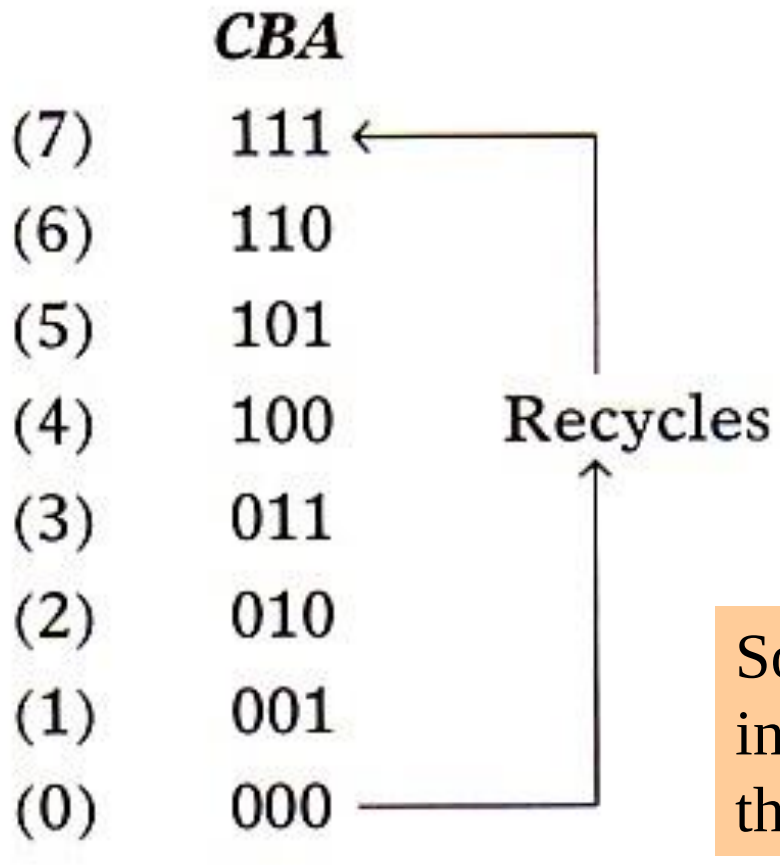
# XMUT 202
# Digital Electronics

A/Prof. Pawel Dmochowski

School of Engineering and Computer Science
Victoria University of Wellington

Victoria
UNIVERSITY OF WELLINGTON
Te Whare Wānanga
o te Ūpoko o te Ika a Māui
CAPITAL CITY UNIVERSITY

