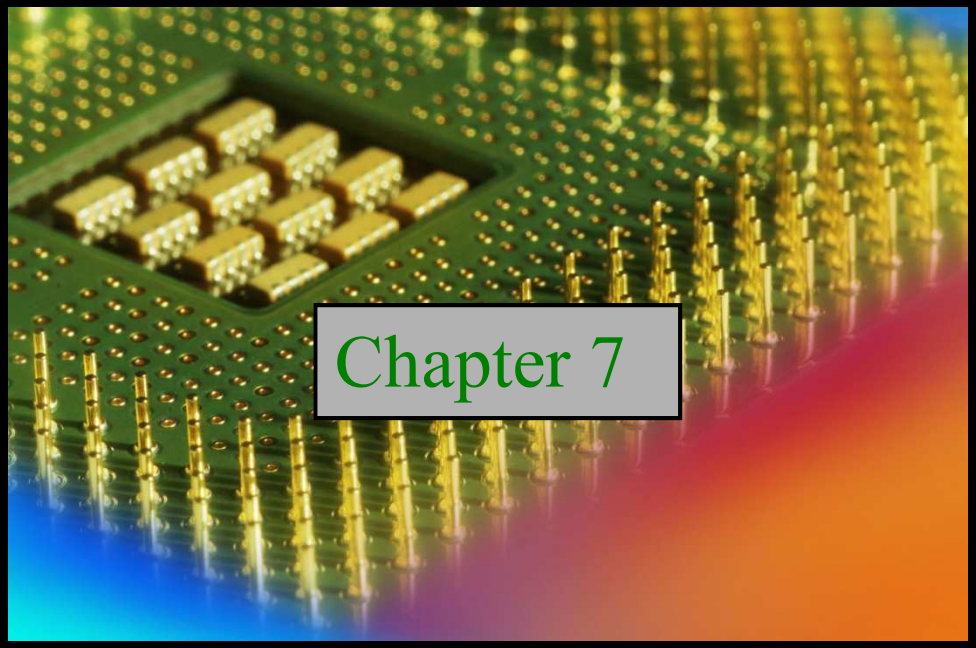
Digital Fundamentals

Tenth Edition

Floyd



© Modified by Yuttapong Jiraraksopakun
ENE, KMUTT 2009

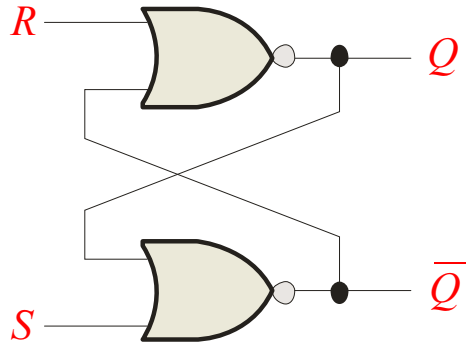
© 2008 Pearson Education

Summary

Latches

A **latch** is a temporary storage device that has two stable states (bistable). It is a basic form of memory.

The S-R (Set-Reset) latch is the most basic type. It can be constructed from NOR gates or NAND gates. With NOR gates, the latch responds to active-HIGH inputs; with NAND gates, it responds to active-LOW inputs.



NOR Active-HIGH Latch

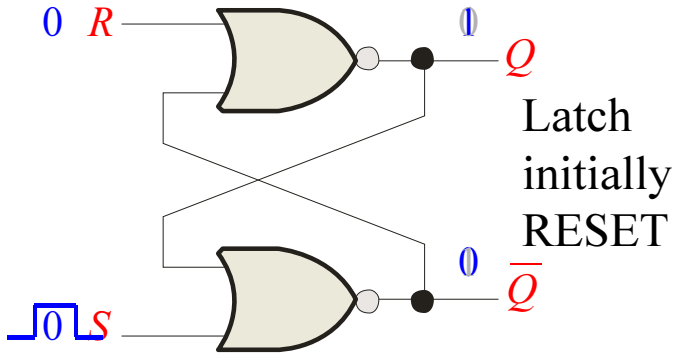
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Latches

The active-HIGH *S*-*R* latch is in a stable (latched) condition when both inputs are LOW.

Assume the latch is initially RESET ($Q = 0$) and the inputs are at their inactive level (0). To SET the latch ($Q = 1$), a momentary HIGH signal is applied to the *S* input while the *R* remains LOW.



To RESET the latch ($Q = 0$), a momentary HIGH signal is applied to the *R* input while the *S* remains LOW.

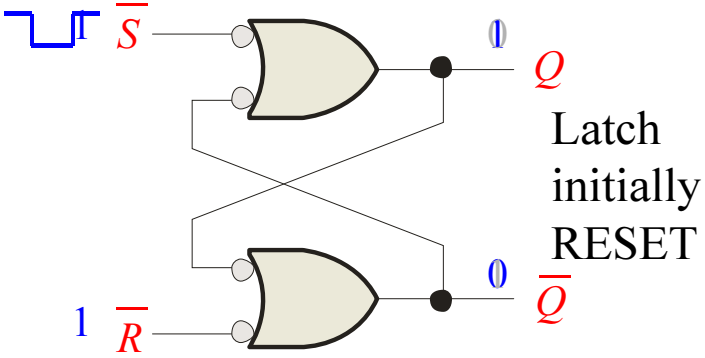
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Latches

The active-LOW $\overline{S}$-$\overline{R}$ latch is in a stable (latched) condition when both inputs are HIGH.

Assume the latch is initially RESET ($Q = 0$) and the inputs are at their inactive level (1). To SET the latch ($Q = 1$), a momentary LOW signal is applied to the $\overline{S}$ input while the $\overline{R}$ remains HIGH.



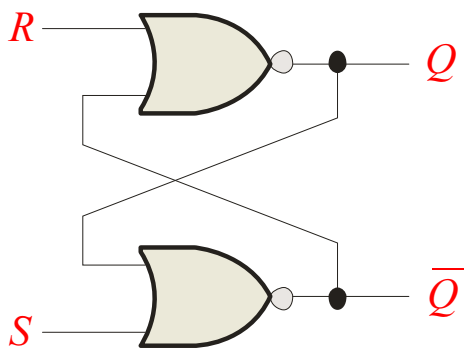
To RESET the latch a momentary LOW is applied to the $\overline{R}$ input while $\overline{S}$ is HIGH.

Never apply an active set and reset at the same time (invalid).

