CMOS LOGIC

MULTIPLEXERS

- Multiplexers are **key components** in CMOS memory elements and data manipulation structures.
- A multiplexer chooses the output from among **several inputs based on a select signal**.
  - A 2-input, or 2:1 multiplexer, chooses input D0 when the select is 0 and input D1 when the select is 1.
  - The truth table is given in Table 1.6
  - The logic function is **Y = S · D0 + S · D1.**

TABLE 1.6 Multiplexer truth table

$S / \bar{S}$	$D1$	$D0$	Y
0 / 1	X	0	0
0 / 1	X	1	1
1 / 0	0	X	0
1 / 0	1	X	1

MULTIPLEXERS

- **Two transmission gates can be tied together to form a compact 2-input multiplexer**, as shown in Figure 1.28(a).
- The **select and its complement enable exactly one of the two transmission gates** at any given time.
  - The complementary select S is often not drawn in the symbol, as shown in Figure 1.28(b).

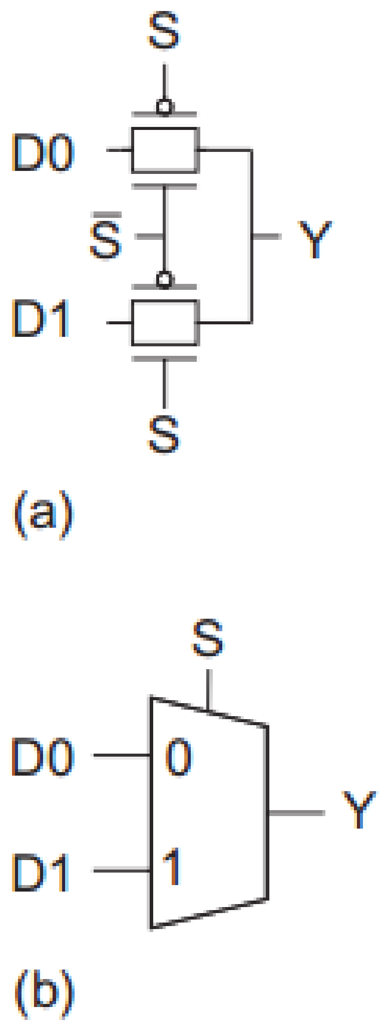


FIGURE 1.28 Transmission gate multiplexer

MULTIPLEXERS

- The **transmission gates produce a non-restoring multiplexer**.
- We could build a **restoring, inverting multiplexer** out of gates in several ways.
  - One is the compound gate of Figure 1.18(e), connected as shown in Figure 1.29(a).
  - Another is to **gang together two tristate inverters**, as shown in Figure 1.29(b).

