CHW 261: Logic Design

Instructors:
Prof. Hala Zayed http://www.bu.edu.eg/staff/halazayed14
Dr. Ahmed Shalaby http://bu.edu.eg/staff/ahmedshalaby14#

Digital Fundamentals

CHAPTER 8
Counters

Counters

Counting in Binary

As you know, the binary count sequence follows a familiar pattern of 0's and 1's.

LSB changes on every number.

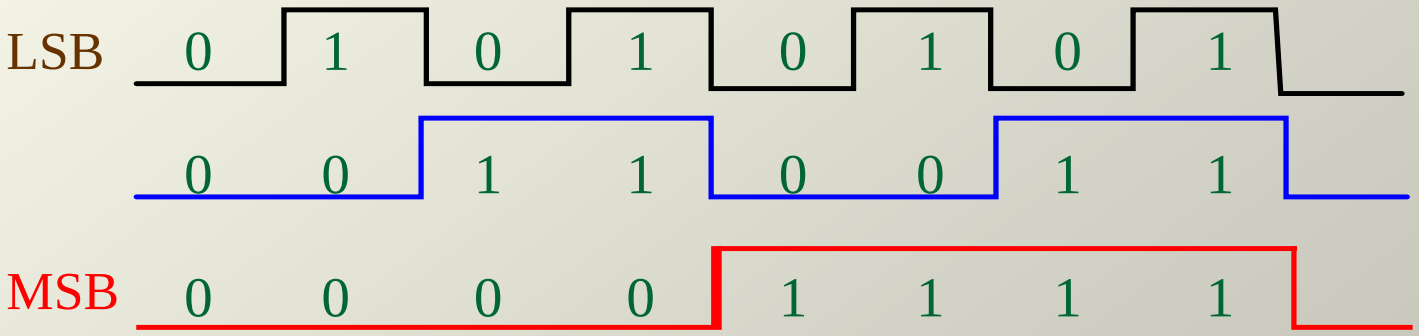
The next bit changes on every other number.

The next bit changes on every fourth number.

Counters

Counting in Binary

A counter can form the same pattern of 0's and 1's with logic levels. The first stage in the counter represents the least significant bit – notice that these waveforms follow the same pattern as counting in binary.



Asynchronous binary counter

In an asynchronous counter, the clock is applied only to the first stage.

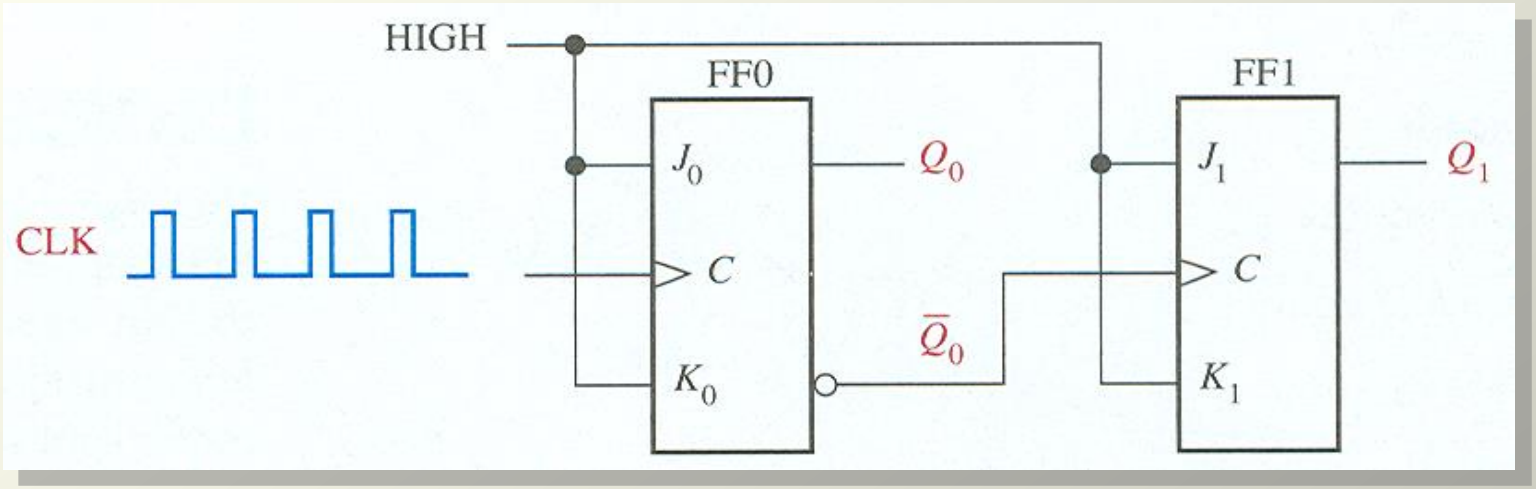
Subsequent stages derive the clock from the previous stage.

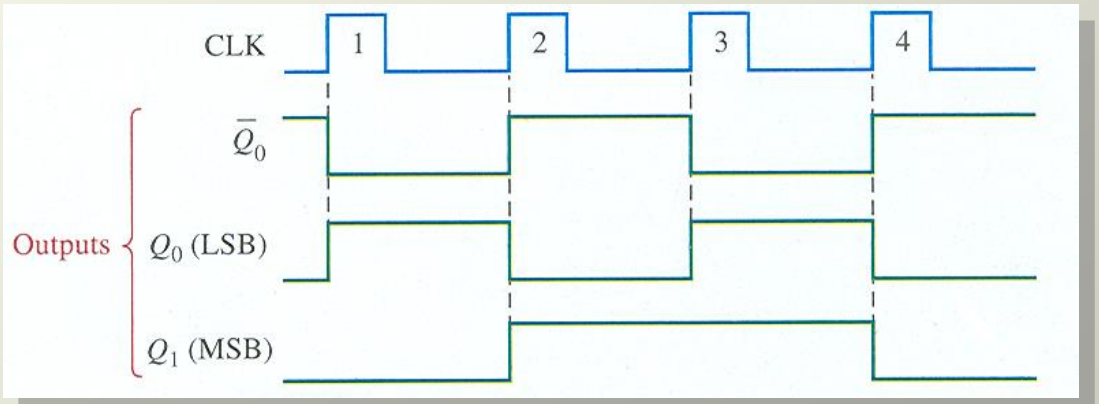
It uses J-K flip-flops in the toggle mode.

Notice that the Q0 output is triggered on the leading edge of the clock signal. The following stage is triggered from $\overline{Q0}$.