# Asynchronous Down Counter




A: Toggles as normal

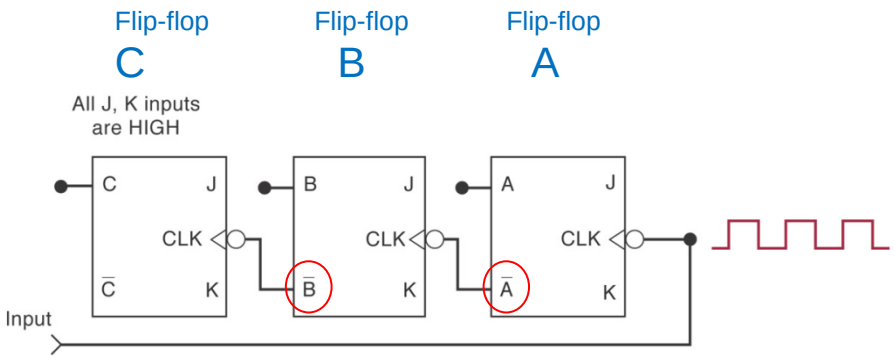
B: Toggles when A goes LO to HI

C: Toggles when B goes LO to HI

Solution: Drive each clock input from the inverted input of the preceding FF

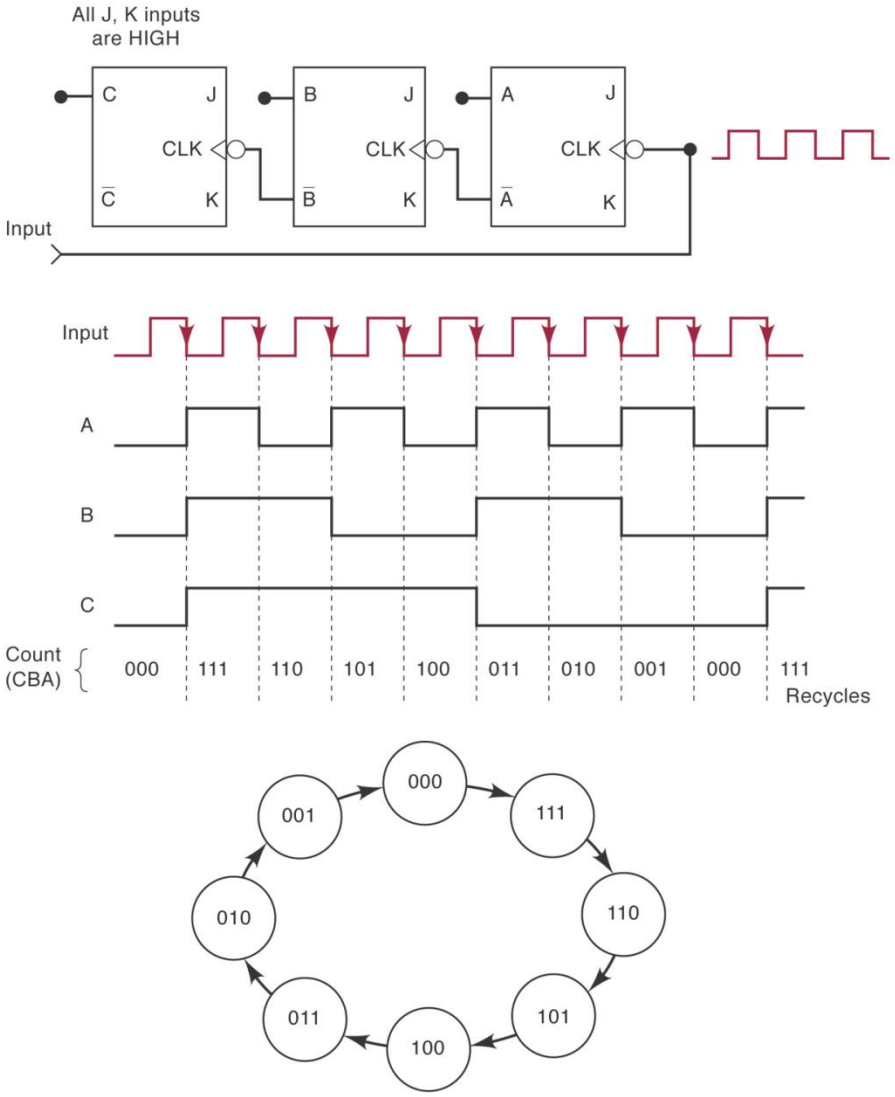
# Asynchronous Down Counter

MOD 8 down counter



# Asynchronous Down Counter

MOD 8 down counter



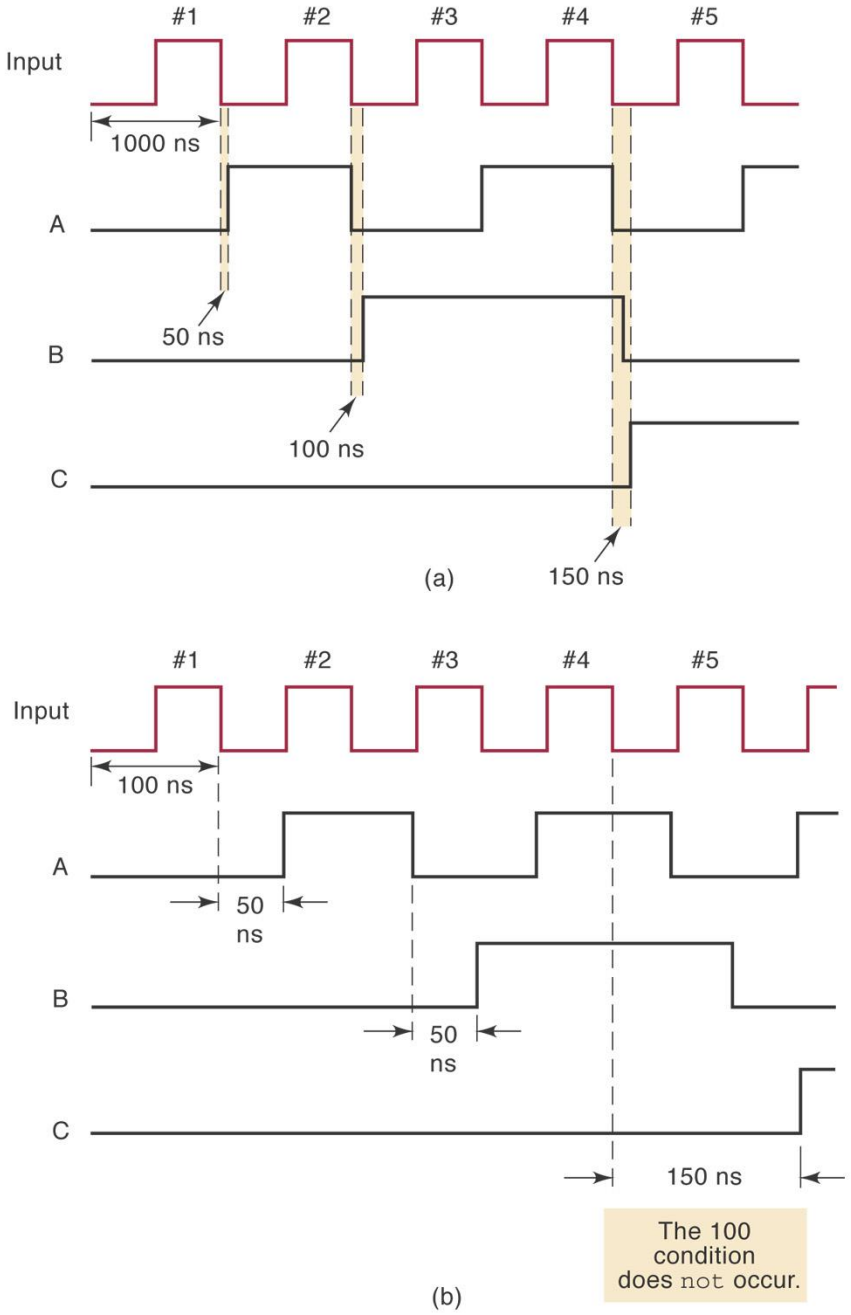
# The Problem with Ripple Counters: Propagation Delay

For proper operation:

$$T_{\text{clock}} \geq N \text{ x } t_{pd}$$

$$F_{max} = 1/(N \text{ x } t_{pd})$$

where N = number of FFs.



Example: Four bit ripple counter constructed from a 74LS112 J-K flip flop.

$t_{PLH}$ = 16 ns and $t_{PHL}$ = 24 ns

What is highest frequency that can be used ?

Example: Four bit ripple counter constructed from a 74LS112 J-K FF.

$t_{PLH}$ = 16 ns and $t_{PHL}$ = 24 ns

What is highest frequency that can be used ?

$f_{max}$ = 1/(4 x 24 ns) = 10.4 MHz

This frequency limitation inhibits the use of asynchronous counters at moderately high frequencies

Example: What is the highest frequency that a ripple counter with a MOD number 60 constructed from a 74LS112 J-K flip flop can operate at?

