuit is to control the output to be changed as needed
- e.g. Suppose we want to change the Flip-flop output from 0 to 1:
- How to apply the inputs:
  - ➢ In In generate the case of D- Flip-flop: Enter *D* with 1
  - ➢ And for JK Flip-flop : Enter *J* = 1 and *K* = 0 or *K* = 1 (*K* = don’t care)
  - ➢ Or for al usage of JK Flip-Flop: May activate the Preset.

Truth Table for a negative edge-triggered JK flip-flop

J	K	C	Q	$\overline{Q}$	State
1	0	↓	1	0	Set
0	1	↓	0	1	Reset
0	0	X	Q	$\overline{Q}$	Unchanged
1	1	↓	$\overline{Q}$ / Q		Toggle

Truth Table for a negative edge-triggered JK flip-flop

Conditions		Control		
Before clock	After clock	Before clock		
Present Q	Next Q	State	J	K
0	0	N.C. Reset	0	×
0	1	Set Toggle	1	×
1	0	Reset Toggle	×	1
1	1	N.C. Set	×	0

Synchronous Counter Design

The Steps for Designing a Synchronous Counter.

1. Assign the desired counting (output) table.
2. Make the truth table for finding the input (*J* and *K*) of every Flip-Flop from the given output, and the excitation table of Flip-Flop (if using JK Flip-Flop).
3. Use the truth table to find the input for the Flip-Flops (in the form of an equation) using Boolean Algebra / Karnaugh Map.
4. Use the equation to create the circuit.

Synchronous Counter Design: Example

1 Assign the desired counting (output) table.

The counting follows this sequence:

0 → 1

1 → 2

2 → 3

3 → 4

⋮

7 → 0

And get the table beside:

Present State			Next State		
Q_C	Q_B	Q_A	Q_C	Q_B	Q_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

Synchronous Counter Design: Example

2 Make the truth table for finding the input (J and K) of every Flip-Flop from the given output, and the excitation table of Flip-Flop (if using JK Flip-Flop).

P.S.			N.S.			C		B		A	
Q_C	Q_B	Q_A	Q_C	Q_B	Q_A	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	1	0	1	X	0	*0*	*X*	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	1	1	1	X	0	X	0	1	X
1	1	1	0	0	0	X	1	X	1	X	1

	00	01	11	10
0				
1				

	00	01	11	10
0				
1				

Synchronous Counter Design: Example

- ❑ Consider the output from Q changing from **0 → 0**

P.S.			N.S.			C		B		A	
Q_C	Q_B	Q_A	Q_C	Q_B	Q_A	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	1	1	1	X	0	X	0	1	X
1	1	1	0	0	0	X	1	X	1	X	1

Synchronous Counter Design: Example

❑ Consider the output from *Q* changing from **1 → 0**

P.S.			N.S.			*C*		*B*		*A*	
Q_C	Q_B	Q_A	Q_C	Q_B	Q_A	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	1	0	1	X	0	*0*	*X*	1	X
1	0	1	1	1	0	X	0	1	X	X	1
1	1	0	1	1	1	X	0	X	0	1	X
1	1	1	0	0	0	X	1	X	1	X	1

Synchronous Counter Design: Example

3 From the truth table, write the Karnaugh Map to find the input equations for the individual Flip-Flops, e.i. J_A, K_A, J_B, K_B, J_C and K_C