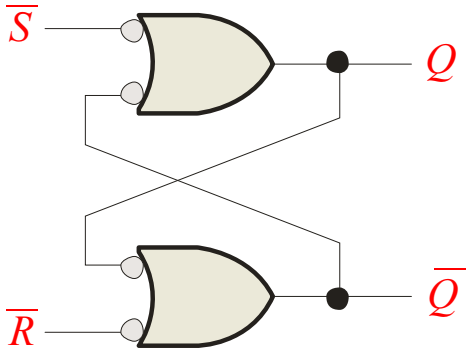
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Latches



NOR Active-HIGH Latch



NAND Active-LOW Latch

INPUTS		OUTPUTS		COMMENTS
$\overline{S}$	$\overline{R}$	Q	$\overline{Q}$	
1	1	NC	NC	No change. Latch remains in present state.
0	1	1	0	Latch SET.
1	0	0	1	Latch RESET.
0	0	1	1	Invalid condition

© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

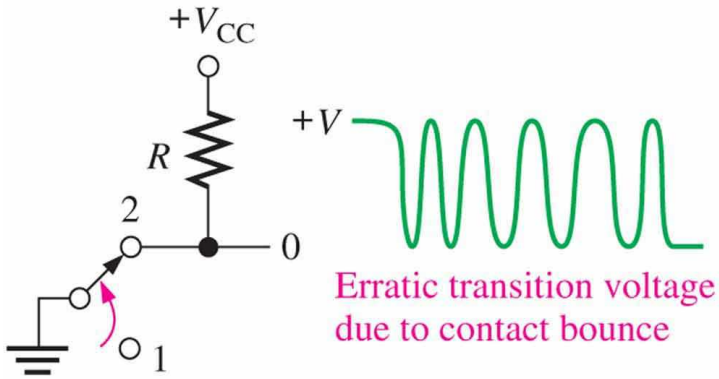
Summary

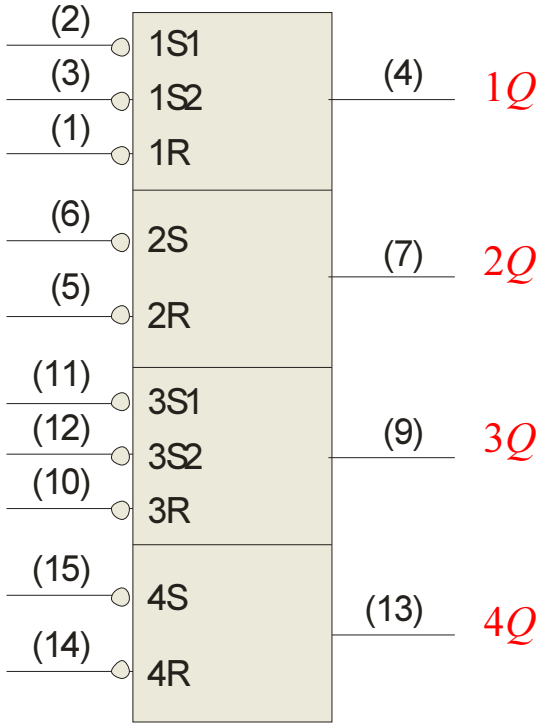
Latches

The active-LOW $\overline{S}$-$\overline{R}$ latch is available as the 74LS279A IC.

It features four internal latches with two having two $\overline{S}$ inputs. To SET any of the latches, the $\overline{S}$ line is pulsed low. It is available in several packages.

$\overline{S}$-$\overline{R}$ latches are frequently used for switch debounce circuits as shown:





74LS279A

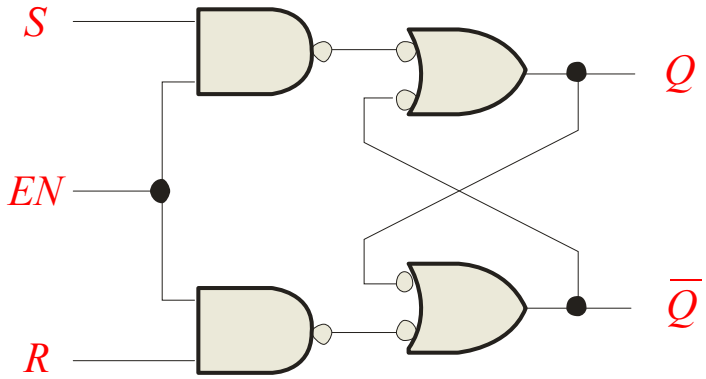
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Gated S-R Latches

A gated latch is a variation on the basic latch.

The gated latch has an additional input, called enable (*EN*) that must be HIGH in order for the latch to respond to the *S* and *R* inputs.

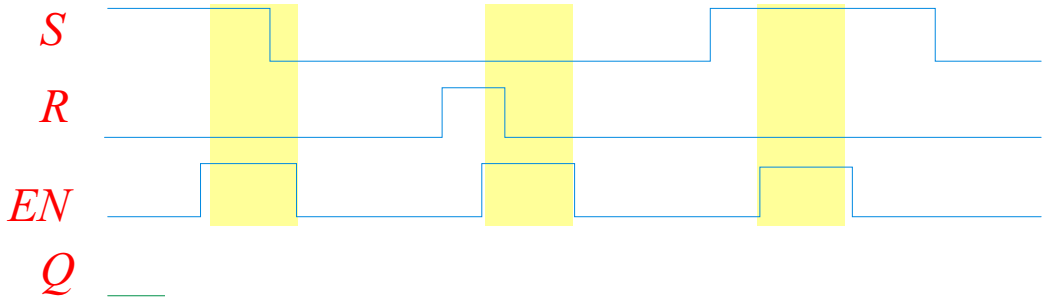


Example

Show the *Q* output with relation to the input signals. Assume *Q* starts LOW.

Solution

Keep in mind that *S* and *R* are only active when *EN* is HIGH.

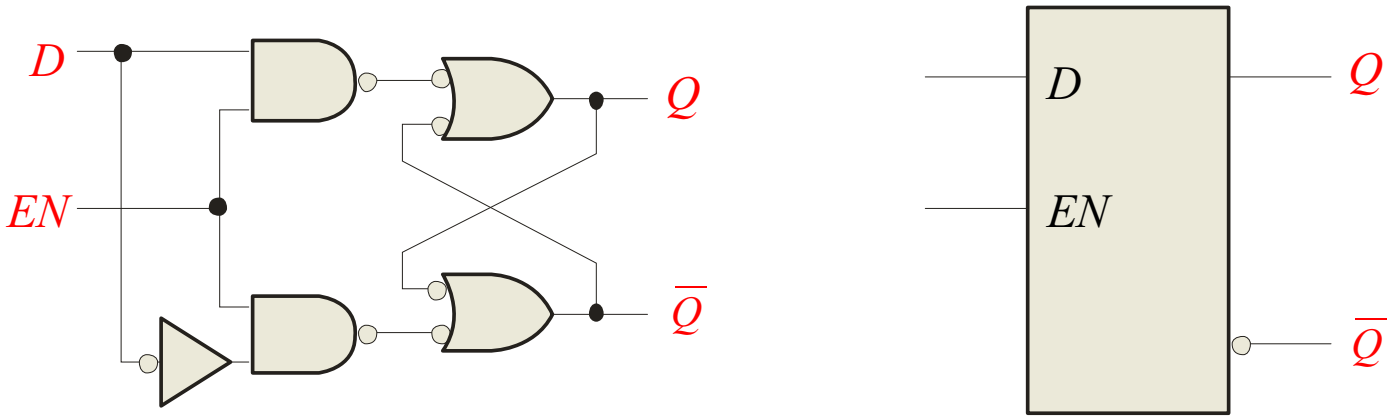


© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Gated D Latches

The *D* latch is an variation of the *S*-*R* latch but combines the *S* and *R* inputs into a single *D* input as shown:



A simple rule for the *D* latch is:

Q follows *D* when the Enable is active.

© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Gated D Latches

The truth table for the *D* latch summarizes its operation. If *EN* is LOW, then there is no change in the output and it is latched.

Inputs		Outputs		
D	*EN*	*Q*	$\overline{Q}$	Comments
0	1	0	1	RESET
1	1	1	0	SET
X	0	Q_0	$\overline{Q}_0$	No change

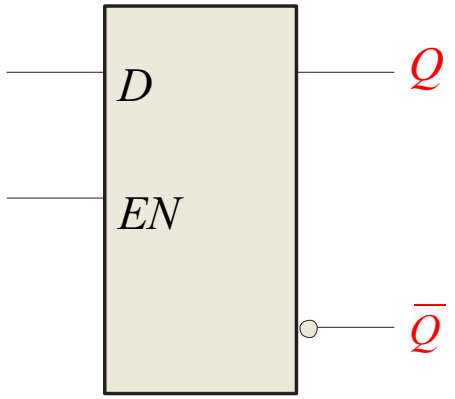
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

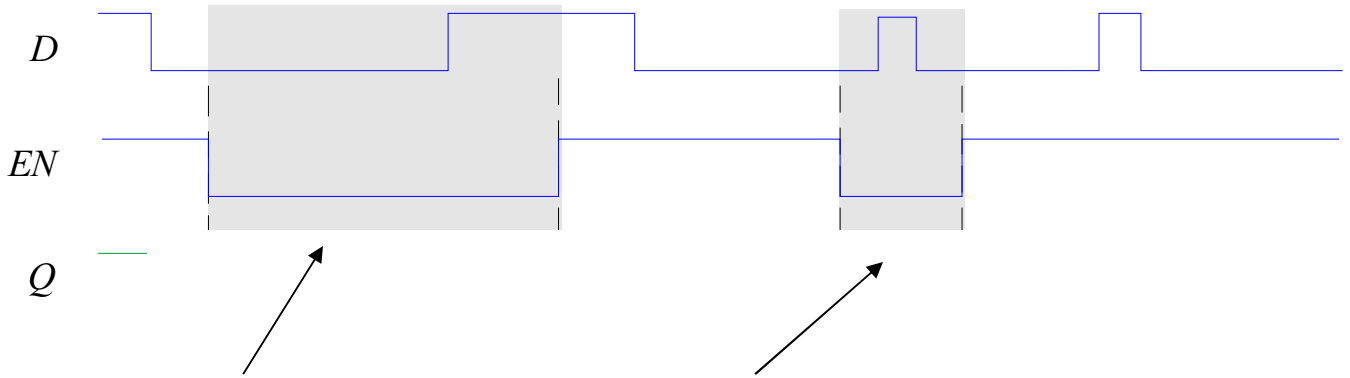
Summary

Gated D Latches

Example

Determine the Q output for the D latch, given the inputs shown.





Notice that the Enable is not active during these times, so the output is latched.

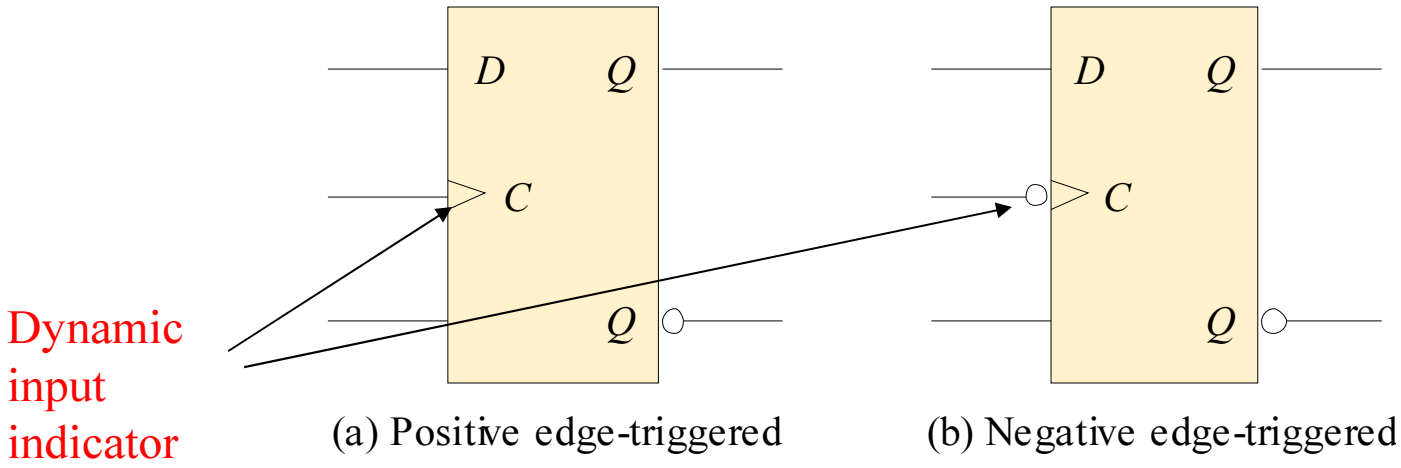
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Flip-flops

A flip-flop differs from a latch in the manner it changes states. A flip-flop is a clocked device, in which only the clock edge determines when a new bit is entered.

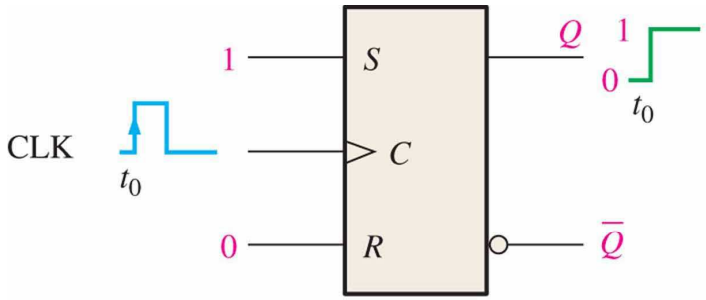
The active edge can be positive or negative.