Asynchronous binary counter

2-bit Asynchronous binary counter

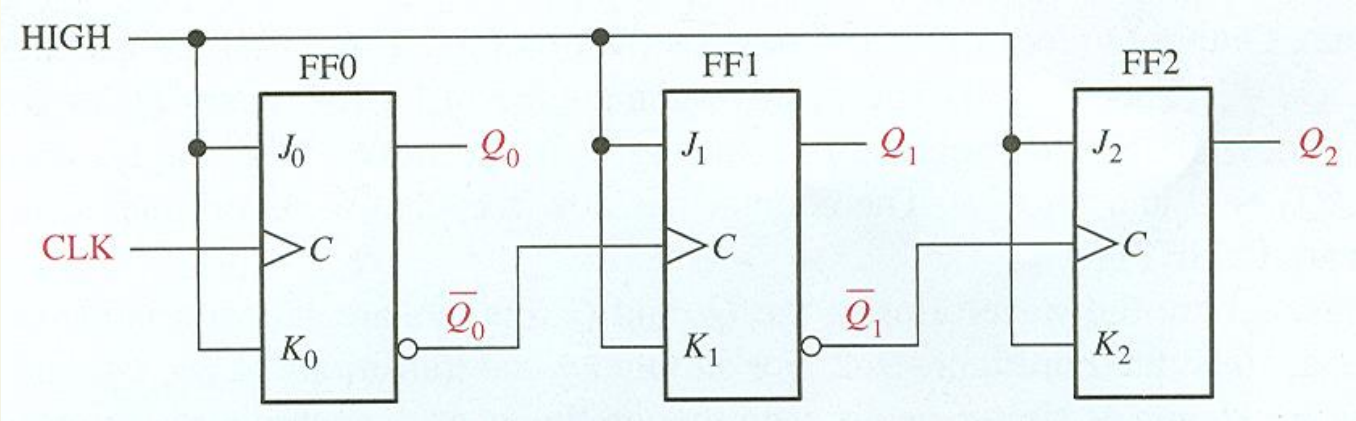


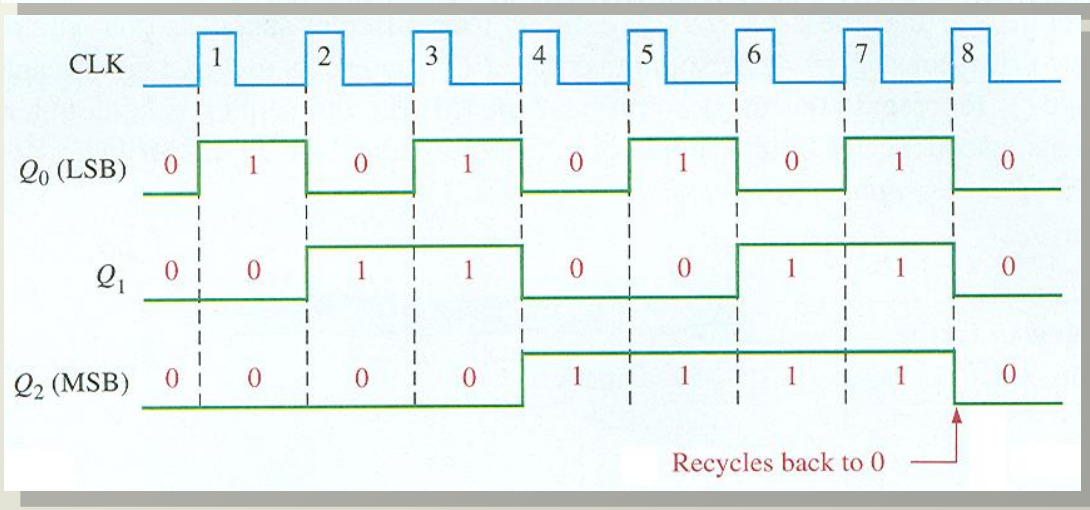


CLOCK PULSE	Q_1	Q_0
Initially	0	0
1	0	1
2	1	0
3	1	1
4 (recycles)	0	0

Asynchronous binary counter

3-bit Asynchronous binary counter

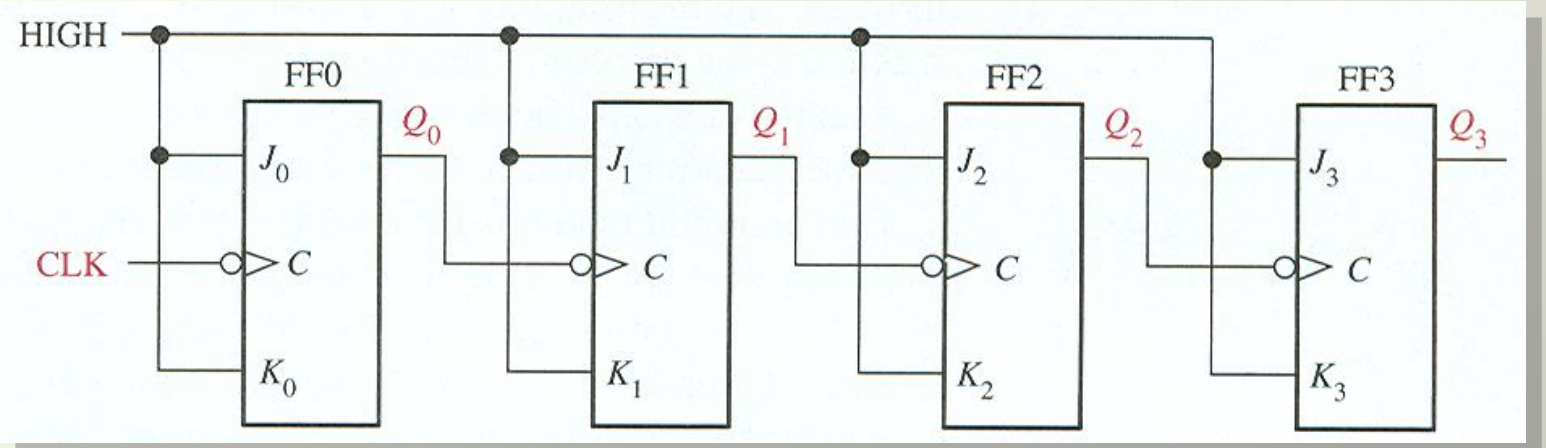


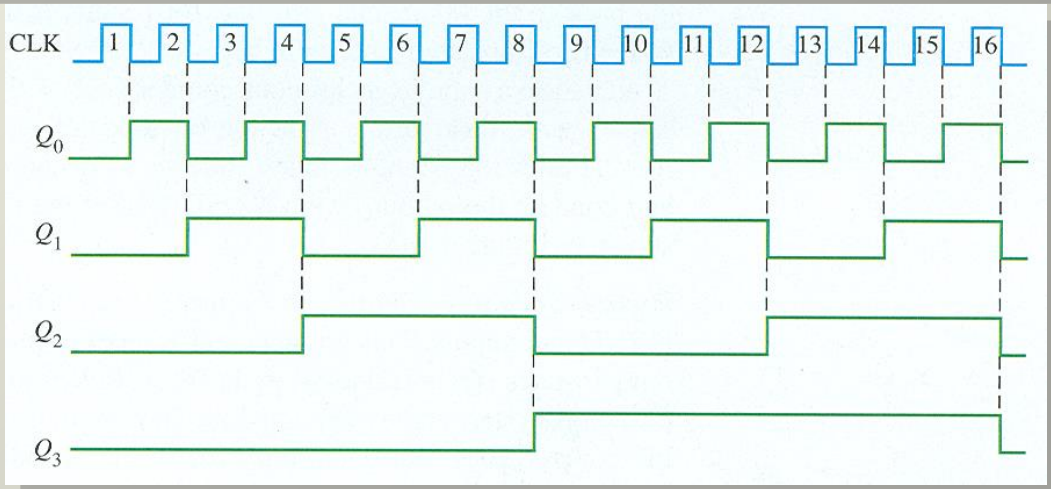


CLOCK PULSE	Q_2	Q_1	Q_0
Initially	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8 (recycles)	0	0	0