J_C

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

$$J_C = AB$$

K_C

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

$$K_C = AB$$

Synchronous Counter Design: Example

J_B

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

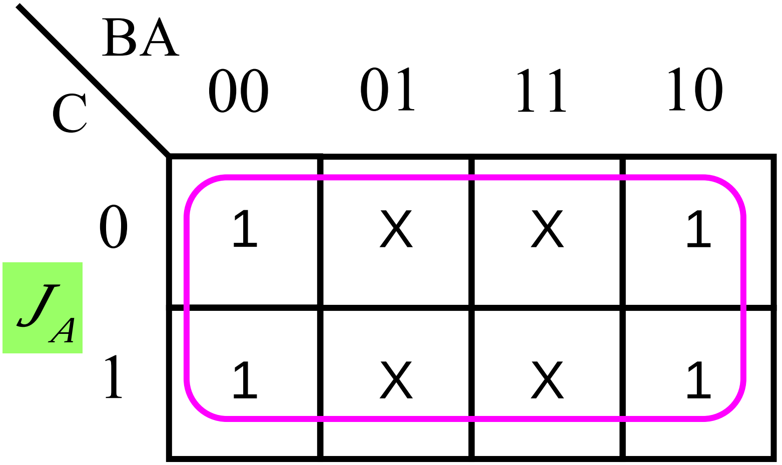
$$J_B = A$$

K_B

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

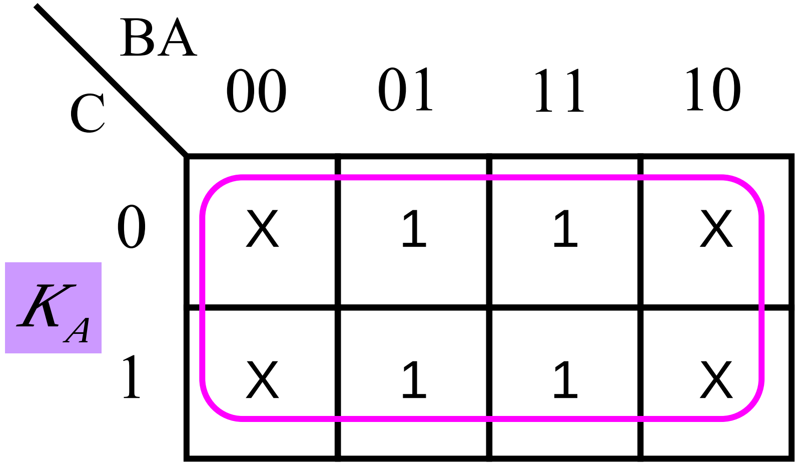
$$K_B = A$$

Synchronous Counter Design: Example



J_A

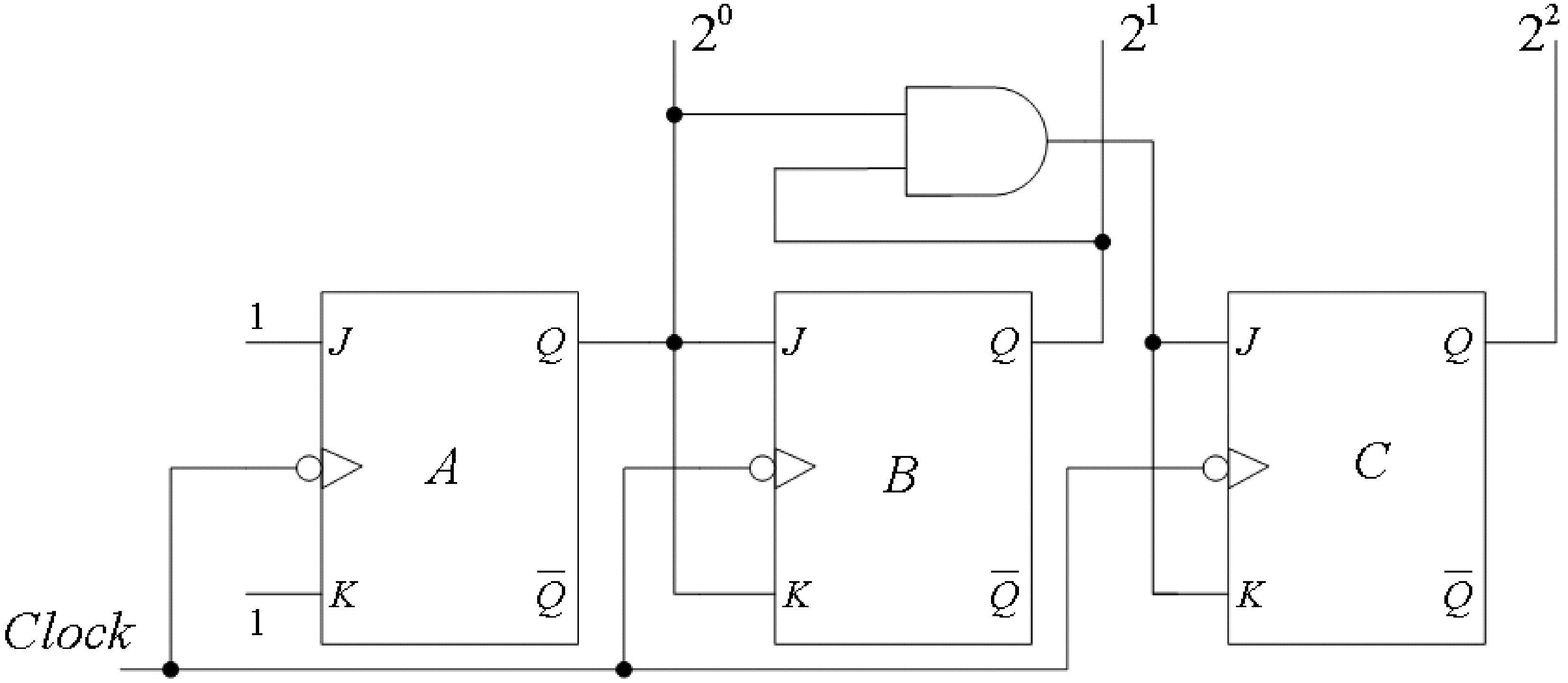
C \ BA	00	01	11	10
0	1	X	X	1
1	1	X	X	1



K_A

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

Synchronous Counter Design: Example



$J_A = 1$ $\quad J_B = A \quad J_C = AB$

$K_A = 1$ $\quad K_B = A \quad K_C = AB$

Modulus *N* (Mod-*N*) Counter

❑ The Mod-*N* counter is the counter that counts *N* numbers:

- Basically, a normal *n*-bit BINARY counter;
- Where $N \leq 2^n$

e.g.

○ The Mod-8 counter is 3-bit counter that produces the sequence of 0, 1, 2, …, 7

○ The Mod-5 counter is 3-bit counter that produces the sequence of 0, 1, 2, 3, 4

○ So, Mod-5 = Mod-8 + Control circuit.

❑ If we want to design Mod-10 counter:

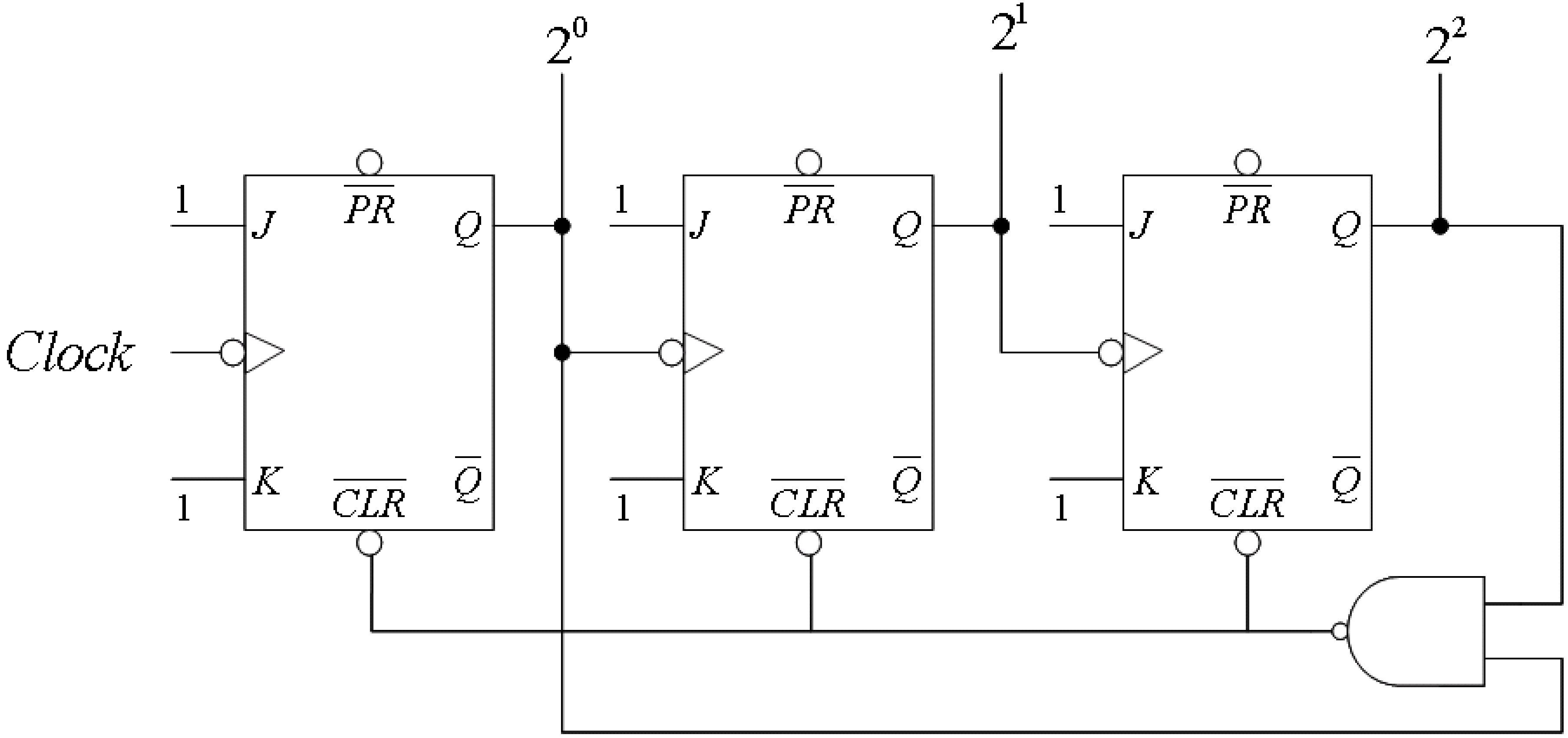
- $10 \leq 2^4$
- Require 4-bit counter with a control circuit.