$t_{PLH}$ = 16 ns and $t_{PHL}$ = 24 ns

Example: What is the highest frequency that a ripple counter with a MOD number 60 constructed from a 74LS112 J-K flip flop can operate at?

$t_{PLH}$ = 16 ns and $t_{PHL}$ = 24 ns

$f_{max}$ = 1/(6 x 24 ns) = 6.9 MHz

# Synchronous (Parallel) Counters

- Solution to the previous problem - All FFs are triggered by CPs simultaneously

# Synchronous (Parallel) Counters

- Solution to the previous problem - All FFs are triggered by CPs simultaneously
- Each FF has J and K inputs connected so they are HIGH only when the outputs of all lower-order FFs are HIGH.

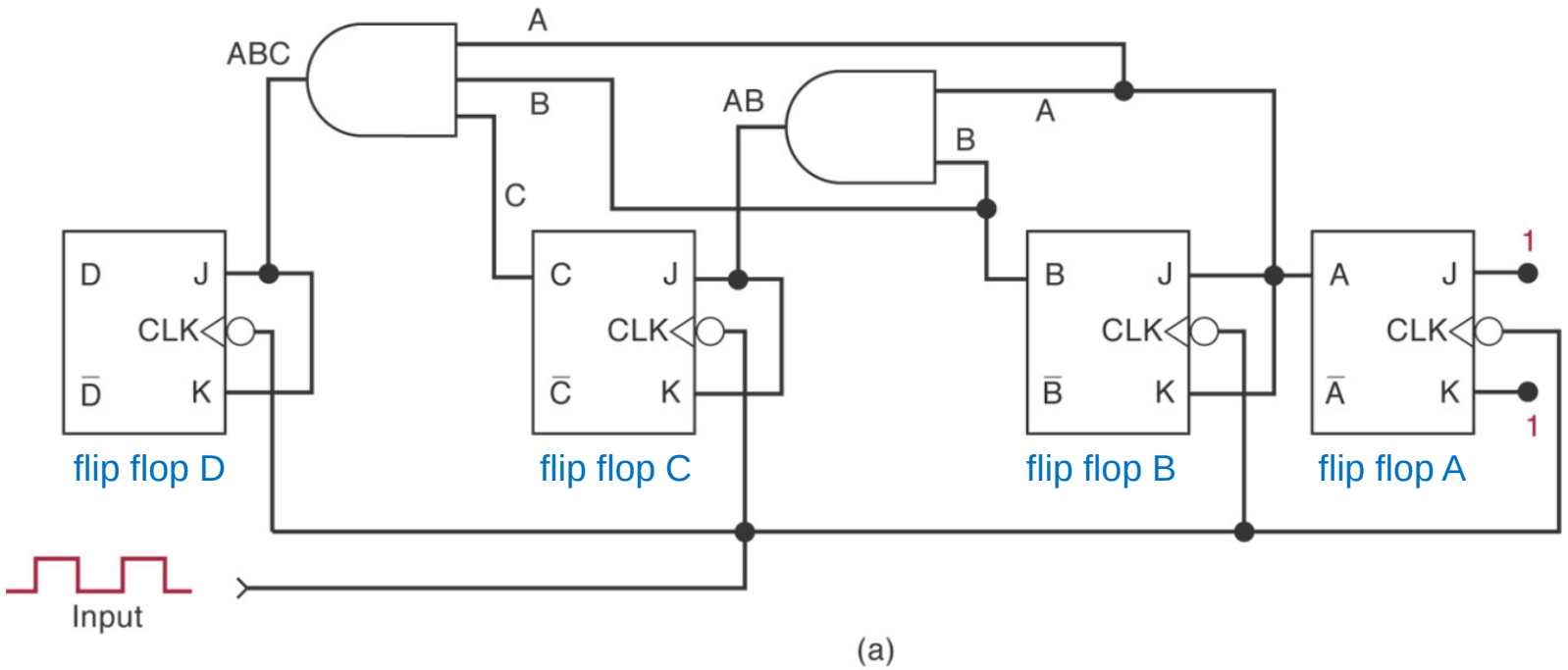
# Synchronous (Parallel) Counters

- Solution to the previous problem - All FFs are triggered by CPs simultaneously
- Each FF has J and K inputs connected so they are HIGH only when the outputs of all lower-order FFs are HIGH.
- The total propagation delay will be the same for any number of FFs.

# Synchronous (Parallel) Counters

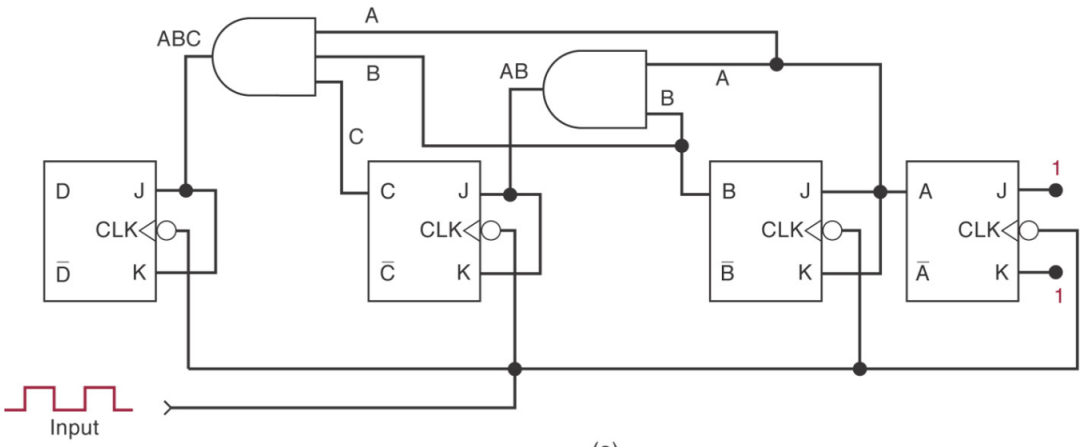
- Solution to the previous problem - All FFs are triggered by CPs simultaneously
- Each FF has J and K inputs connected so they are HIGH only when the outputs of all lower-order FFs are HIGH.
- The total propagation delay will be the same for any number of FFs.
- Synchronous counters can operate at much higher frequencies than asynchronous counters.

