31 LATCHES AND FLIP-FLOPS

In this chapter *latches* and *flip-flops* are introduced and described in detail. These devices are basic building blocks of sequential logic systems which are systems for which the output voltage levels depend upon past as well as present voltage levels. Hence, these types of logic gates are memory elements. The operation of sequential logic gates is such that when new inputs are applied, the output responds according to the new inputs and the previous inputs. Upon removal of the inputs, the outputs then remain unchanged.

Such memory elements are made up of the combinational logic gates NOT, AND, OR, NAND, and NOR and feedback. Feedback is a term used to indicate that a portion of the output voltage is fed back to the input. In addition to combinational logic gates, CMOS latches and flip-flops are commonly constructed with the tri-state inverters and transmission gates introduced in Chapter 25.

This chapter begins with definitions and descriptions of properties used to describe latches and flip-flops. The basic digital memory element, the cross coupled inverter latch, is then introduced and analyzed. The different types of latches and flip-flops including RS, JK, and D types are then introduced.

Some sections of this chapter describe latch and flip-flop configurations realized with complex logic function AND-OR-invert gates. All sections, figures, and examples that refer to circuit configurations realized with AOIs are noted with the superscript ^{AOI}. These sections can be skipped by the reader without loss of continuity.

31.1 BASIC DEFINITIONS FOR SEQUENTIAL LOGIC GATES

Single Cell Memory Elements (*Latches and Flip-Flops*)

Single cell memory elements are called **latches** or **flip-flops**. Since the output of a latch or flip-flop can have either of two logic states, these devices are also called **bistable memory elements**. The difference between latches and flip-flops is as follows:

- a *latch* can change output states continuously corresponding to input changes in any instant, whereas
- a *flip-flop* changes output states only at precise instants controlled by a train of equally spaced pulses called a clock pulse train

The use of flip-flops insures that the system components change at the correct instants. The clock control terminal is an additional input that acts as an enabling input only at precise instants of time.

Periodic Clock Signal

A typical square-wave clock signal is displayed in Figure 31.1. Note that this signal is a precise string of periodic voltage pulses that alternate between logic level **0** and **1**. Note further that the *period* of this square wave is T and the *frequency* of the wave shape is v. The relation between the period and frequency is

$$v = \frac{1}{T}$$

The period T for the clock signal in Figure 31.1 is indicated.

Transitions (Edges)

A clock pulse has two types of **transitions** (or **ticks**). When a flip-flop is controlled by a clock pulse, either one or the other of these transitions enable a change in the outputs. These two types of transitions are classified as follows:

1. **Low-to-high transition** (L-to-H) which corresponds to a signal changing from the low logic state to the high logic state as indicated in Figure 31.1. When this type of transition permits change in the outputs, the logic gate is said to be *positive-edge triggered.*

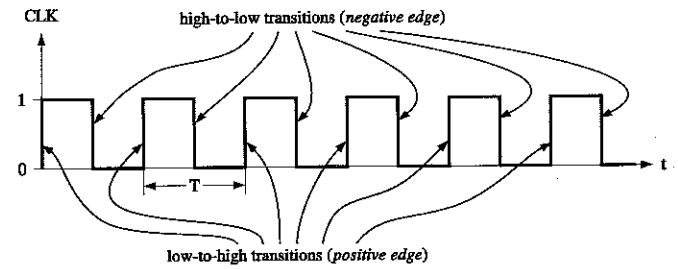

FIGURE 31.1 Typical Periodic Clock Signal

2. **High-to-low transitions** (H-to-L) which corresponds to a signal changing from the logic high state to the logic low state as indicated in Figure 31.1. When this type of transition permits change in the outputs, the logic gate is said to be *negative-edge triggered.*

Example 31.1 ***Level-Triggered Versus Edge-Triggered***

Consider the signal in Figure 31.2a to be a periodic clock signal and let the signal in Figure 31.2b be the input to three different types of latches or flip-flops, whose outputs are shown in Figure 31.2c, d, and e. Are the outputs in (c), (d), and (e) outputs of level-triggered or edge-triggered latches or flip-flops?