Asynchronous Mod-*N* Counter

- Asynchronous Mod-*N* Counter is the same as the typical binary counter;
- When it reaches the desire maximum, then use it to reset counting to 0;
- By activating the clear input (Sending 0 to $\overline{\text{CLR}}$ pin);

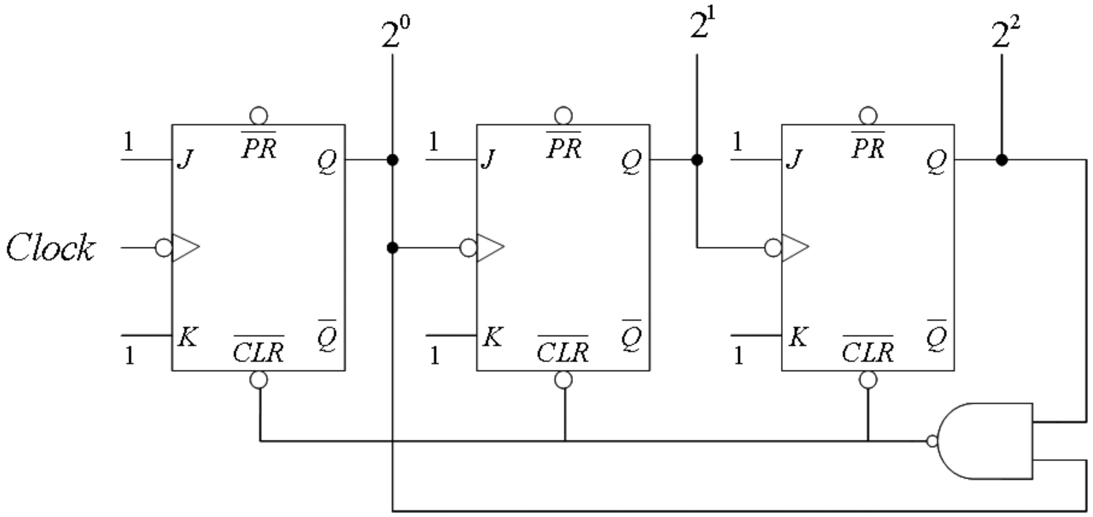
[Need to observe the desire maximum to produce a control signal for $\overline{\text{CLR}}$ pin]

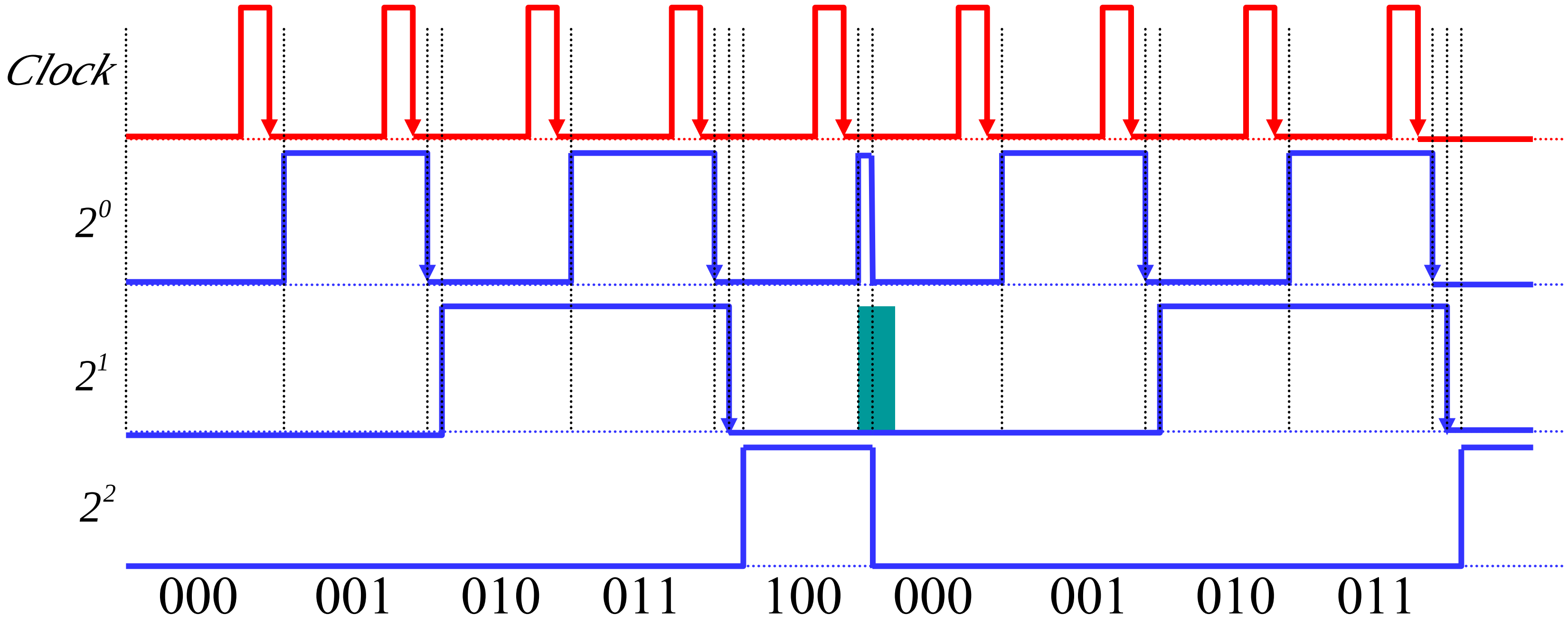
Asynchronous Mod-5 Counter



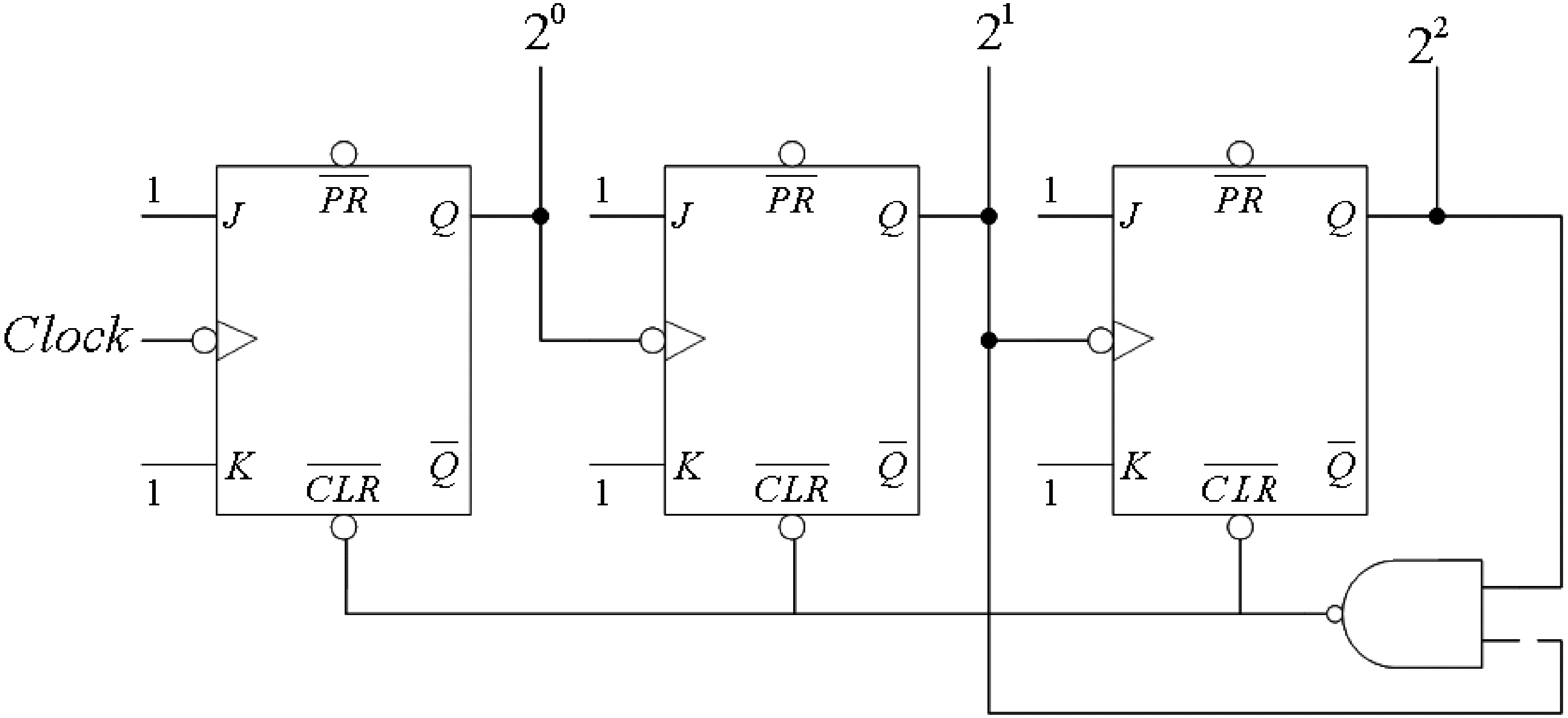
Q_C	Q_B	Q_A
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	**0**	**1**

