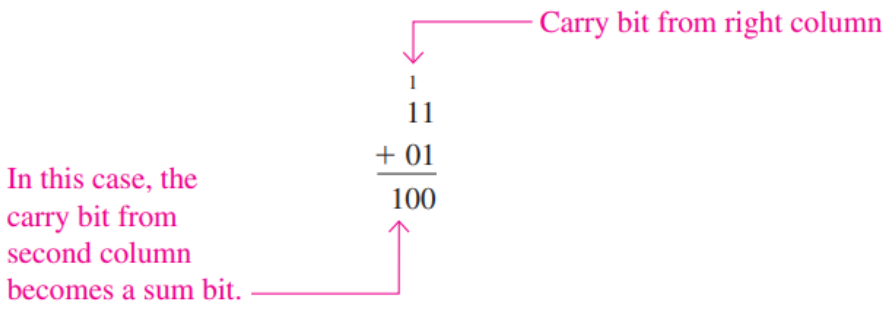
Parallel Binary Adders:



To add two binary numbers, a full-adder (FA) is required for each bit in the numbers. So for 2-bit numbers, two adders are needed; for 4-bit numbers, four adders are used; and so on. The carry output of each adder is connected to the carry input of the next higher-order adder, as shown in Figure 6–7 for a 2-bit adder. Notice that either a half-adder can be used for the least significant position or the carry input of a full-adder can be made 0 (grounded) because there is no carry input to the least significant bit position.

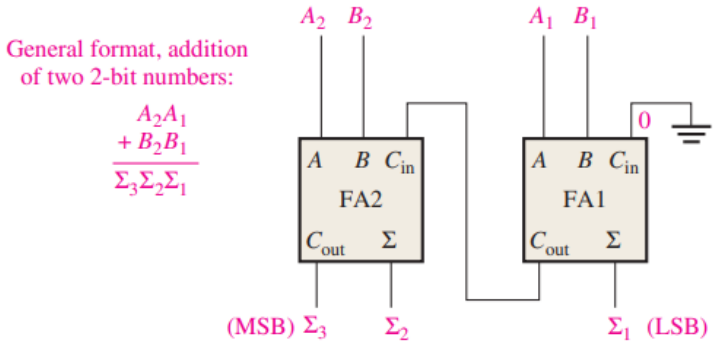


FIGURE 6–7 Block diagram of a basic 2-bit parallel adder using two full-adders.

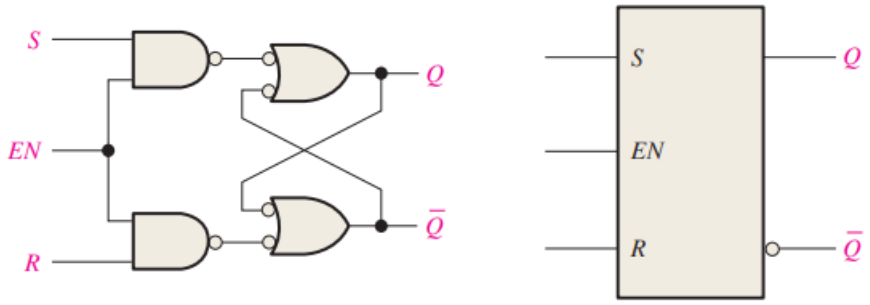
Latches:

The **latch** is a type of temporary storage device that has two stable states (bistable) and is normally placed in a category separate from that of flip-flops. Latches are similar to flip-flops because they are bistable devices that can reside in either of two states using a feedback arrangement, in which the outputs are connected back to the opposite inputs. The main difference between latches and flip-flops is in the method used for changing their state.

For Simple latches, you will consult youtube, they will be included in quizzes, and finals as well. You must have knowledge of their workings, truth table, timing diagrams etc.