# Synchronous MOD 16 counter




(a)

# Synchronous MOD 16 counter




(a)

Truth table

| Count | D | C | B | A |
|---|---|---|---|---|
| 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 |
| 10 | 1 | 0 | 1 | 0 |
| 11 | 1 | 0 | 1 | 1 |
| 12 | 1 | 1 | 0 | 0 |
| 13 | 1 | 1 | 0 | 1 |
| 14 | 1 | 1 | 1 | 0 |
| 15 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 |
| . | . | . | . | . |
| . | . | . | . | . |
| . | . | etc. | . | . |

(b)

# Synchronous Down and Up Counters

- The synchronous counter can be converted to a down counter by using the inverted FF outputs to drive the JK inputs.

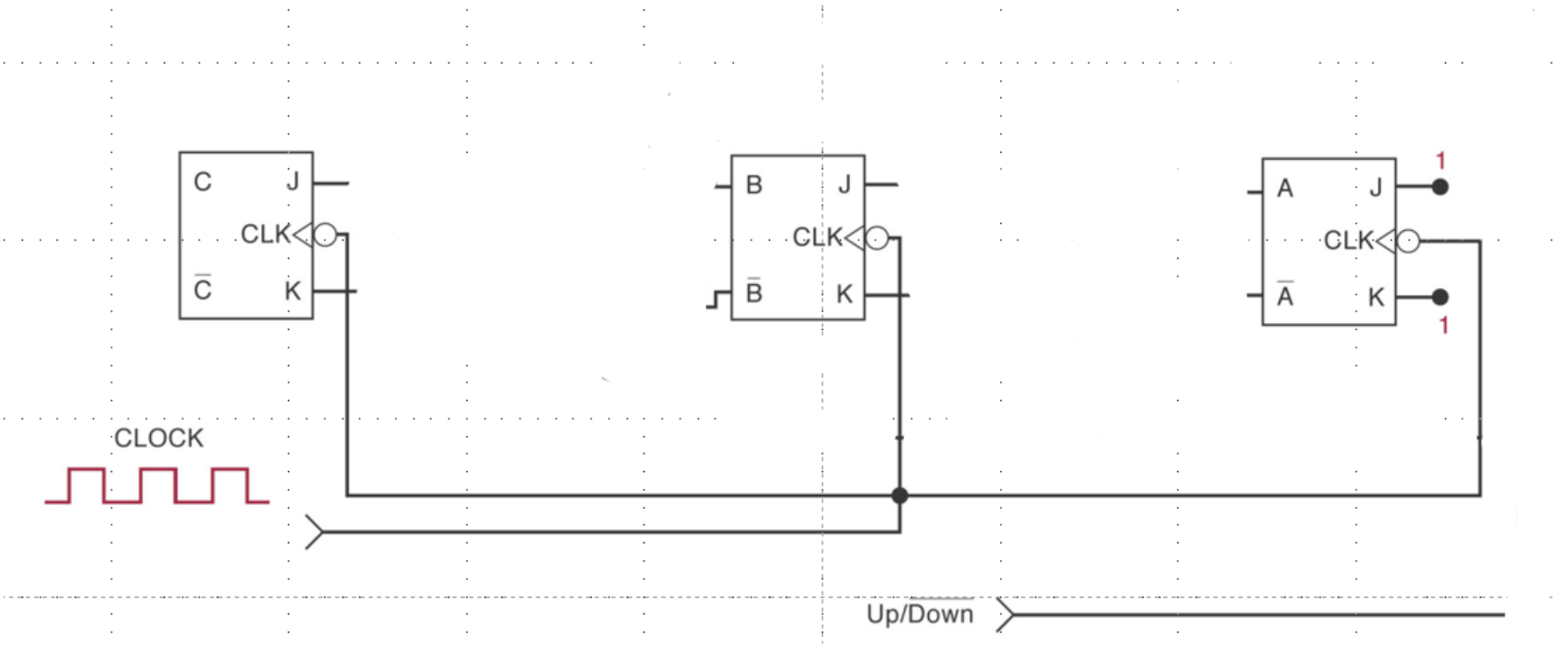
# Synchronous Down and Up Counters

- The synchronous counter can be converted to a down counter by using the inverted FF outputs to drive the JK inputs.
- A synchronous counter can be made an up/down counter by suitable connections.