Asynchronous Mod-5 Counter



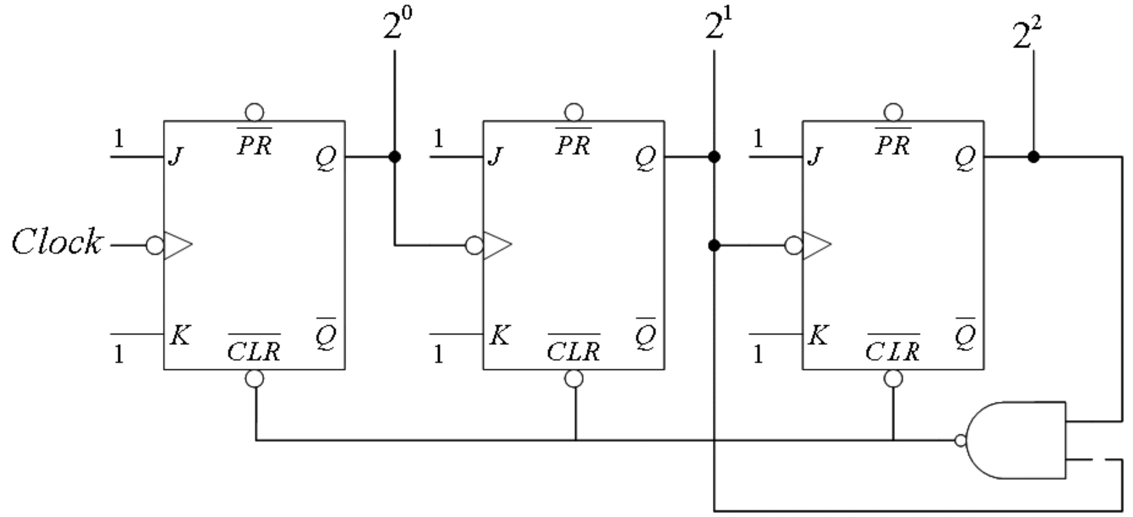


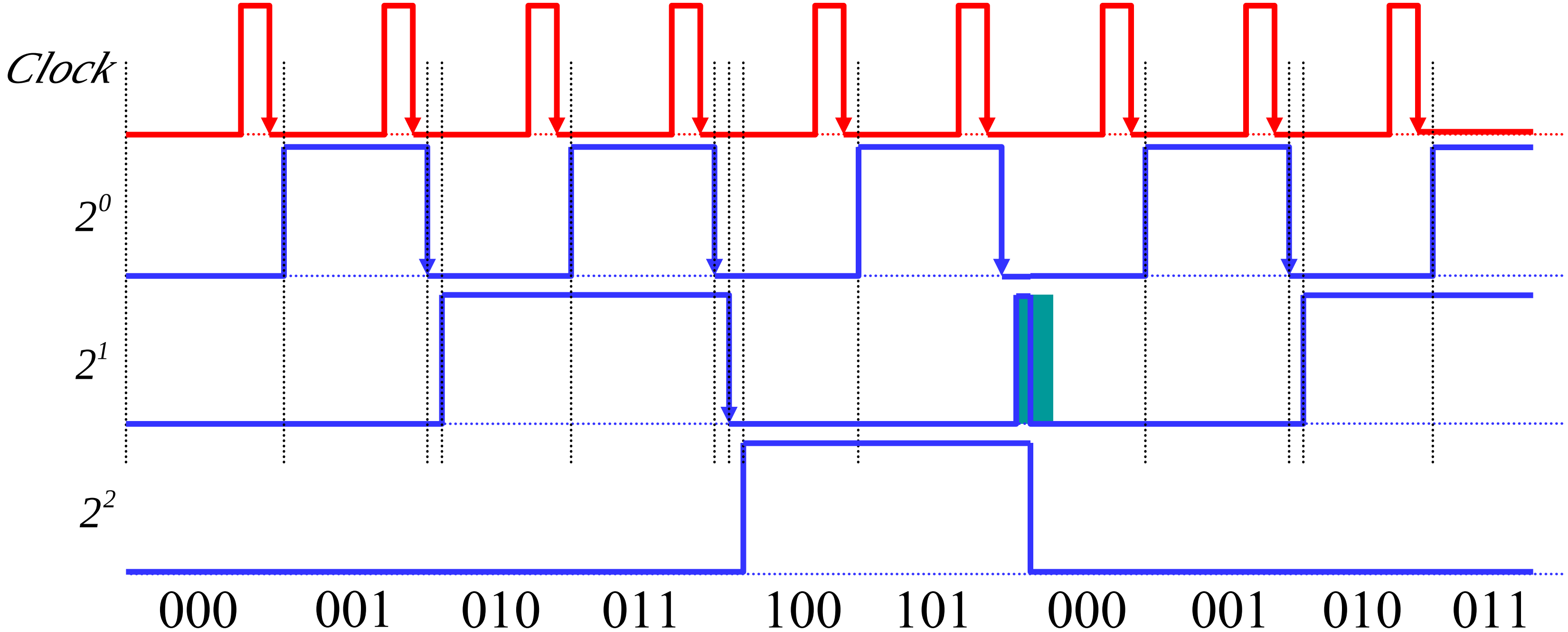
Asynchronous Mod-6 Counter



Q_C	Q_B	Q_A
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	**1**	**0**

Asynchronous Mod-6 Counter





Synchronous Mod-*N* Counter

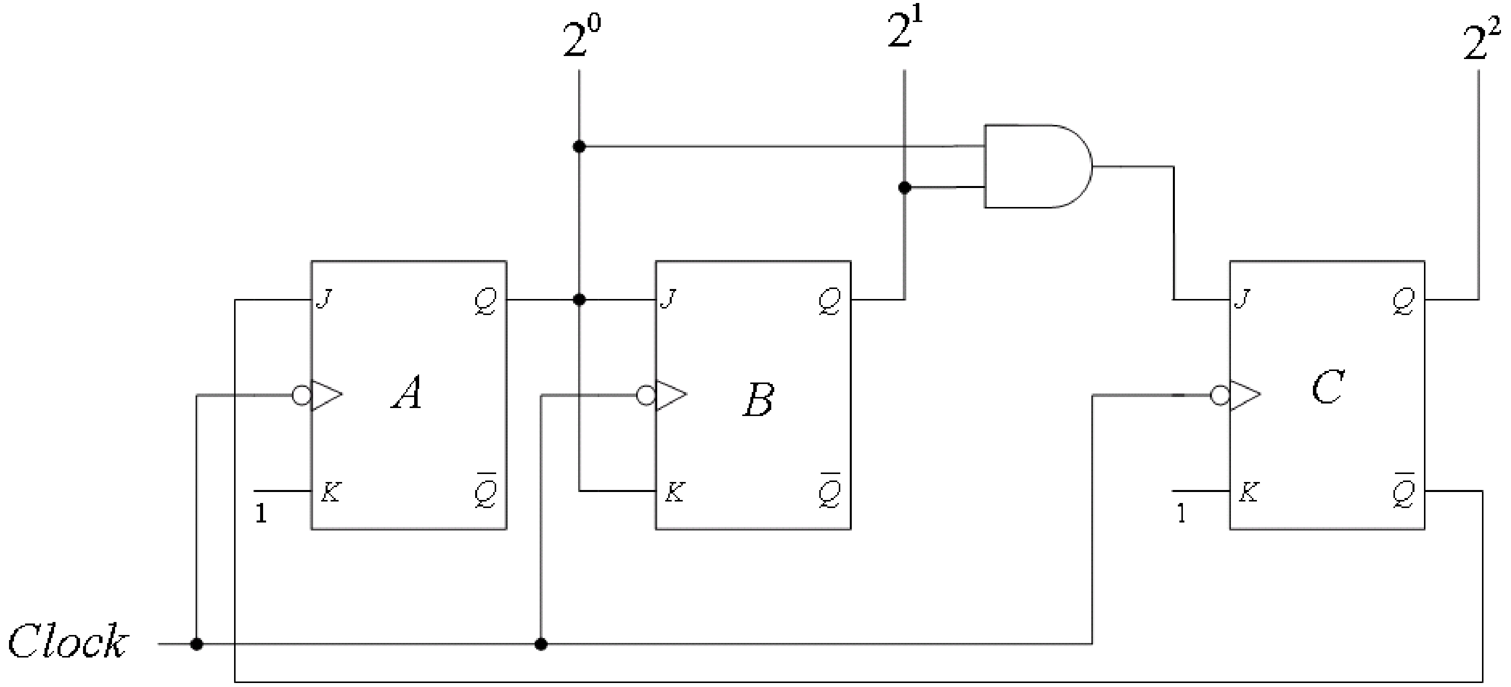
- ○ The design is the same as normal synchronous counter circuit design,
- ○ by counting the number as assigned.

Qc	Qb	Qa	Qc	Qb	Qa	Jc	Kc	Jb	Kb	Ja	Ka