Asynchronous binary counter

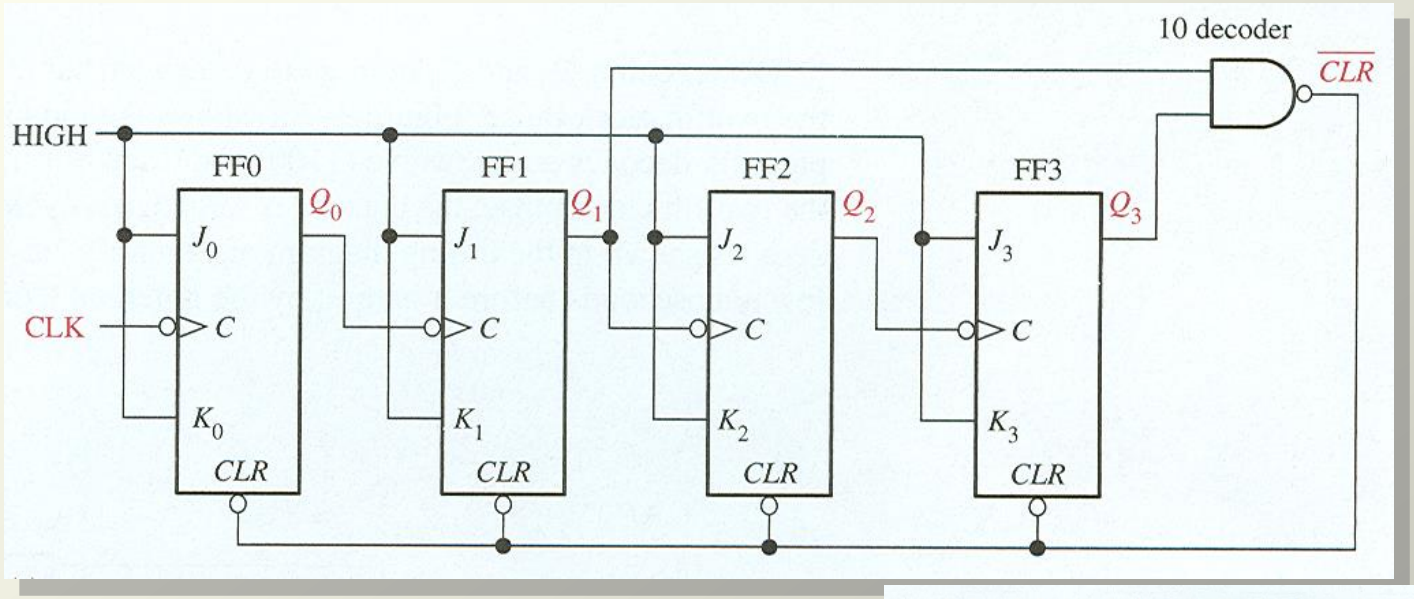
4-bit Asynchronous binary counter

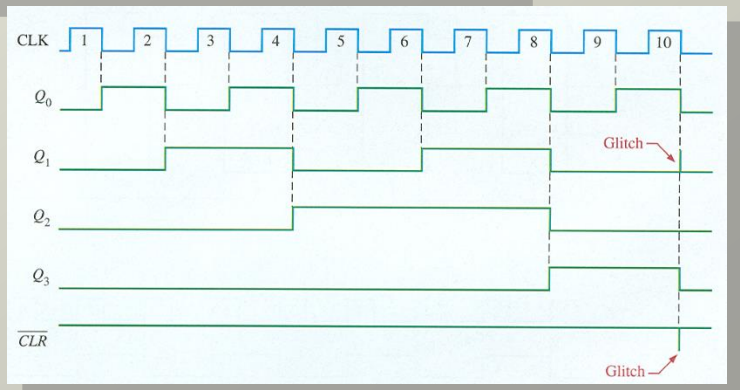




Asynchronous binary counter

Asynchronous decade counter



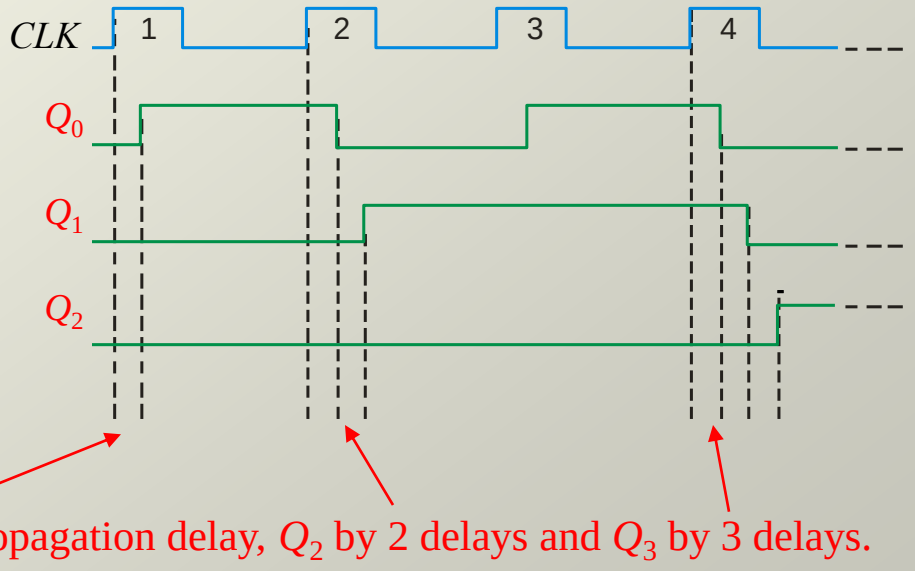


Asynchronous binary counter

Propagation Delay

Asynchronous counters are sometimes called **ripple** counters, because the stages do not all change together. For certain applications requiring high clock rates, this is a major disadvantage.

Notice how delays are cumulative as each stage in a counter is clocked later than the previous stage.