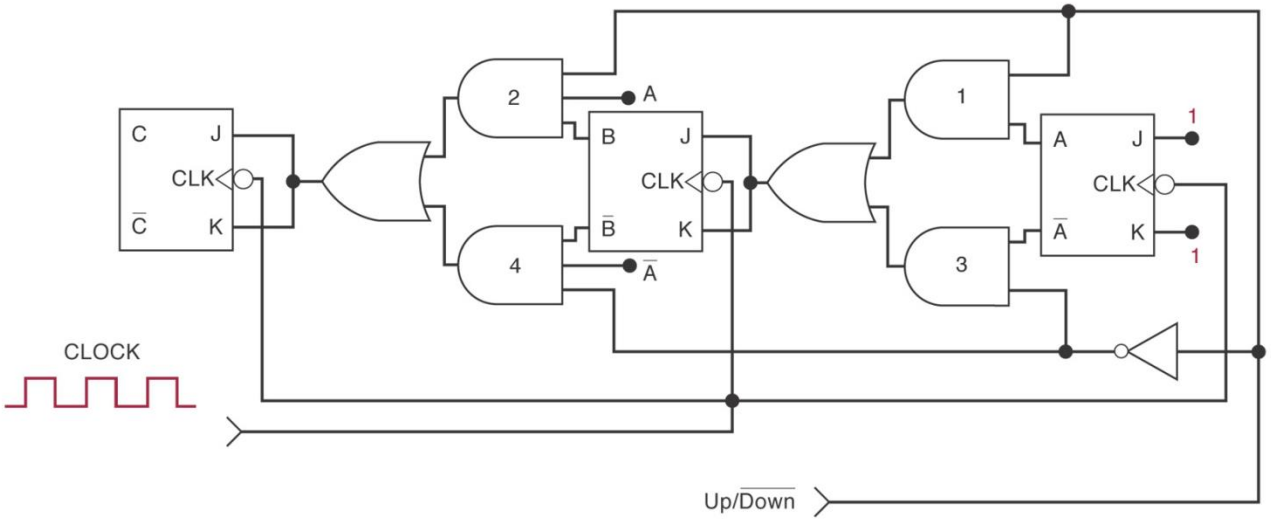
# Synchronous Down and Up Counters

- The synchronous counter can be converted to a down counter by using the inverted FF outputs to drive the JK inputs.
- A synchronous counter can be made an up/down counter by suitable connections.
- Available in IC form with Up/Down control input.

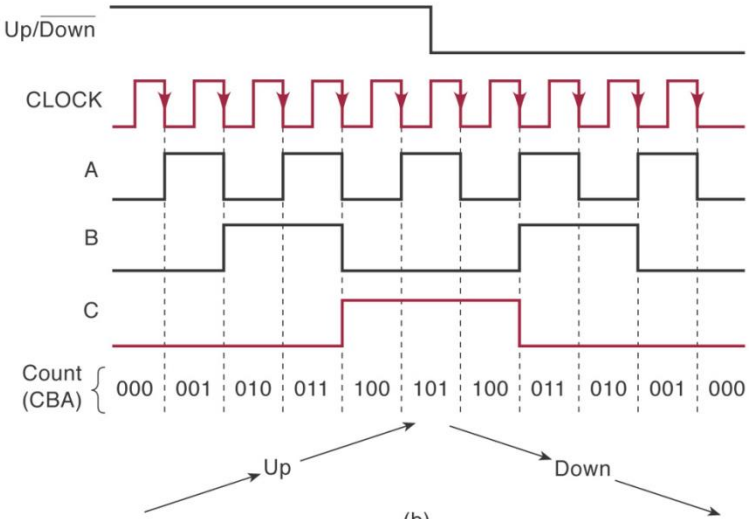
# Synchronous Down and Up Counters



# MOD 8 Up/Down Counter




(a)

