CMOS Inverter:
A Programmer's View

B. L. Massingill

March 27, 2019

1 Introduction

This note describes a simple CMOS circuit from a programmer's point of view, as a way of showing how this technology can be used to build circuits that perform Boolean-algebra operations. It is most definitely a programmer's take on the subject and ignores the electrical-engineering aspects of the subject. It is based on recollections from an upper-division course in VLSI design I took in graduate school many years ago[1], supplemented by the Wikipedia article `https://en.wikipedia.org/wiki/CMOS`.

2 CMOS technology

CMOS (complementary metal-oxide semiconductor) is a widely-used technology for designing *integrated circuits* as used in microprocessors and other applications. From a programmer's perspective, basic building blocks of this technology include the following:

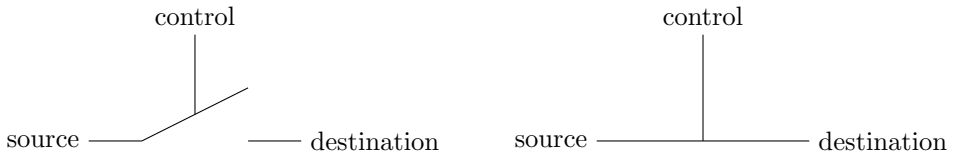
- "Wires" along which bits (1s and 0s) flow.
- Transistors that function as switches that allow bits to flow or not flow.
- A strong source of 1s (called V_{dd}).
- A strong source of 0s (called Ground).

A typical circuit has one or more input bits and one or more output bits, where the output bits represent Boolean-algebra operations on the input bits. (Some circuits also include a notion of persistent state; this note doesn't address that.) In EE terms, 1s and 0s are represented by two distinct voltages, and the flow of bits mentioned above is propagation of electrical signals.

Figure 1 shows a schematic representation of a switch. The value of the control input controls whether the source and destination are connected; if the control input is 1, the switch is "closed" and bits flow; if it is 0, the switch is "open" and they do not. (It might be worth noting that while the description is in terms of a source and a destination and suggests that bits only flow in one direction, one thing I learned from that long-ago course is that the circuit itself would allow bits (current) to flow in either direction; the actual flow is from whichever side is stronger to the weaker side. V_{dd} and Ground are "strong" sources; if one of them is connected to something else, bits flow from it to the other side of the switch.

CMOS uses two types of transistors, represented as shown in Figure 2. Again from a programmer's point of view, one of them is closed when its control input is 0 and conducts 1s well, while the other is closed when its control input is 1 and conducts 0s well. In designing a circuit, the goal is that for any combination of inputs there will either be a path through closed switches from the source of 1s (V_{dd}) to the output or a path through closed switches from the source of 0s (Ground) to the output, but not both.

[1] I thank its instructor and TAs for teaching me everything I know about this subject. Any errors are mine!



(a) Open switch (value of control is 0).

(b) Closed switch (value of control is 1).

Figure 1: Open and closed switches.

(a) Switch that is closed when its control is 0 and conducts 1s well.

(b) Switch that is closed when its control is 1 and conducts 0s well.

Figure 2: Two types of CMOS switches.

3 Inverter

Figure 3 shows a CMOS inverter, i.e., something that implements the Boolean "not" operator. I think this is pretty nifty!

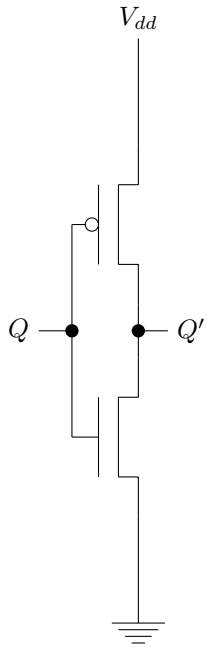


Figure 3: CMOS inverter. (The symbol at the bottom represents Ground.) When input Q is 0, the top switch is closed and bits flow from the source of 1s to the output. When input Q is 1, the bottom switch is closed and bits flow from the source of 0s to the output.