The Gated S-R Latch

A gated latch requires an enable input, *EN* (*G* is also used to designate an enable input). The logic diagram and logic symbol for a gated S-R latch are shown in Figure 7–8. The *S* and *R* inputs control the state to which the latch will go when a HIGH level is applied to the *EN* input. The latch will not change until *EN* is HIGH; but as long as it remains HIGH, the output is controlled by the state of the *S* and *R* inputs. The gated latch is a *level-sensitive* device. In this circuit, the invalid state occurs when both *S* and *R* are simultaneously HIGH and *EN* is also HIGH.

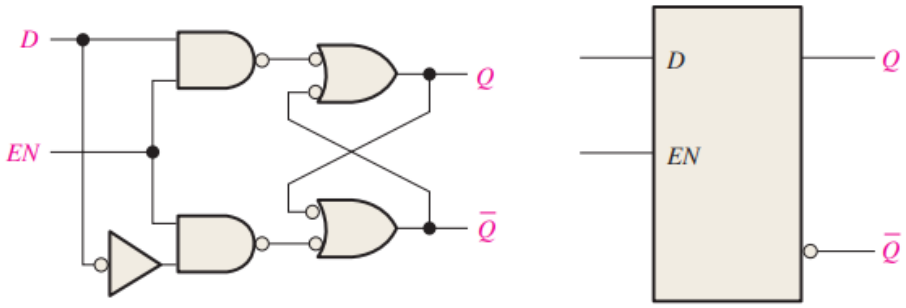


(a) Logic diagram (b) Logic symbol

FIGURE 7–8 A gated S-R latch.

The Gated D Latch

Another type of gated latch is called the D latch. It differs from the S-R latch because it has only one input in addition to *EN*. This input is called the *D* (data) input. Figure 7–10 contains a logic diagram and logic symbol of a D latch. When the *D* input is HIGH and the *EN* input is HIGH, the latch will set. When the *D* input is LOW and *EN* is HIGH, the latch will reset. Stated another way, the output *Q* follows the input *D* when *EN* is HIGH.



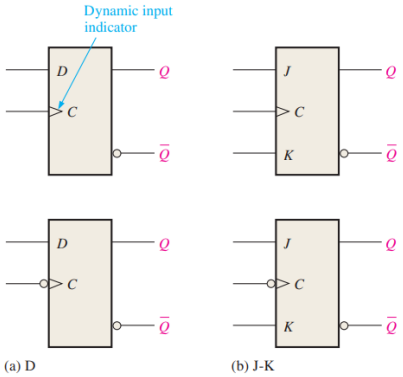
(a) Logic diagram (b) Logic symbol

FIGURE 7–10 A gated D latch. Open file F07-10 and verify the operation.

Flip Flops:

Flip-flops are synchronous bistable devices, also known as *bistable multivibrators*. In this case, the term *synchronous* means that the output changes state only at a specified point (leading or trailing edge) on the triggering input called the **clock** (CLK), which is designated as a control input, *C*; that is, changes in the output occur in synchronization with the clock. Flip-flops are edge-triggered or edge-sensitive whereas gated latches are level-sensitive.

An edge-triggered flip-flop changes state either at the positive edge (rising edge) or at the negative edge (falling edge) of the clock pulse and is sensitive to its inputs only at this transition of the clock. Two types of edge-triggered flip-flops are covered in this section: D and J-K. The logic symbols for these flip-flops are shown in Figure 7–13. Notice that each type can be either positive edge-triggered (no bubble at C input) or negative edge-triggered (bubble at C input). The key to identifying an edge-triggered flip-flop by its logic symbol is the small triangle inside the block at the clock (C) input. This triangle is called the dynamic input indicator



(a) D (b) J-K

FIGURE 7–13 Edge-triggered flip-flop logic symbols (top: positive edge-triggered; bottom: negative edge-triggered).

The dynamic input indicator means the flip-flop changes state only on the edge of a clock pulse.

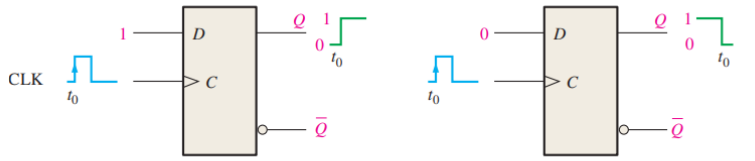
The D Flip-Flop:

The D input of the D flip-flop is a synchronous input because data on the input are transferred to the flip-flop's output only on the triggering edge of the clock pulse.