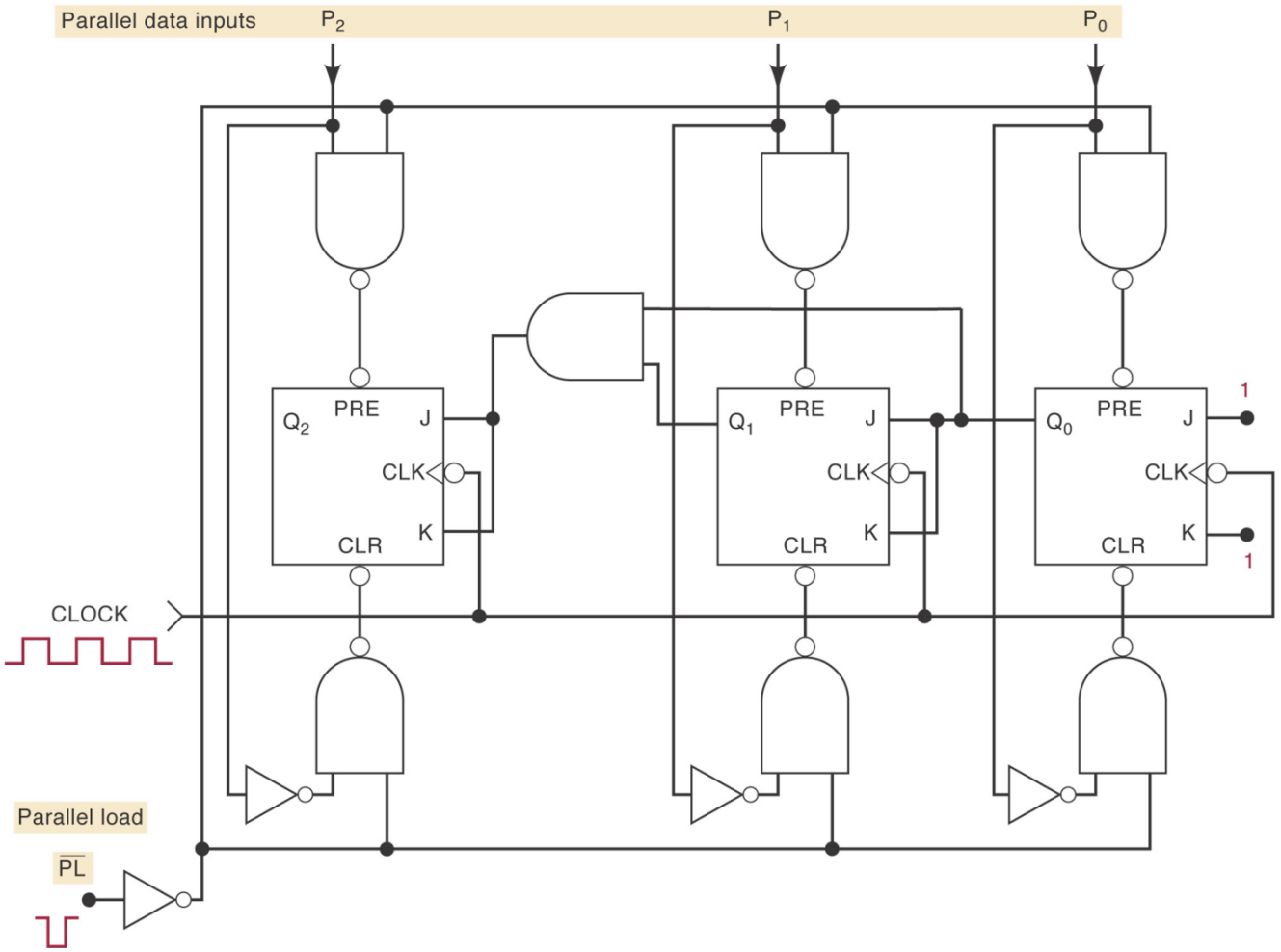



(b)

# Presettable Counters

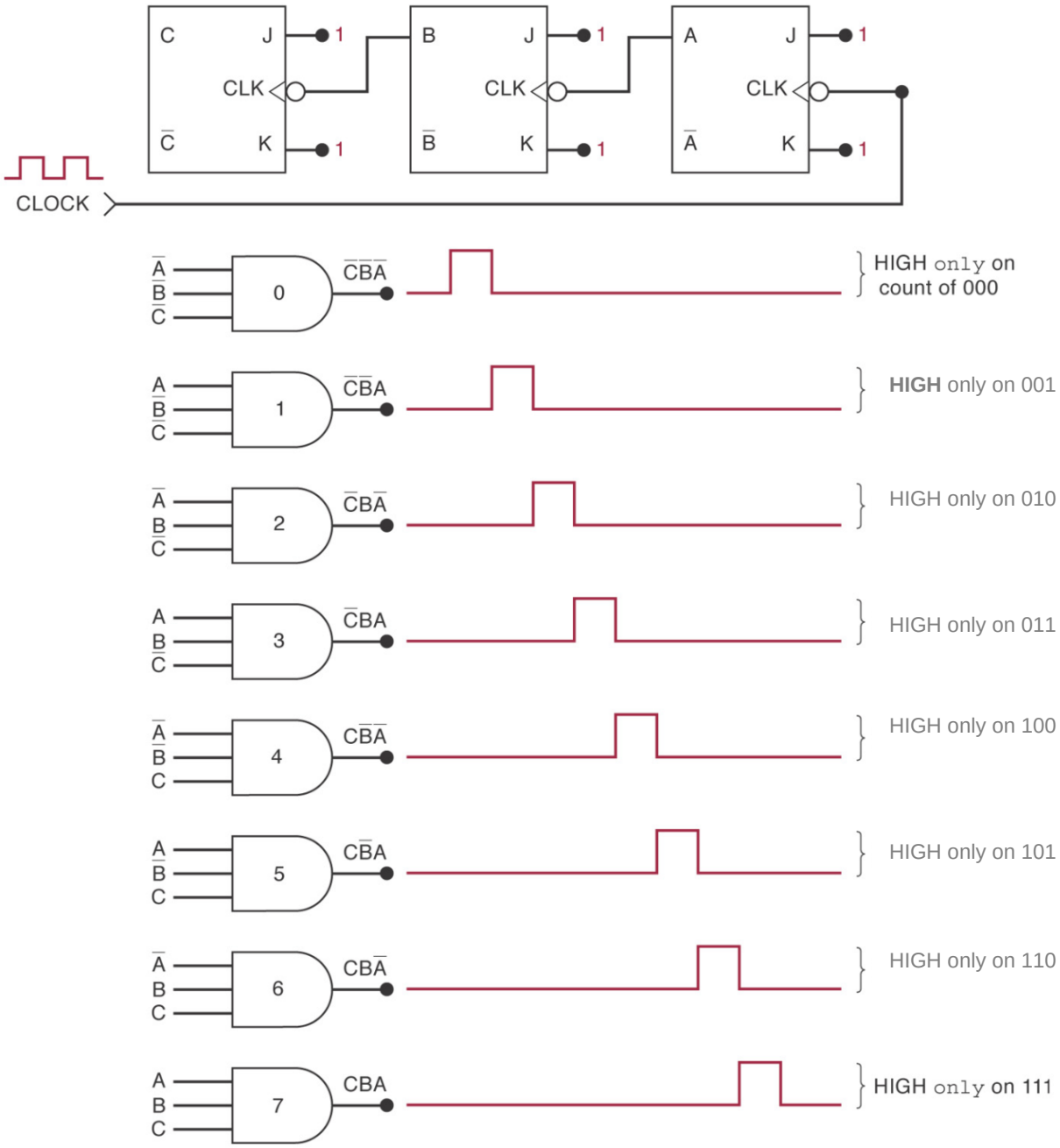
- A presettable counter can be set to any desired starting point either asynchronously or synchronously.
- The preset operation is also called parallel loading the counter.
- The next figure illustrates an asynchronous preset.
- There are several TTL and CMOS devices that provide both synchronous and asynchronous presetting.

# Presettable counter with asynchronous reset.



# Decoding a Counter

- Decoding is the conversion of a binary output to a decimal value.
- The active high decoder shown in the next figure could be used to light an LED representing each decimal number 0 to 7
- Active low decoding is obtained by replacing the AND gates with NAND gates.