Synchronous binary counter

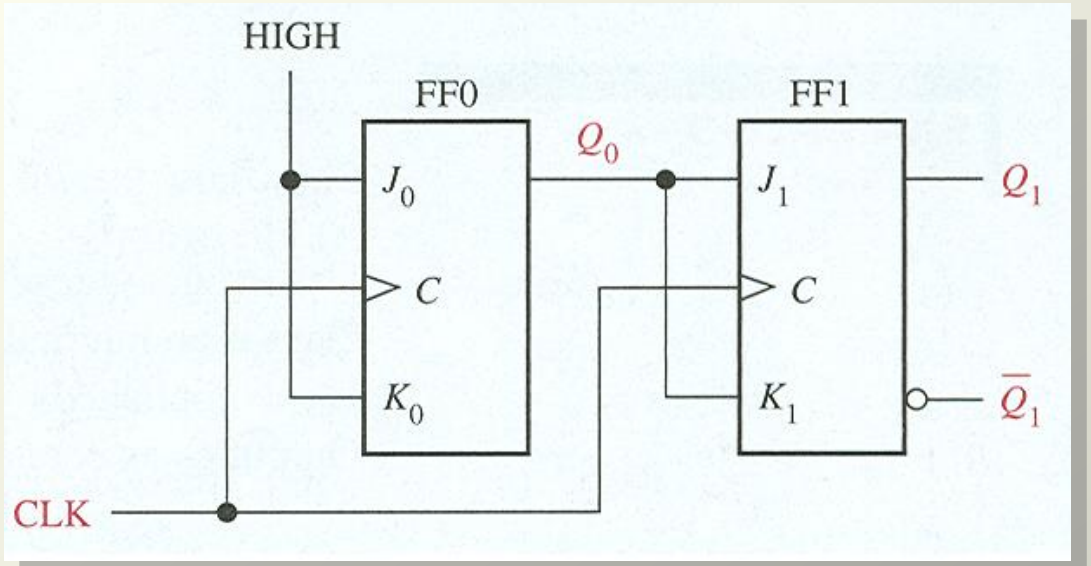
Synchronous Counters

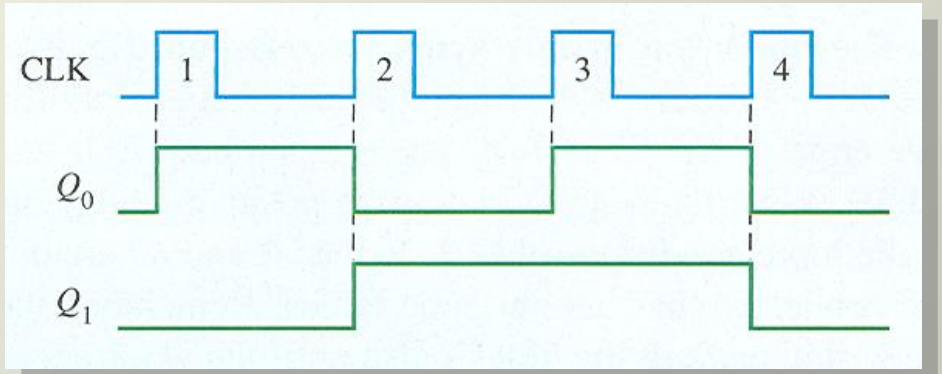
In a **synchronous counter** all flip-flops are clocked together with a common clock pulse.

Synchronous counters overcome the disadvantage of accumulated propagation delays, but generally they require more circuitry to control states changes.

Synchronous binary counter

2-bit Synchronous binary counter

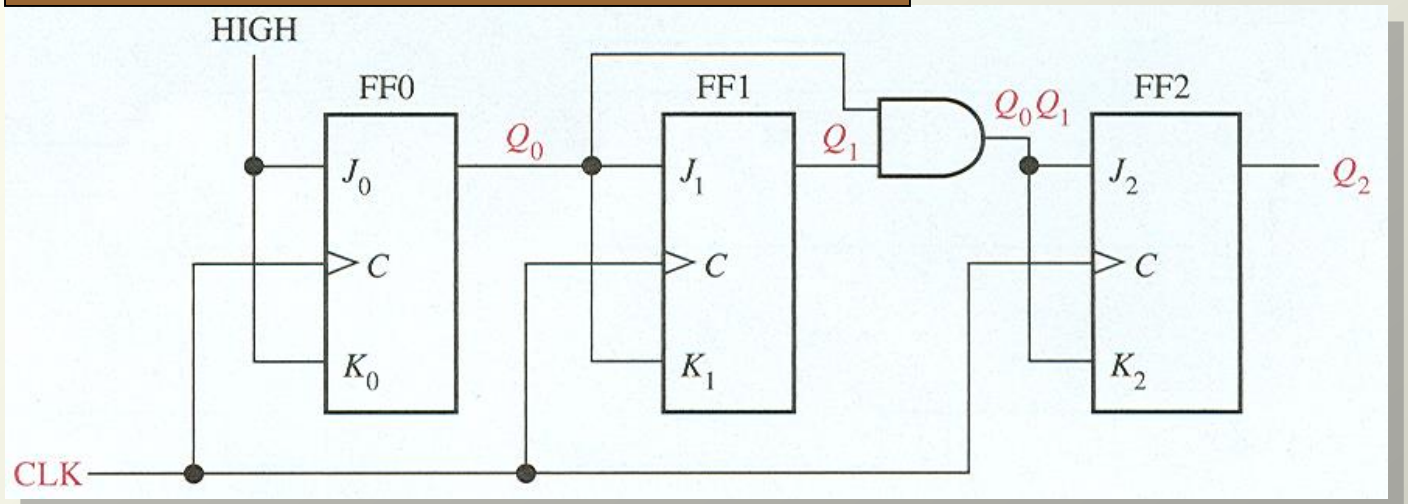


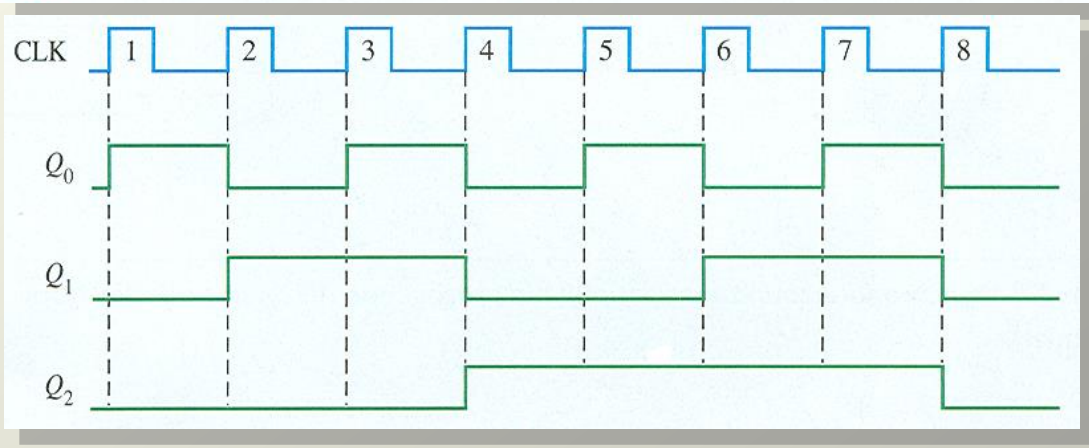


Inputs			Outputs		
J	K	CLK	Q	$\overline{Q}$	Comments
0	0	↑	Q_0	$\overline{Q}_0$	No change
0	1	↑	0	1	RESET
1	0	↑	1	0	SET
1	1	↑	$\overline{Q}_0$	Q_0	Toggle