- When D is HIGH, the Q output goes HIGH on the triggering edge of the clock pulse, and the flip-flop is SET.
- When D is LOW, the Q output goes LOW on the triggering edge of the clock pulse, and the flip-flop is RESET.

This basic operation of a positive edge-triggered D flip-flop is illustrated in Figure 7–14, and Table 7–2 is the truth table for this type of flip-flop. Remember, the flip-flop cannot change state except on the triggering edge of a clock pulse.

The D input can be changed at any time when the clock input is LOW or HIGH (except for a very short interval around the triggering transition of the clock) without affecting the output. **Just remember, Q follows D at the triggering edge of the clock.**



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

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

FIGURE 7–14 Operation of a positive edge-triggered D flip-flop.

TABLE 7–2

Truth table for a positive edge-triggered D flip-flop.

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

↑ = clock transition LOW to HIGH

The operation and truth table for a negative edge-triggered D flip-flop are the same as those for a positive edge-triggered device except that the falling edge of the clock pulse is the triggering edge.

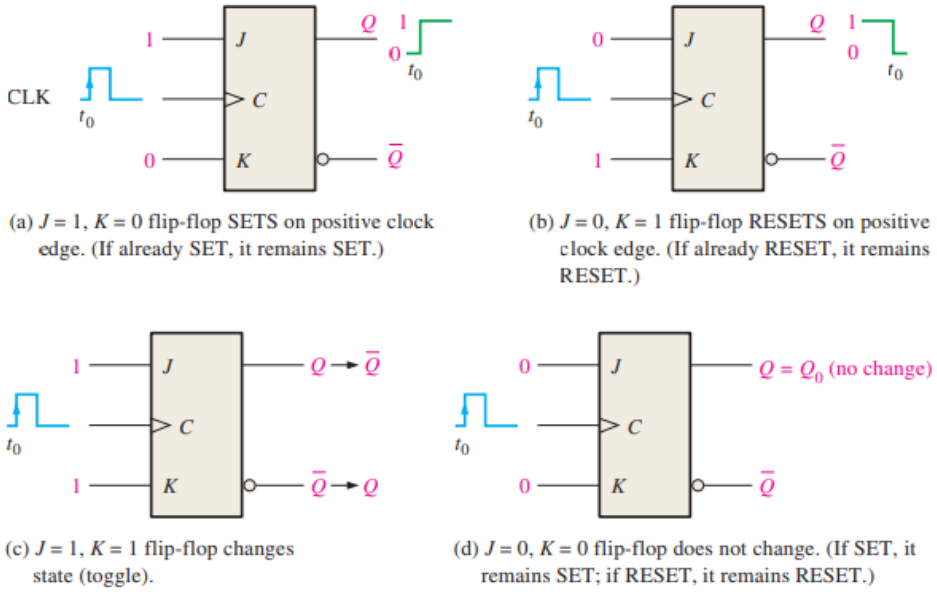
The J-K Flip-Flop

The J and K inputs of the J-K flip-flop are synchronous inputs because data on these inputs are transferred to the flip-flop's output only on the triggering edge of the clock pulse.

- When J is HIGH and K is LOW, the Q output goes HIGH on the triggering edge of the clock pulse, and the flip-flop is SET.
- When J is LOW and K is HIGH, the Q output goes LOW on the triggering edge of the clock pulse, and the flip-flop is RESET.
- When both J and K are LOW, the output does not change from its prior state.
- When J and K are both HIGH, the flip-flop changes state. This called the toggle mode.

This basic operation of a positive edge-triggered flip-flop is illustrated in Figure 7–17, and Table 7–3 is the truth table for this type of flip-flop. Remember, the flip-flop cannot change state except on the triggering edge of a clock pulse.