0	0	0	0	0	1	0	x	0	x	1	x
0	0	1	0	1	0	0	x	1	x	x	1
0	1	0	0	1	1	0	x	x	0	1	x
0	1	1	1	0	0	1	x	x	1	x	1
1	0	0	0	0	0	x	1	0	x	0	x

	00	01	11	10
0				
1				

	00	01	11	10
0				
1				

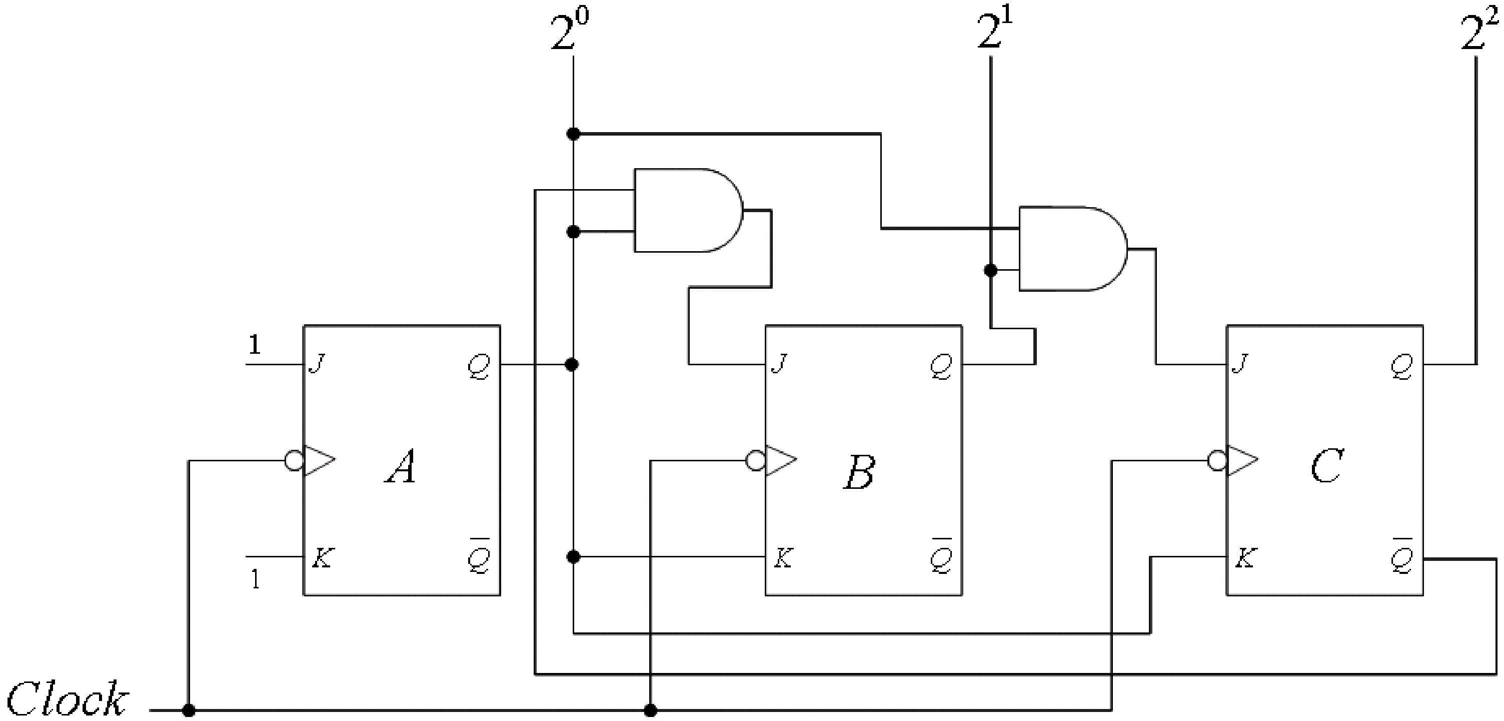
Synchronous Mod-5 Counter



$$J_A = \overline{C} \qquad J_B = A \qquad J_C = AB$$

$$K_A = 1 \qquad K_B = A \qquad K_C = 1$$

Synchronous Mod-6 Counter



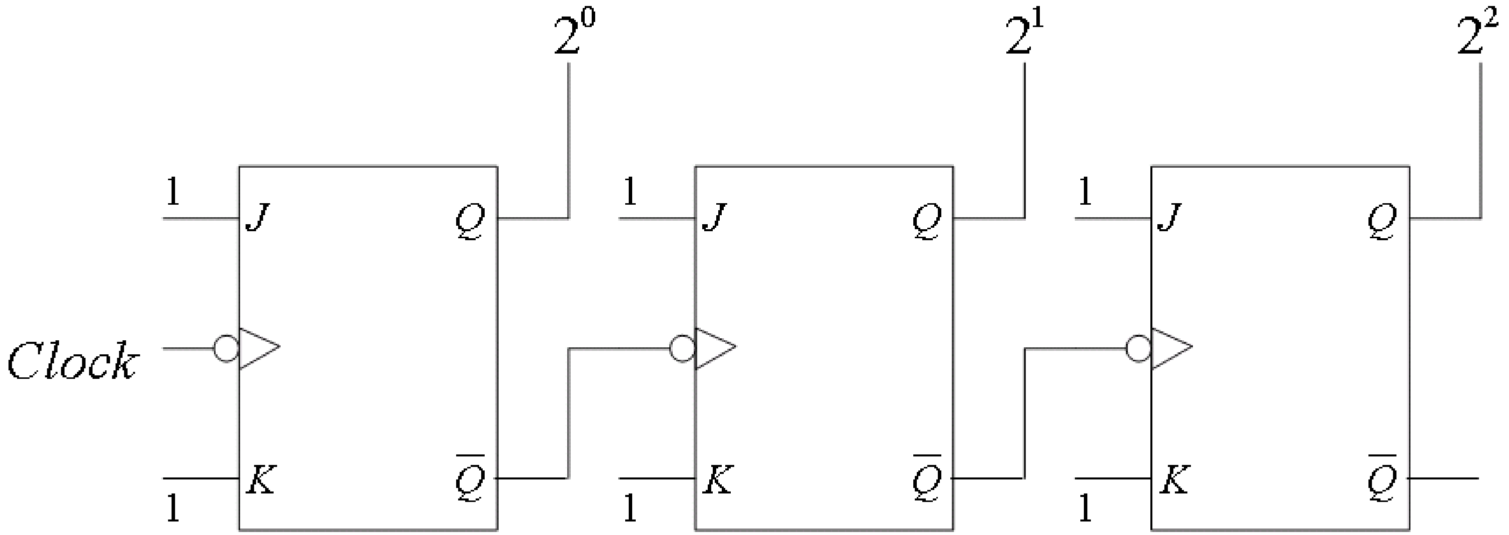
$J_A = 1$ $J_B = A\overline{C}$ $J_C = AB$