Synchronous binary counter

3-bit Synchronous binary counter

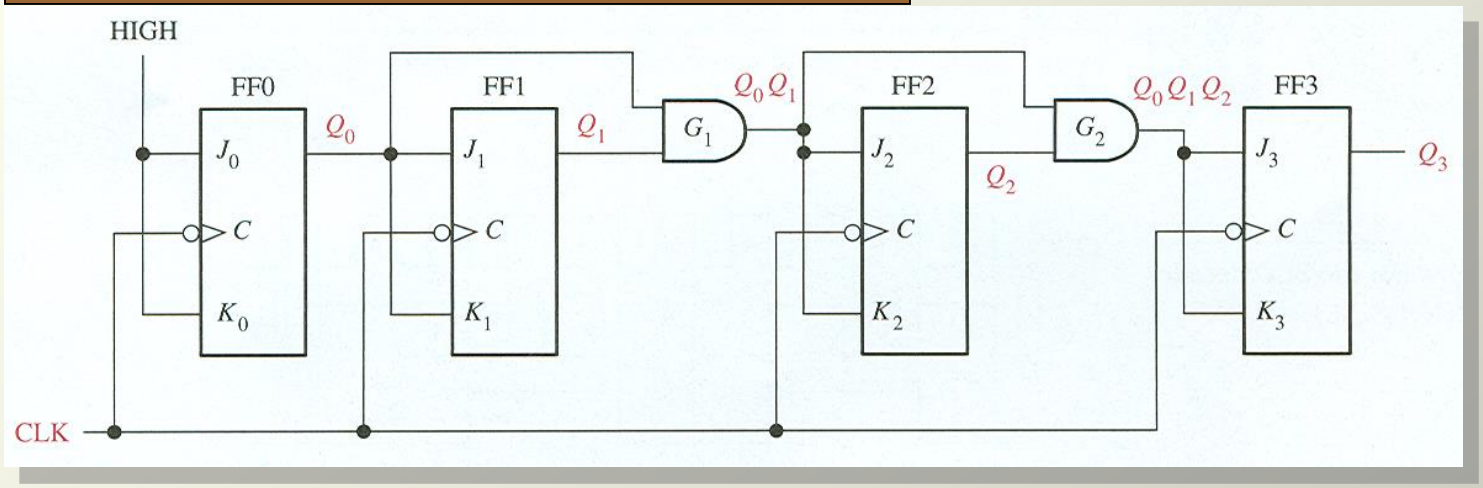


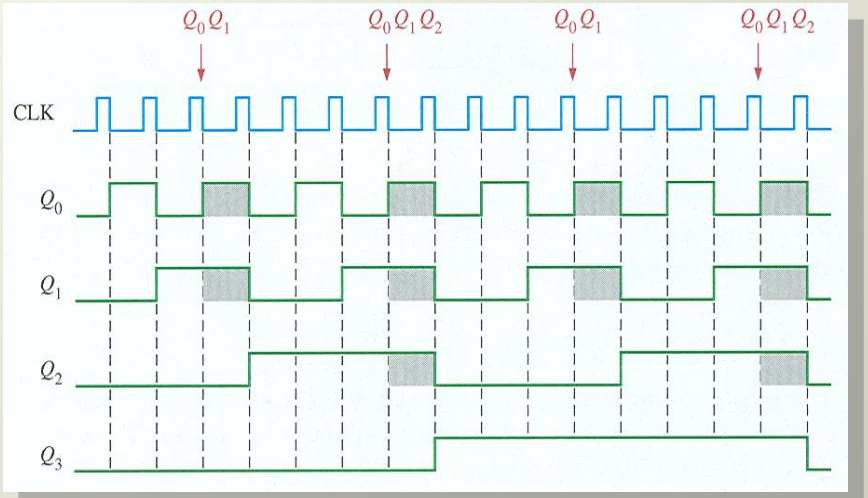


CLOCK PULSE	Q_2	Q_1	Q_0
Initially	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8 (recycles)	0	0	0

Synchronous binary counter

4-bit Synchronous binary counter

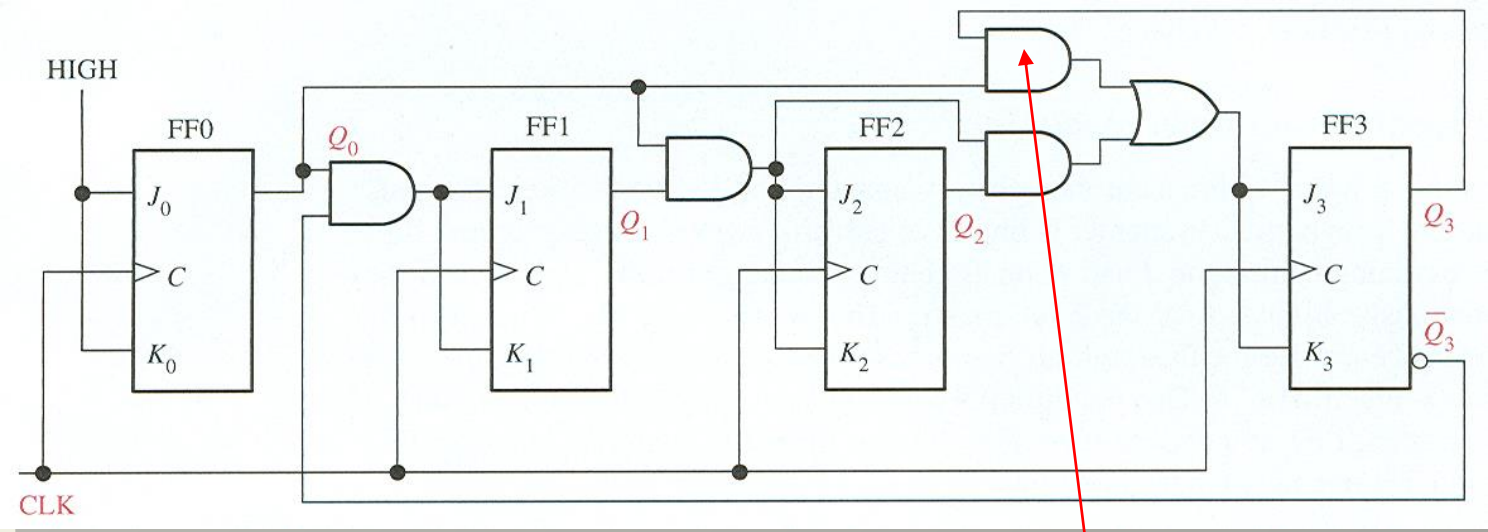


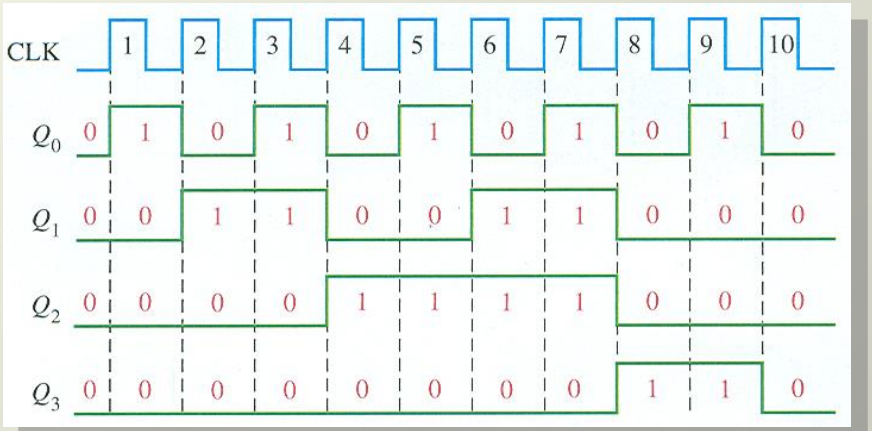


The 4-bit binary counter has one more AND gate than the 3-bit counter just described. The shaded areas show where the AND gate outputs are HIGH causing the next FF to toggle.