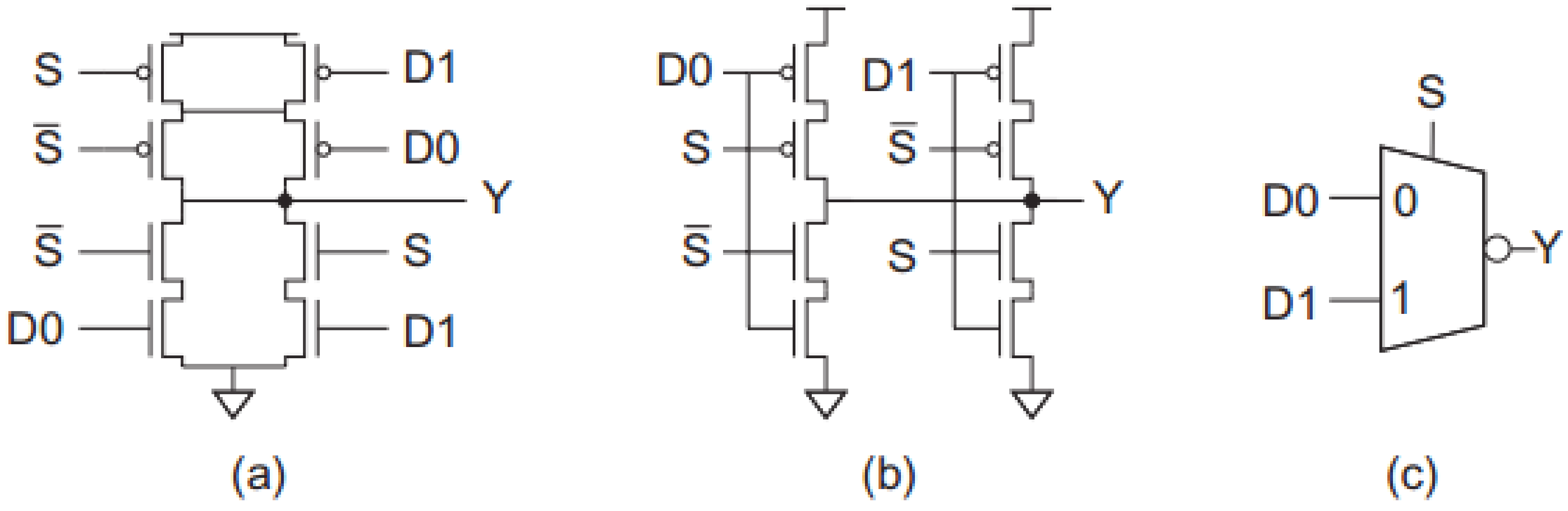


FIGURE 1.29 Inverting multiplexer

MULTIPLEXERS

- Notice that the schematics of these two approaches are nearly identical, save that the pull-up network has been slightly simplified and permuted in Figure 1.29(b). This is possible because the select and its complement are mutually exclusive.
- The tristate approach is slightly more **compact and faster** because it require**s less internal wire**. Again, if the complementary select is generated within the cell, it is omitted from the symbol (Figure 1.29(c)).

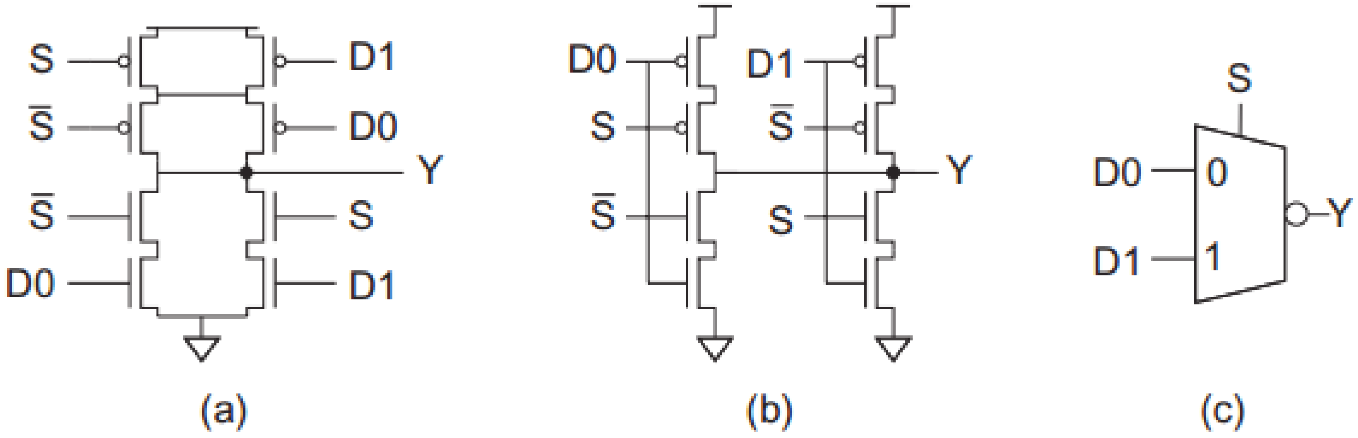


FIGURE 1.29 Inverting multiplexer

MULTIPLEXERS

- **Larger multiplexers** can be built from **multiple 2-input** multiplexers or by **directly ganging together several tristates.**
- The latter approach requires **decoded enable signals for each tristate**.
- **The enables should switch simultaneously to prevent contention.**
  - 4-input (4:1) multiplexers using each of these approaches are shown in Figure 1.30.

