XMUT 202
Digital Electronics

A/Prof. Pawel Dmochowski

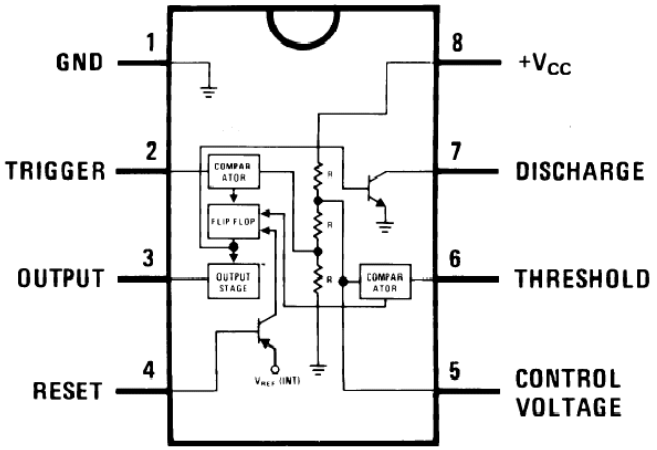
School of Engineering and Computer Science
Victoria University of Wellington

The 555 Timer

The 555 timer is a very useful IC as it can be configured as either a monostable or astable multivibrator.

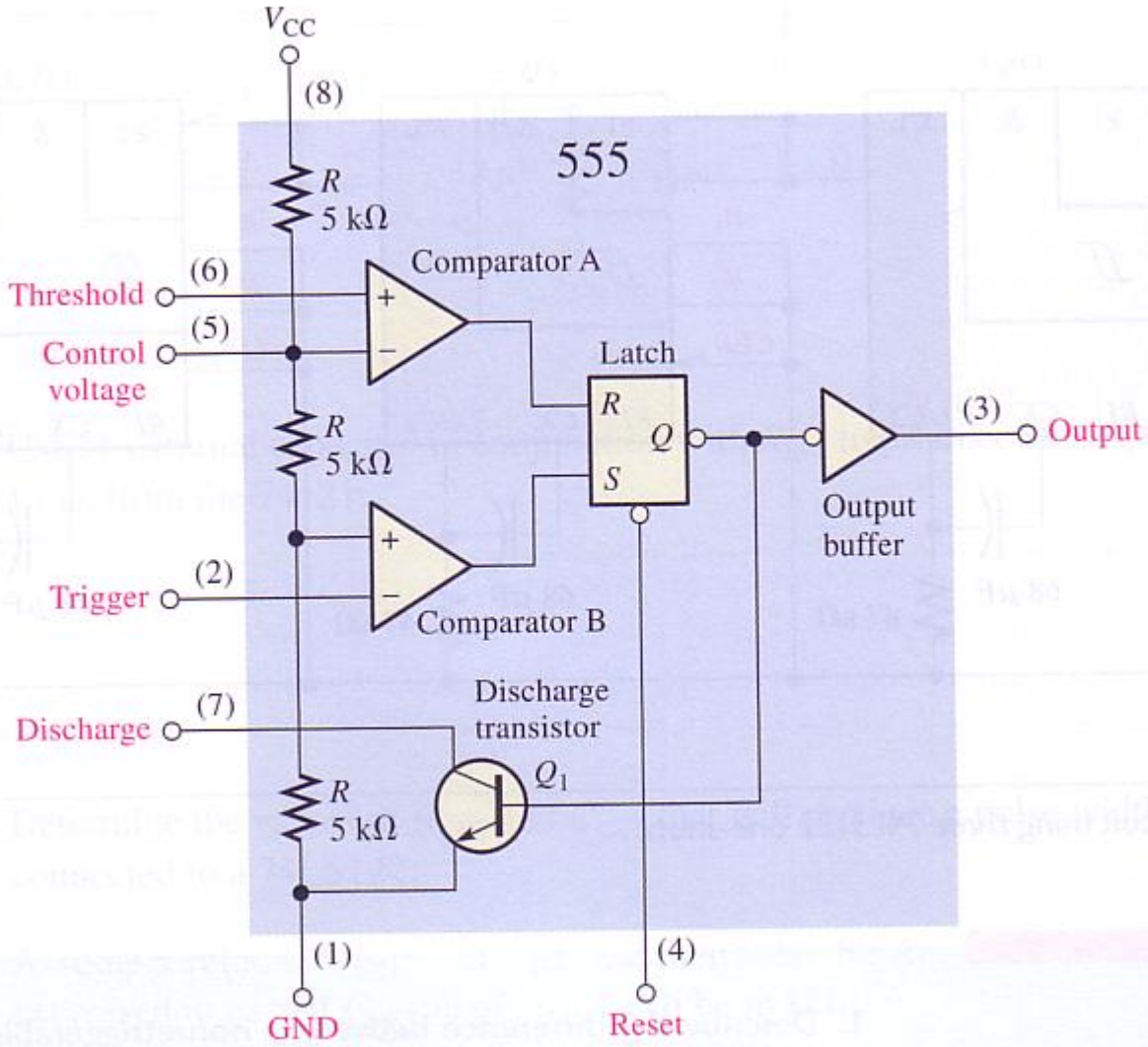
•Basic design ~1971. Now available in CMOS and in 2-in-1 version (556)

• Use 2 internal comparators: O/P is HI when the voltage on the non-inverting (+) input is higher than the voltage on the inverting (-) input.



LM555 Timer

Internal construction of a 555 Timer



Comparators: O/P is HI when the voltage on the non-inverting (+) input is higher than the voltage on the inverting (-) input.

Voltage divider consisting of **three** 5kΩ resistors – providing

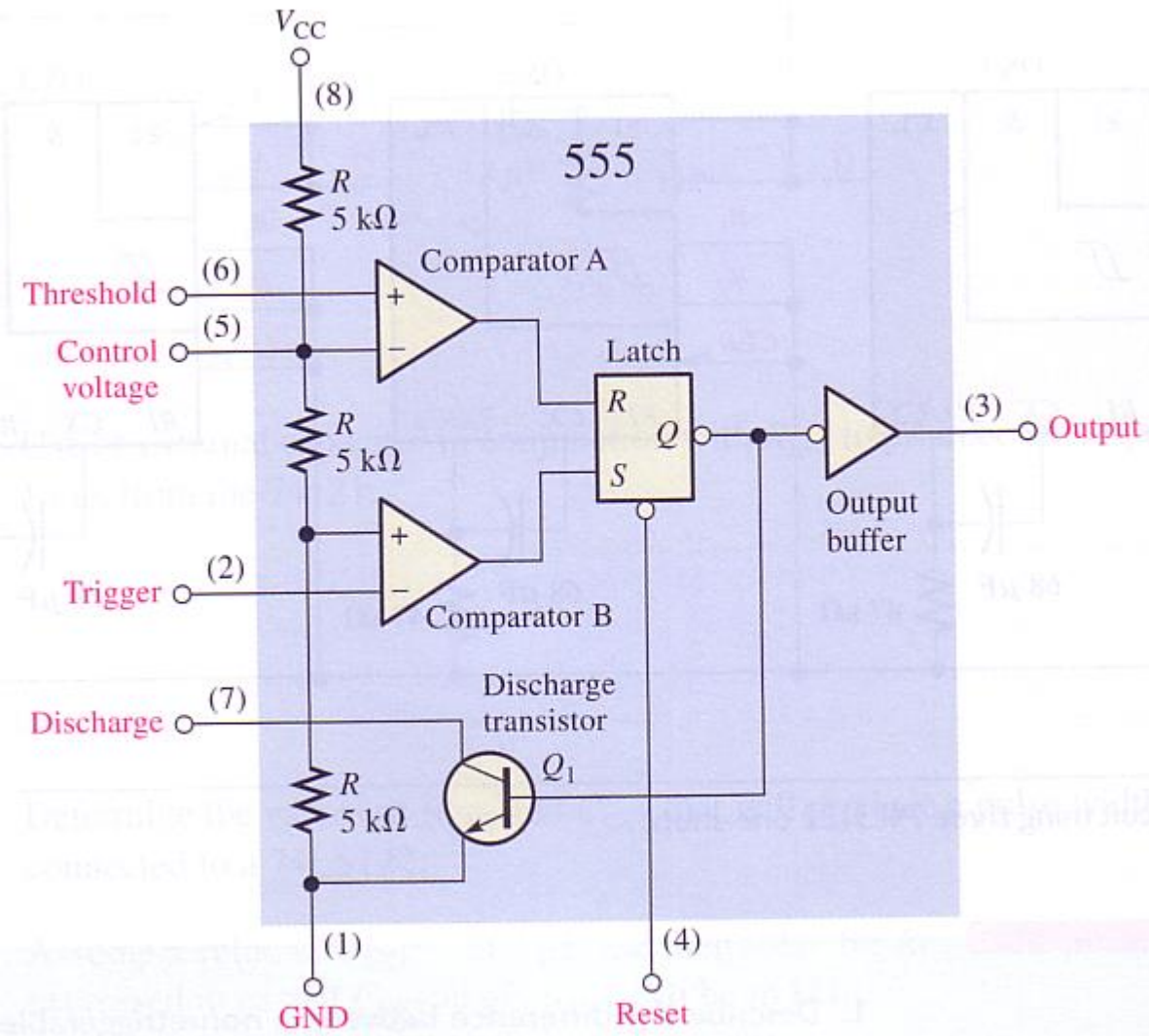
a trigger level = 1/3 V_{cc} and a threshold level = 2/3 V_{cc}

Trigger level normally HI – when drops below 1/3Vcc the comparator B O/P goes HI – sets the S-R latch

Will cause O/P (pin 3) to go HI

Also turn OFF the transistor Q_1 base current

Internal construction of a 555 Timer



O/P will stay HI until the normally LO threshold voltage goes > 2/3Vcc – Causes comparator A to switch and reset the latch

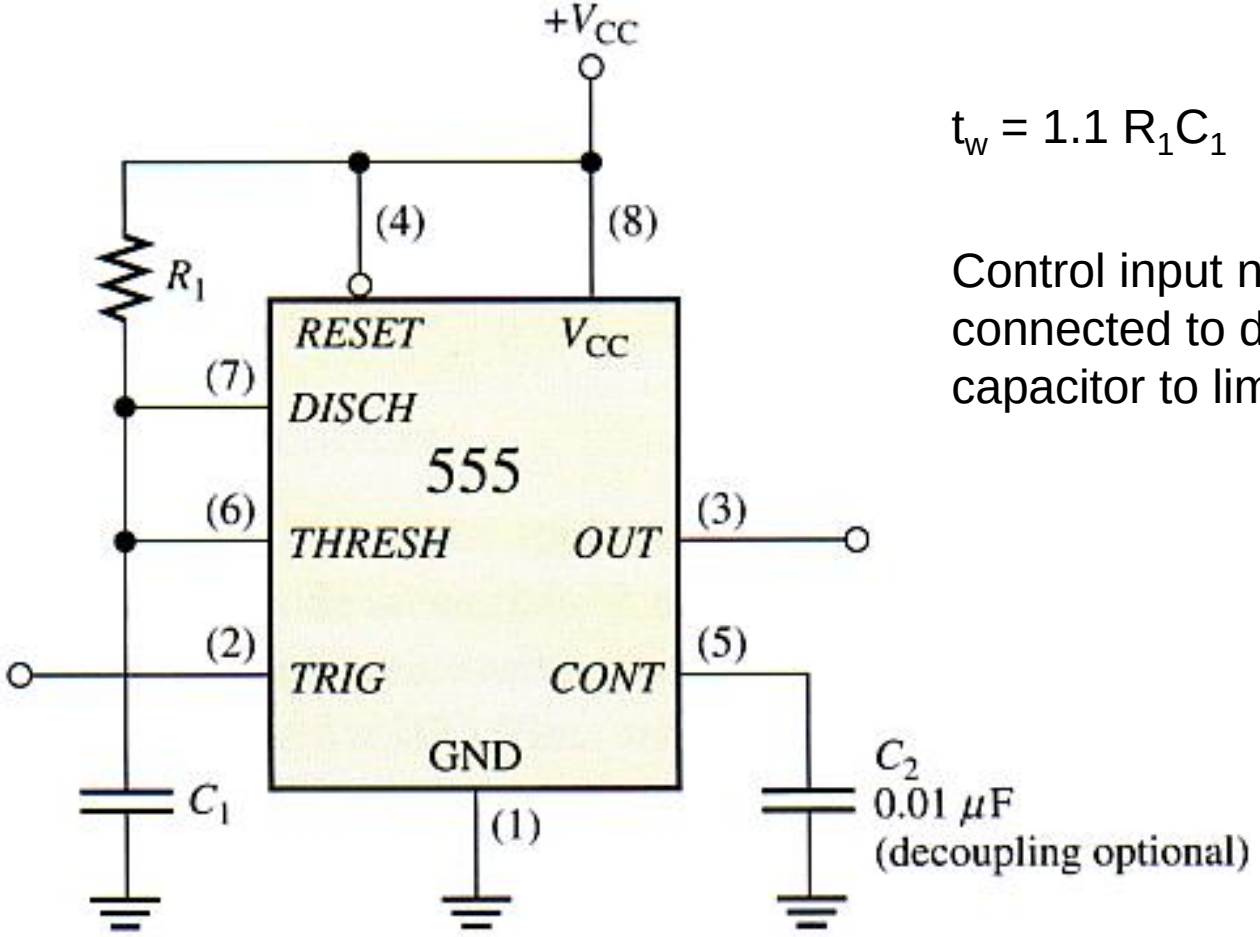
This turns O/P to LO again and switches on the discharge transistor.

External reset input can be used to reset the latch independent of internal operation.

Trigger (2) and threshold (6) pins are controlled by external components that can be connected for either monostable or astable operation.

555 Timer connected as a one-shot (monostable)

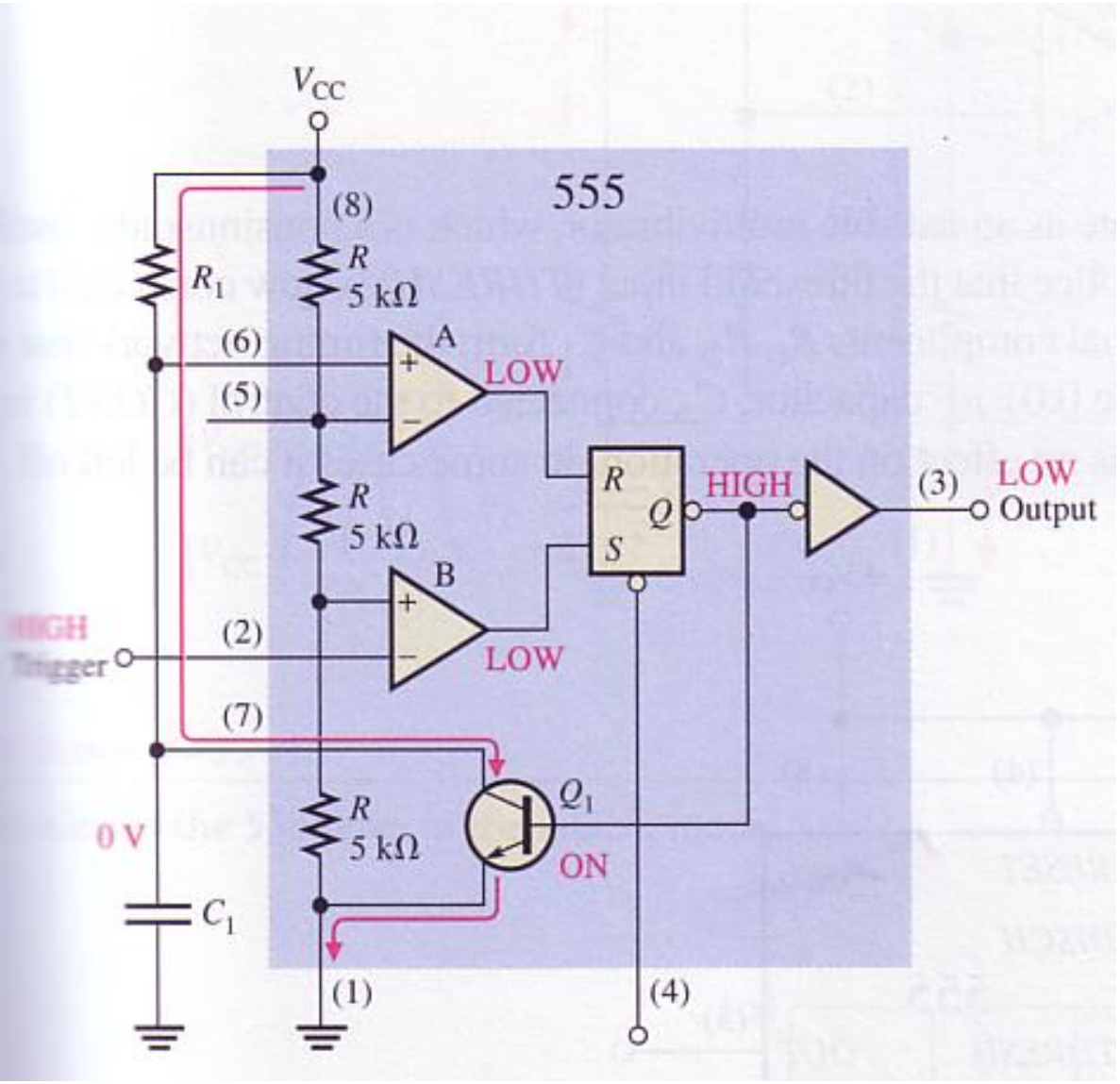
External R_1 and C_1 connected as shown below.



$t_w = 1.1\ R_1C_1$

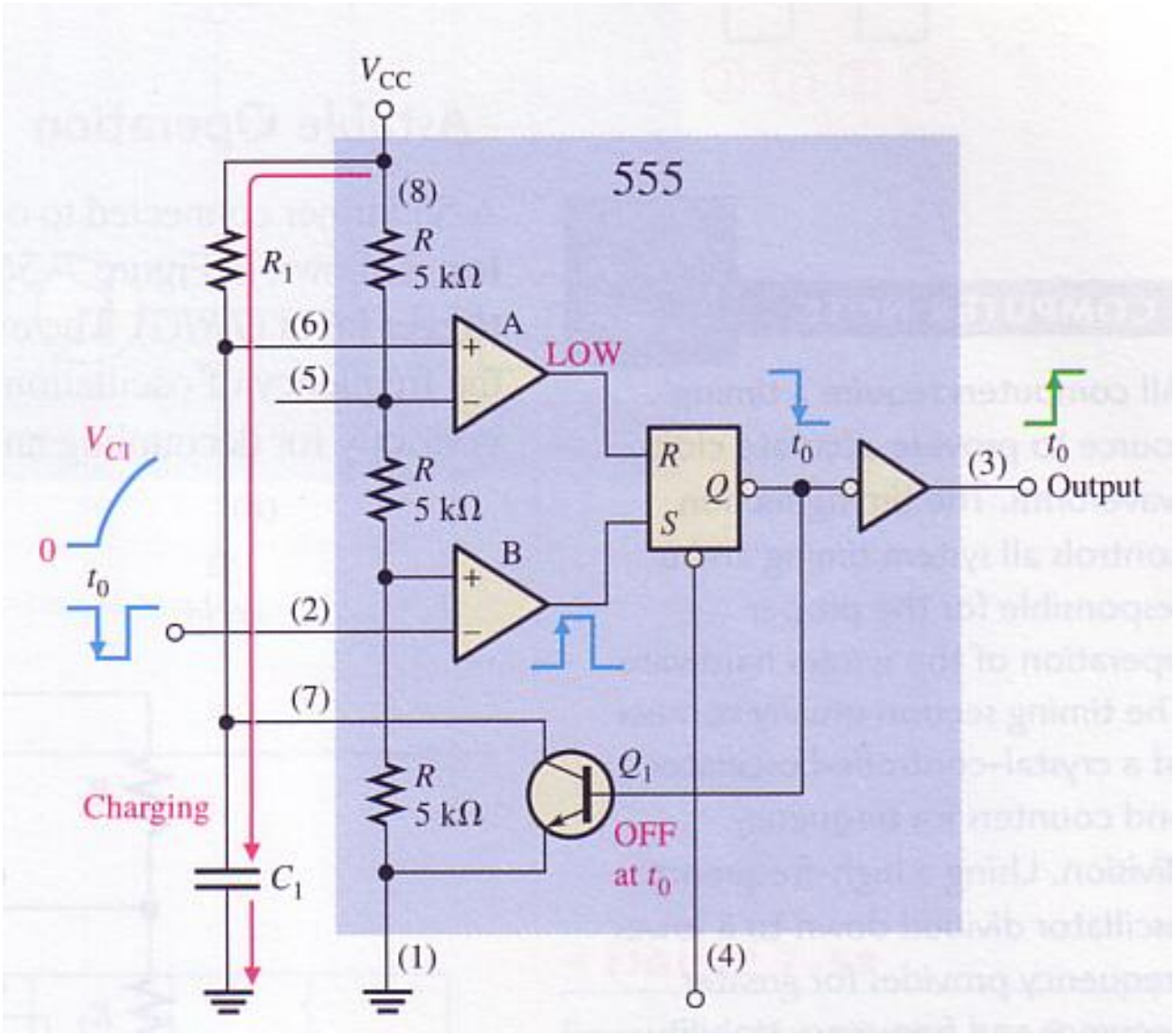
Control input not used – connected to decoupling capacitor to limit noise.