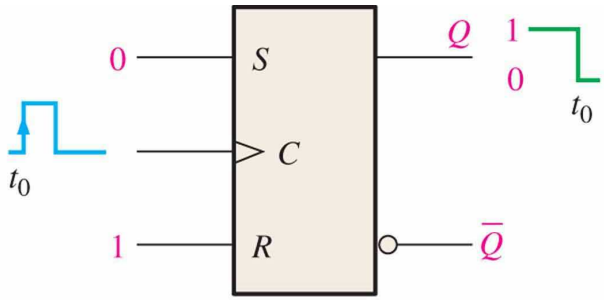
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

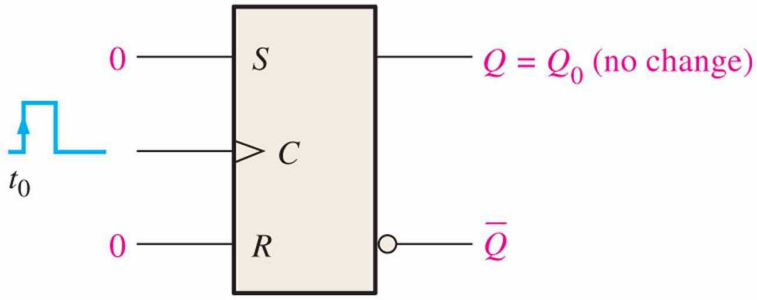
S-R Flip-flops



(a) $S = 1$, $R = 0$ flip-flop SETS on positive clock edge. (If already SET, it remains SET.)



(b) $S = 0$, $R = 1$ flip-flop RESETS on positive clock edge. (If already RESET, it remains RESET.)

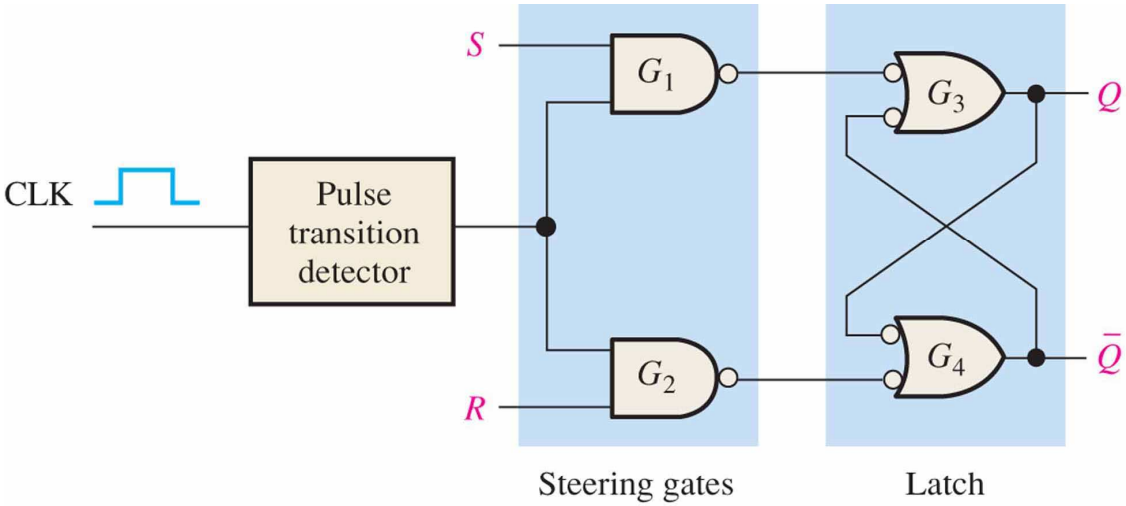


(c) $S = 0$, $R = 0$ flip-flop does not change. (If SET, it remains SET; if RESET, it remains RESET.)

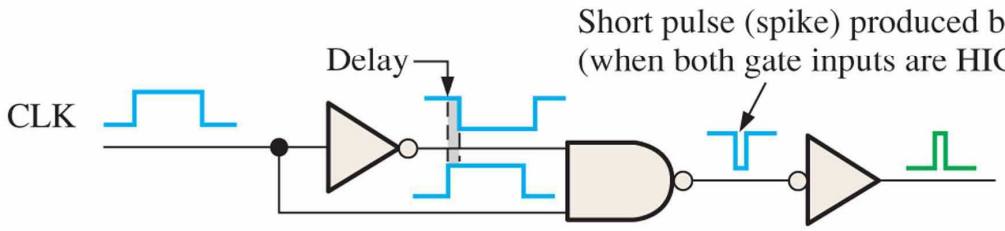
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