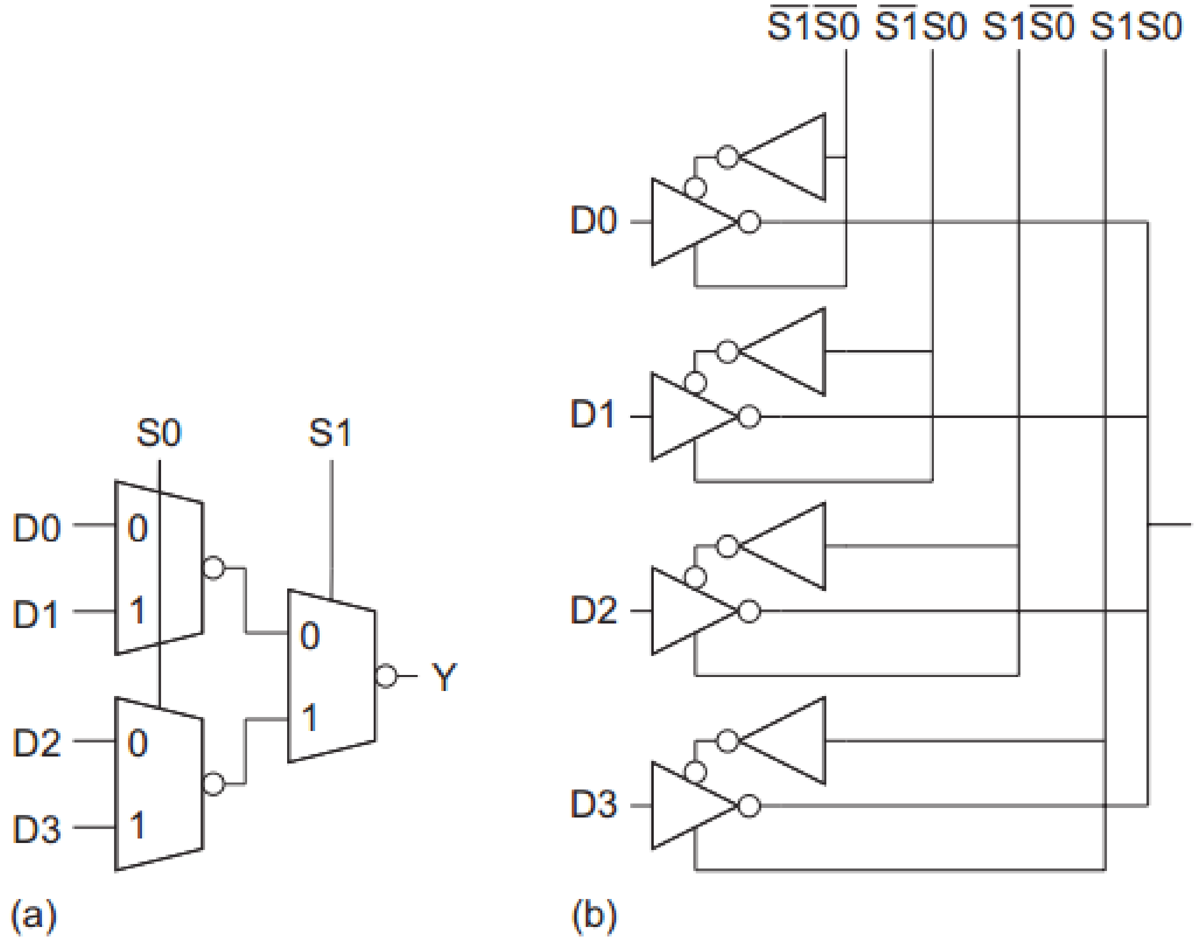
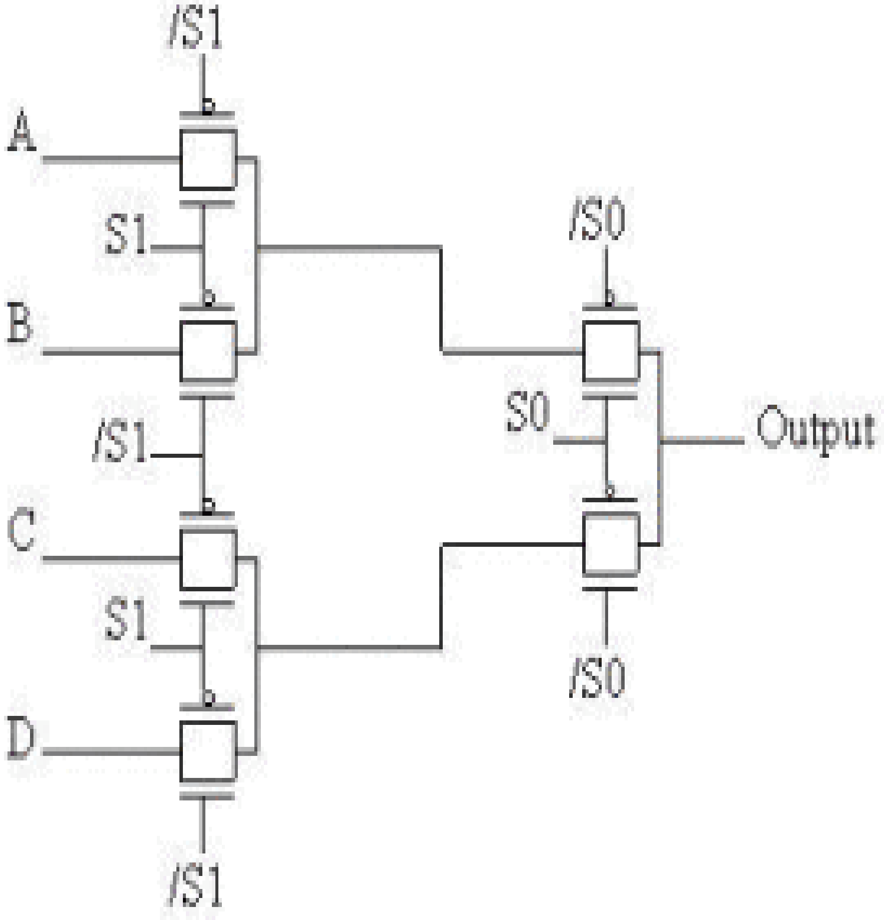