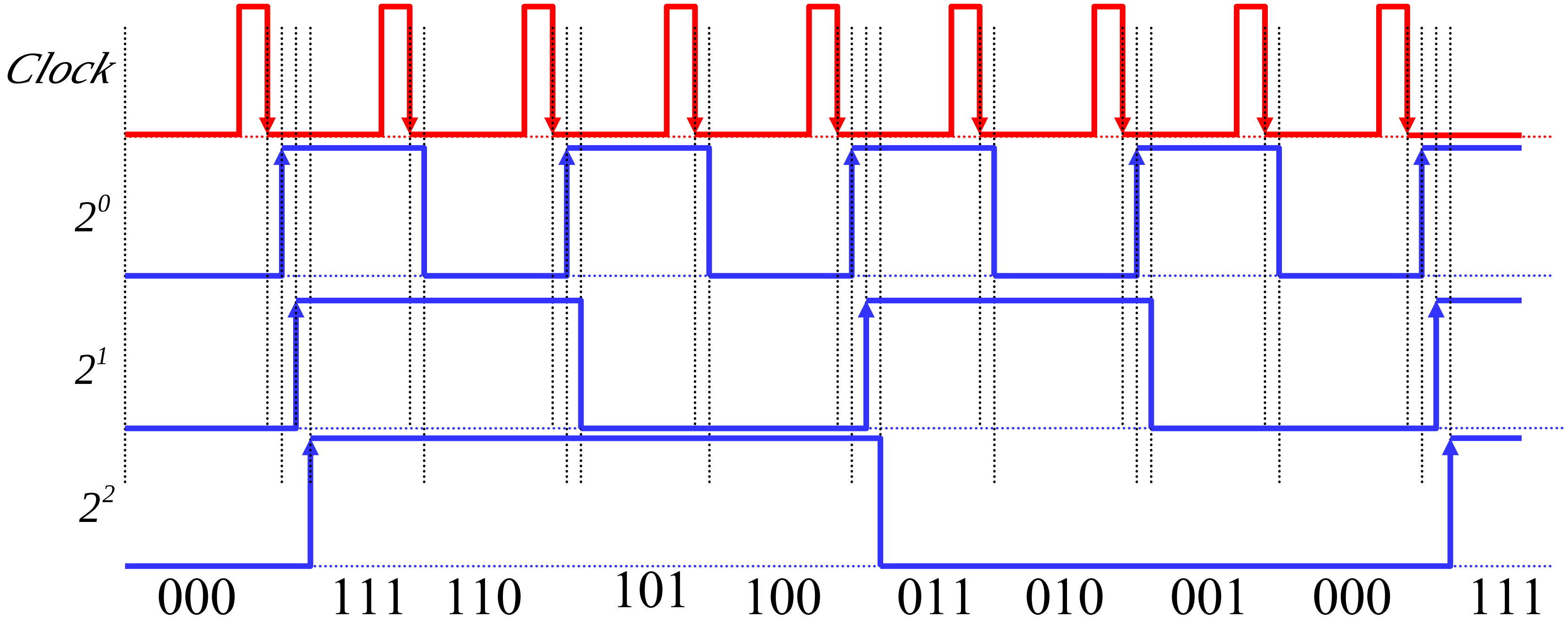
$K_A = 1$ $K_B = A$ $K_C = A$

DOWN Counter

- The circuit that counts downward from max to min e.g. 111, 110, 101,…, 000.
- Asynchronous Down Counter
  - Similar to the Asynchronous Counter,
  - Using $\overline{Q}$ as a clock signal to trigger the next Flip-Flops, instead of *Q.*
- Synchronous Down Counter
  - The design principles are like a synchronous counter.
  - Follow the truth table assigned to design the control unit.

Asynchronous Down Counter





Synchronous Down Counter

Qc	Qb	Qa	Qc	Qb	Qa	Jc	Kc	Jb	Kb	Ja	Ka
0	0	0	1	1	1	1	x	1	x	1	x
1	1	1	1	1	0	x	0	x	0	x	1
1	1	0	1	0	1	x	0	x	1	1	x
1	0	1	1	0	0	x	0	0	x	x	1
1	0	0	0	1	1	x	1	1	x	1	x
0	1	1	0	1	0	0	x	x	0	x	1
0	1	0	0	0	1	0	x	x	1	1	x
0	0	1	0	0	0	0	x	0	x	x	1

	00	01	11	10
0				
1				

	00	01	11	10
0				
1				

How to Design Synchronous 3-bit Down Counter

①

P.S			N.S			C		B		A	
C	B	A	C	B	A	J_C	K_C	J_B	K_B	J_A	K_A
1	1	1	1	1	0	X	0	X	0	X	1
1	1	0	1	0	1	X	0	X	1	1	X
1	0	1	1	0	0	X	0	0	X	X	1
1	0	0	0	1	1	X	1	1	X	1	X
0	1	1	0	1	0	0	X	X	0	X	1
0	1	0	0	0	1	0	X	X	1	1	X
0	0	1	0	0	0	0	X	0	X	X	1
0	0	0	1	1	1	1	X	1	X	1	X