The J and K inputs can be changed at any time when the clock input is LOW or HIGH (except for a very short interval around the triggering transition of the clock) without affecting the output.



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

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

(c) $J = 1$, $K = 1$ flip-flop changes state (toggle).

(d) $J = 0$, $K = 0$ flip-flop does not change. (If SET, it remains SET; if RESET, it remains RESET.)

FIGURE 7–17 Operation of a positive edge-triggered J-K flip-flop.

TABLE 7–3

Truth table for a positive edge-triggered J-K flip-flop.

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

↑ = clock transition LOW to HIGH

Q_0 = output level prior to clock transition

T Flip Flop:

Do it yourself. Learn from different sources.

Registers:

A register can consist of one or more flip-flops used to store and shift data.

A **register** is a digital circuit with two basic functions: data storage and data movement. The storage capability of a register makes it an important type of memory device. Figure 8–1 illustrates the concept of storing a 1 or a 0 in a D flip-flop. A 1 is applied to the data input as shown, and a clock pulse is applied that stores the 1 by *setting* the flip-flop. When the 1 on the input is removed, the flip-flop remains in the SET state, thereby storing the 1. A similar procedure applies to the storage of a 0 by *resetting* the flip-flop, as also illustrated in Figure 8–1.

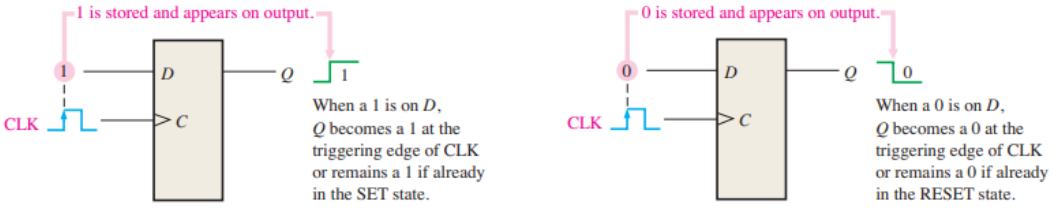


FIGURE 8–1 The flip-flop as a storage element.

The *storage capacity* of a register is the total number of bits (1s and 0s) of digital data it can retain. Each **stage** (flip-flop) in a shift register represents one bit of storage capacity; therefore, the number of stages in a register determines its storage capacity.

The *shift capability* of a register permits the movement of data from stage to stage within the register or into or out of the register upon application of clock pulses. Figure 8–2

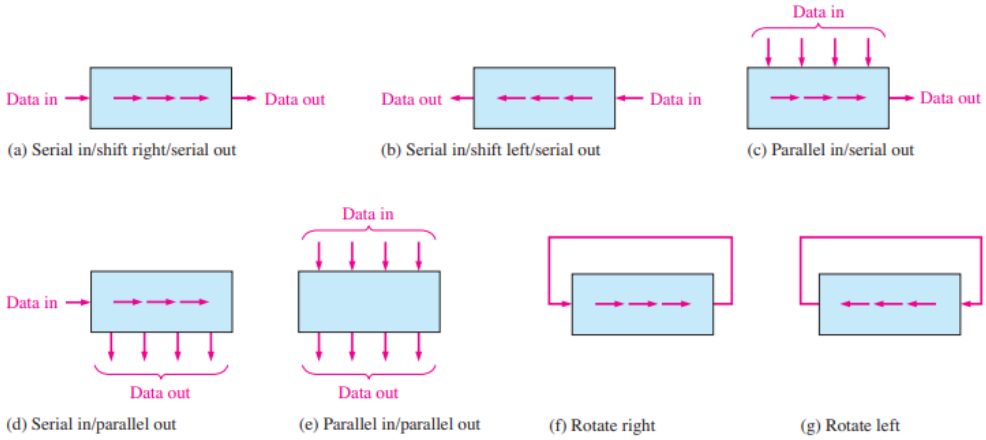


FIGURE 8–2 Basic data movement in shift registers. (Four bits are used for illustration. The bits move in the direction of the arrows.)