C
J
1
CLK
C̄
K
1
B
J
1
CLK
B̄
K
1
A
J
1
CLK
Ā
K
1
CLOCK
Ā
B̄
C̄
0
C̄B̄Ā
HIGH only on count of 000
Ā
B̄
C̄
1
C̄B̄A
HIGH only on 001
Ā
B̄
C̄
2
C̄BĀ
HIGH only on 010
A
B̄
C̄
3
C̄BA
HIGH only on 011
Ā
B̄
C
4
C̄B̄Ā
HIGH only on 100
A
B̄
C
5
C̄B̄A
HIGH only on 101
Ā
B
C
6
CBĀ
HIGH only on 110
A
B
C
7
CBA
HIGH only on 111

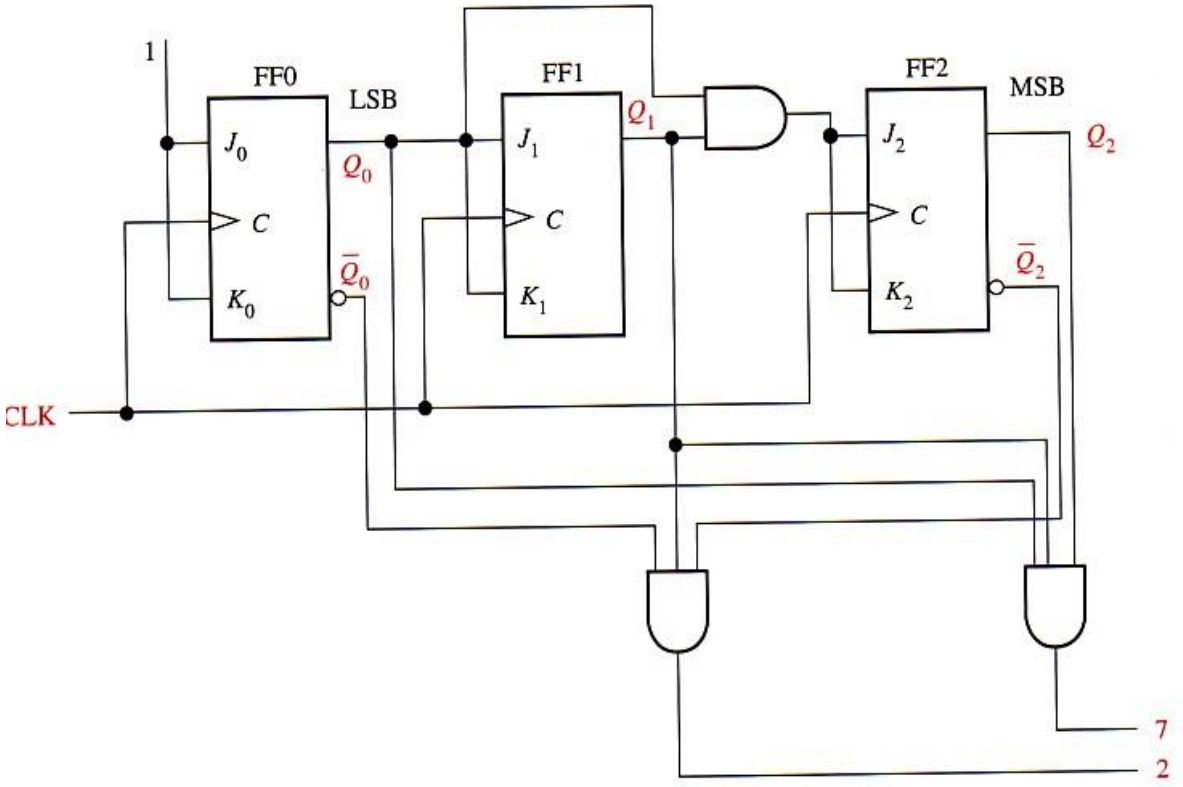
## Problem:

Design a 3 – bit synchronous counter that will decode binary state 2 and binary state 7.