Prior to triggering



O/P is LO and Q1 is ON – keeps C1 discharged.

When triggered

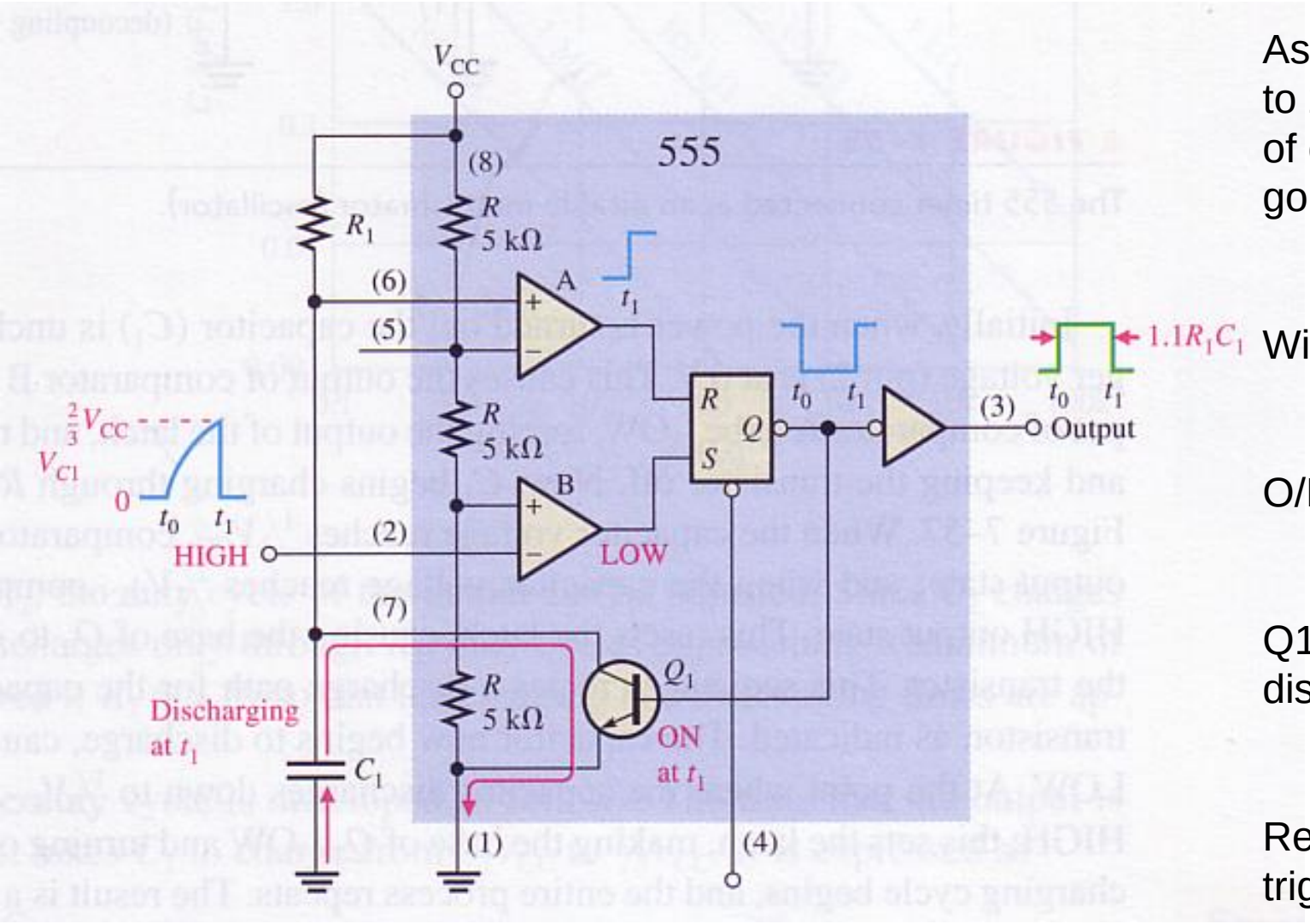


NG trigger pulse is applied – Comparator B switches to HI and the O/P switches to HI

Q1 turns off

Capacitor C1 starts to charge through R1

At end of charging interval



As C1 charges up to 2/3 Vcc, the O/P of comparator A will go HI at t1

Will reset the latch

O/P goes LO at t1

Q1 will turn ON and discharge C1

Ready for next trigger

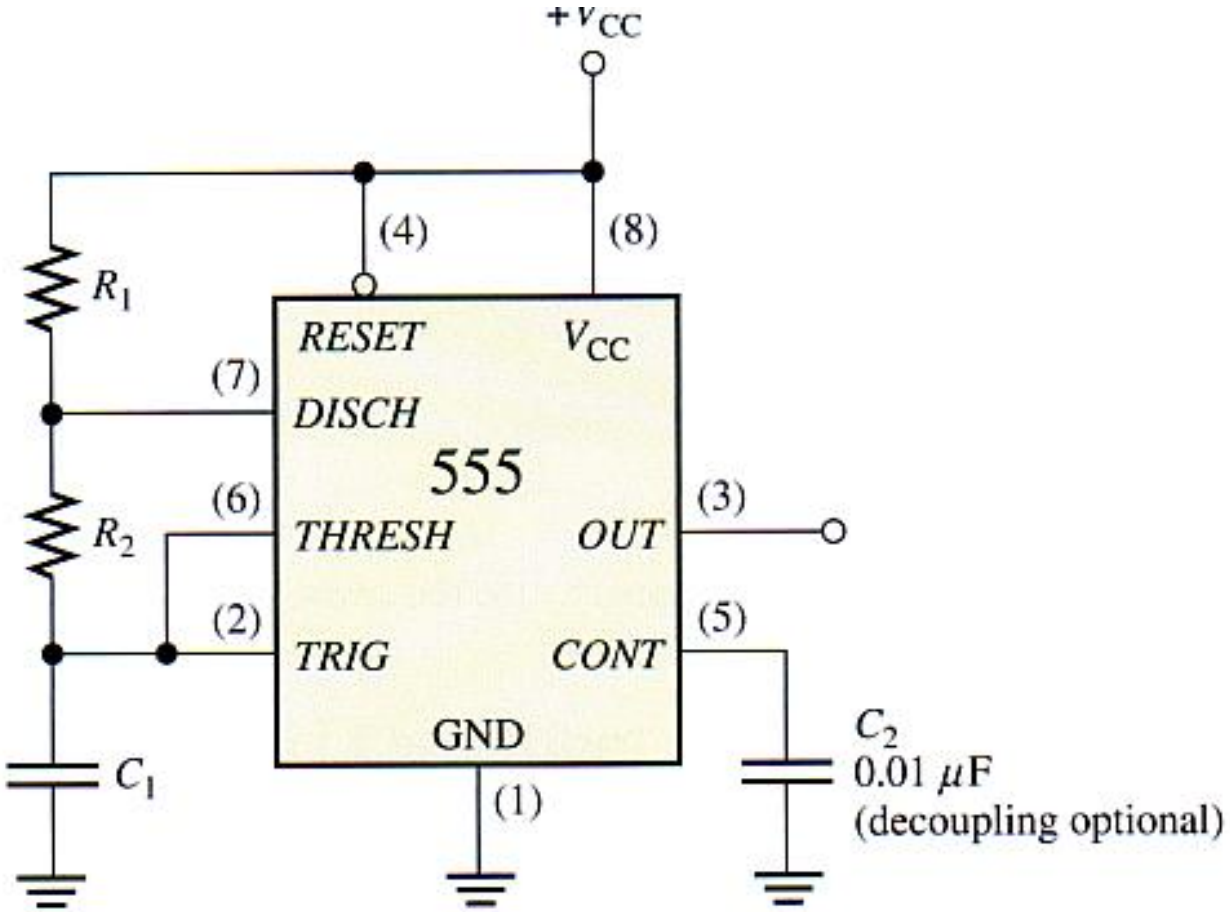
Astable Multivibrators = Clocks

Astable or **free-running multivibrators** switch back and forth between two unstable states. This makes it useful for generating clock signals for synchronous circuits.

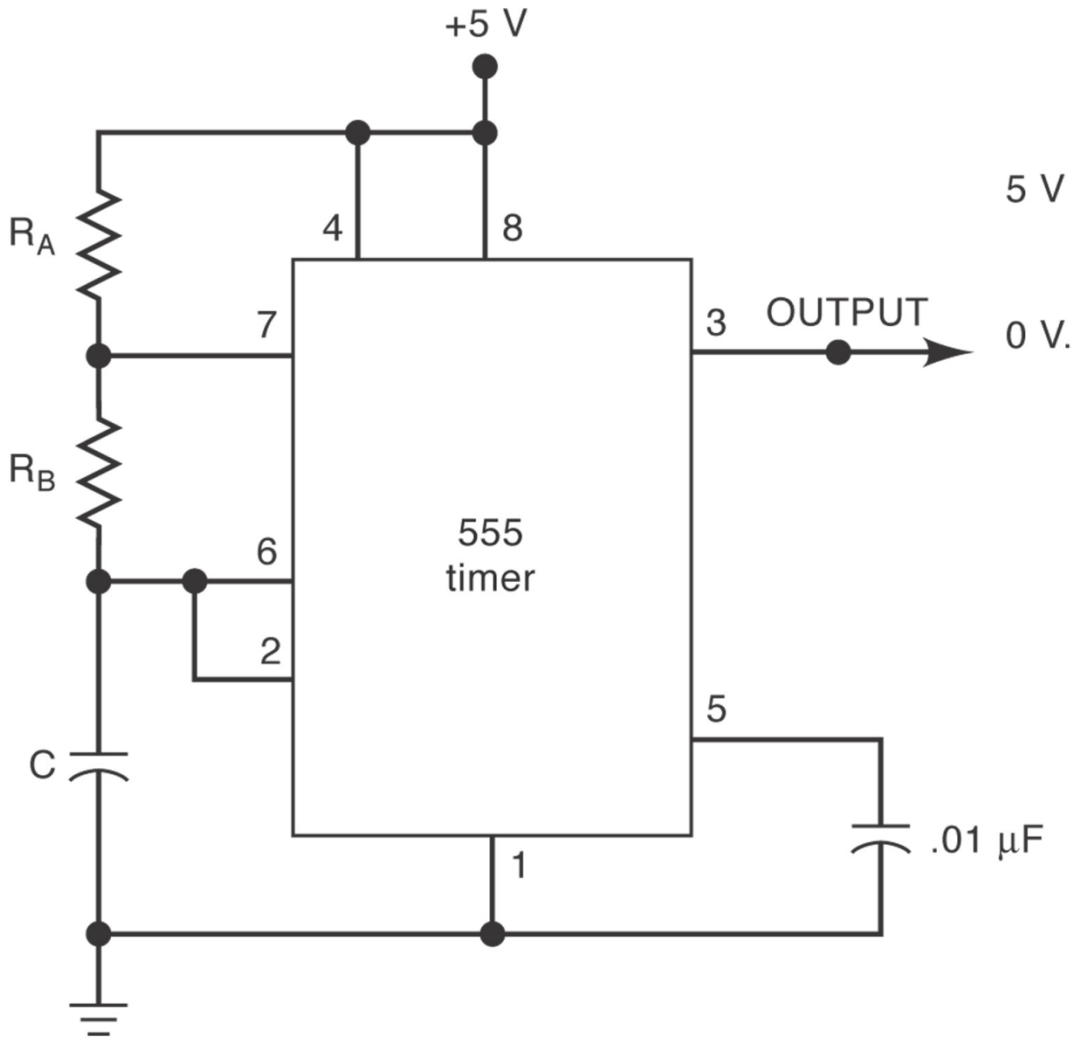
Different applications place varying demand on clock circuits in terms of precision and long term stability

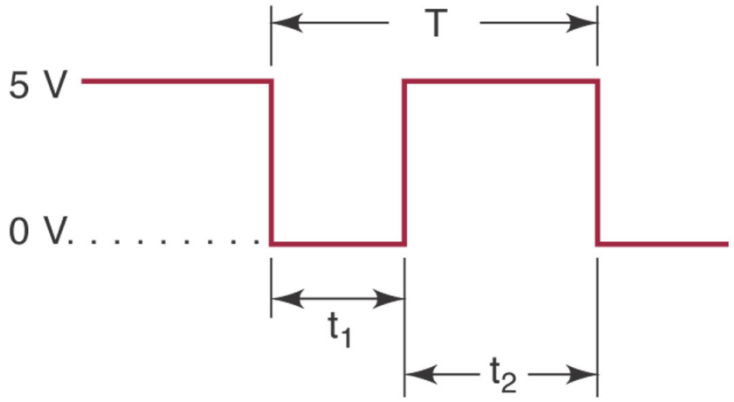
- Low end – Digital IC based clocks
- Medium – Crystal oscillator circuits
- Temperature controlled crystal oscillators, atomic clocks

The 555 connected as an astable (free running) multivibrator

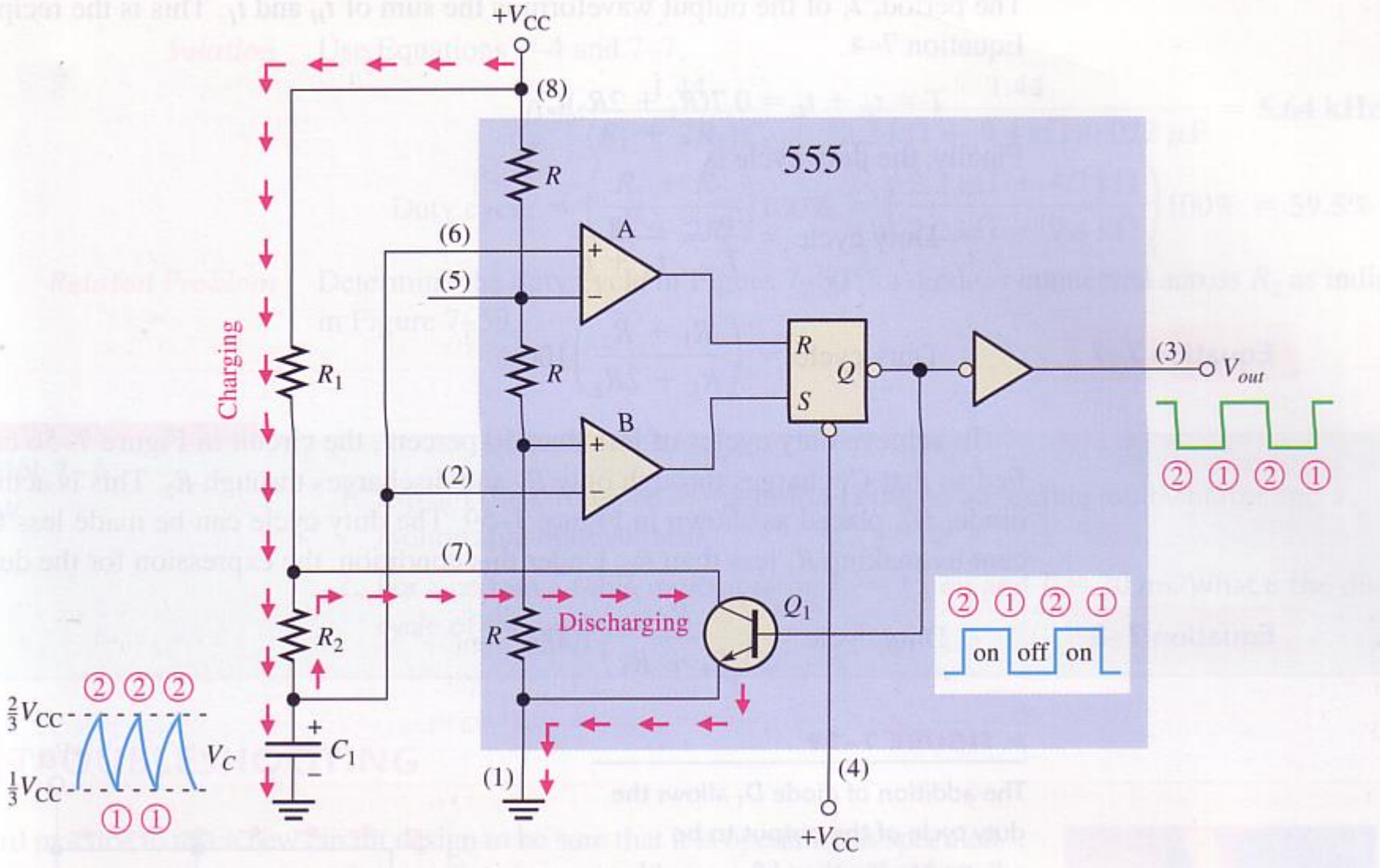


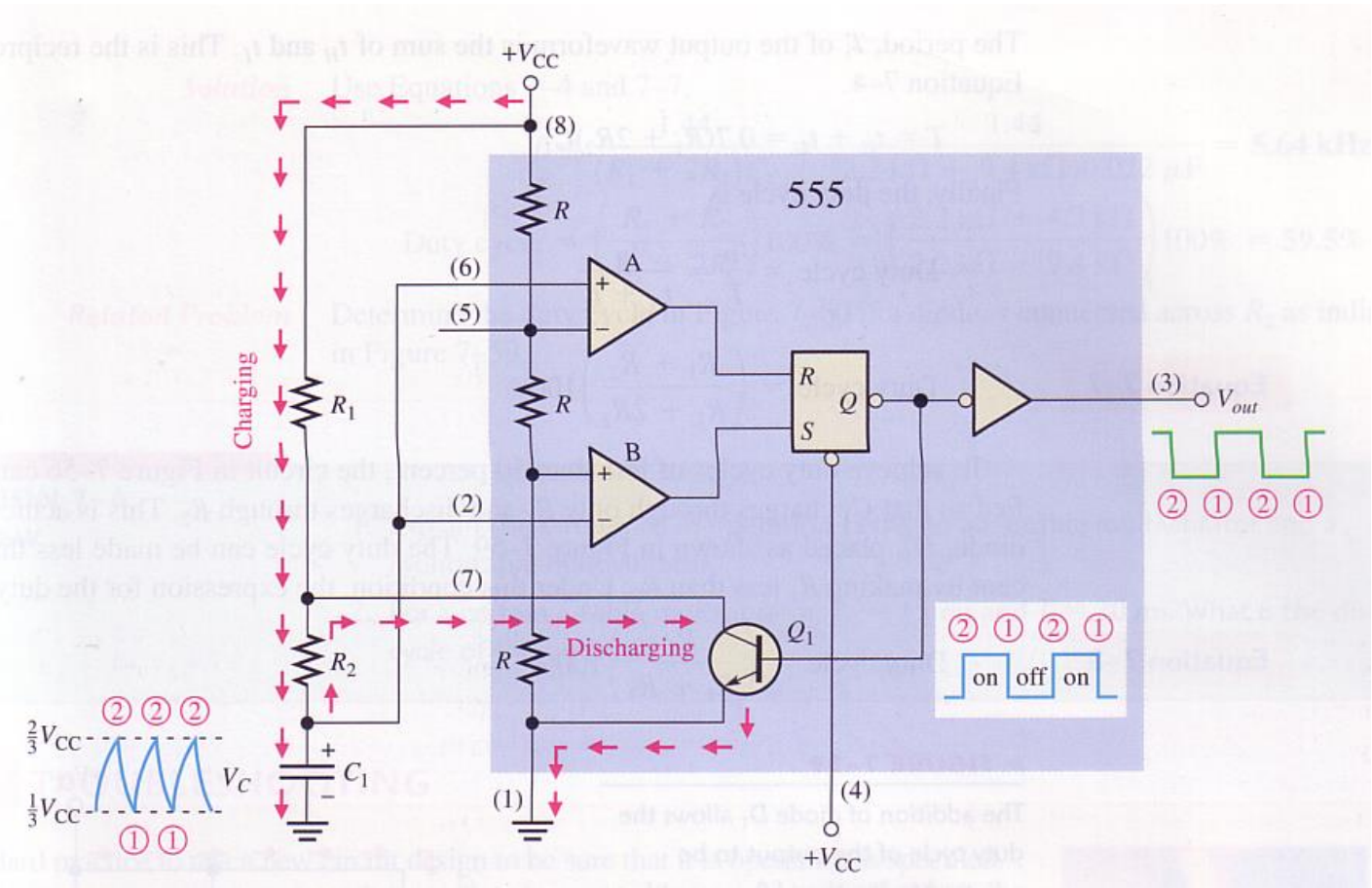
The 555 Timer as an Astable Multivibrator.





$t_1 = 0.693\ R_B C$
$t_2 = 0.693\ (R_A + R_B)C$
$T = t_1 + t_2$
frequency = 1/T
duty cycle = $t_2/T \times 100\%$
R_A] 1 kV
$R_A + R_B$ * 6.6 MV
C] 500 pF


+VCC
(8)
555
R
(6)
A
(5)
R
R1
Charging
R
Q
S
(3)
Vout
B
(2)
(7)
Q1
R2
R
Discharging
on
off
on
(1)
(4)
+VCC
C1
VC
2/3VCC
1/3VCC



Astable operation

Threshold (6) now connected to the trigger (2) input

Two external resistors and a capacitor

2nd Capacitor (on Control) for decoupling only – can be left off

Initially capacitor is uncharged – trigger pin (2) is at 0V

This causes O/P of Comp B = HI and Comp A = LO This will keep Q1 OFF

Capacitor will now start to charge through R1 and R2

When it reaches 1/3 Vcc Comp B will switch to LO, when it reached 2/3 Vcc Comp A will switch to HI – Will now reset the latch.

