FLIP-FLOPS

- By **combining two level-sensitive** latches, **one negative-sensitive and one positive-sensitive**, we construct the **edge-triggered flip-flop** shown in Figure 1.32(a–b).
- The first latch stage is called the **master** and the **second** is called the slave.

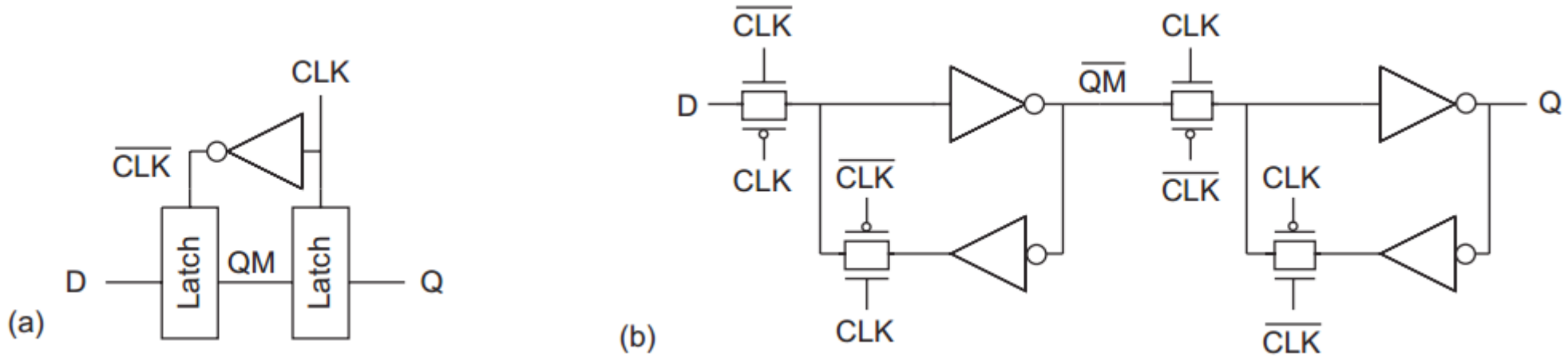

FIGURE 1.32 CMOS positive-edge-triggered *D* flip-flop

FLIP-FLOPS

- While **CLK is low**, the **master negative-level-sensitive** latch **output** (QM) follows the D input while the **slave positive-level-sensitive latch holds** the previous value (Figure 1.32(c)).
- When the **clock transitions from 0 to 1**, the **master latch becomes opaque** and **holds** the D value at the time of the clock transition.
  - The **slave latch becomes transparent**, passing the **stored** master value (QM) to the **output** of the slave latch (Q).

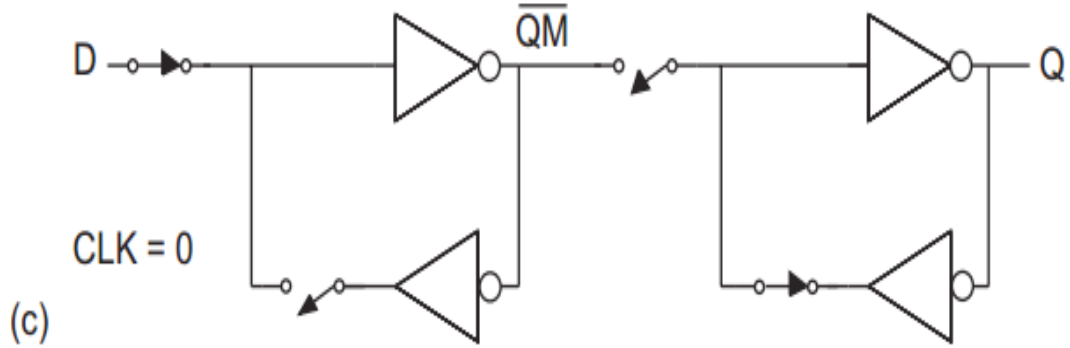

FIGURE 1.32 CMOS positive-edge-triggered *D* flip-flop

FLIP-FLOPS

- When the **clock transitions from 0 to 1**, the **master latch becomes opaque** and **holds** the D value at the time of the clock transition.
  - The **slave latch becomes transparent**, passing the **stored** master value (QM) to the **output** of the slave latch (Q).
  - The D input is blocked from affecting the output because the master is disconnected from the D input (Figure 1.32(d)).
- When the clock transitions from 1 to 0, the slave latch holds its value and the master starts sampling the input again.

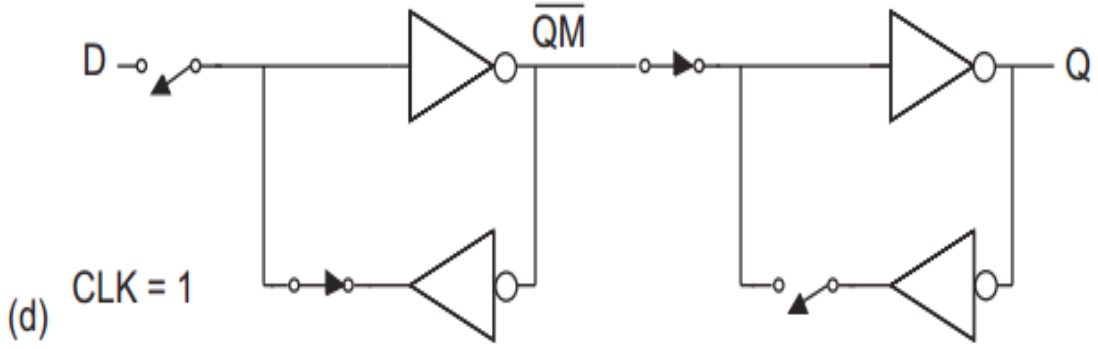

FIGURE 1.32 CMOS positive-edge-triggered *D* flip-flop

FLIP-FLOPS

- In summary, this flip-flop **copies D to Q on the rising edge** of the clock, as shown in Figure 1.32(f).
- Thus, this device is called a **positive-edge triggered** flip-flop (also called a D flip-flop, D register, or master–slave flip-flop). Figure 1.32(g) shows the circuit symbol for the flip-flop.
- By **reversing** the **latch polarities**, a **negative-edge triggered** flip-flop may be constructed.