FIGURE 1.30 4:1 multiplexer



When S1=0, /S1=1,
B and D turns ON, and,
When S0=0, /S0=1,
D is selected and
appears at the output.

When S1=1, /S1=0,
A and C turns ON, and,
When S0=0, /S0=1,
C is selected and
appears at the output.

When S1=0, /S1=1,
B and D turns ON, and,
When S0=1, /S0=0,
B is selected and
appears at the output.

When S1=1, /S1=0,
A and C turns ON, and,
When S0=1, /S0=0,
A is selected and
appears at the output.

SEQUENTIAL CIRCUITS

- Sequential circuits have **memory.**
- Their outputs depend on **both current and previous inputs (or outputs)**.
- Using the combinational circuits, we can now build sequential circuits such as:
  - **Latches**
  - **Flip-flops**

- These elements receive a **clock**, CLK, and a **data input**, D, and produce an **output**, Q.
  - A D latch is **transparent** when **CLK = 1**, meaning that **Q follows D**.
  - It becomes **opaque** when **CLK = 0**, meaning **Q retains its previous value and ignores changes in D**.
  - An **edge-triggered** flip-flop **copies D to Q on the rising edge of CLK** and remembers its old value at other times.

LATCHES

- A D latch built from a **2-input multiplexer** and **two inverters** is shown in Figure 1.31(a).
- The multiplexer can be built from **a pair of transmission gates**, shown in Figure 1.31(b)
  - because the **inverters are restoring**.
    - [Yet here may be some problems]
- This latch also produces a complementary output, Q’.

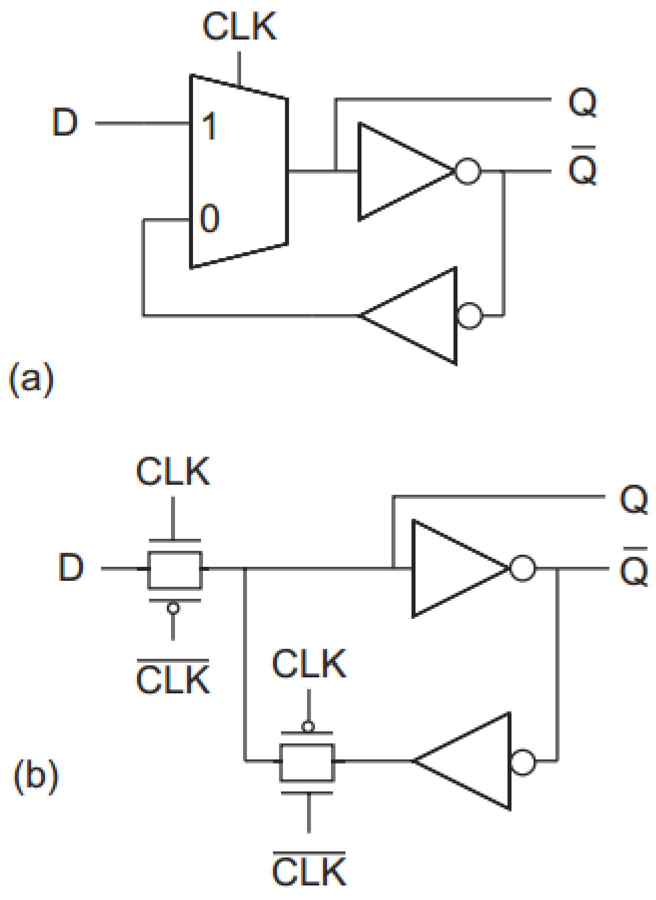


FIGURE 1.31 CMOS positive-level-sensitive *D* latch

LATCHES

- When **CLK = 1**, the latch is **transparent** and **D flows through to Q** (Figure 1.31(c)).
- When **CLK falls to 0**, the latch becomes **opaque**.
- A **feedback path** around the **inverter pair is established** (Figure 1.31(d)) to **hold the current state of Q indefinitely**.
  - Why?