This switches on Q1 and discharges capacitor through R2.

When capacitor discharges to 1/3Vcc Comp B goes HI and sets the latch again.

This turns off Q1 – Another charge cycle of the capacitor starts

Calculate the frequency and the duty cycle of the 555 astable multivibrator output for $C = 0.001\ \mu\text{F}$, $R_A = 2.2\ \text{k}\Omega$, and $R_B = 100\ \text{k}\Omega$.

Design a 555 free-running oscillator to produce an approximate square wave at 40 kHz. C should be kept at 500 pF or greater.

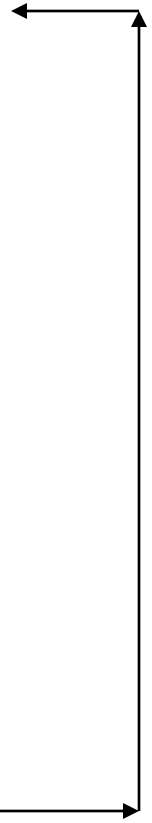
Counters and Register

Synchronous vs Asynchronous Counters

- Combining flip-flops and logic gates to form various counters and registers.
- Asynchronous (Ripple) Counters
- Synchronous (Parallel) Counters

A look at the standard up counting sequence

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



A: Toggles as normal (every step)

B: Toggles when A goes HI to LO

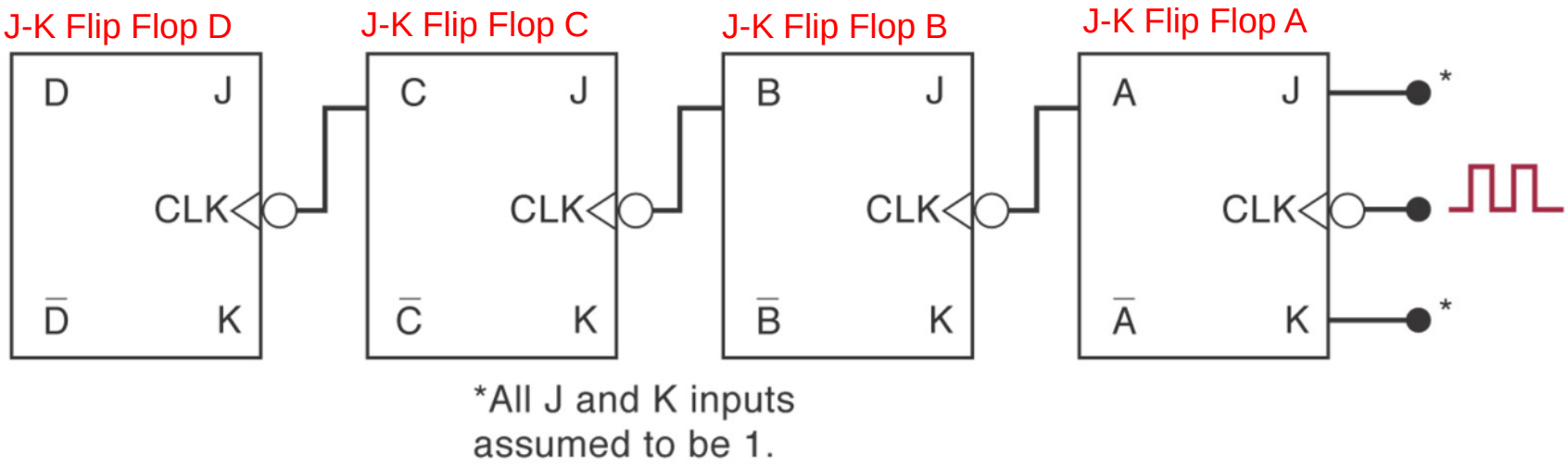
C: Toggles when B goes HI to LO

Asynchronous (Ripple) Counters

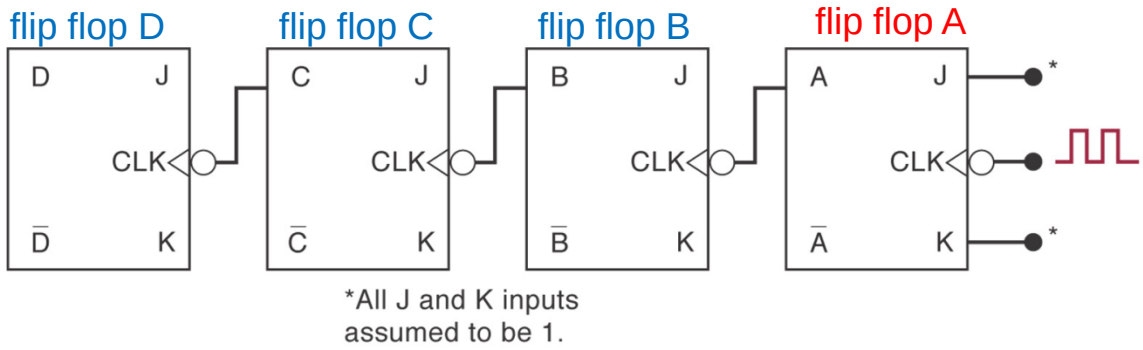
Counters that are made of flip flops that are supplied with different clock signals.

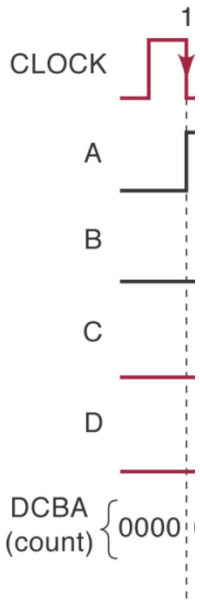
Asynchronous (Ripple) Counters

Counters that are made of flip flops that are supplied with different clock signals.