S-R Flip-flops



(a) A simplified logic diagram for a positive edge-triggered S-R flip-flop

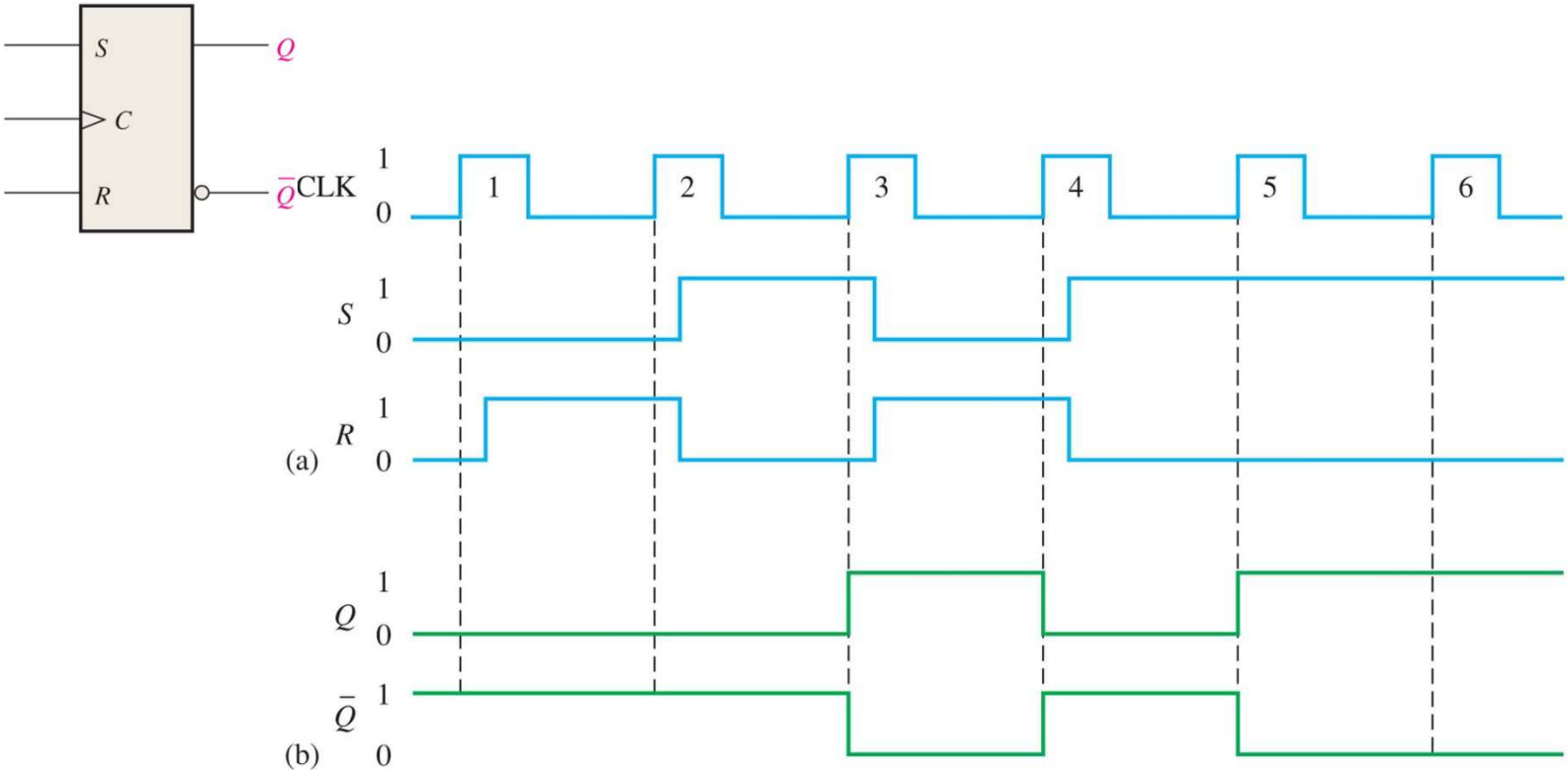


(b) A type of pulse transition detector

© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

S-R Flip-flops



© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

D Flip-flops

The truth table for a positive-edge triggered D flip-flop shows an up arrow to remind you that it is sensitive to its *D* input only on the rising edge of the clock; otherwise it is latched. The truth table for a negative-edge triggered D flip-flop is identical except for the direction of the arrow.

Inputs		Outputs		
D	CLK	*Q*	$\overline{Q}$	Comments
1	↑	1	0	SET
0	↑	0	1	RESET

(a) Positive-edge triggered

Inputs		Outputs		
D	CLK	*Q*	$\overline{Q}$	Comments
1	↓	1	0	SET
0	↓	0	1	RESET

(b) Negative-edge triggered

© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

J-K Flip-flops

The J-K flip-flop is more versatile than the D flip flop. In addition to the clock input, it has two inputs, labeled *J* and *K*. When both *J* and *K* = 1, the output changes states (toggles) on the active clock edge (in this case, the rising edge).

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

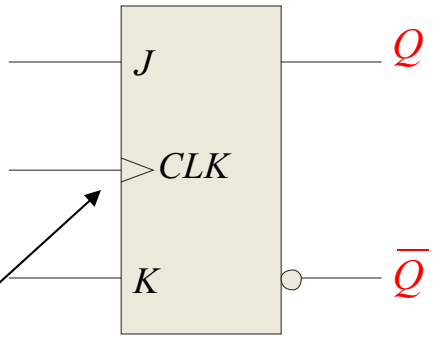
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

J-K Flip-flops

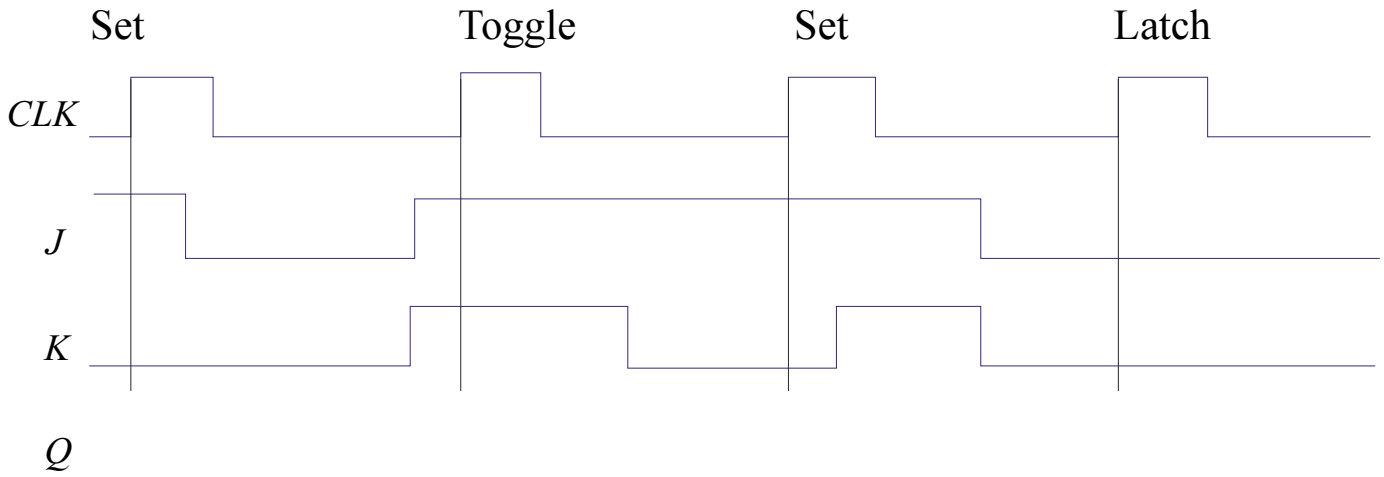
Example

Determine the *Q* output for the *J-K* flip-flop, given the inputs shown.



Notice that the outputs change on the leading edge of the clock.

Solution



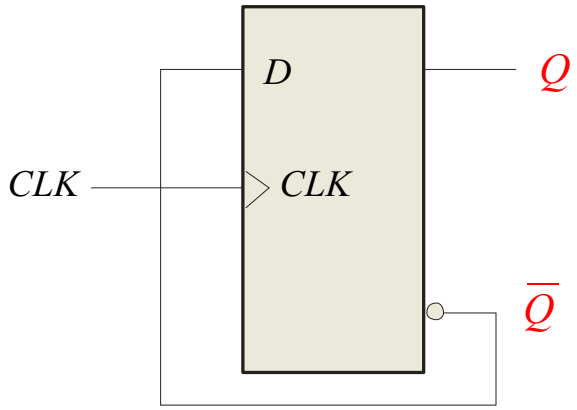
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Flip-flops

A D-flip-flop does not have a toggle mode like the J-K flip-flop, but you can hardwire a toggle mode by connecting $\overline{Q}$ back to D as shown. This is useful in some counters as you will see in Chapter 8.