Solution (Figure 31.2c) Level-Triggered Latch The output signal in Figure 31.2c changes with the input signal in (b) whenever the clock signal in (a) is high. At times t_1, t_2, t_3, t_6, t_7, and t_8 output (c) is seen to follow the input in this manner. When the clock signal goes low, the output signal (c) stores or "*latches*" the input signal value until the next time the clock signal goes high. Note that at time t_5 the input signal changes but the output signal in (c) does not. At time t_6 the clock signal again goes high and the (c) output signal attains the logic value of the input (b). The output signal shown in Figure 31.2c demonstrates a latch that is triggered when the clock is high and is referred to as a *positive level-triggered latch.*

Solution (Figure 31.2d) Positive-Edge Triggered Flip-Flop The output signal in Figure 31.2d is initially low. At time t_1, the clock signal in (a) goes low-to-high and the output signal in (d) attains the logic value of the input signal in (b). At time 6_6, the clock signal again goes low-to-high and the output signal in (d) again attains the logic value of (b). Note that the output signal in (d) never changes state when the input signal changes state nor when the clock goes high-to-low. Since this output signal changes state only on the rising edge of the clock, the signal of Figure 31.2d represents the output of a *positive edge-triggered flip-flop.*

Solution (Figure 31.2e) Negative Edge-Triggered Flip-Flop The output signal shown in Figure 31.2e is seen to change logic states only when the clock signal goes high-to-low. The output signal in Figure 31.2e is therefore the output of a *negative edge-triggered flip-flop.*

Types of Sequential Logic Circuits

There are two types of clocked digital logic systems. These sequential logic circuits are called:

1. **synchronous logic circuits:** those in which the same clock is used to cause all logic variables to change simultaneously, and
2. **asynchronous logic circuits:** those that are unclocked or portions of the system are either unclocked or run off independent clocks (i.e. all variables are not clocked together)

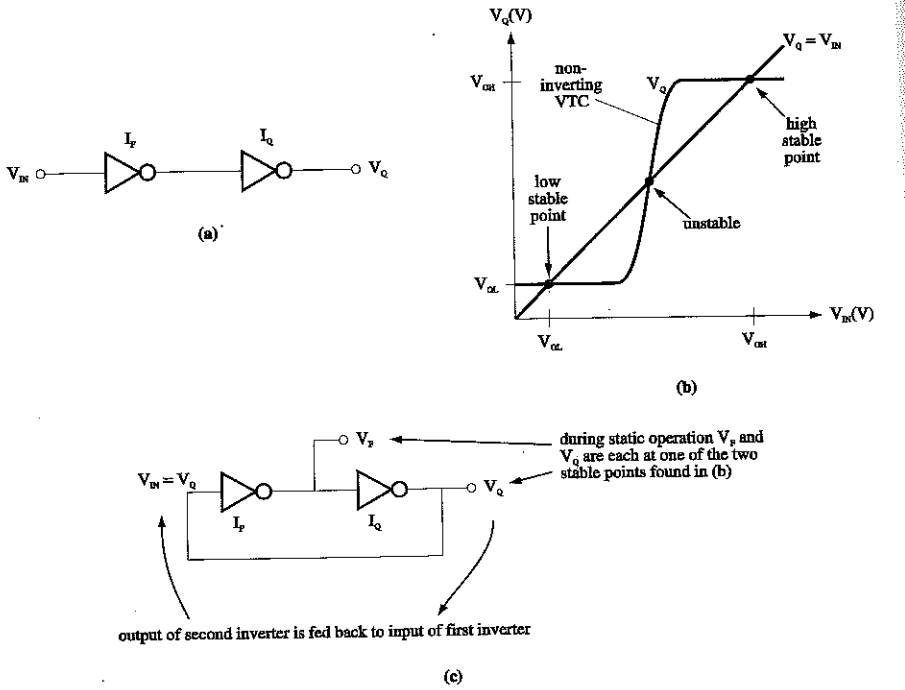

FIGURE 31.5 Analysis of Cross Coupled Inverter Latch: (a) Two cascaded inverters, (b) Voltage transfer characteristic V_Q versus V_{IN} superimposed over $V_Q = V_{IN}$ gives two stable solutions, (c) Output V_Q fed back to V_{IN} gives positive feedback inverter loop of Figure 31.4