Synchronous binary counter

Synchronous decade counter





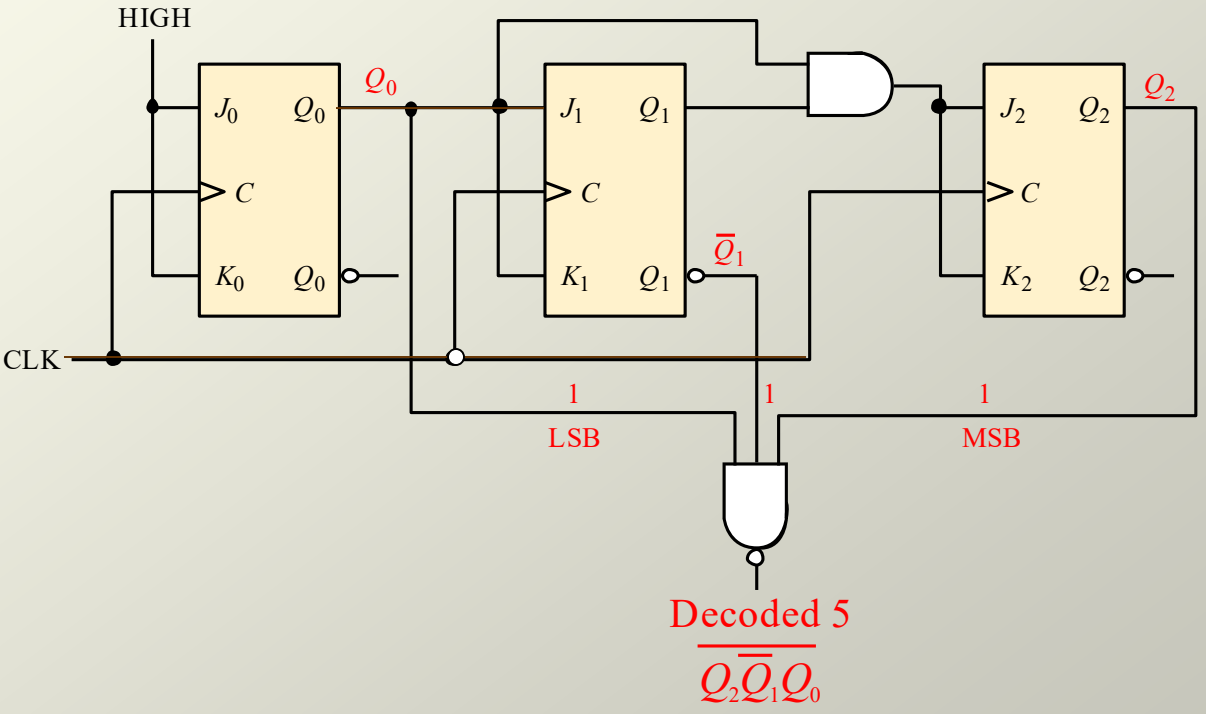
This gate detects 1001, and causes FF3 to toggle on the next clock pulse. FF0 toggles on every clock pulse. Thus, the count starts over at 0000.

Synchronous binary counter

Counter Decoding

Example

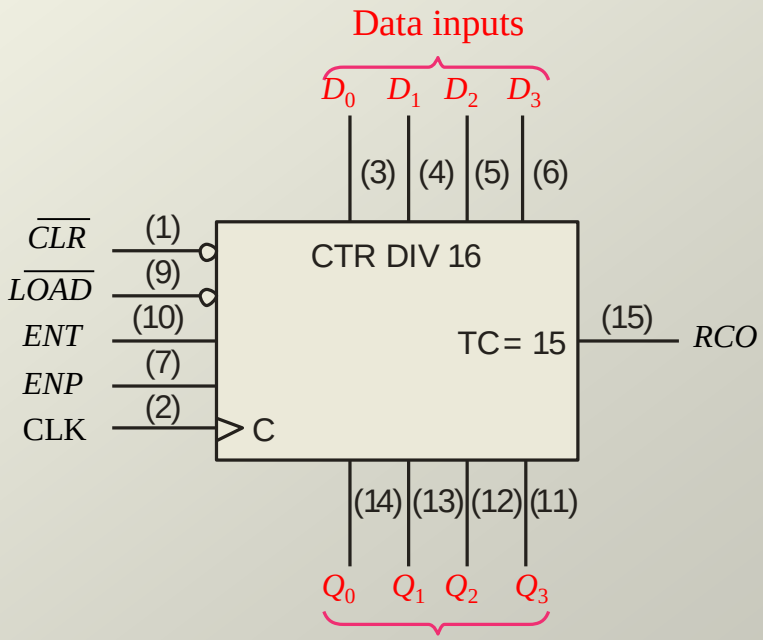
Show how to decode state 5 with an active LOW output.