For example, if Q is LOW, $\overline{Q}$ is HIGH and the flip-flop will toggle on the next clock edge. Because the flip-flop only changes on the active edge, the output will only change once for each clock pulse.



D flip-flop hardwired for a toggle mode

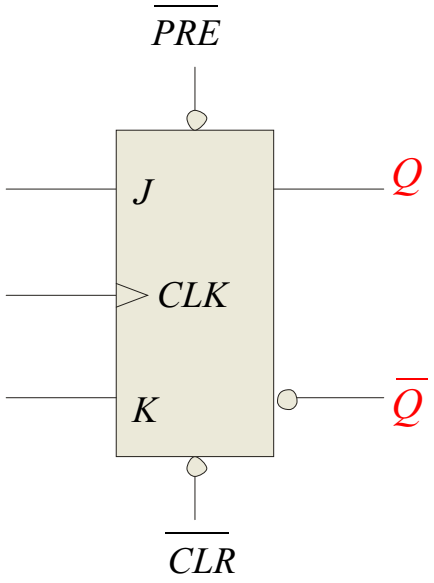
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Asynchronous Preset and Clear

Synchronous inputs are transferred in the triggering edge of the clock (for example the *D* or *J-K* inputs). Most flip-flops have other inputs that are *asynchronous*, meaning they affect the output independent of the clock.

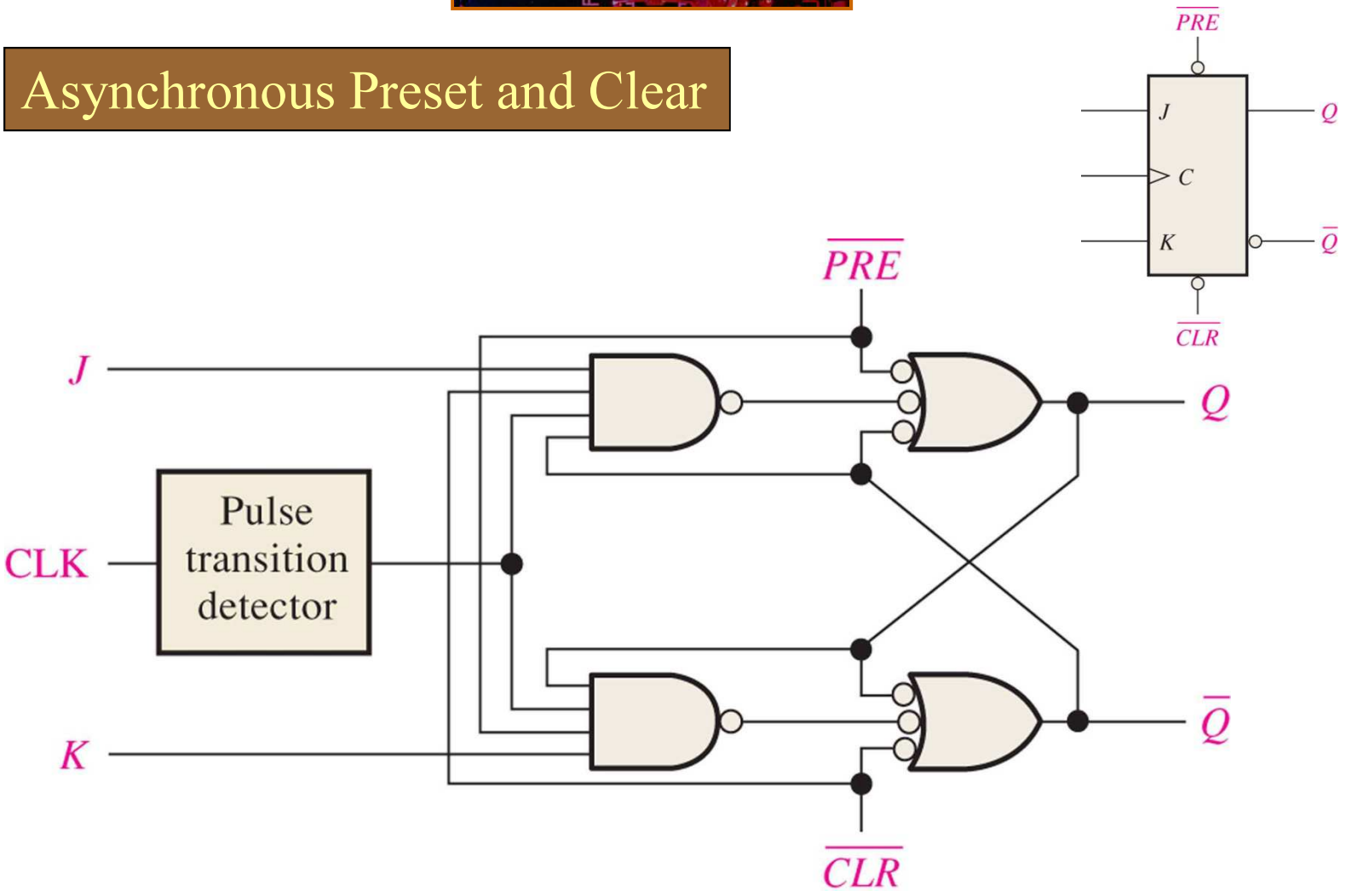
Two such inputs are normally labeled preset (*PRE*) and clear (*CLR*). These inputs are usually active LOW. A J-K flip flop with active LOW preset and CLR is shown.



© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Asynchronous Preset and Clear



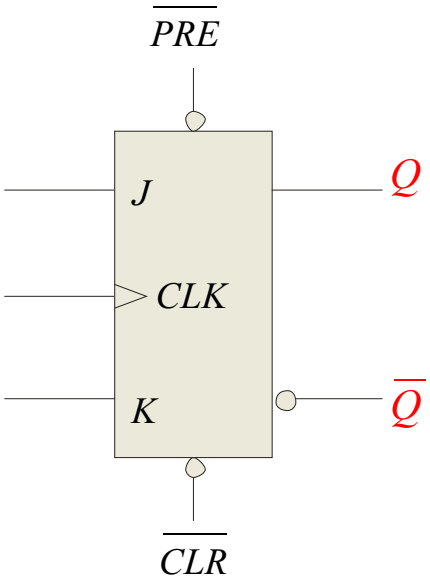
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

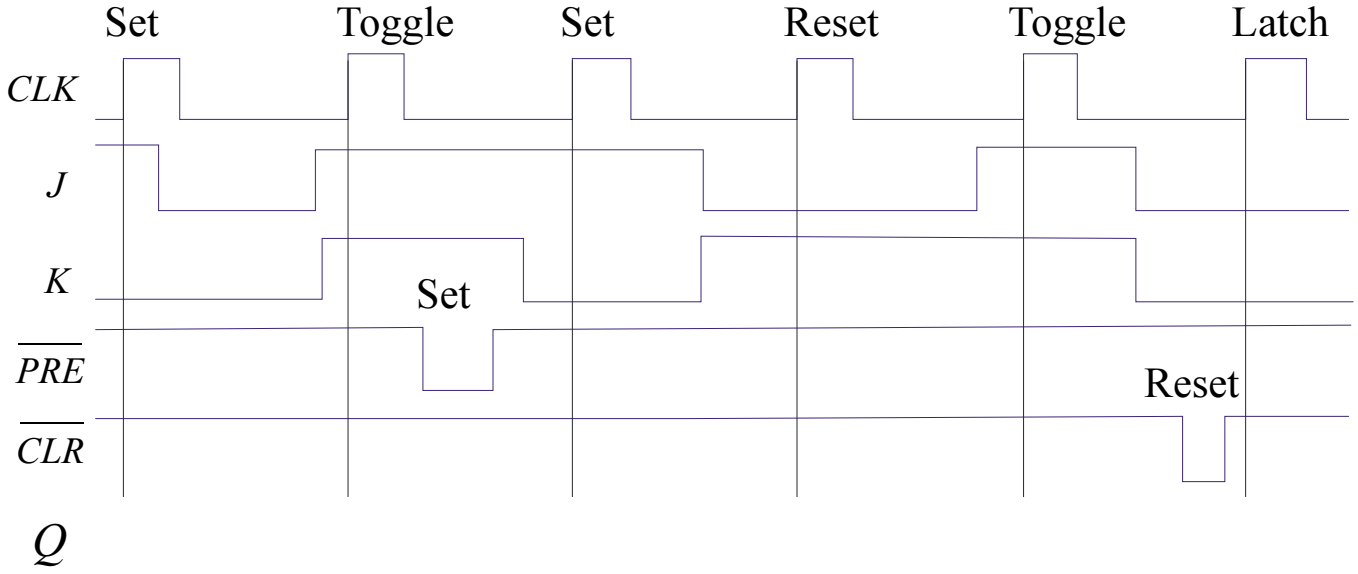
Flip-flops

Example

Determine the Q output for the J-K flip-flop, given the inputs shown.



Solution

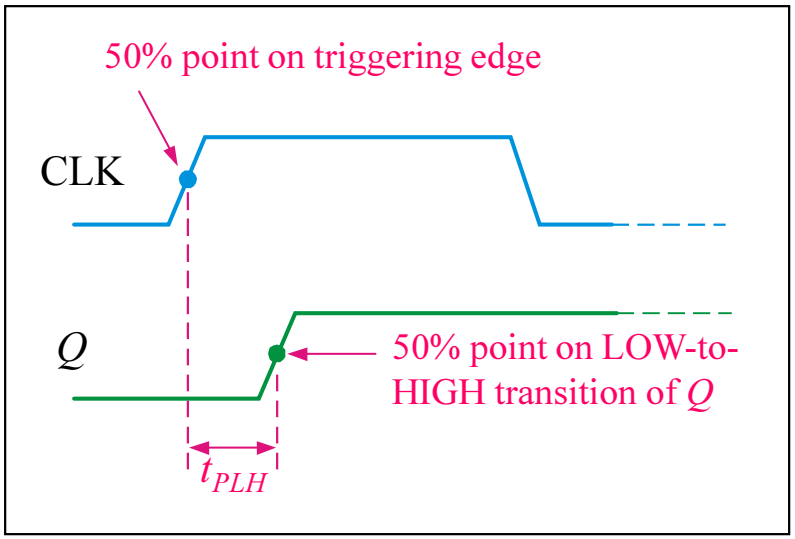


© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Flip-flop Characteristics

Propagation delay time is specified for the rising and falling outputs. It is measured between the 50% level of the clock to the 50% level of the output transition.



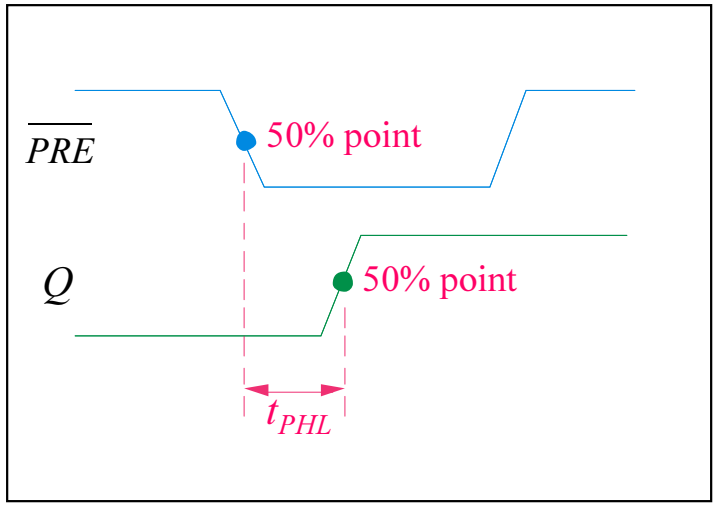
The typical propagation delay time for the 74AHC family (CMOS) is 4 ns. Even faster logic is available for specialized applications.

© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Flip-flop Characteristics

Another **propagation delay time** specification is the time required for an *asynchronous* input to cause a change in the output. Again it is measured from the 50% levels. The 74AHC family has specified delay times under 5 ns.