Decode

$2 = 010_2$

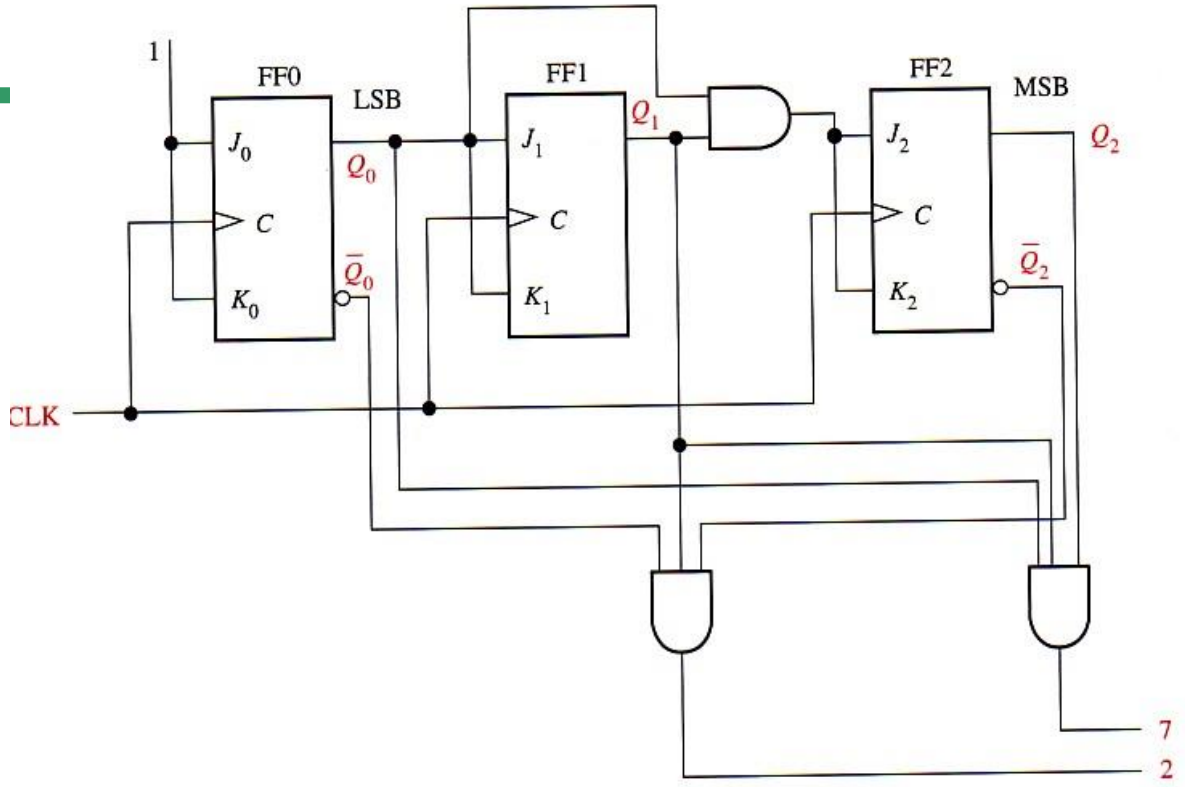
$7 = 111_2$

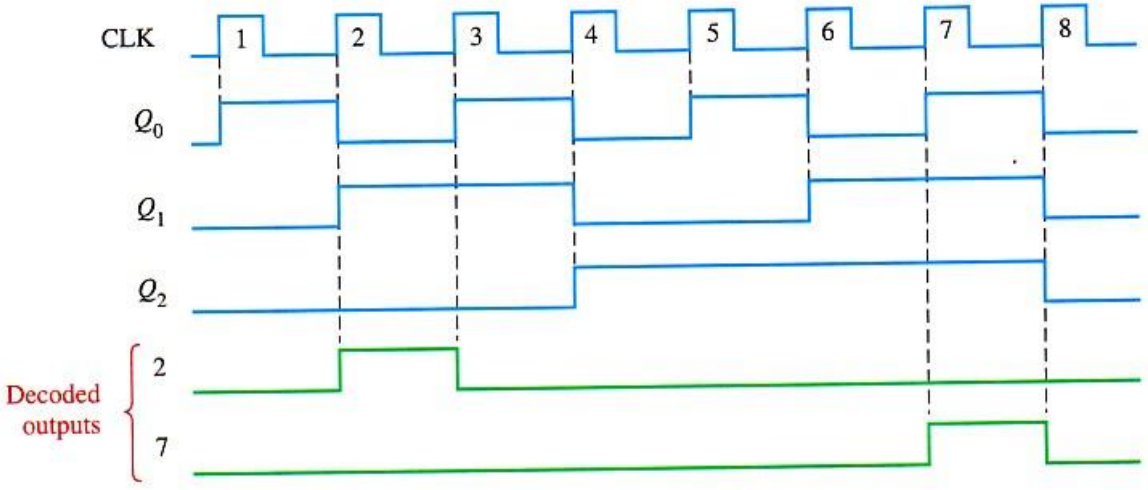


Decode

$2 = 010_2$

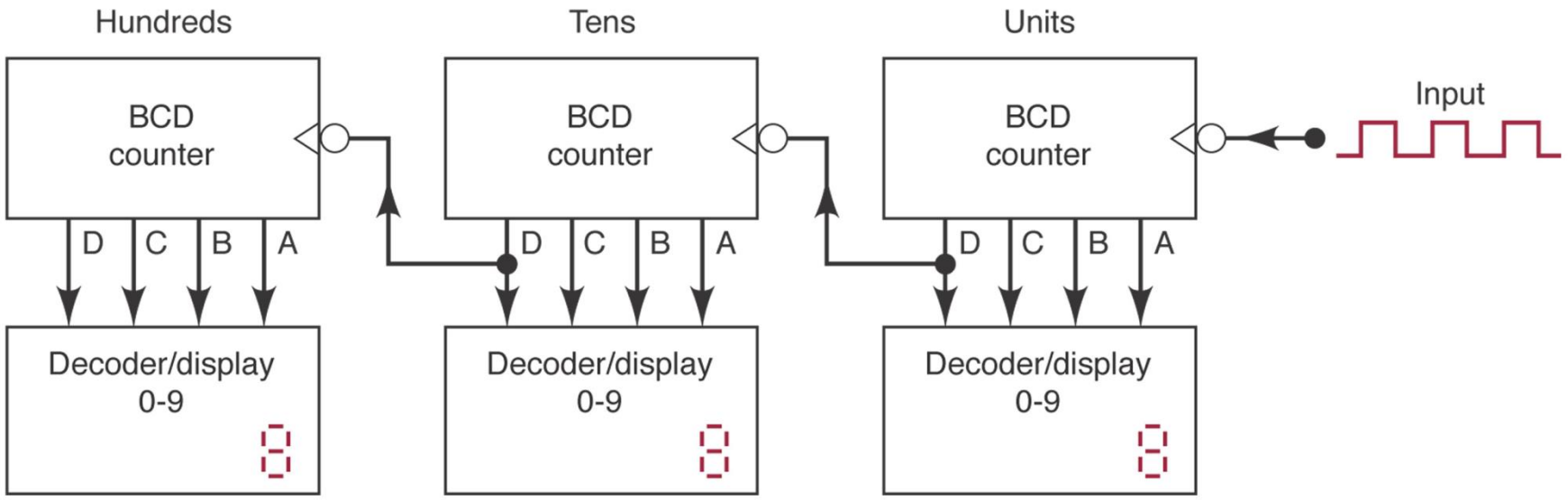
$7 = 111_2$



# Cascading BCD Counters

- Operation of a decimal 000 to 999 BCD counter



# Cascading BCD Counters

- Initially all counters are reset to 0
- Each input pulse advances the first counter once.
- The $10^{th}$ input pulse causes the counter to recycle, which advances the second counter 1.
- This continues until the second counter (10's digit) recycles, which advances the third counter 1.
- The cycle repeat until 999 is reached and all three counters start again at zero.