$J_C = \bar{A}\bar{B}$

$K_C = \bar{A}\bar{B}$

$J_B = \bar{A}$

$K_B = \bar{A}$

$J_A = 1$

$K_A = 1$

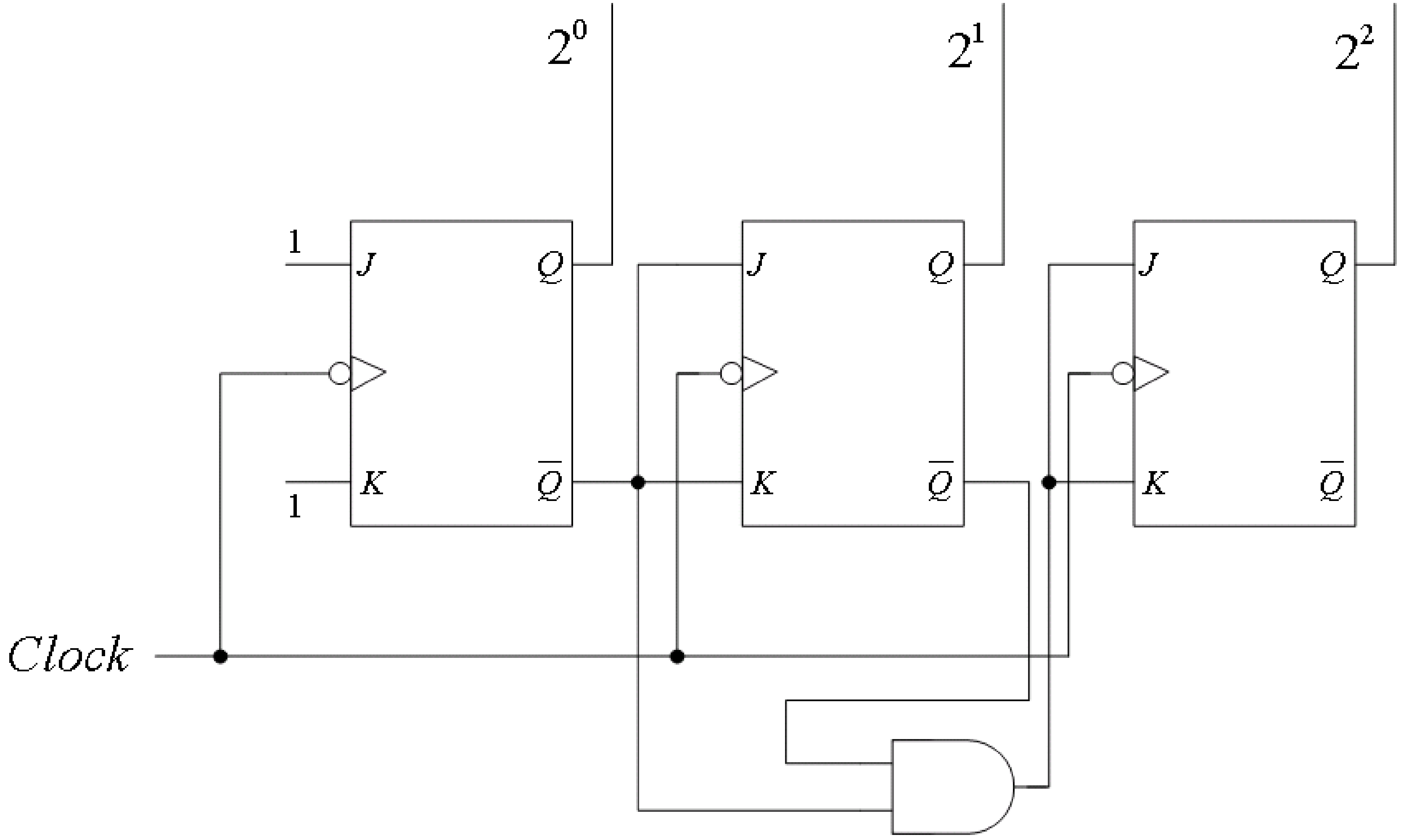
J

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

A

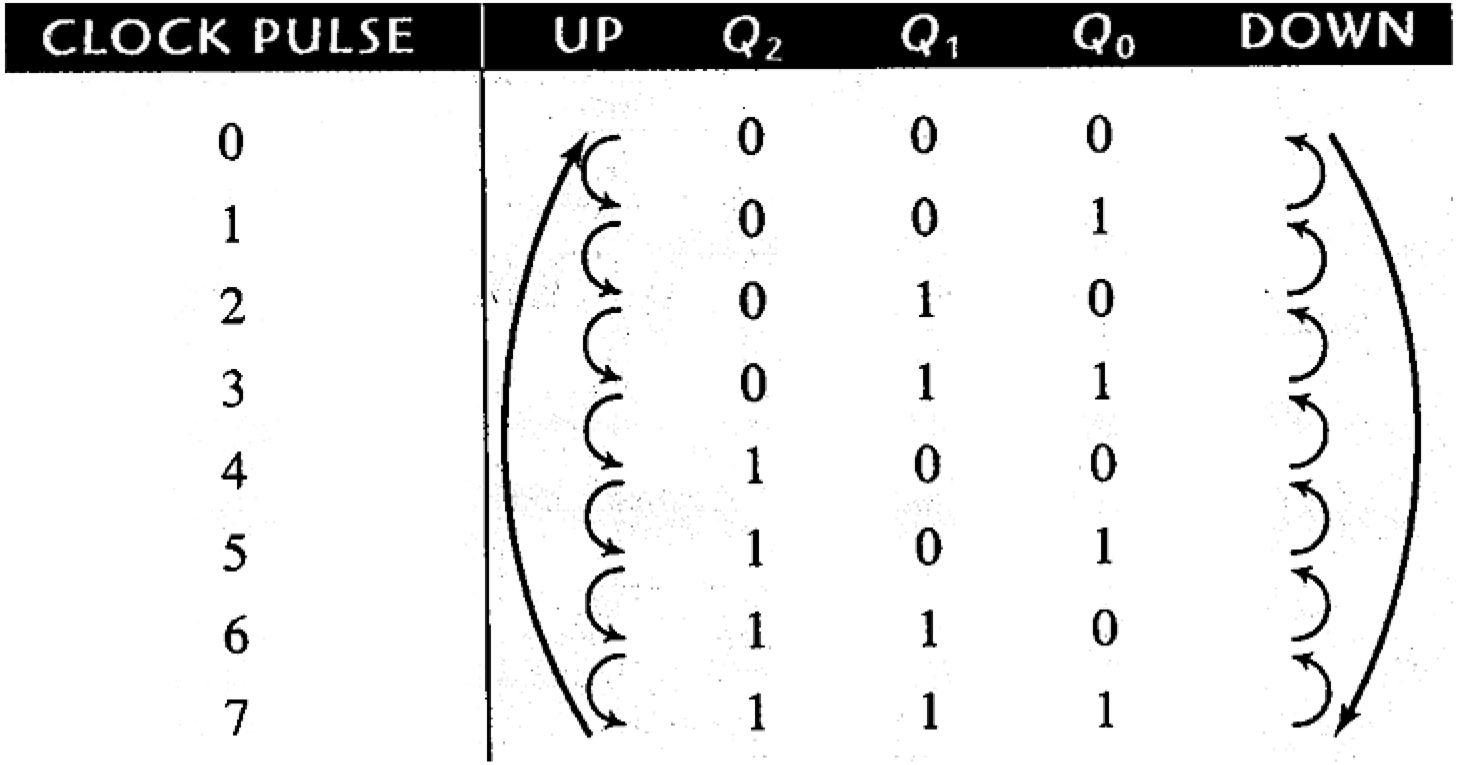
CB \ A	00	01	11	10
0	X	1	1	X
1	X	1	1	X

Synchronous Down Counter



Synchronous Up-Down Counter

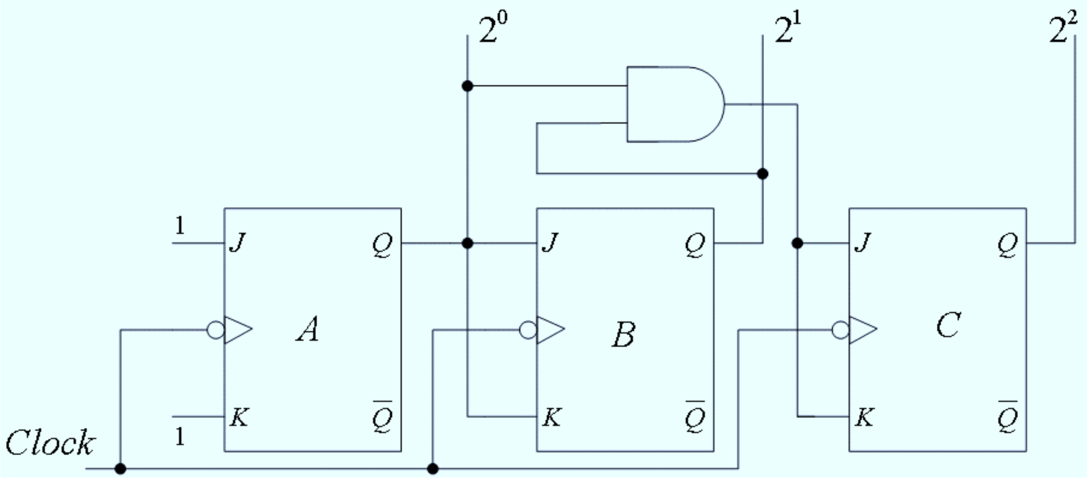
- Able to count UP or DOWN in the same circuit;
- By a control unit to the selection the direction to count (up or down);
- This counter could be uses counting the number of cars in the car park.