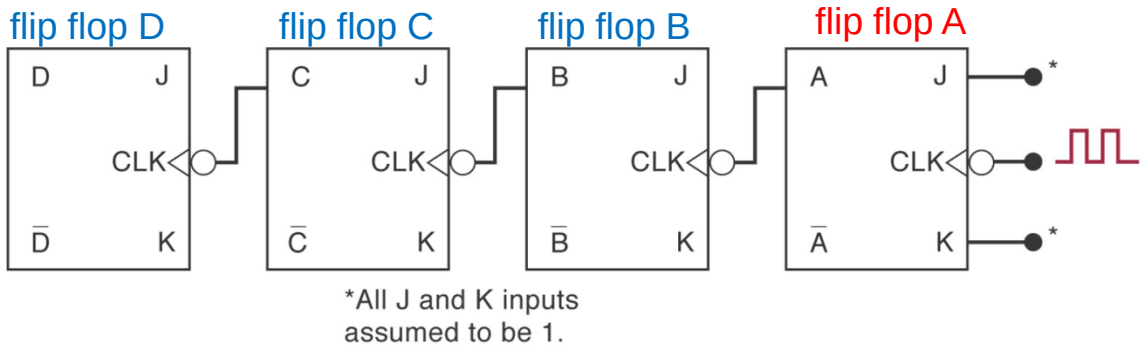


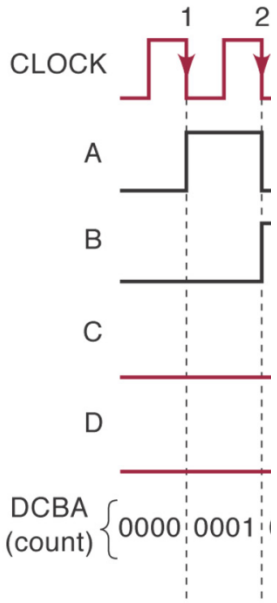
Asynchronous (Ripple) Counters



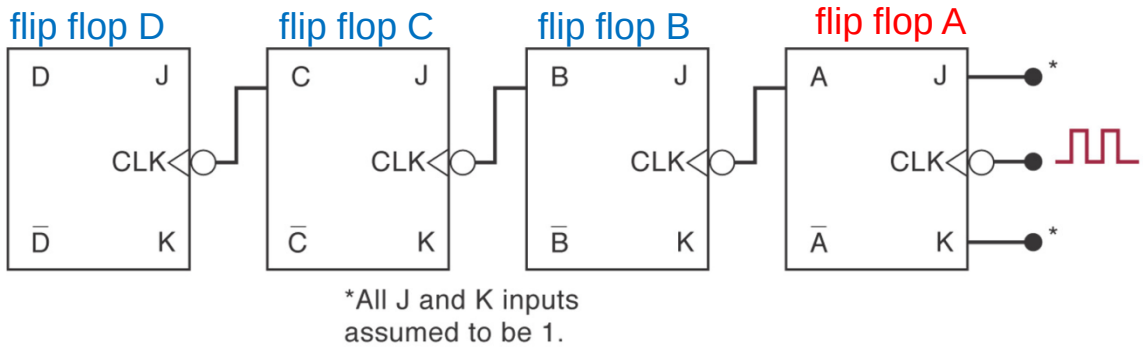


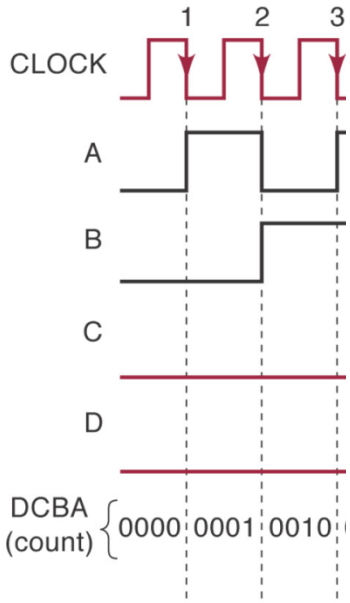
Asynchronous (Ripple) Counters



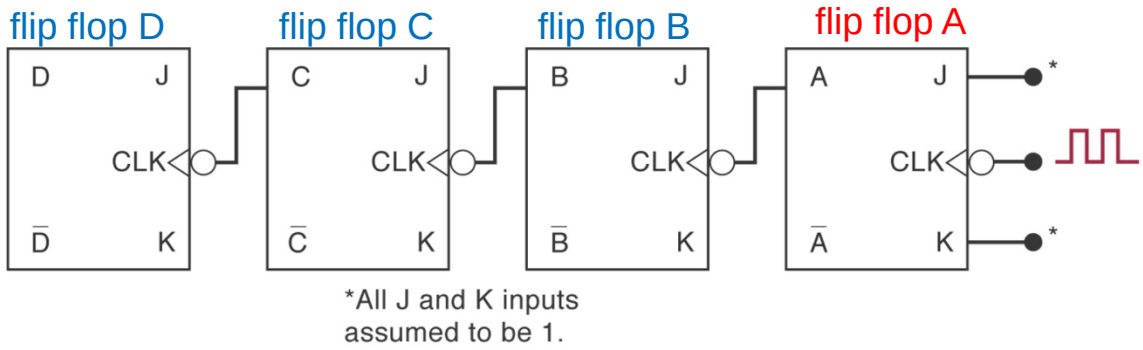


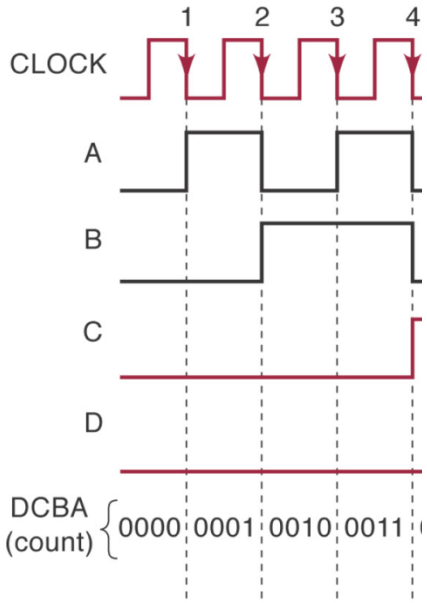
Asynchronous (Ripple) Counters



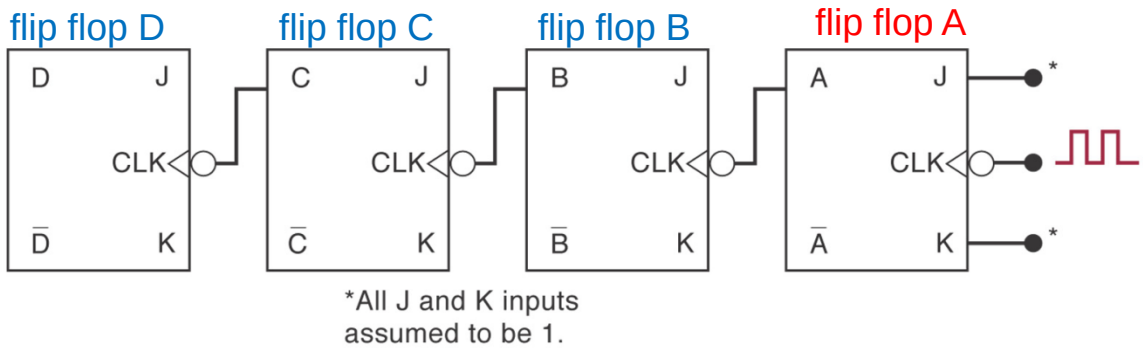


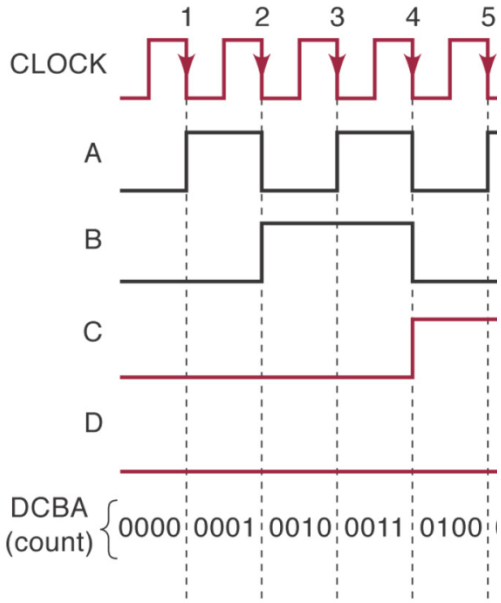
Asynchronous (Ripple) Counters



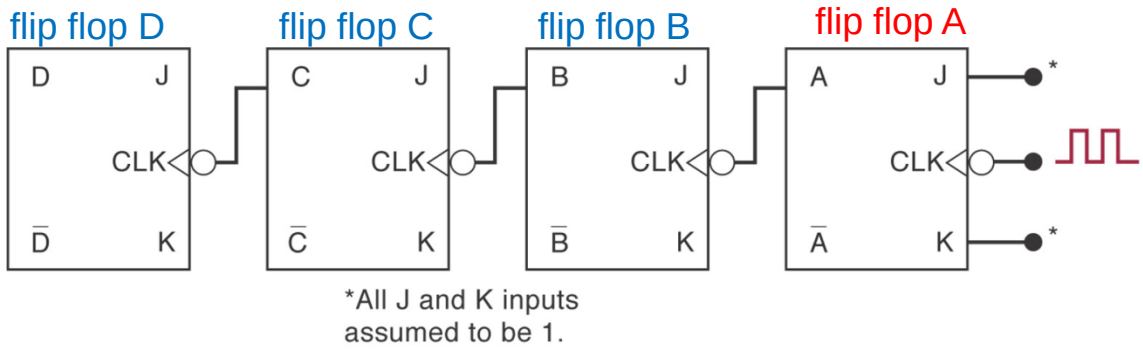


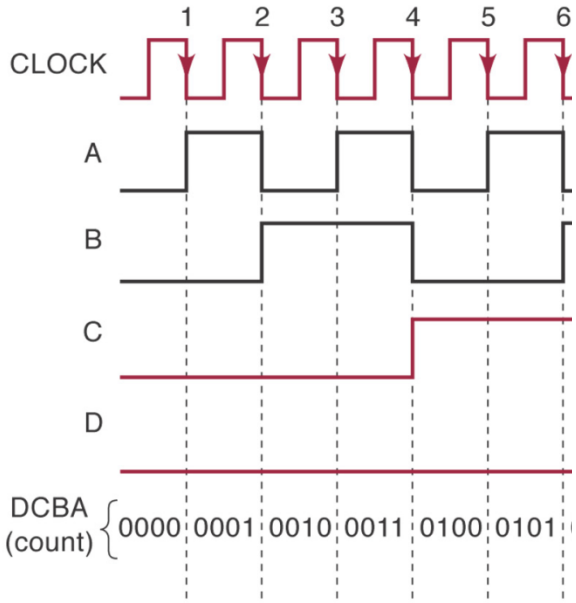
Asynchronous (Ripple) Counters



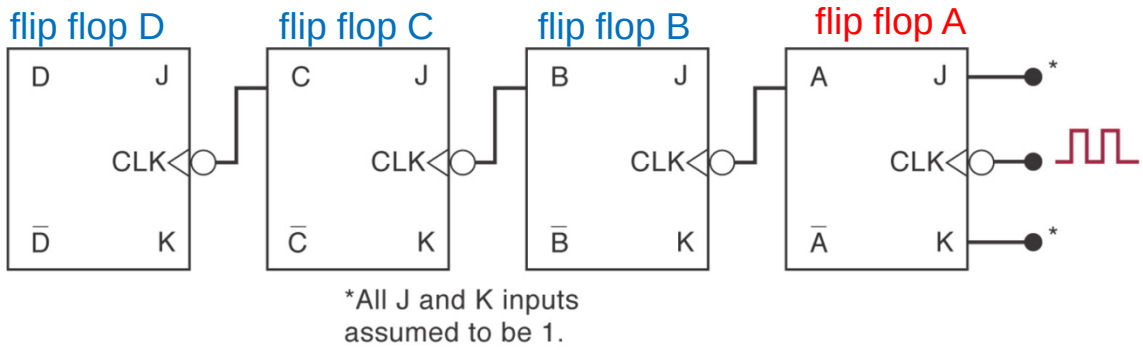


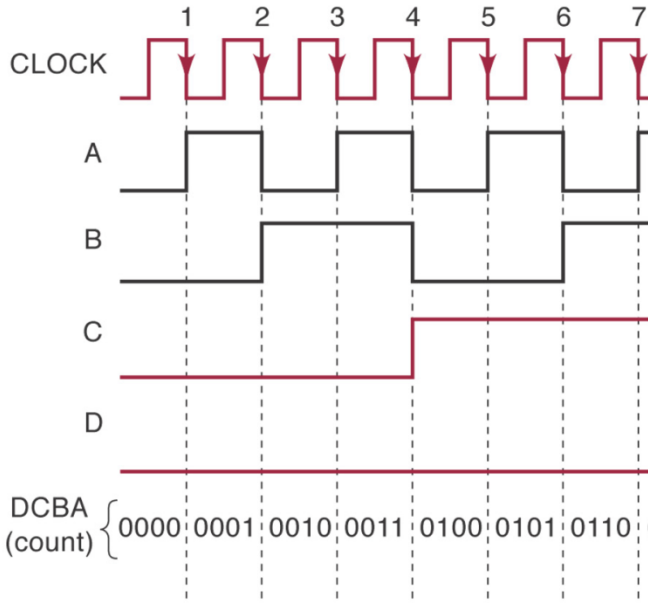
Asynchronous (Ripple) Counters



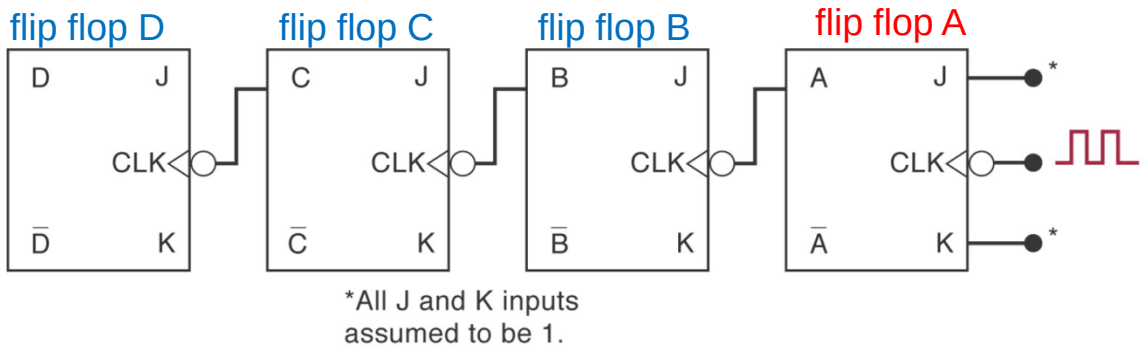


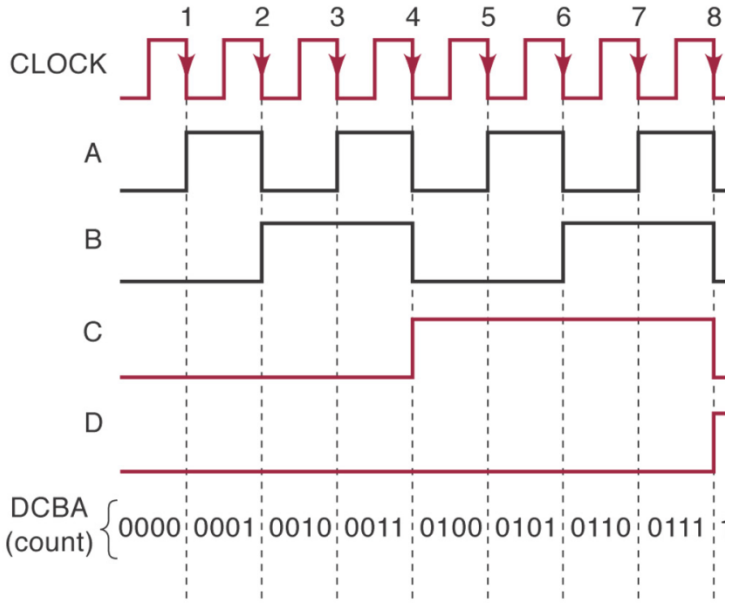
Asynchronous (Ripple) Counters



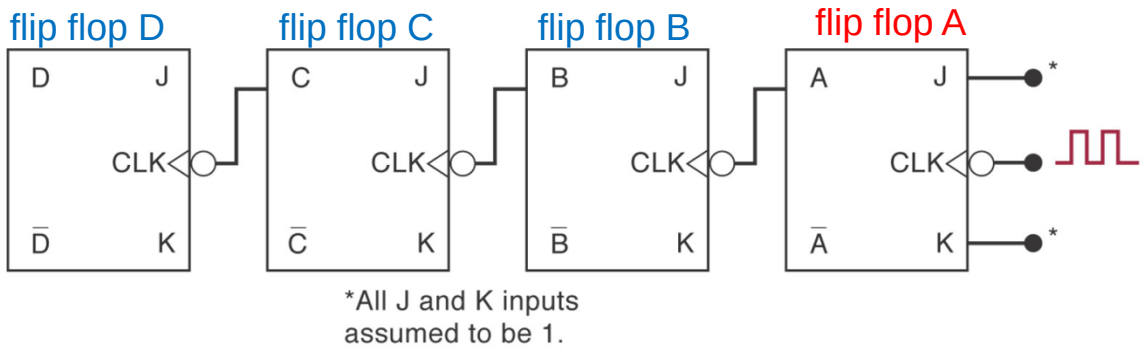


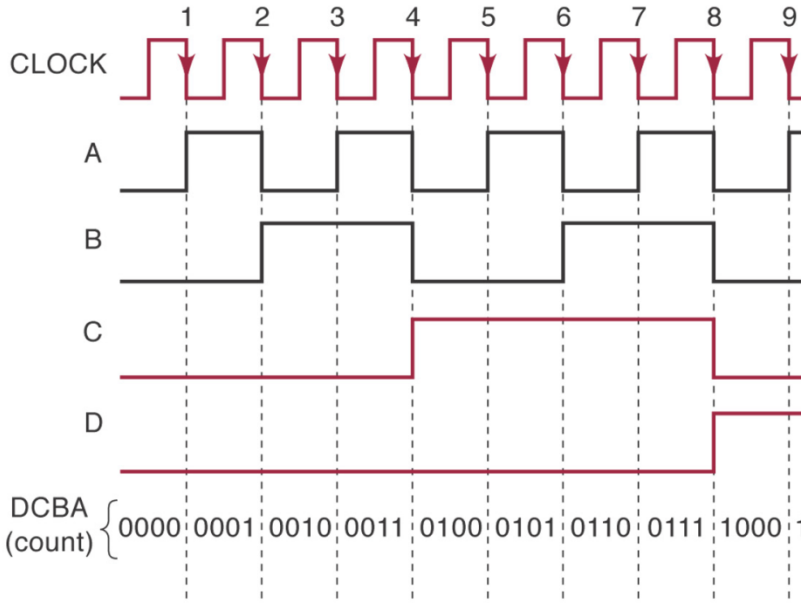
Asynchronous (Ripple) Counters



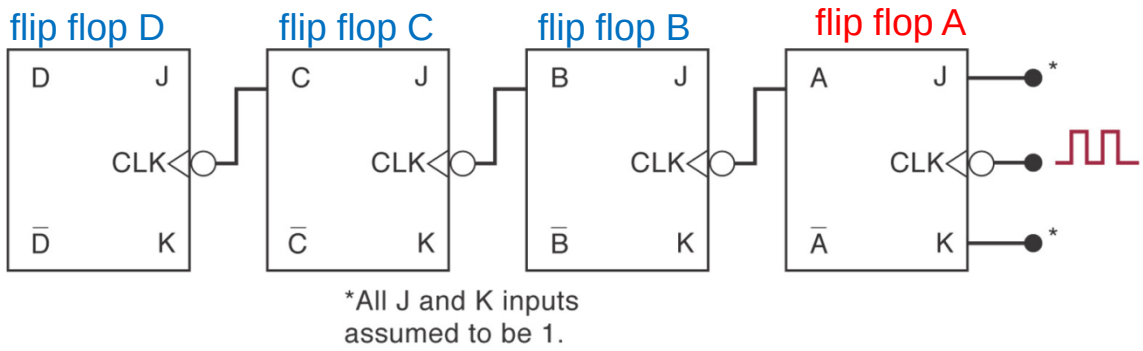


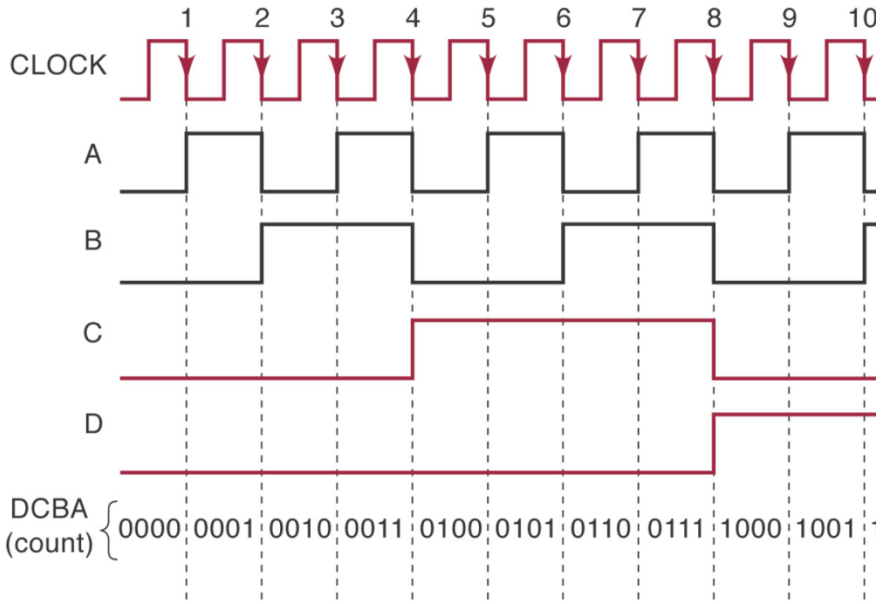
Asynchronous (Ripple) Counters



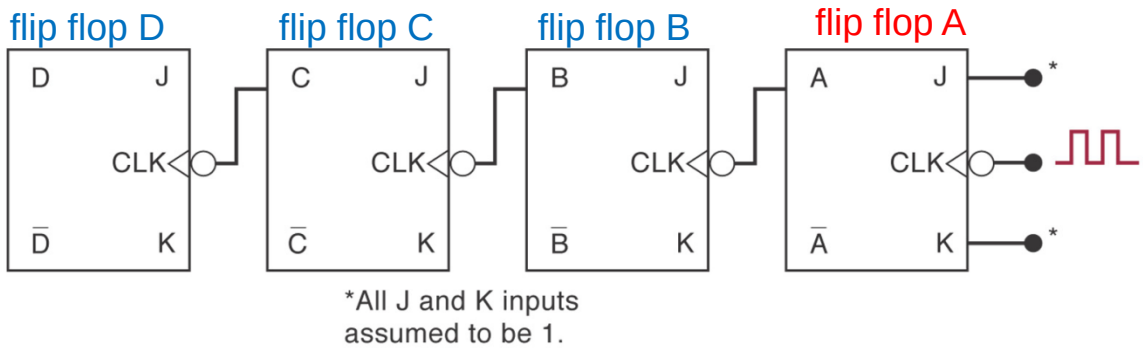


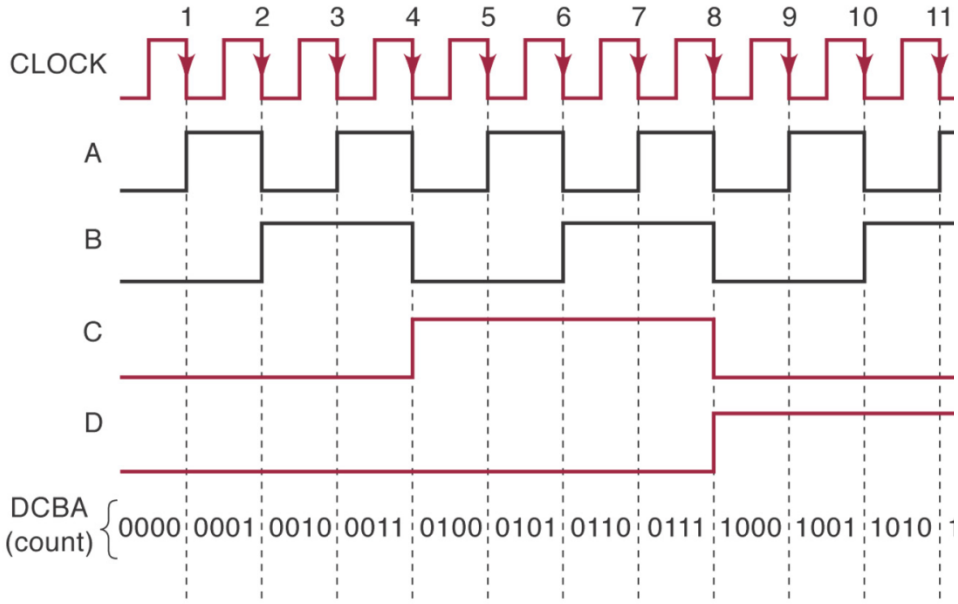
Asynchronous (Ripple) Counters



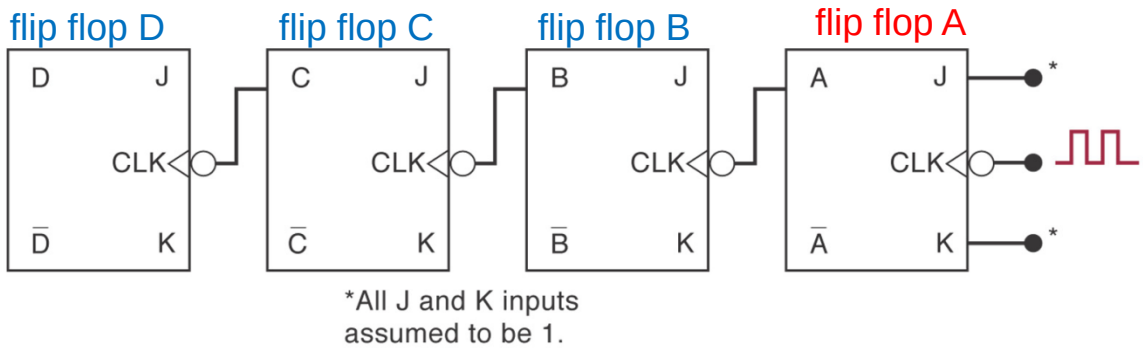


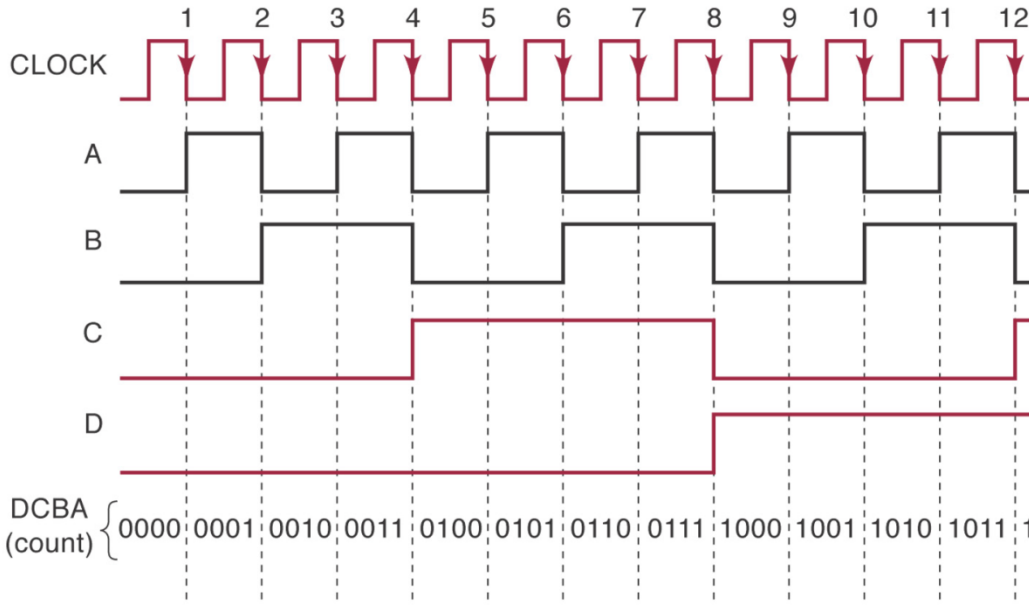
Asynchronous (Ripple) Counters



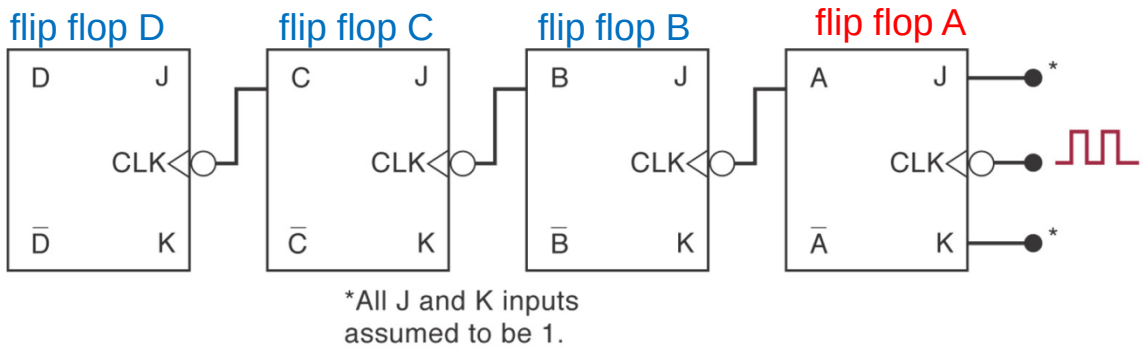


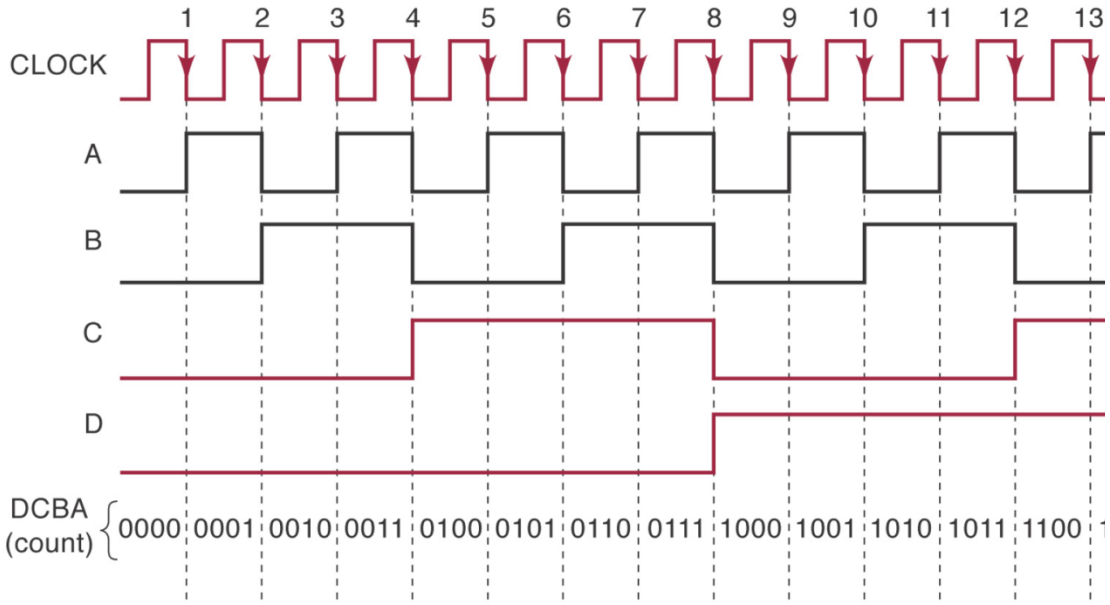
Asynchronous (Ripple) Counters



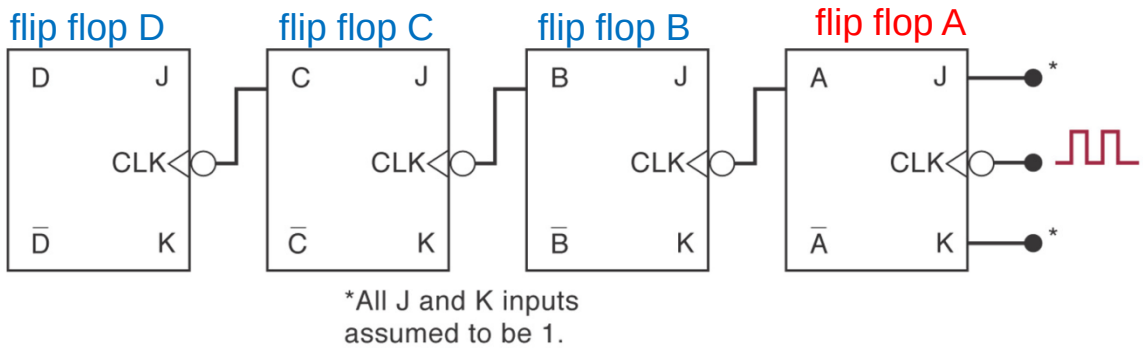


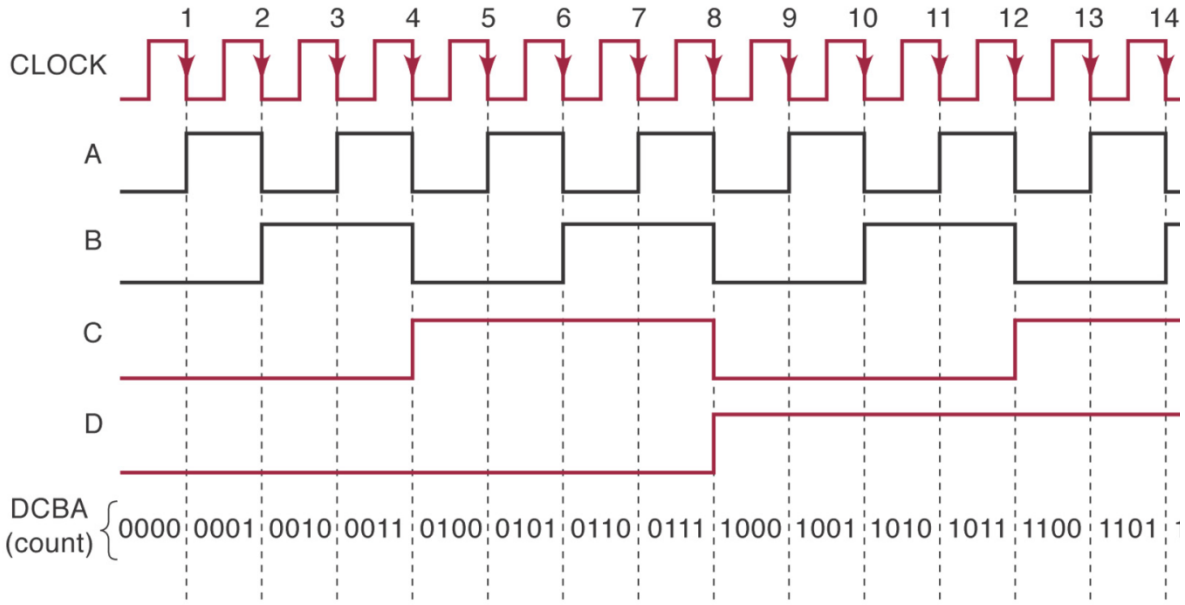
Asynchronous (Ripple) Counters



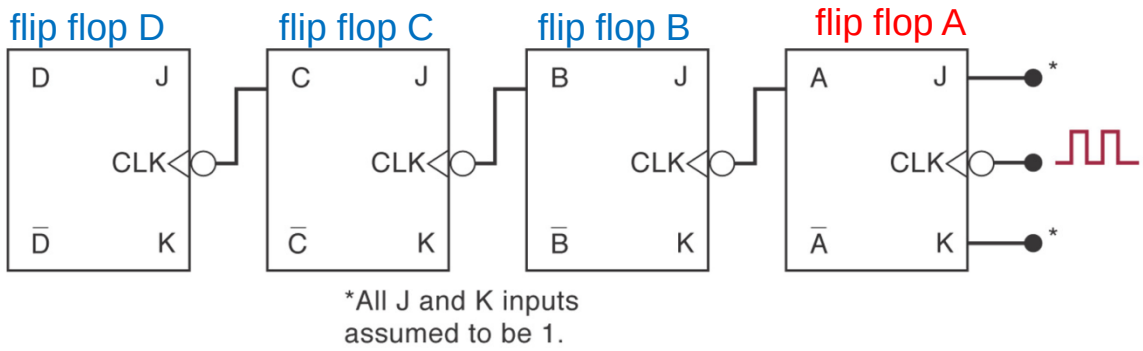


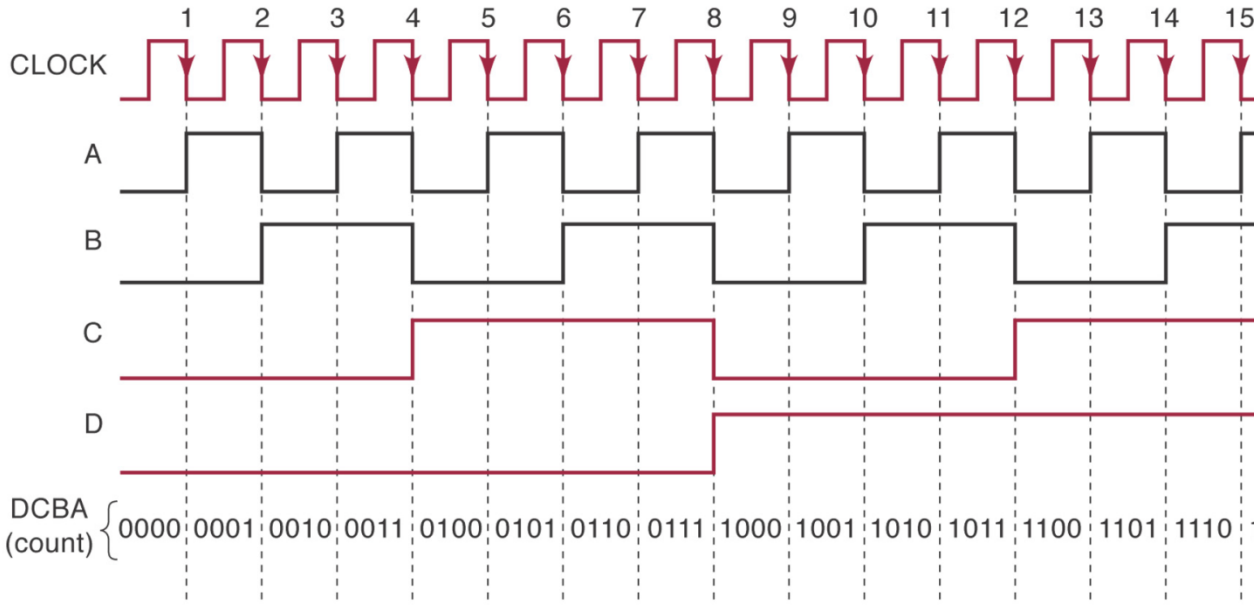
Asynchronous (Ripple) Counters



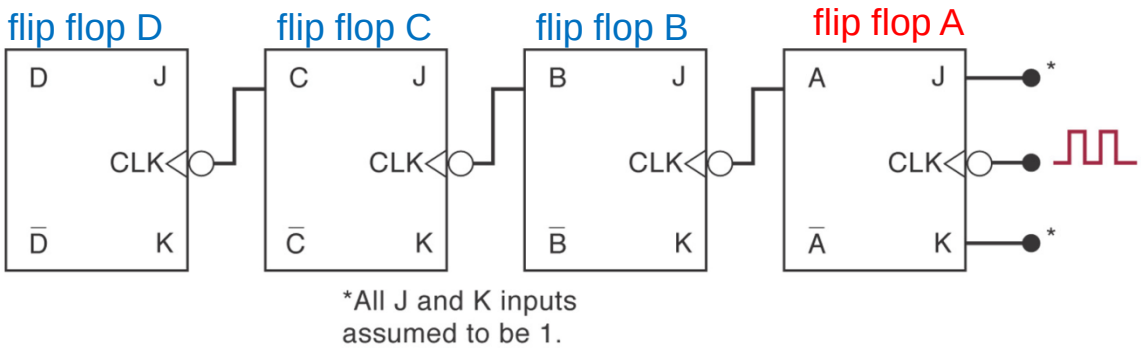


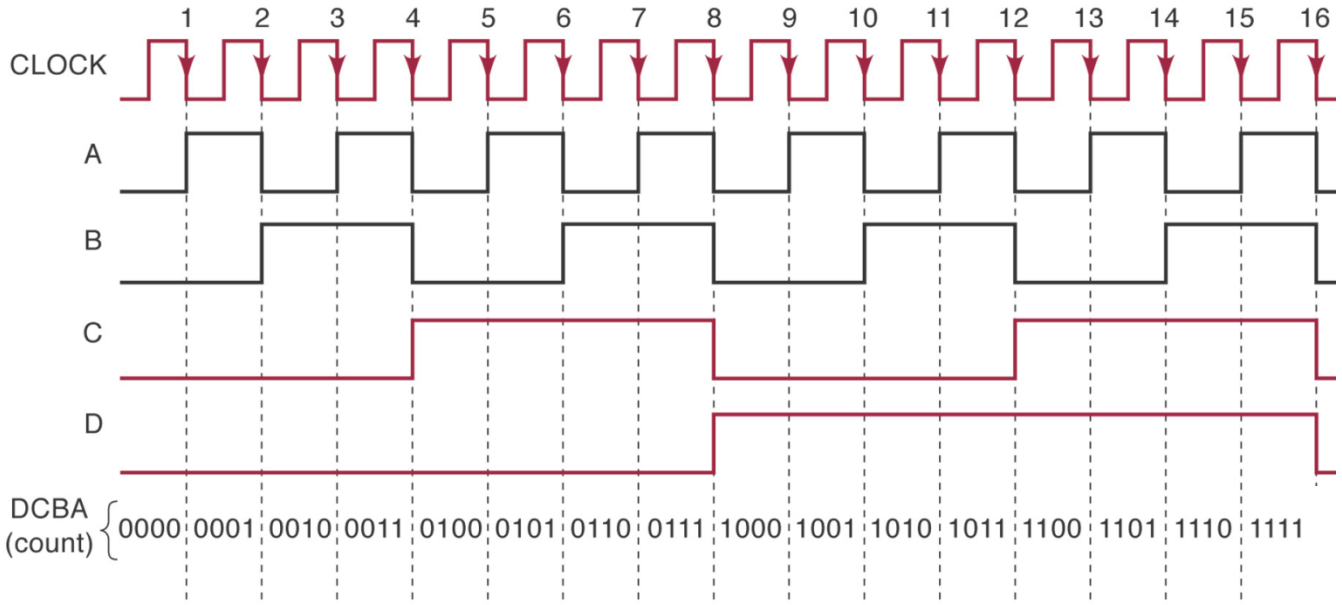
Asynchronous (Ripple) Counters



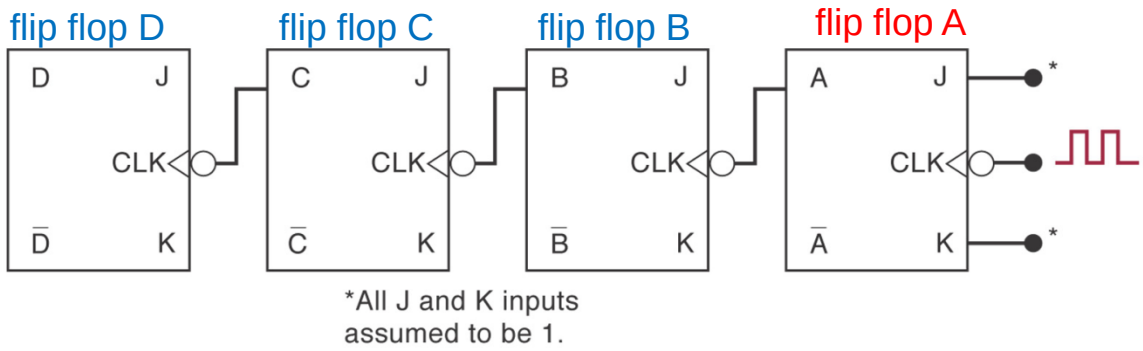


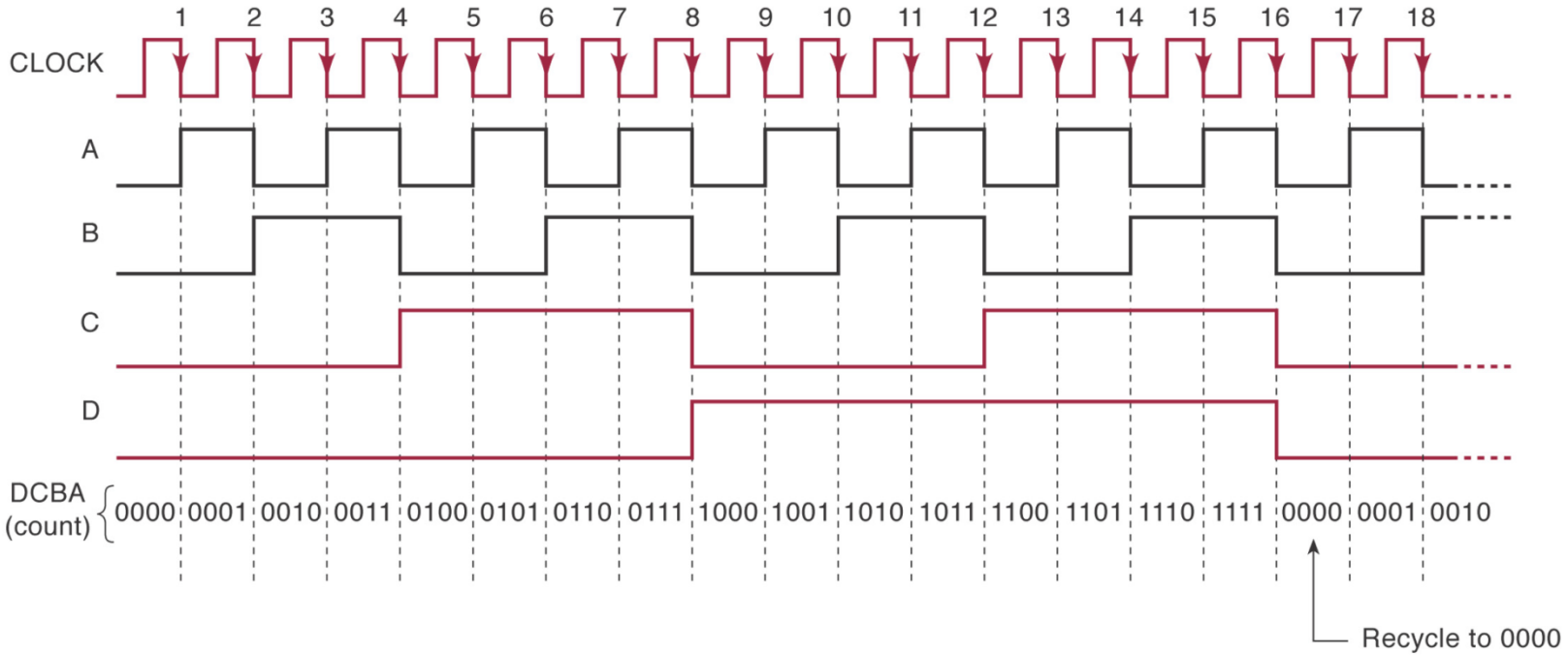
Asynchronous (Ripple) Counters





Asynchronous (Ripple) Counters