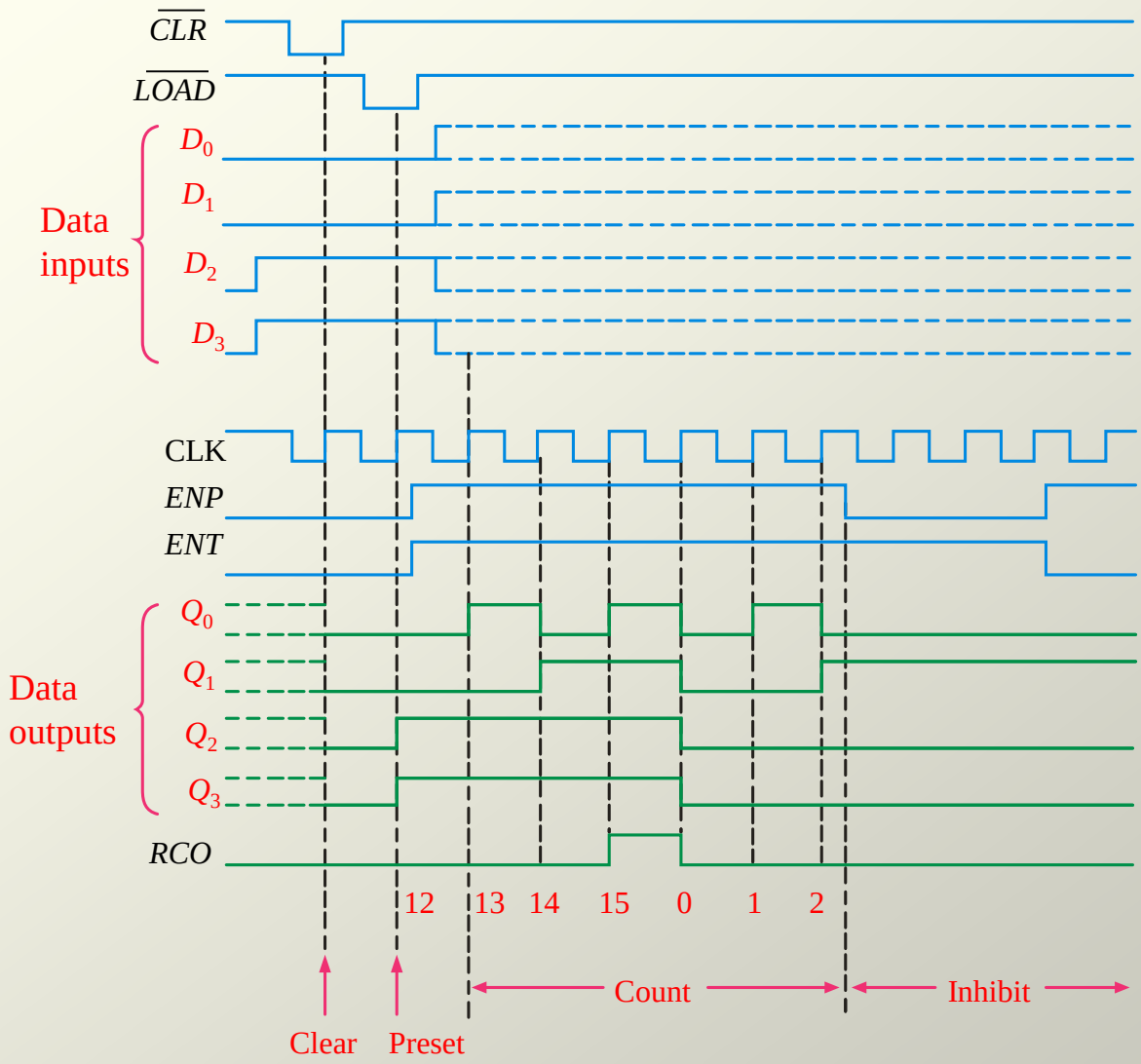
Synchronous binary counter

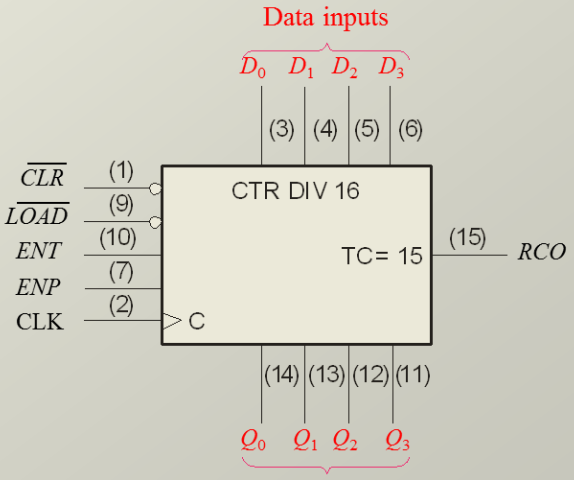
A 4-bit Synchronous Binary Counter

The 74LS163 is a 4-bit IC synchronous counter with additional features over a basic counter. It has parallel load, a $\overline{CLR}$ input, two chip enables, and a ripple count output that signals when the count has reached the terminal count.



A 4-bit Synchronous Binary Counter

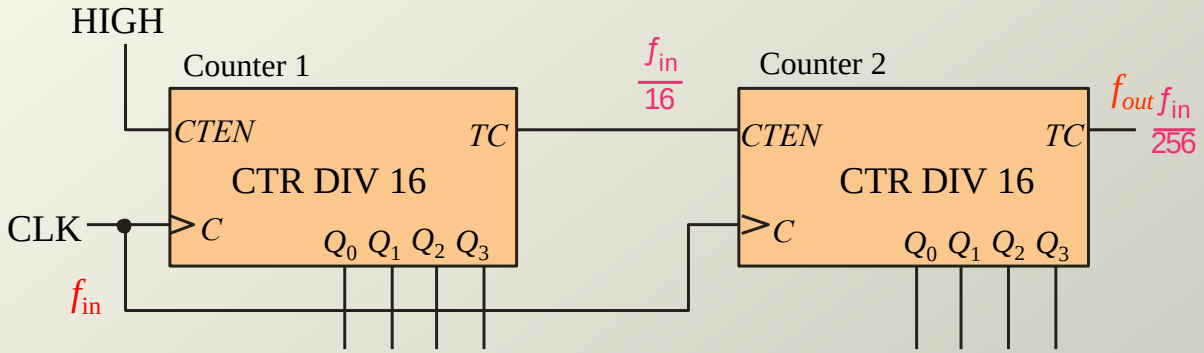




Synchronous binary counter

Cascaded counters

Cascading is a method of achieving higher-modulus counters. For synchronous IC counters, the next counter is enabled only when the terminal count of the previous stage is reached.



Example

a) What is the modulus of the cascaded DIV 16 counters?

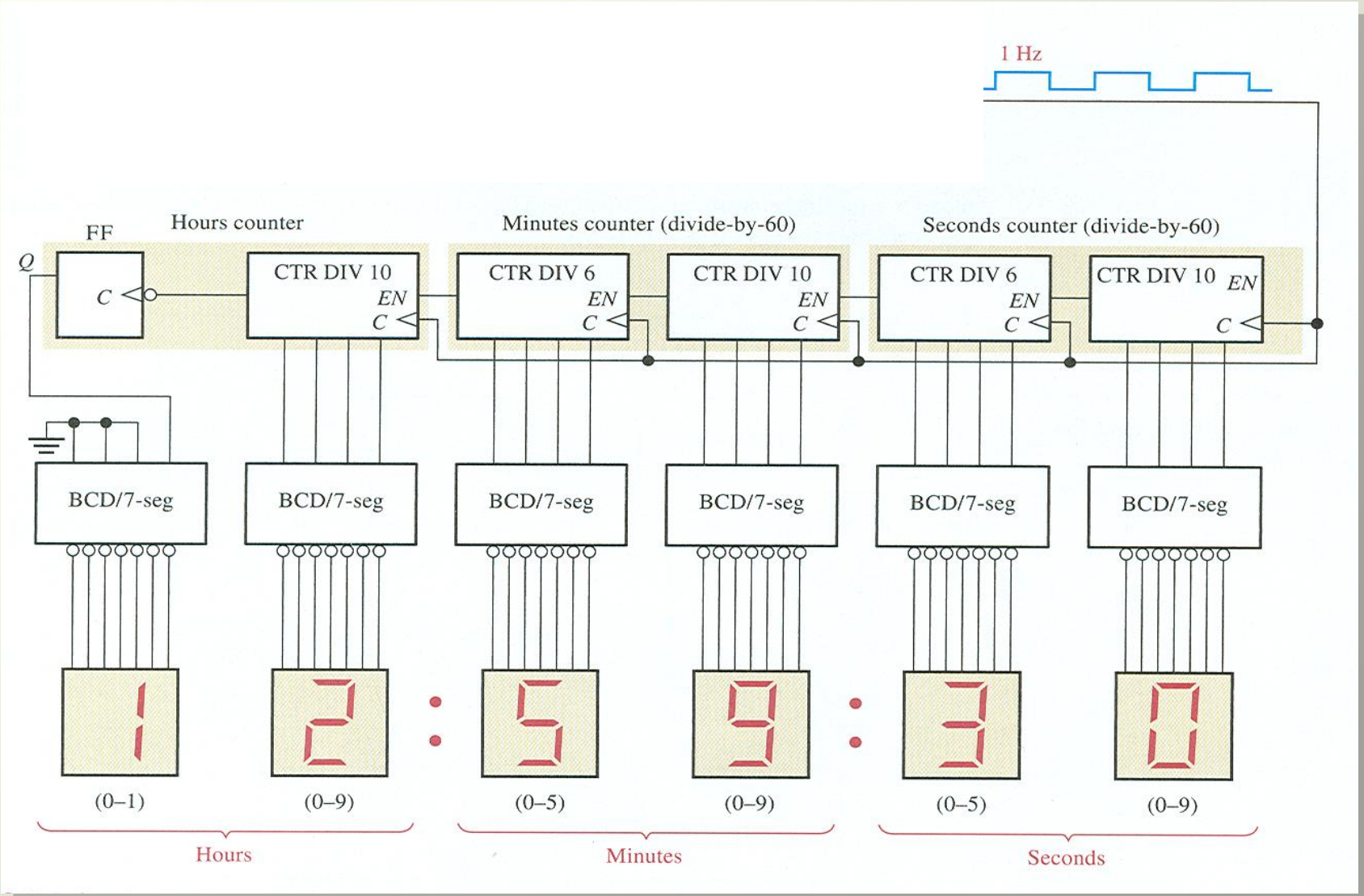
b) If f_{in} =100 kHz, what is f_{out}?