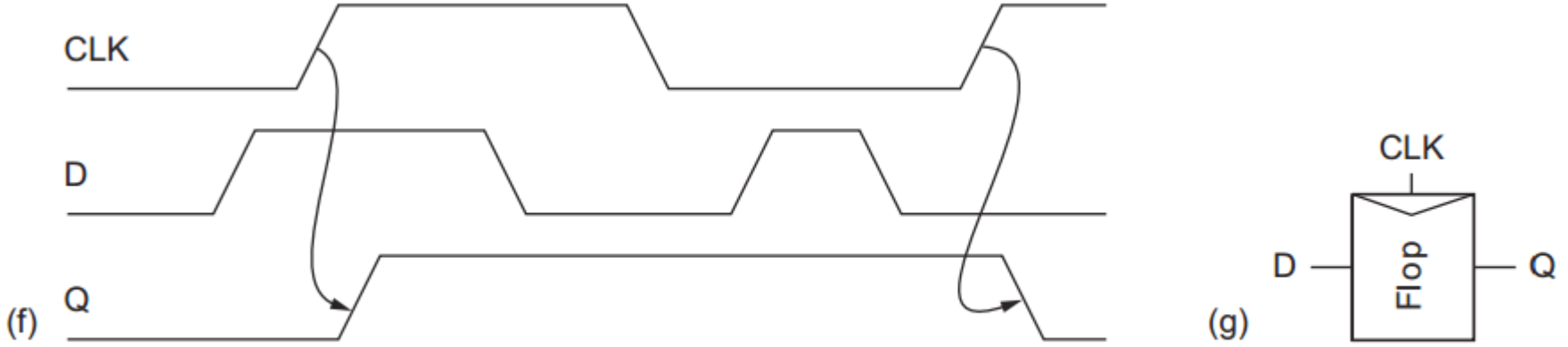

FIGURE 1.32 CMOS positive-edge-triggered *D* flip-flop

FLIP-FLOPS

- While we have shown a **transmission gate multiplexer** as the input stage, **good design** practice would **buffer** the input and output with **inverters**, as shown in Figure 1.32(e), to preserve what we call "**modularity**."

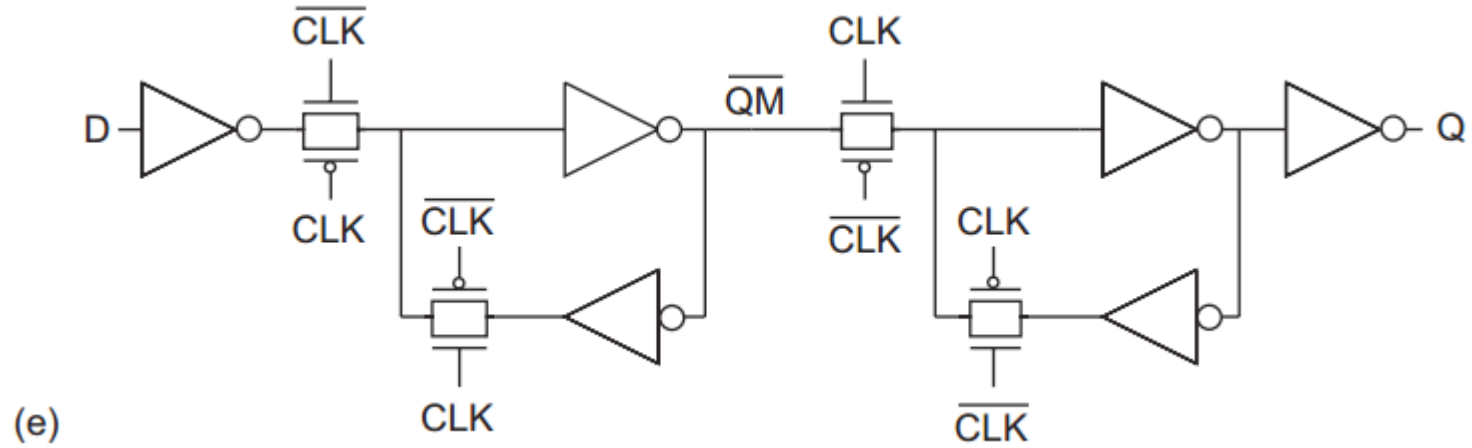

FIGURE 1.32 CMOS positive-edge-triggered *D* flip-flop

THREE TERMS!

- **Regularity**
  - **Decomposition** of a large system in **simple and similar blocks** as much as possible.
- **Modularity**
  - Modularity in design means that the various **functional blocks** which make up the larger system must have **well-defined functions and interfaces**.
  - Modularity allows that **each block or module can be designed relatively independently** from each other.
  - All of the blocks can be **combined with ease at the end of the design process**, to form the large system.
  - The concept of modularity enables the **parallelization of the design process.**

THREE TERMS!

- **Locality**:
  - The concept of locality also ensures that **connections are mostly between neighboring modules**, **avoiding long-distance connections** as much as possible.

REFERENCE

- Weste
  - 1.4.9
- Slide