Asynchronous (Ripple) Counters

- Review of four bit counter operation
  - Clock is applied only to flip flop A. J and K are high in all flip flops.
  - Output of flip flop A is CLK of flip flop B and so forth.
  - Flip flop outputs D, C, B, and A are a 4 bit binary number with D as the MSB.
  - After the NGT of the 16^{th} clock pulse the counter recycles to 0000.
- This is an asynchronous counter because state is not changed in exact synchronism with the clock.

Asynchronous (Ripple) Counters

- **MOD number is equal to the number of states that the counter goes through before recycling.** Adding flip flops will increase the MOD number.

Asynchronous (Ripple) Counters

- **MOD number is equal to the number of states that the counter goes through before recycling.** Adding flip flops will increase the MOD number.

- MOD number = 2^n, where n = number of flip-flops.

Asynchronous (Ripple) Counters

- **MOD number is equal to the number of states that the counter goes through before recycling.** Adding flip flops will increase the MOD number.

- MOD number = 2^n, where n = number of flip-flops.

- The output of the last flip flop (MSB) divides the input clock frequency by the MOD number.

Frequency Division

- Each flip flop will have an output frequency of ½ the input.

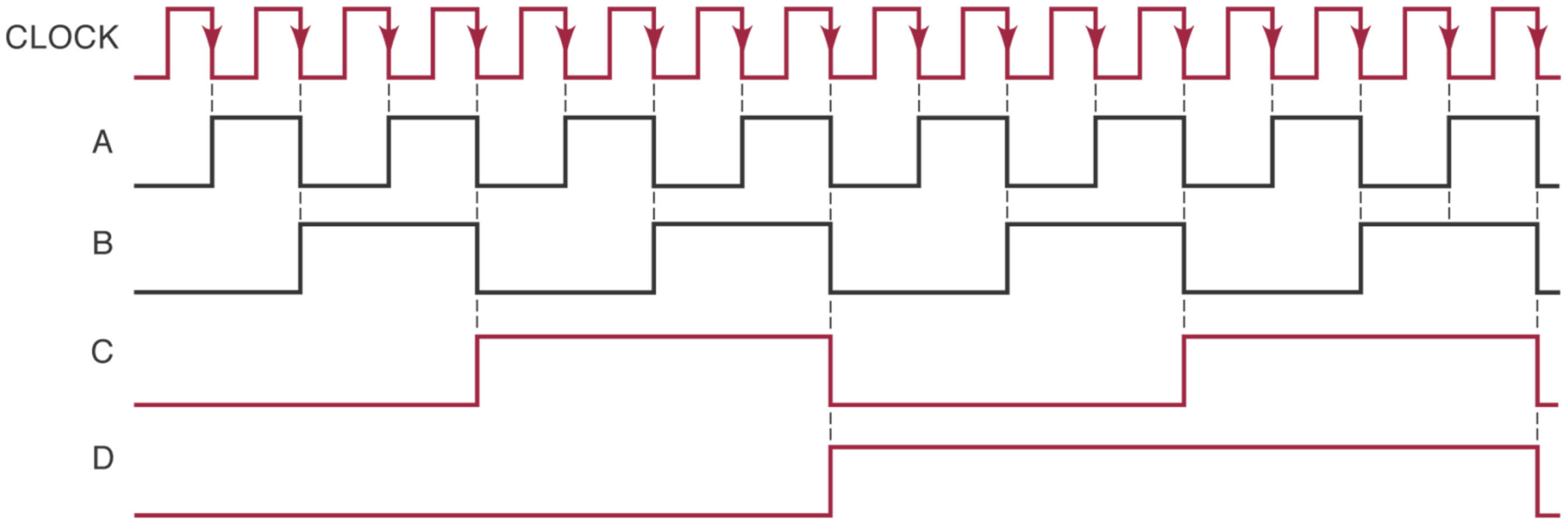
Frequency Division

- Each flip flop will have an output frequency of ½ the input.
- The output frequency of the last flip flop of any counter will be the clock frequency divided by the MOD of the counter.

Frequency Division

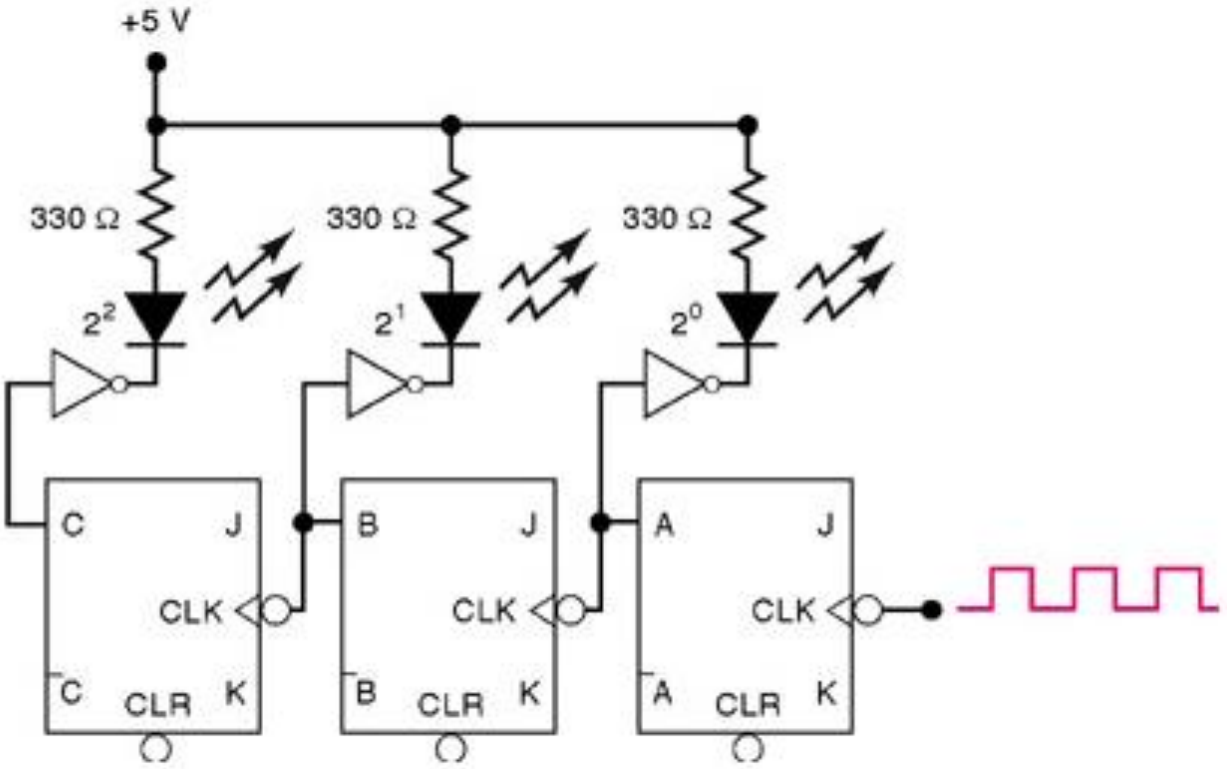
- Each flip flop will have an output frequency of ½ the input.
- The output frequency of the last flip flop of any counter will be the clock frequency divided by the MOD of the counter.

Timing Diagram



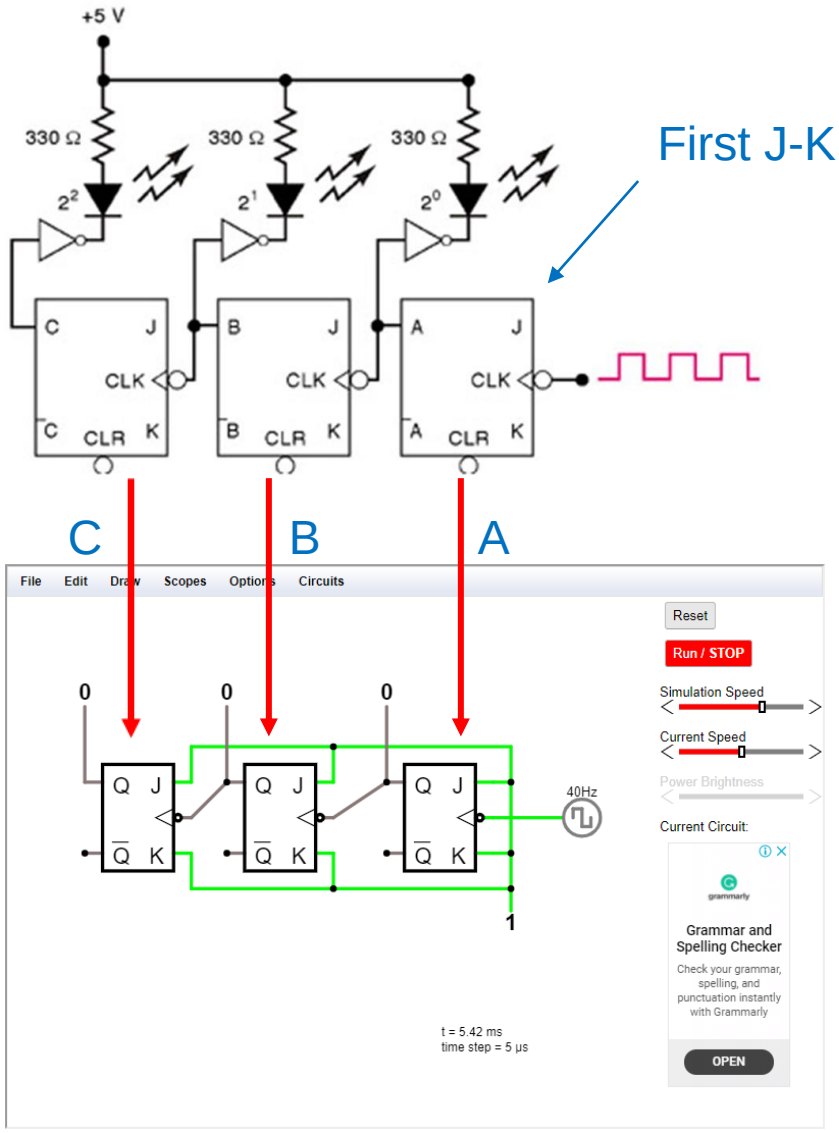
Displaying counter states

The arrangement below will result in an LED on when the output is high.