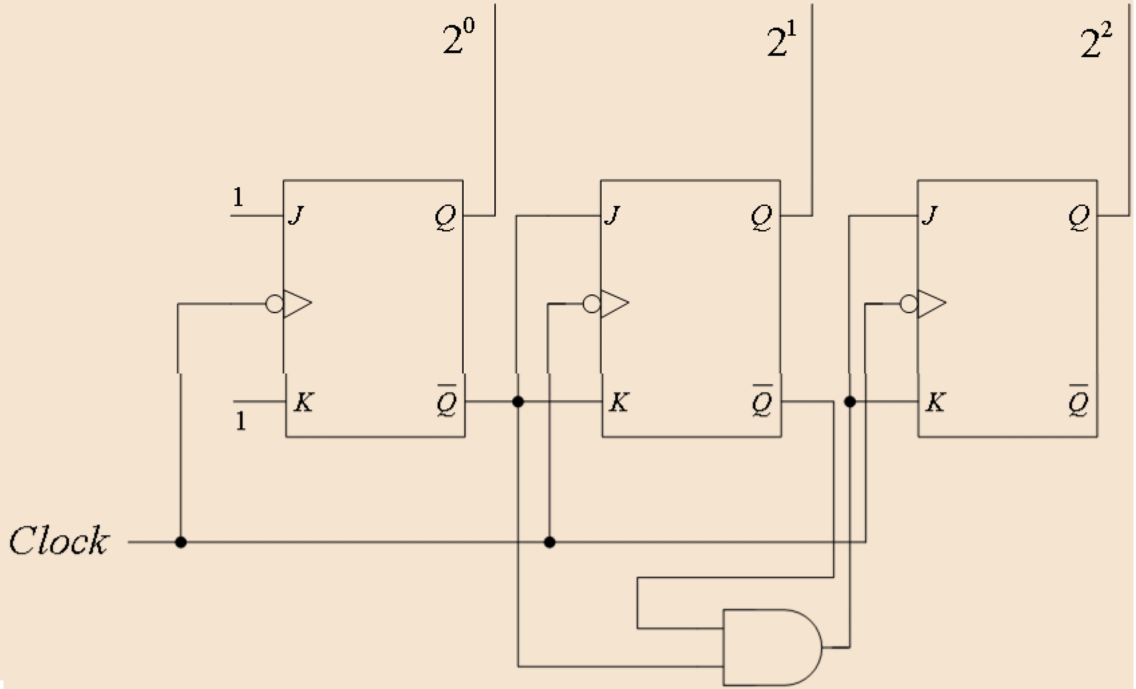
CLOCK PULSE	UP	Q_2	Q_1	Q_0	DOWN
0		0	0	0	
1		0	0	1	
2		0	1	0	
3		0	1	1	
4		1	0	0	
5		1	0	1	
6		1	1	0	
7		1	1	1	

Synchronous Up-Down Counter

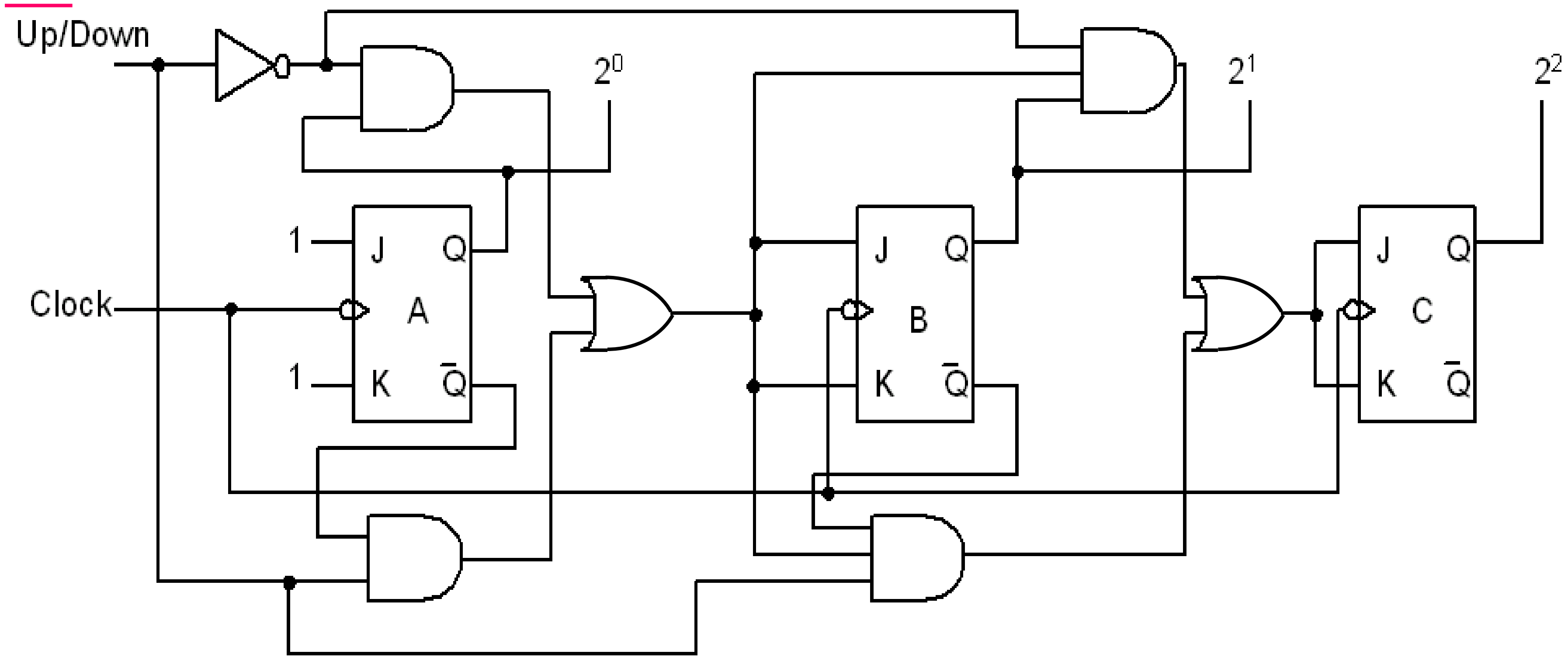
Mod-8 UP Counter



Mod-8 DOWN Counter



Synchronous Up-Down Counter



$\overline{\text{Up}}$/Down
Clock
1
1
J
Q
A
K
$\bar{Q}$
2^0
J
Q
B
K
$\bar{Q}$
2^1
J
Q
C
K
$\bar{Q}$
2^2