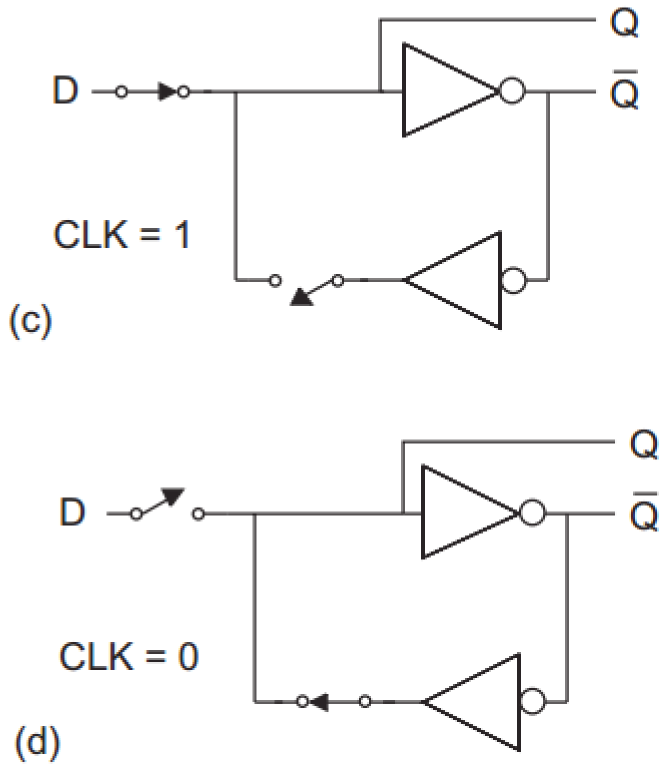


FIGURE 1.31 CMOS positive-level-sensitive *D* latch

LATCHES

- The D latch is also known as a **level-sensitive latch** because the **state of the output is dependent** on the **level of the clock signal**, as shown in Figure 1.31(e).
- The latch shown is a **positive-level-sensitive** latch, represented by the symbol in Figure 1.31(f).
- By **inverting the control connections** to the multiplexer, the latch becomes **negative-level-sensitive**. (Do it yourself!)

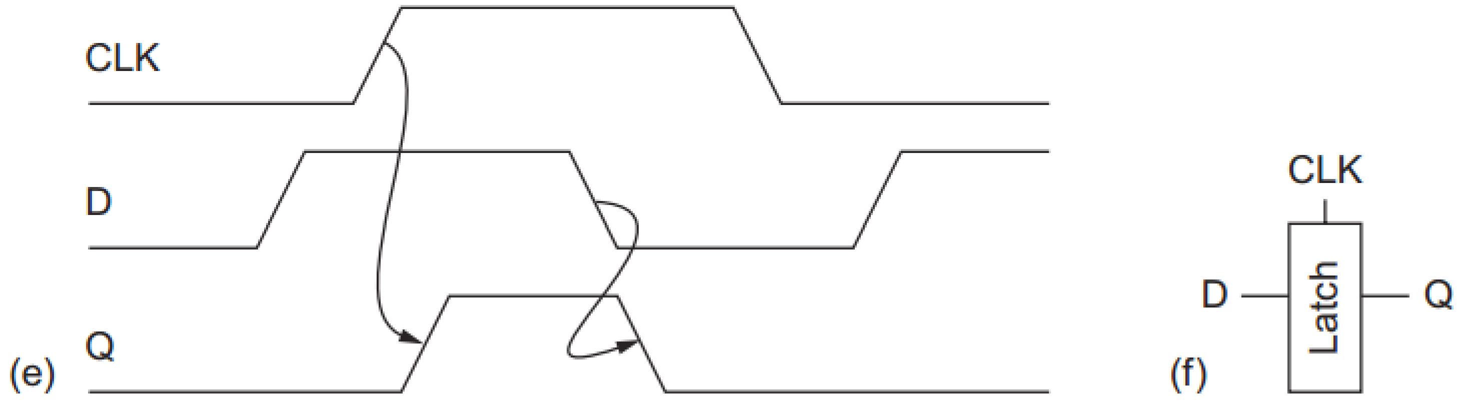


FIGURE 1.31 CMOS positive-level-sensitive *D* latch

REFERENCE

- Weste
  - 1.4.6
  - 1.4.7
  - 1.4.8
  - 1.4.9
  - 1.4.9.1