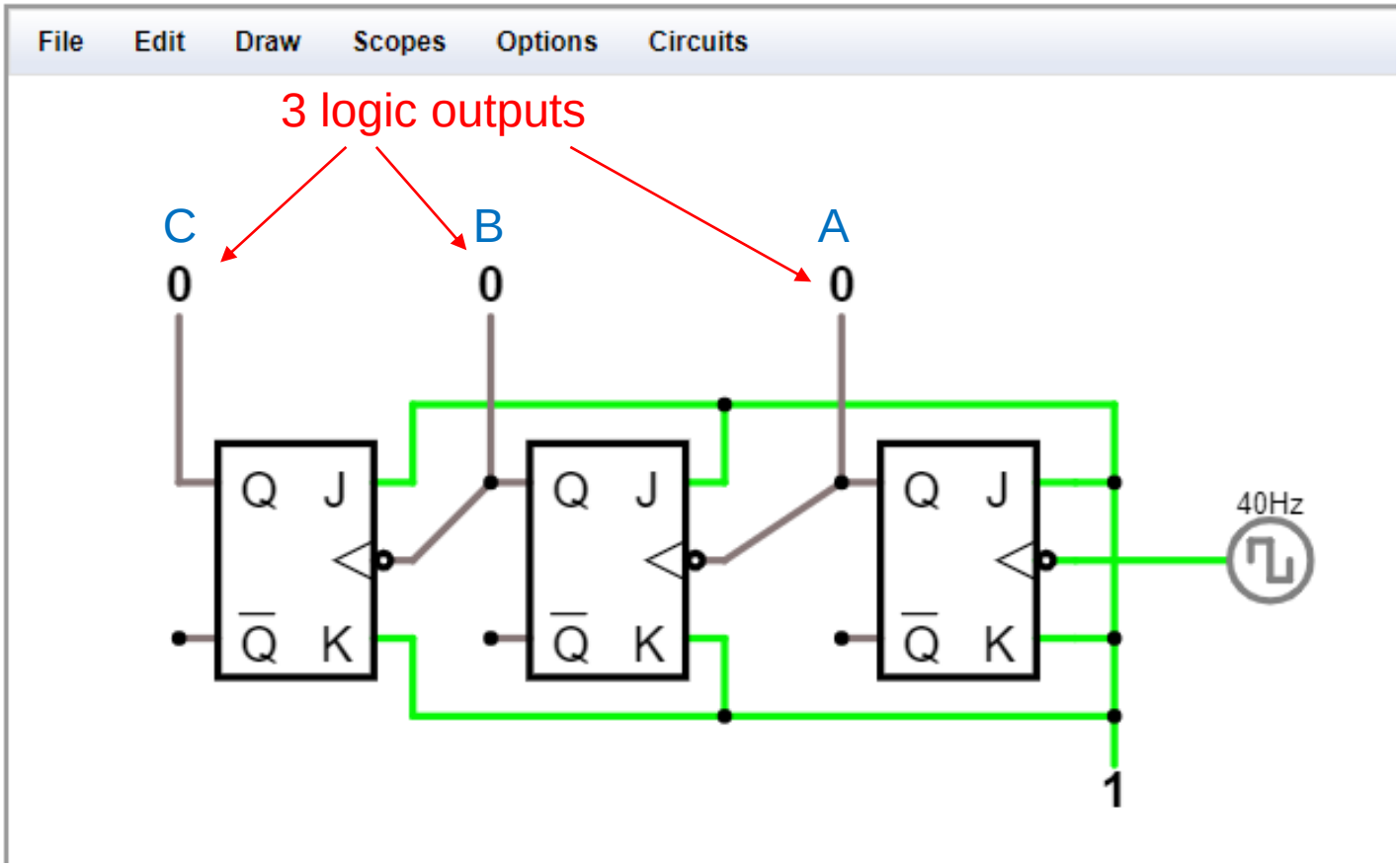
Displaying counter states

Using the circuit simulator to observe the counter in action



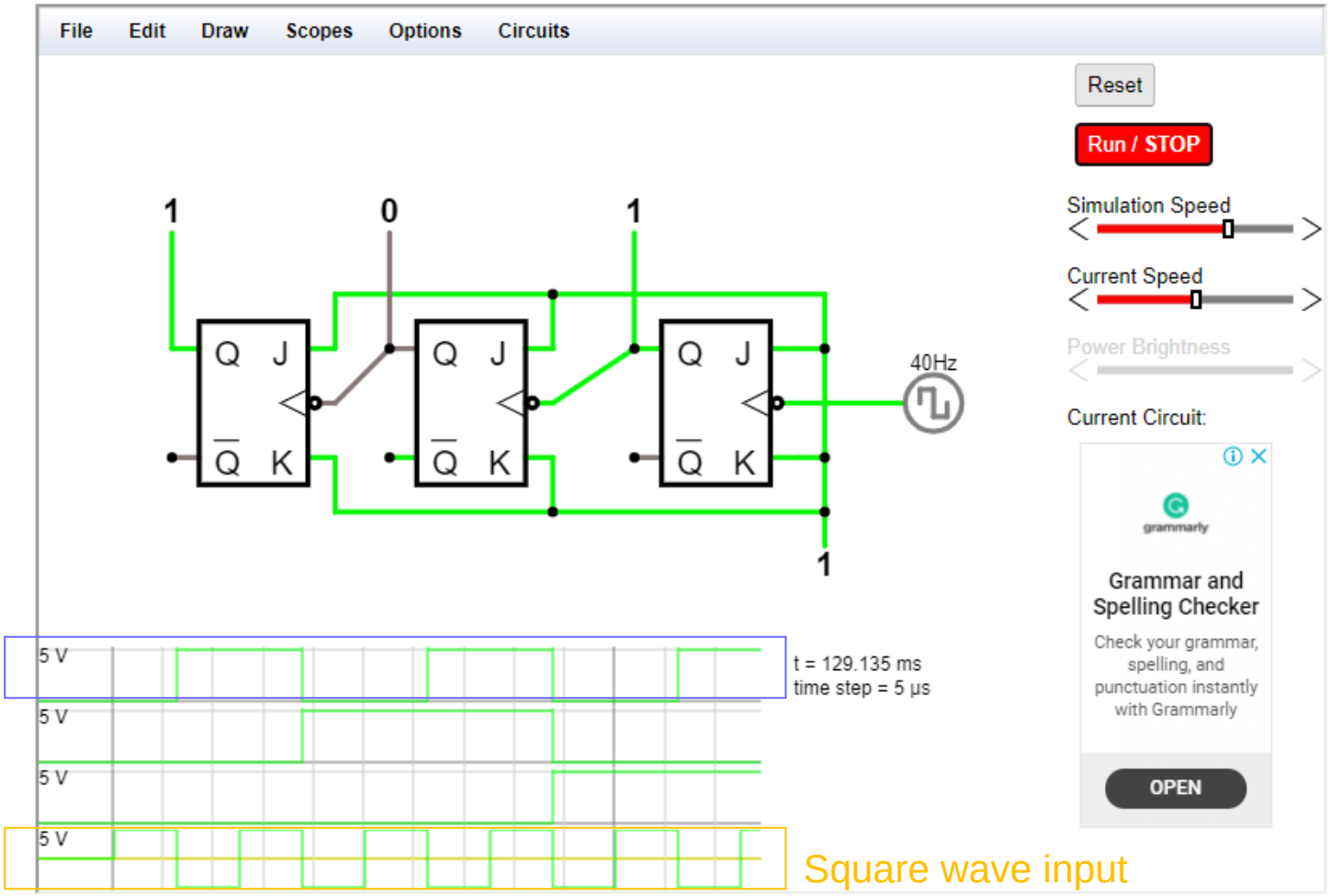
Displaying counter states

Using the circuit simulator to observe the counter in action



Displaying counter states

Using the circuit simulator to observe the counter in action

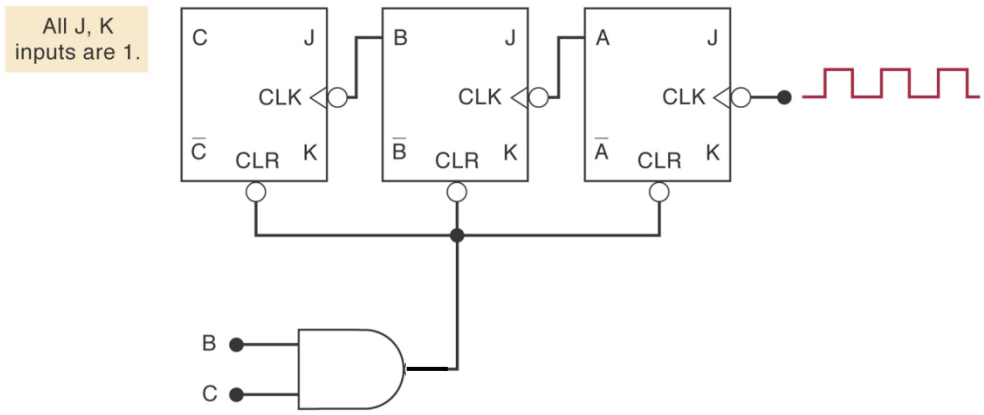


Counters with MOD Number $<2^N$

The MOD of a counter can be changed by designing the counter to reset on a specific counter state.

Counters with MOD Number <2^N

The MOD of a counter can be changed by designing the counter to reset on a specific counter state.

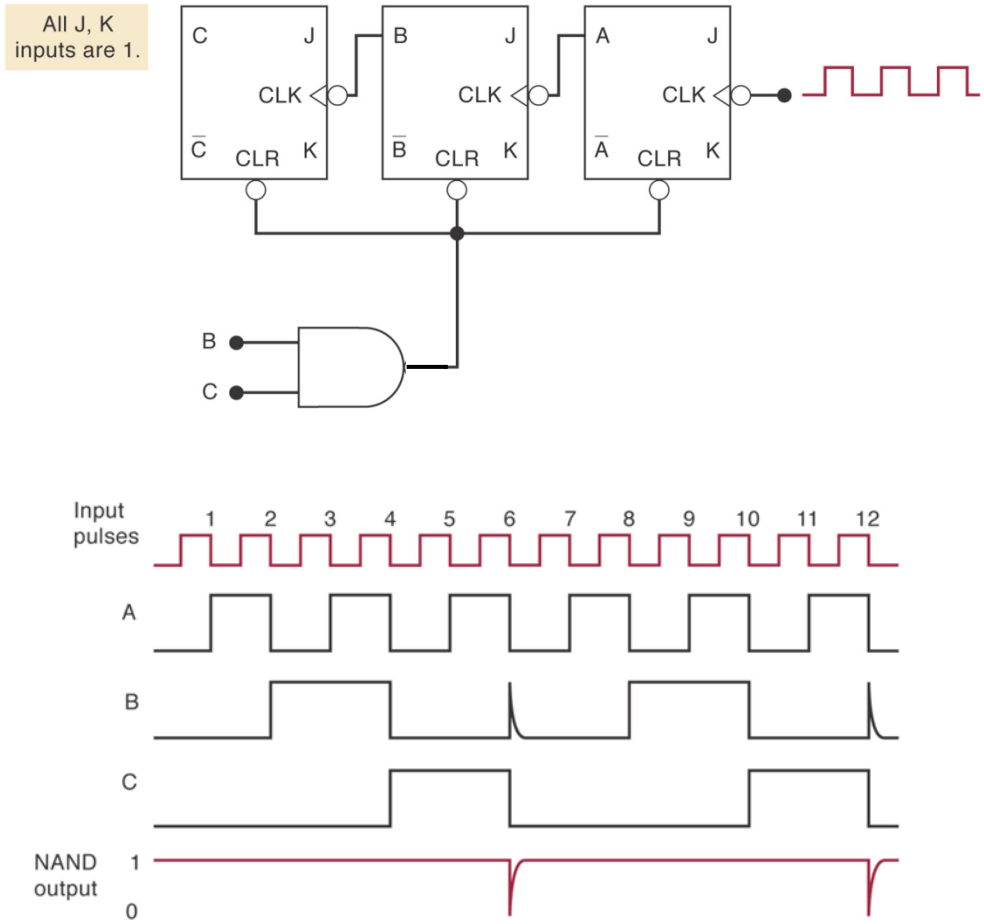


Turns MOD 8 counter into a MOD 6 counter

Counters with MOD Number $<2^N$

The MOD of a counter can be changed by designing the counter to reset on a specific counter state.

Turns MOD 8 counter into a MOD 6 counter



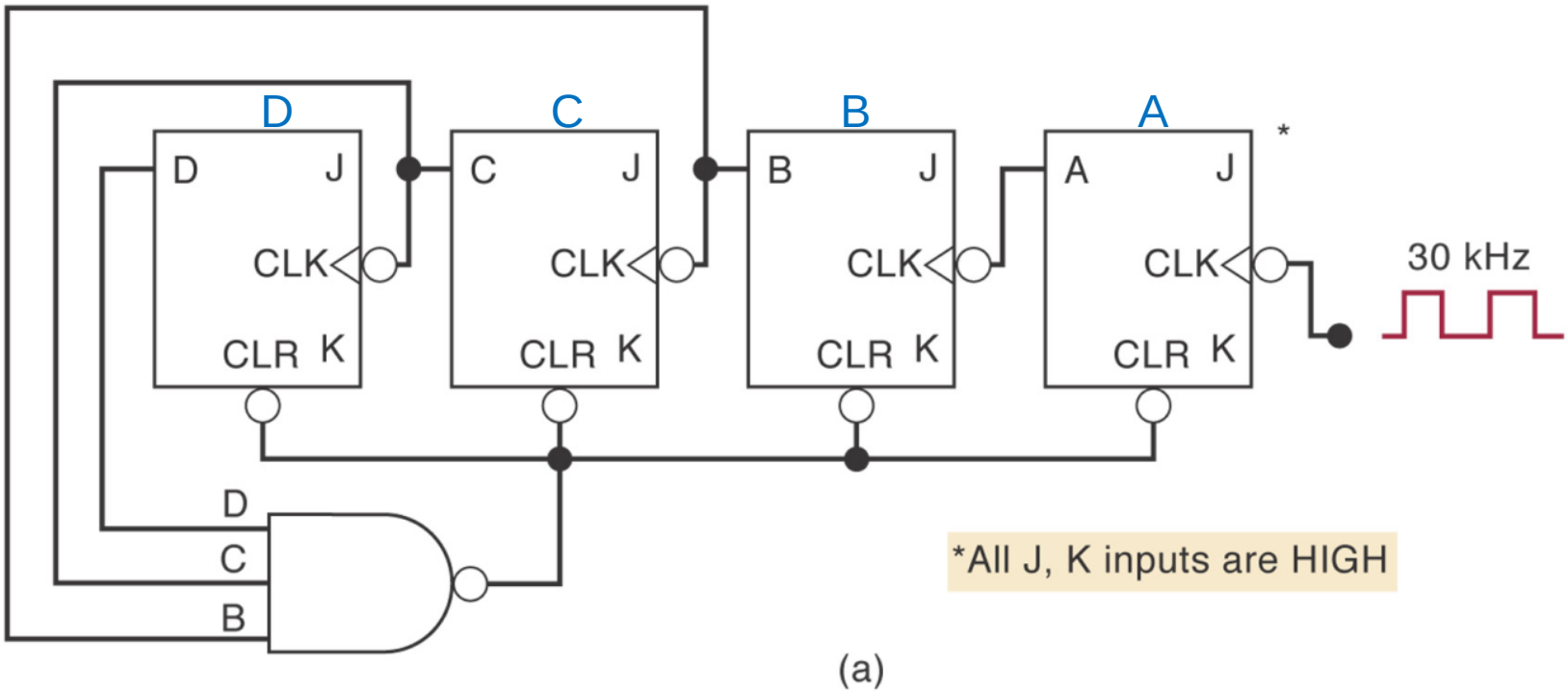
Changing the MOD number:

1. Find the smallest MOD required so that 2^N is greater than or equal to the requirement.

2. Connect a NAND gate to the asynchronous CLEAR inputs of all flip flops.

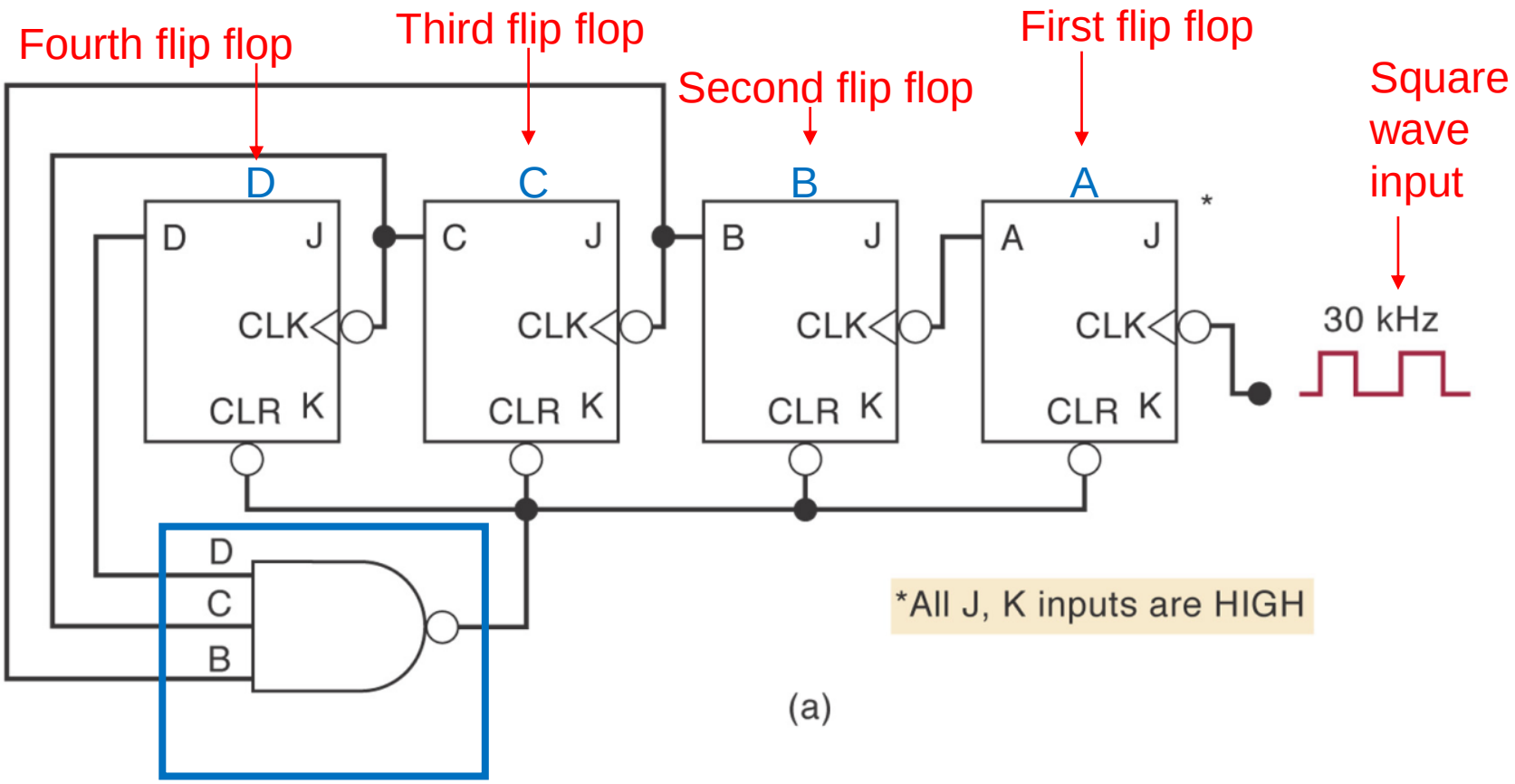
3. Determine which flip flops are HIGH at the first undesired count and connect the outputs of these flip flops to the NAND gate inputs.

What is the MOD number and frequency at O/P D for counter below?