Solution

a) Each counter divides the frequency by 16. Thus the modulus is $16^2 = 256$.

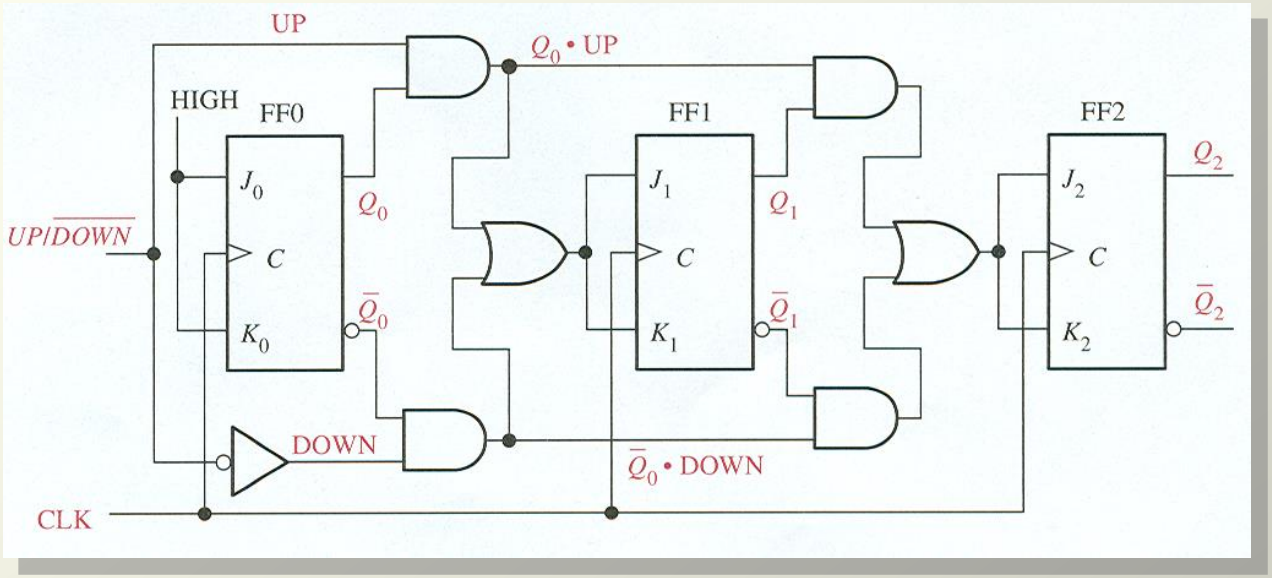
b) The output frequency is 100 kHz/256 = 391 Hz

Counter Applications – Digital Clock



Synchronous binary counter

Up/Down Synchronous Counters



An up/down counter is capable of progressing in either direction depending on a control input.

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	

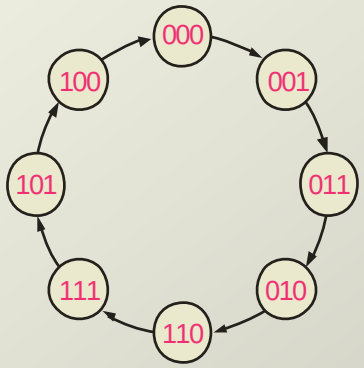
Synchronous binary counter

Synchronous Counter Design

Most requirements for synchronous counters can be met with available ICs. In cases where a special sequence is needed, you can apply a step-by-step design process.

Start with the desired sequence and draw a state diagram and next-state table. The gray code sequence from the text is illustrated:

State diagram:



Next state table:

Present State			Next State		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	0	1	0
0	1	0	1	1	0
1	1	0	1	1	1
1	1	1	1	0	1
1	0	1	1	0	0
1	0	0	0	0	0

Synchronous binary counter

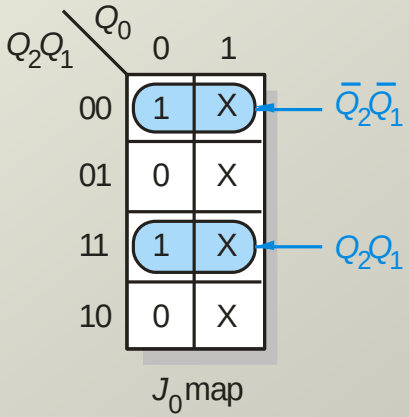
Synchronous Counter Design

The J-K transition table lists all combinations of present output (Q_N) and next output (Q_{N+1}) on the left. The inputs that produce that transition are listed on the right.

Output Transitions Q_N	Q_{N+1}	Flip-Flop Inputs J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Each time a flip-flop is clocked, the J and K inputs required for that transition are mapped onto a K-map.

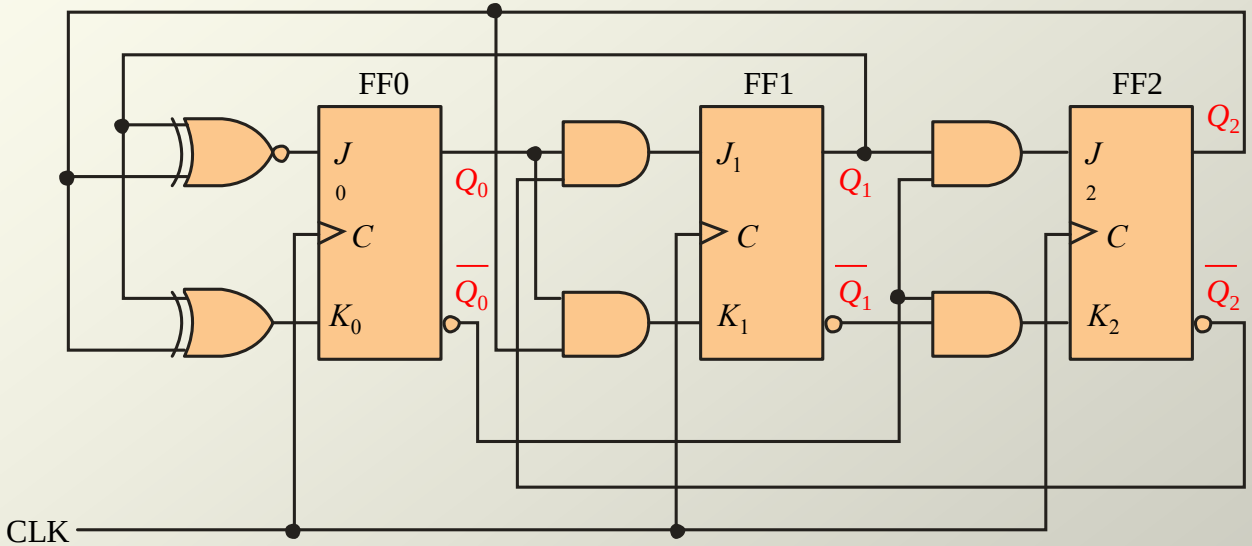
An example of the J_0 map is:



Present State Q_2	Q_1	Q_0	Next State Q_2	Q_1	Q_0
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	0	1	0
0	1	0	1	1	0
1	1	0	1	1	1
1	1	1	1	0	1
1	0	1	1	0	0
1	0	0	0	0	0

Synchronous binary counter

Synchronous Counter Design



The logic for each input is read and the circuit is constructed. The slide shows the circuit for the gray code counter…