the straight line for $V_{IN} = V_Q$ must also be satisfied and this line is also plotted in Figure 31.5b. The three intersection points of the two curves indicate that the circuit of Figure 31.5b can statically operate at any of these intersections. However, the middle intersection point is unstable and even the most minor anomaly or circuit noise forces the inverter loop out of that state and into one of the other states. Hence, the inverter loop of Figure 31.5c has only two stable operating states.

The basic cross coupled inverter latch presented in this section and Figure 31.4d is used extensively in the following sections. This element is the basis for latches and flip-flops.

31.3 RESET-SET (RS) LATCH

NOR Realized RS Latch

The first practical digital latch is displayed in Figure 31.6a. This circuit provides two additional inputs to the basic cross coupled inverter latch of the previous section and Figure 31.4d by replacing the inverters

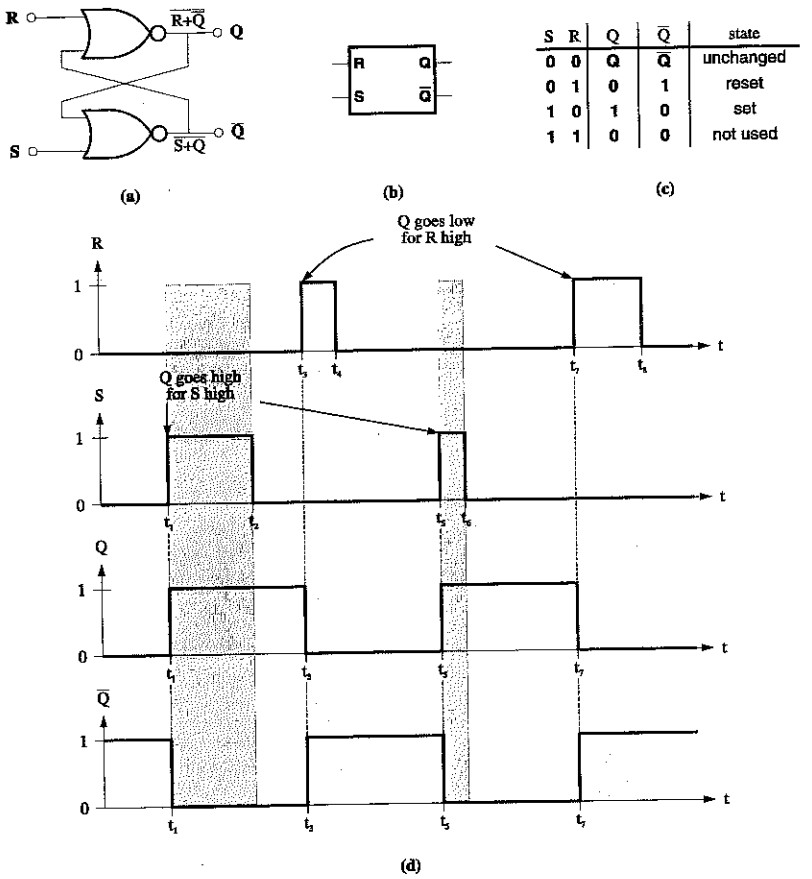

FIGURE 31.6 RS Latch (NOR realization): (a) Cross coupled NOR gates, (b) Circuit symbol, (c) Truth table, (d) Time waveforms demonstrating operation

with two-input NOR gates. The feedback connection of the two NOR gates represents one implementation of the memory element called a **reset-set latch**, or **RS latch**. Each NOR gate output is fed back to one input of the other NOR gate and the remaining NOR gate inputs are the two inputs to this RS latch. The outputs are defined as Q and $\overline{Q}$, the inverses of each other. The inputs are defined as R and S which stand for *reset* and *set*, respectively. The *reset condition* of R = 1 and S = 0 corresponds to Q being reset