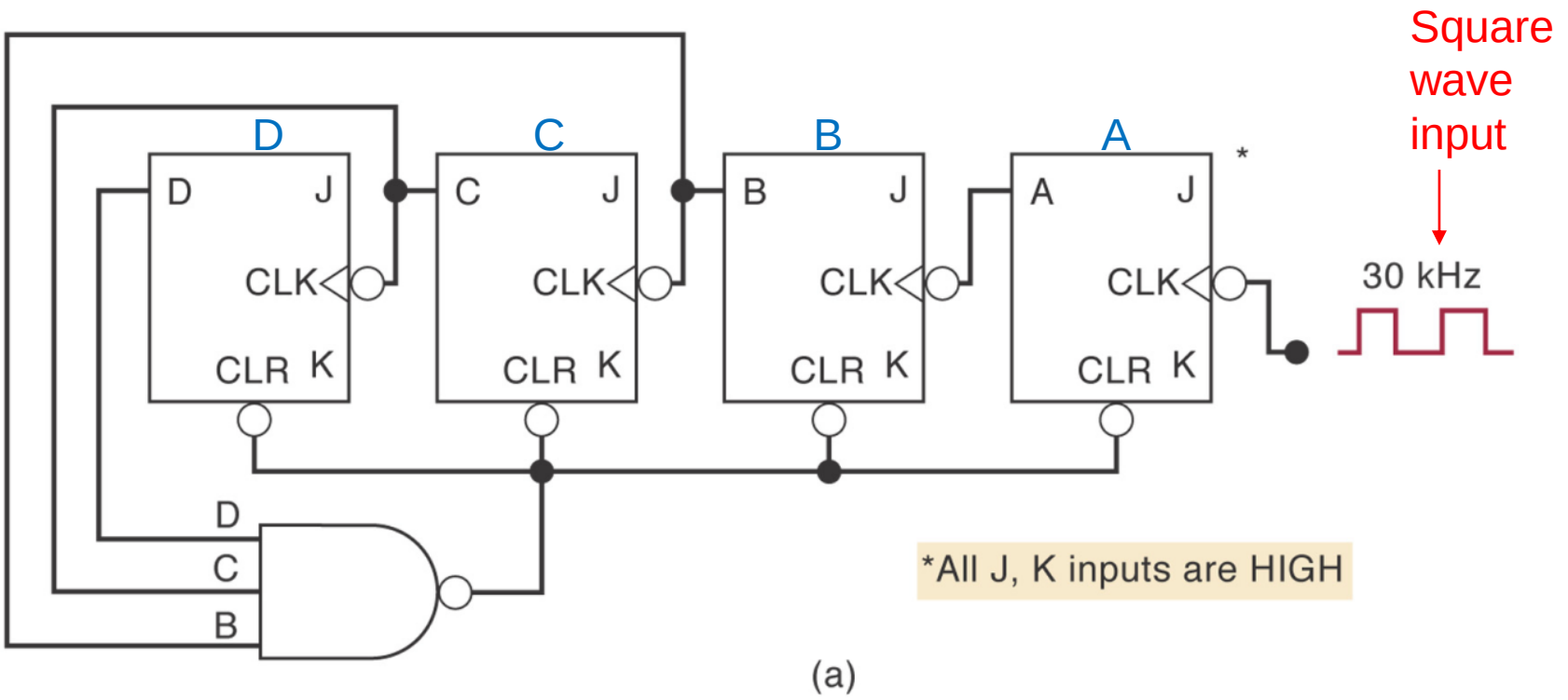


(a)

What is the MOD number and frequency at O/P D for counter below?



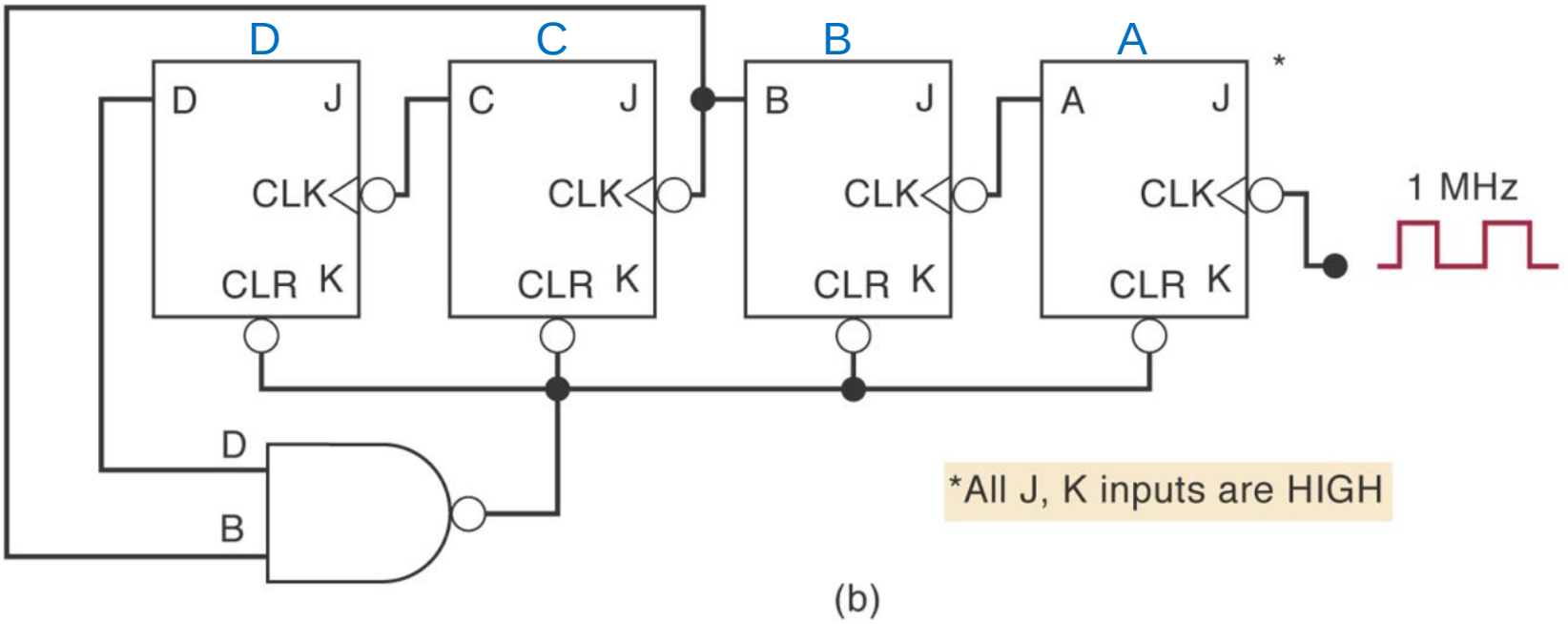
What is the MOD number and frequency at O/P D for counter below?



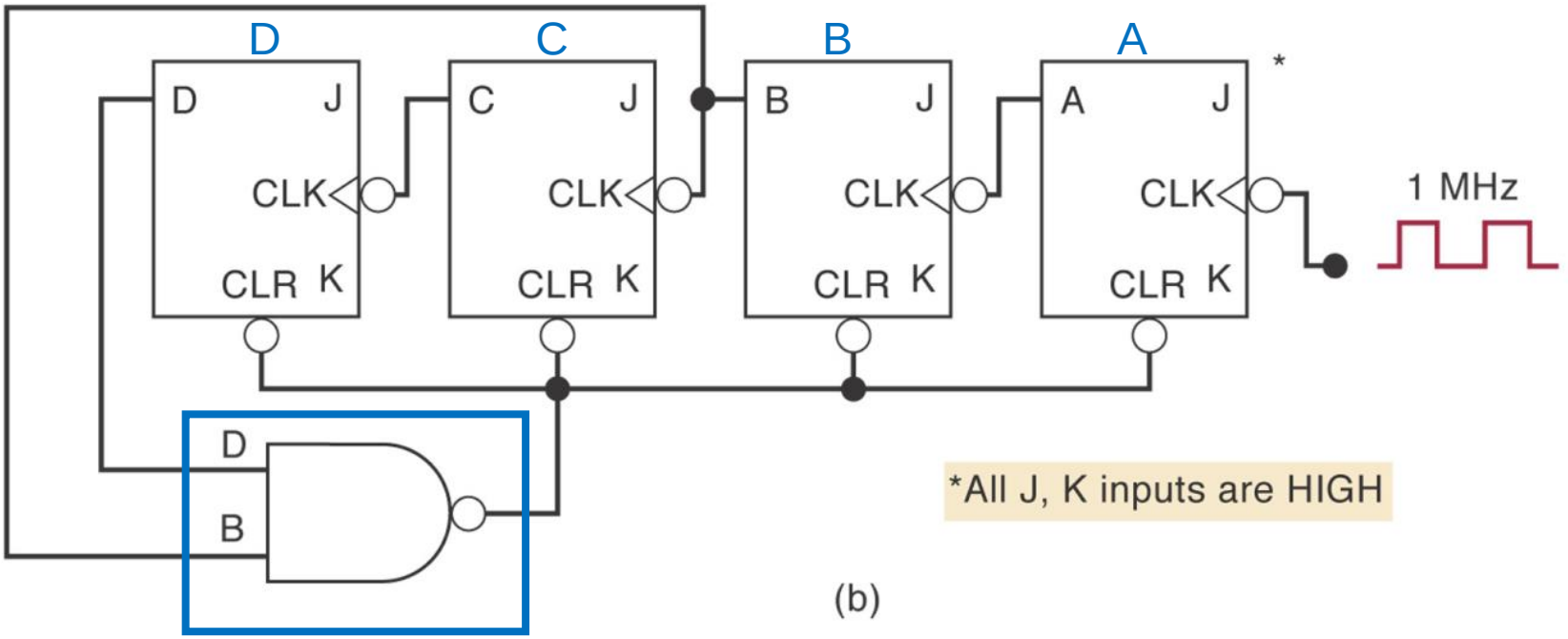
Reset on 1110 (14_{10}) therefore it is a MOD 14 counter

Frequency = 30kHz/14 = 2.14 kHz

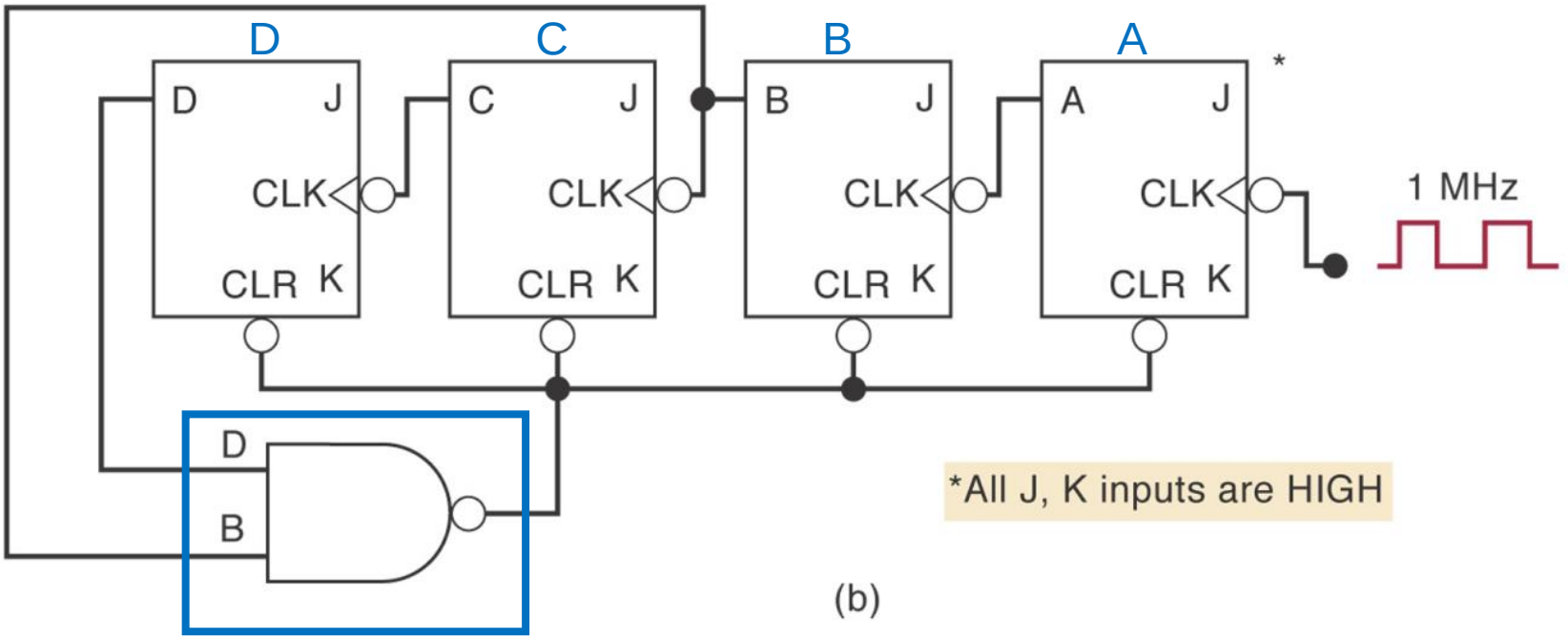
What is the MOD number and frequency at O/P D for counter below?



What is the MOD number and frequency at O/P D for counter below?



What is the MOD number and frequency at O/P D for counter below?



Reset on 1010 (10_{10}) therefore it is a MOD 10 counter

Frequency = 1 MHz/10 = 100 kHz

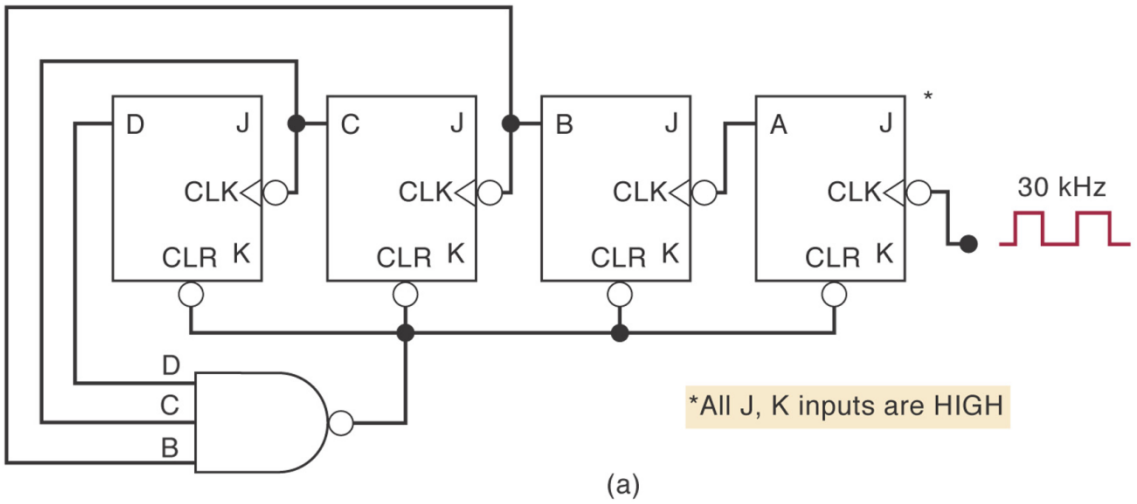
MOD number and frequency @ D for counters below?

Reset on 1110 (14_{10})

Thus MOD 14 cntr*

F = 30kHz/14

= 2.14 kHz



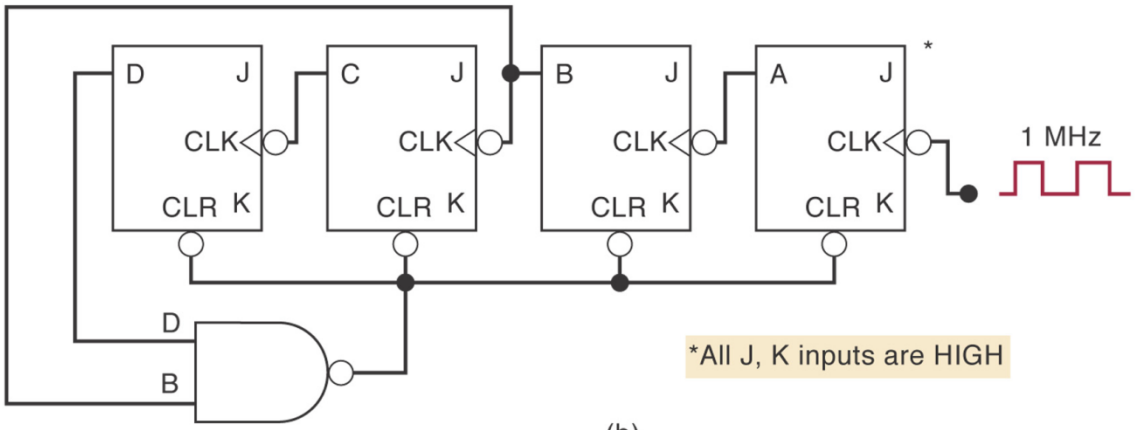
(a)

Reset on 1010 (10_{10})

Thus MOD 10 cntr*

F = 1 MHz/10

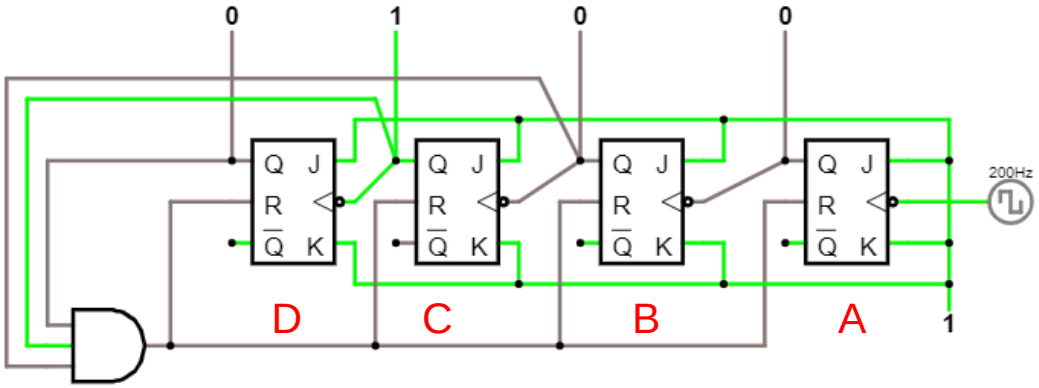
= 100 kHz



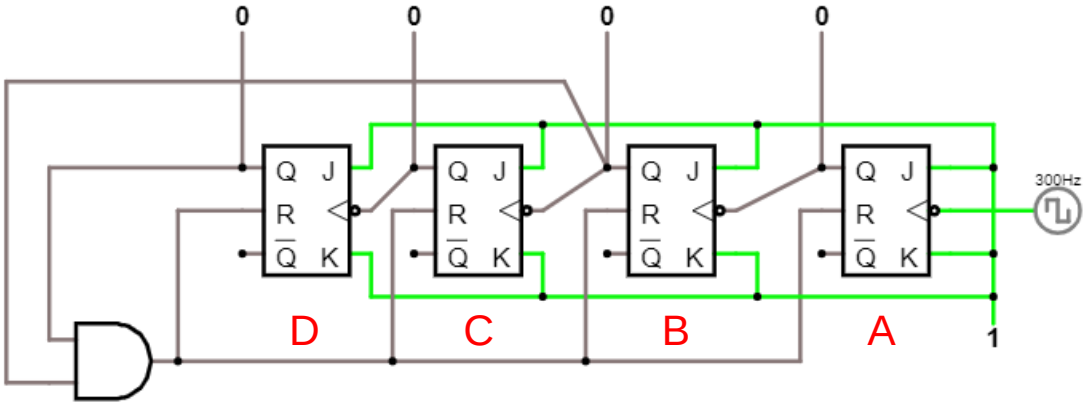
(b)

* The reset state is not a counting state

Using the circuit simulator:



All J, K and inputs are HIGH

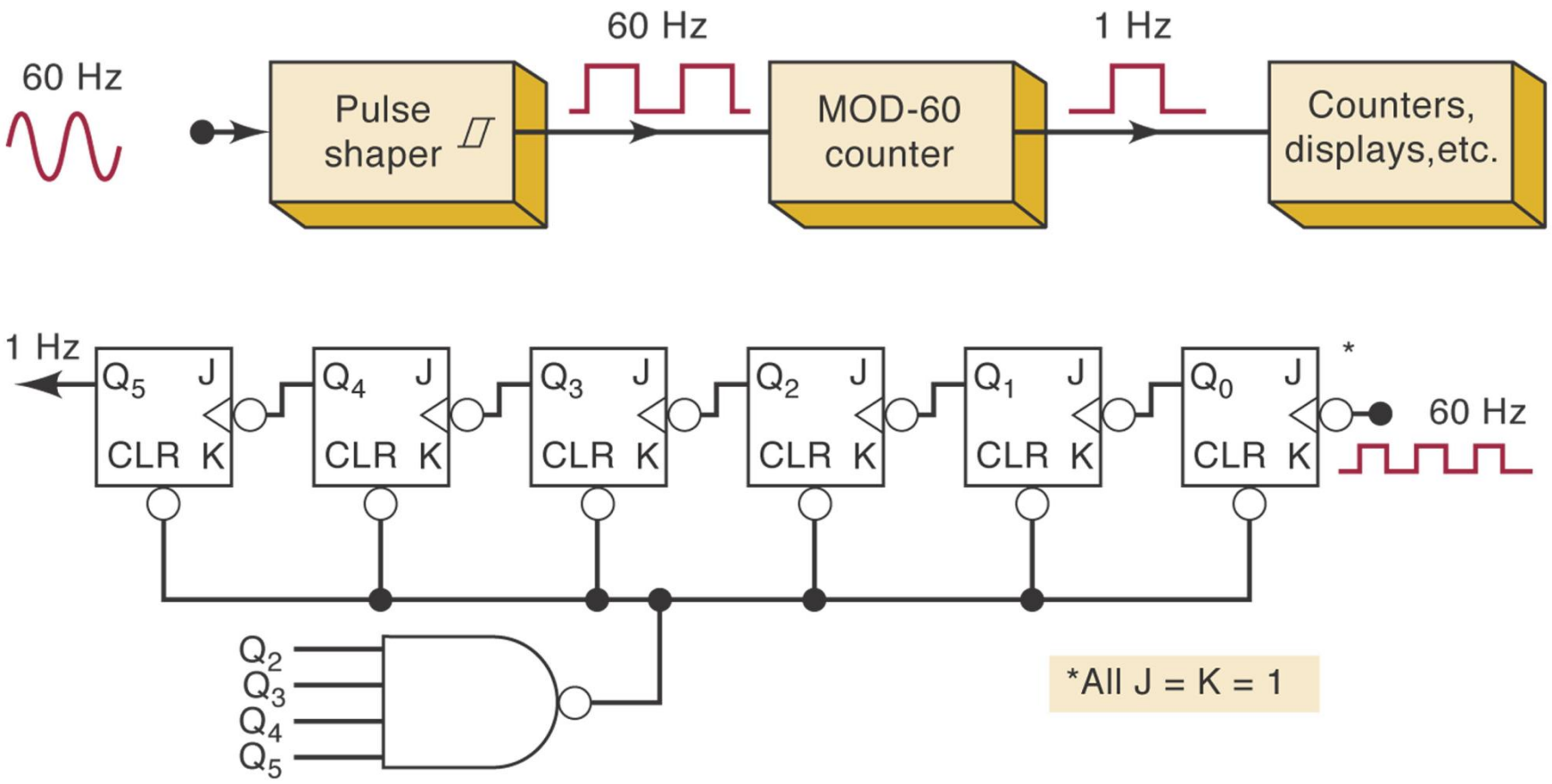


Examples of counters with MOD Number $<2^N$

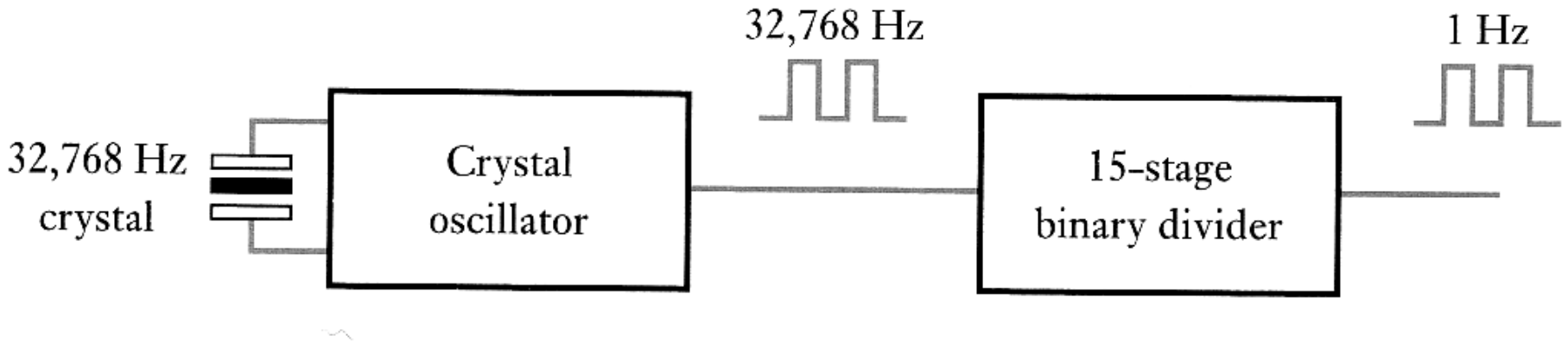
1. Decade counters or BCD counters
   - A decade counter is any counter with 10 distinct states, regardless of the sequence. Any MOD-10 counter is a decade counter.
   - A BCD counter is a decade counter that counts from binary 0000 to 1001.

Decade counters are widely used for counting events and displaying results in decimal form.

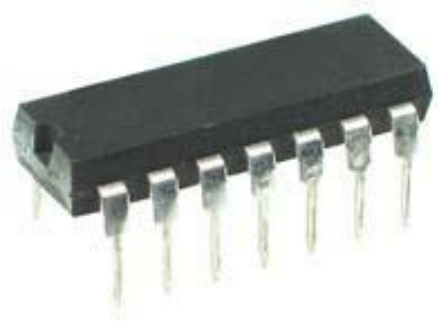
2. Designing a MOD 60 Counter

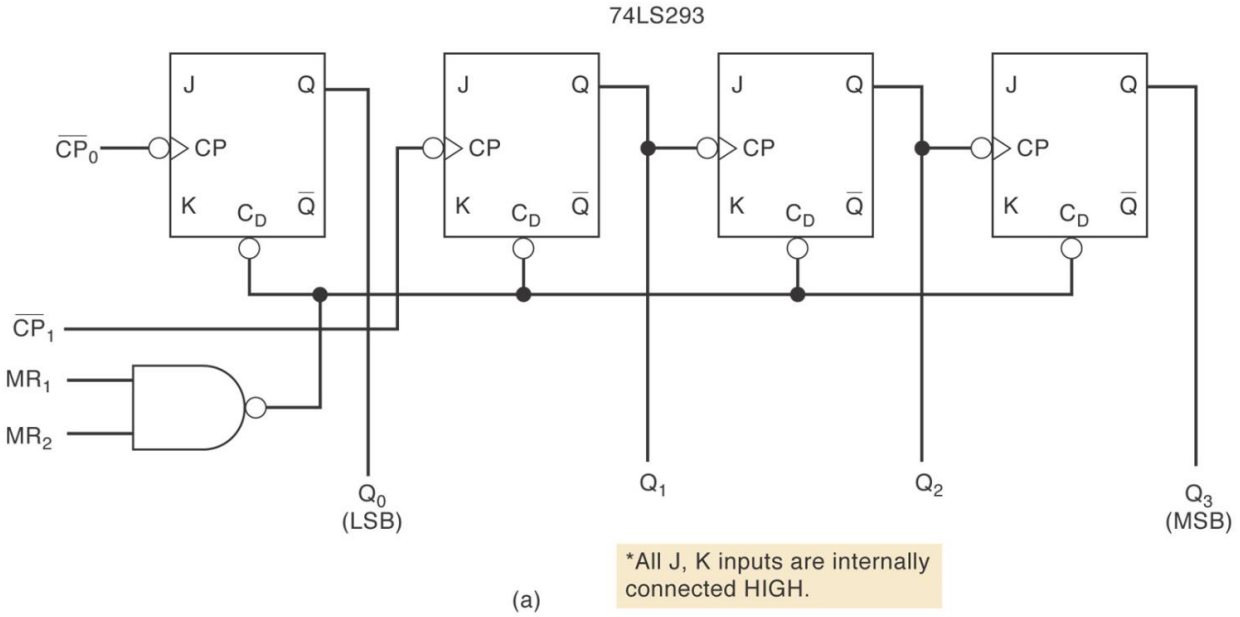


3. Your electronic watch will most likely contain a quartz crystal resonator oscillating at a frequency of 32 768 Hz ?

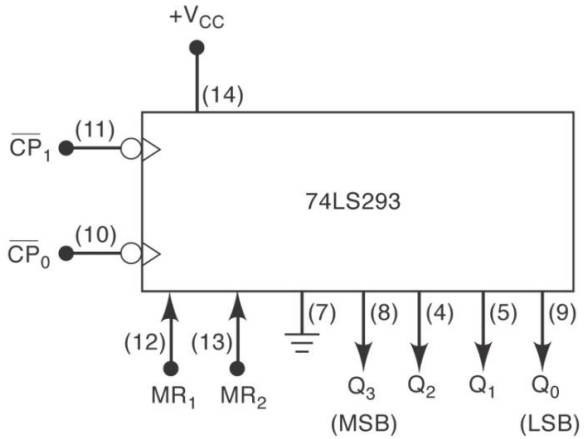


An IC asynchronous counter – the 74LS293





(a)



(b)

An IC asynchronous counter – the 74LS293

MOTOROLA

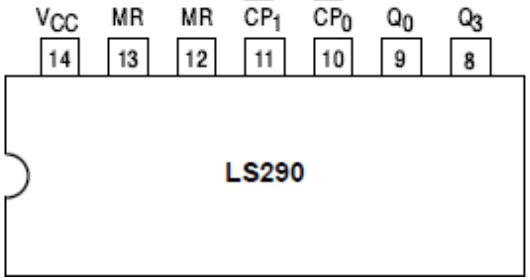
SN54/74LS290
SN54/74LS293

DECADE COUNTER; 4-BIT BINARY COUNTER

The SN54/74LS290 and SN54/74LS293 are high-speed 4-bit ripple type counters partitioned into two sections. Each counter has a divide-by-two section and either a divide-by-five (LS290) or divide-by-eight (LS293) section which are triggered by a HIGH-to-LOW transition on the clock inputs. Each section can be used separately or tied together (Q to CP)to form BCD, Bi-quinary, or Modulo-16 counters. Both of the counters have a 2-input gated Master Reset (Clear), and the LS290 also has a 2-input gated Master Set (Preset 9).

- Corner Power Pin Versions of the LS90 and LS93
- Low Power Consumption . . . Typically 45 mW
- High Count Rates . . . Typically 42 MHz
- Choice of Counting Modes . . . BCD, Bi-Quinary, Binary
- Input Clamp Diodes Limit High Speed Termination Effects

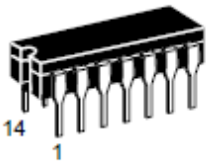
CONNECTION DIAGRAM DIP (TOP VIEW)



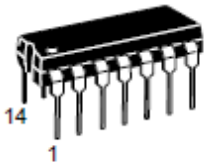
NOTE:
The Flatpak version has the same pinouts (Connection Diagram) as the Dual In-Line Package.

DECADE COUNTER; 4-BIT BINARY COUNTER

LOW POWER SCHOTTKY



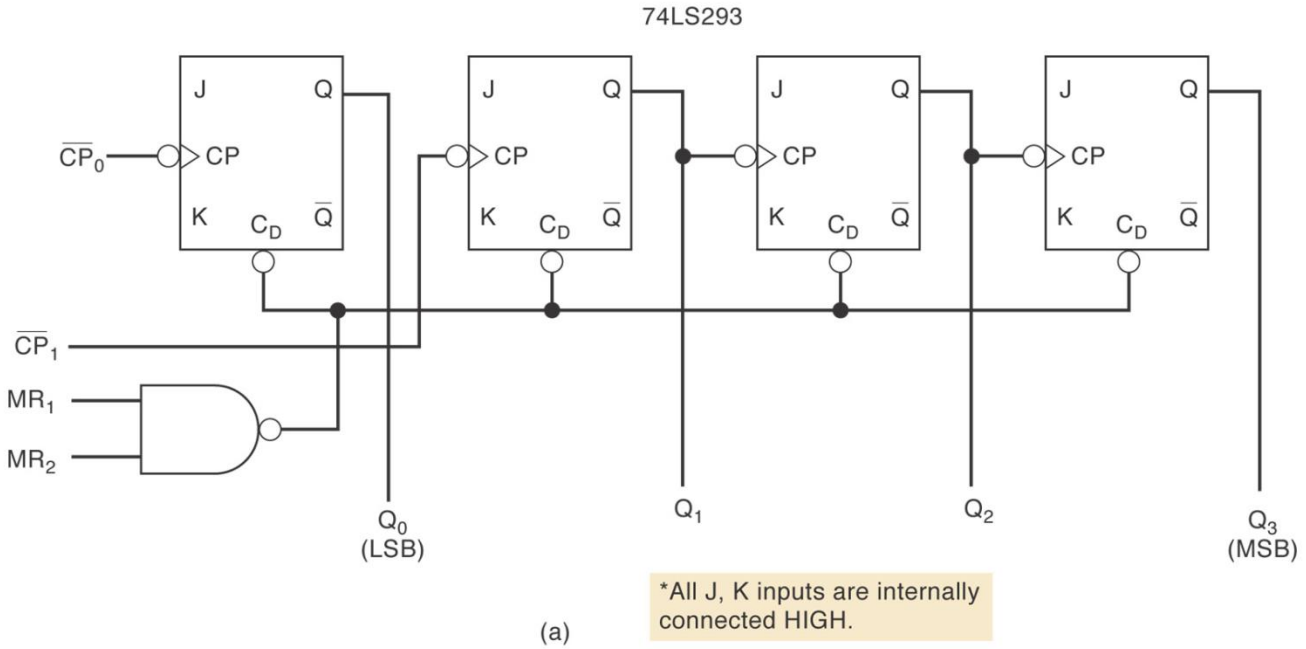
J SUFFIX
CERAMIC
CASE 632-08



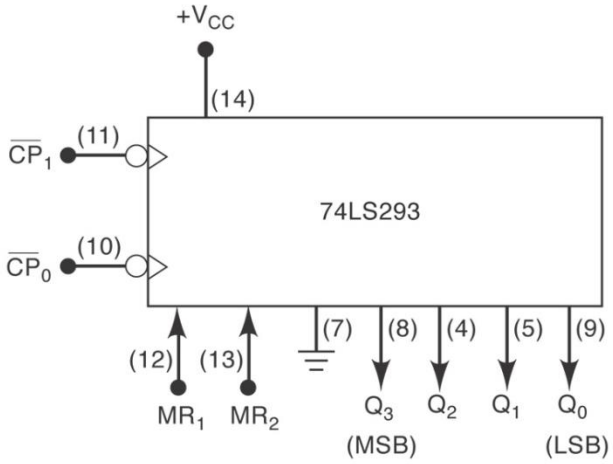
N SUFFIX
PLASTIC
CASE 646-06

D SUFFIX
SOIC

An IC asynchronous counter – the 74LS293



(a)



(b)

An IC asynchronous counter – the 74LS293

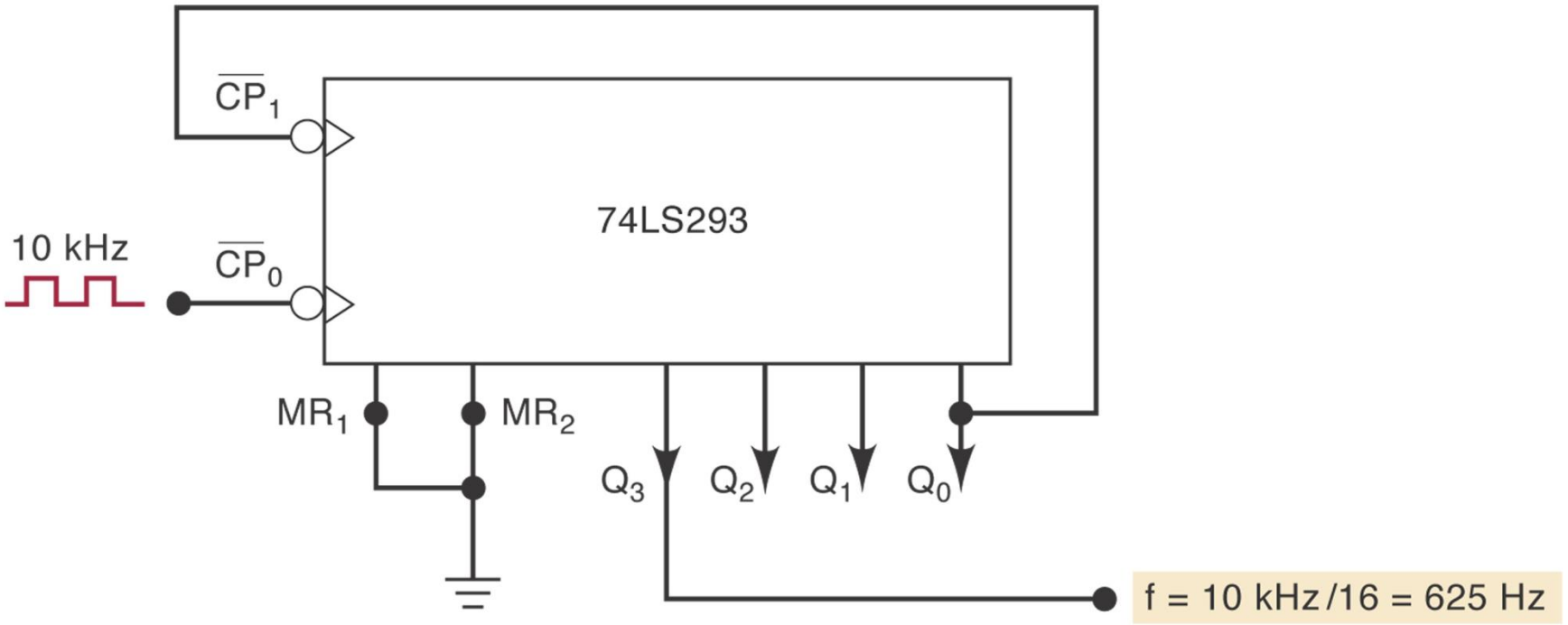
- 4 JK flip flop (FFs)

– Each FF has NGT CP (CLK) input

– Each FF has active low asynchronous CLEAR inputs (MR_1 and MR_2 through NAND gate). Both MR I/P's must be HI to clear the counter to 0000.

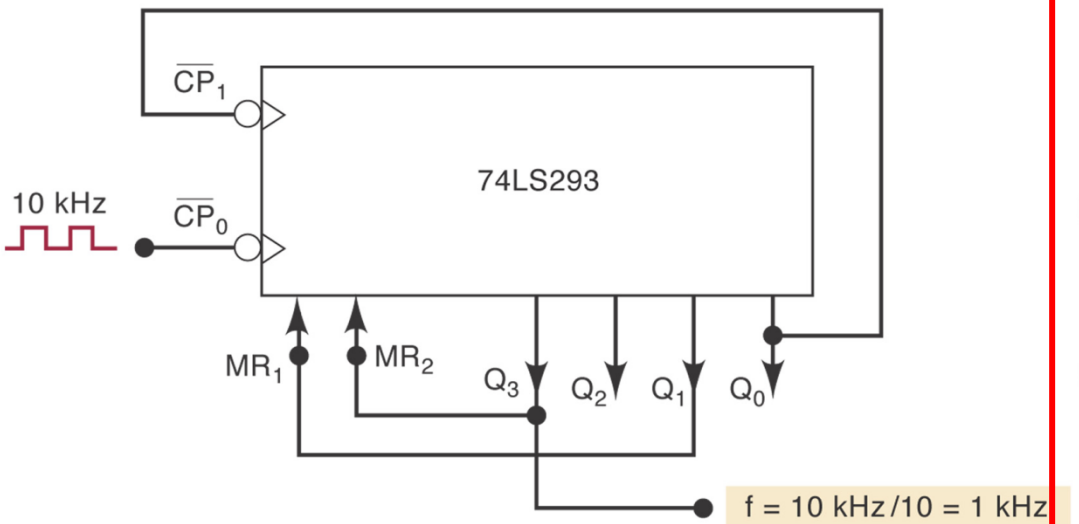
– FF's Q_1, Q_2, and Q_3 are connected as a 3 bit ripple counter. Q_0 is not connected internally, which allows flexibility in design.

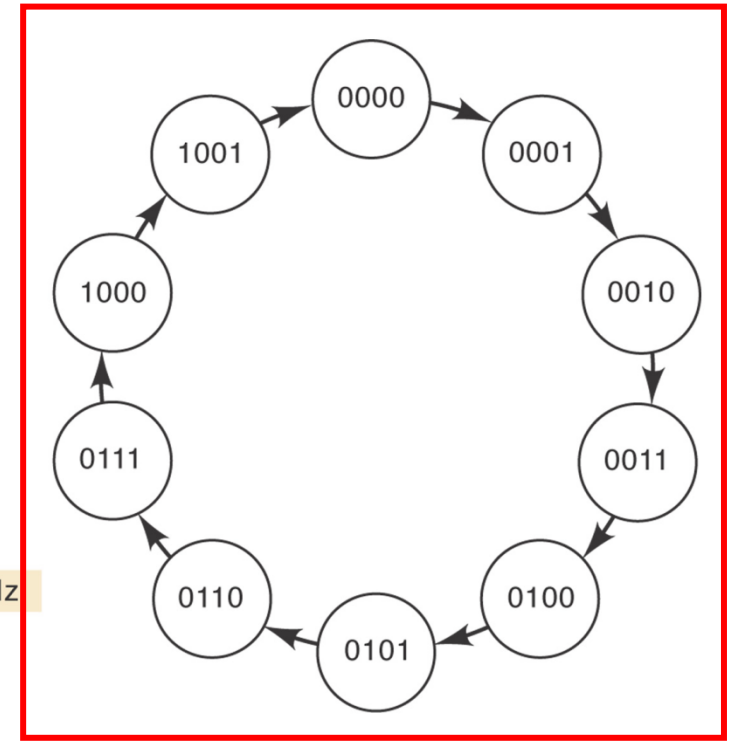
74LS293 as a MOD 16 counter



74LS293 as a MOD 10 counter

Should reset on 1010





74LS293 as a MOD 14 counter

Should reset on 1110