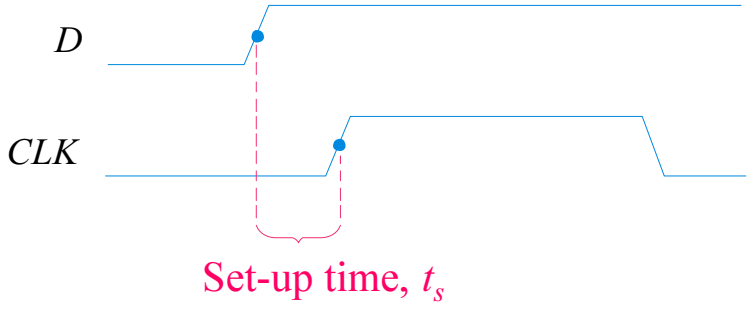
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

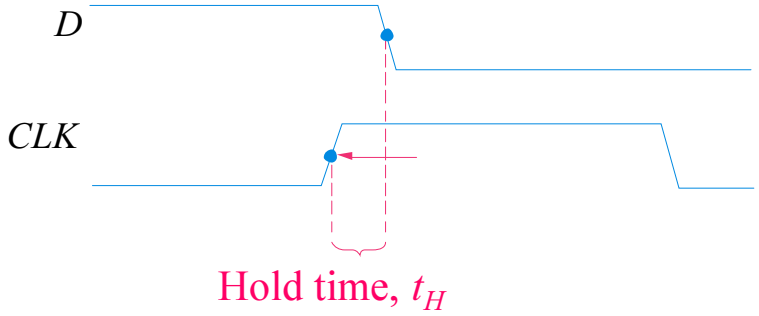
Flip-flop Characteristics

Set-up time and **hold time** are times required before and after the clock transition that data must be present to be reliably clocked into the flip-flop.

Setup time is the minimum time for the data to be present *before* the clock.



Hold time is the minimum time for the data to *remain* after the clock.



© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Flip-flop Characteristics

Other specifications include maximum clock frequency, minimum pulse widths for various inputs, and power dissipation. The power dissipation is the product of the supply voltage and the average current required.

A useful comparison between logic families is the **speed-power product** which uses two of the specifications discussed: the average propagation delay and the average power dissipation. The unit is energy.

Example What is the speed-power product for 74AHC74A? Use the data from Table 7-5 to determine the answer.

Solution From Table 7-5, the average propagation delay is 4.6 ns. The quiescent power dissipated is 1.1 mW. Therefore, the speed-power product is 5 pJ

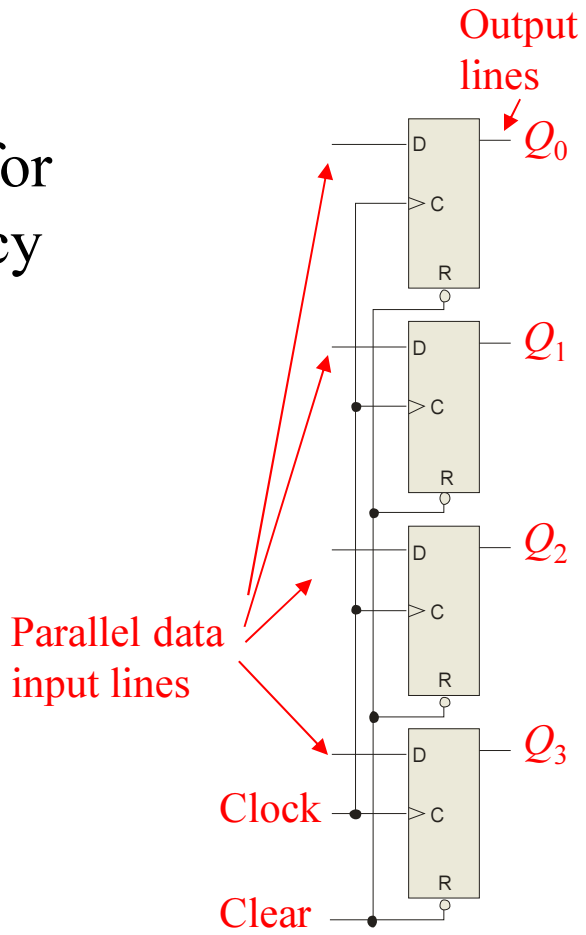
© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Summary

Flip-flop Applications

Principal flip-flop applications are for temporary data storage, as frequency dividers, and in counters (which are covered in detail in Chapter 8).

Typically, for **data storage** applications, a group of flip-flops are connected to parallel data lines and clocked together. Data is stored until the next clock pulse.



© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

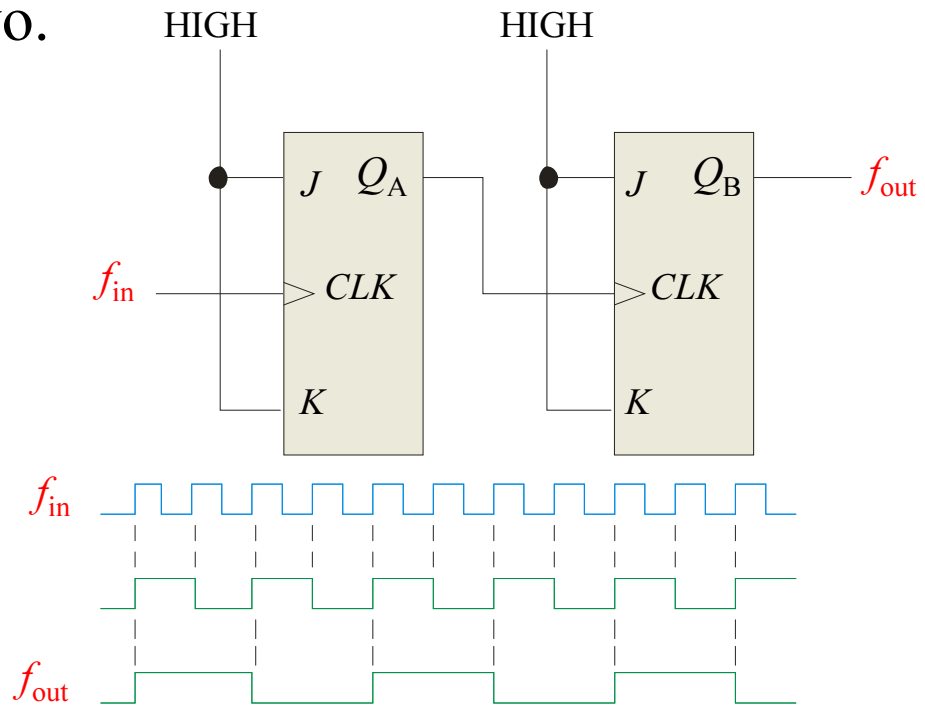
Summary

Flip-flop Applications

For **frequency division**, it is simple to use a flip-flop in the toggle mode or to chain a series of toggle flip flops to continue to divide by two.

One flip-flop will divide f_{in} by 2, two flip-flops will divide f_{in} by 4 (and so on). A side benefit of frequency division is that the output has an exact 50% duty cycle.

Waveforms:



© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved

Homework 10

- Chapter 7 (5, 8, 16, 17, 25)

© 2009 Pearson Education, Upper Saddle River, NJ 07458. All Rights Reserved