LAB 10

TTL and CMOS Logic Gates

Reading: Hayes and Horowitz, Class 13 and Lab 13.

Today you will be introduced to the circuits of digital electronics. We will start with some circuits made with discrete electronics to perform logical AND, OR and NOT functions. Next, properties of the most commonly used integrated circuit series, the LS-TTL and HC CMOS, are studied. Finally you will use these basic chips to construct more complex circuits.

1. Mickey Mouse Logic and the Totem Pole Output

These are among the simplest logic devices. They are useful in their own right from time to time. Also, they demonstrate the input circuitry to the most commonly used TTL chips, the low-power Schottky (LS-TTL) family. Here we use standard 1N914 signal diodes instead of the faster Schottky diodes used in LS-TTL gates.

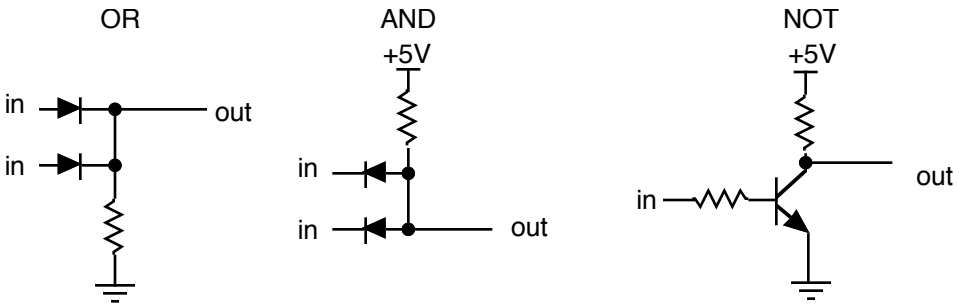


Fig 12.1

The AND circuit illustrated in Fig 12.1 is similar to the input of the 74LS00 NAND gates. Build it and confirm the logic function experimentally and record your results in a truth table. When you test this gate, use 0 V and 5 V for logical false and true.

TTL expects at least 2.0 V input for a high (true) input and guarantees at least 2.4 V for a high output. CMOS requires at least 3.5 V for a high input and delivers at least 4.9 V for high output. In what ways do these circuits disobey these criteria?

Try driving an LED. First connect an LED from the output to ground and observe what happens. Next connect the LED from the output to +5 V through a 2.2 k resistor. This will invert the output signal. Does it work better this way? Why or why not?

Build the NAND gate shown in Fig. 12.2 which is very similar to an LS-TTL circuit. Notice the totem pole output. Make a truth table of its operation showing the voltages on the bases of the totem pole transistors as well as the output voltage. Do the voltage levels conform to TTL criteria? What is the effect of leaving an input unconnected?

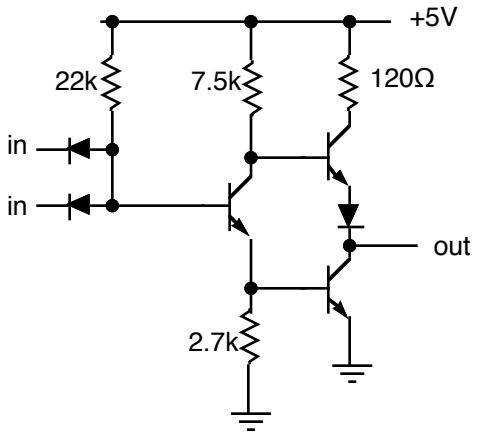
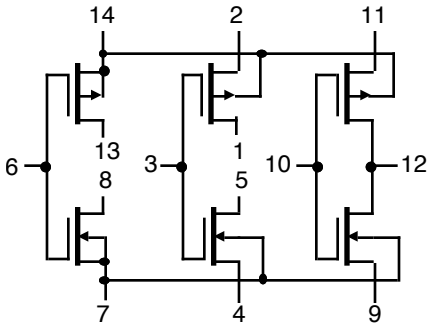


Fig 12.2

2. CMOS Logic Gates

Elementary logic gates are even more easily built from CMOS field effect transistors. Matched complementary pairs of MOSFET transistors are packaged in the CD4007 chip. Fig 12.3 shows its pin arrangement. For the following experiments always tie pin 14 to +5 V and pin 7 to ground.



CD4007 MOS Transistor Array

Fig 12.3

Two inverters are shown in Fig 12.4. The first, using a passive pull-up resistor, is like an "open drain" output. Try it and measure its output voltage as a function of time with a 1 kHz square wave input and a 100 kHz input. Now crank up the frequency as high as you can to see what happens.

If you have 1 kHz and 100 kHz outputs on you breadboard use them. Otherwise, use the F34 at 5 Vpp with a dc offset of 2.5 V to give 0 and 5 V logic levels.

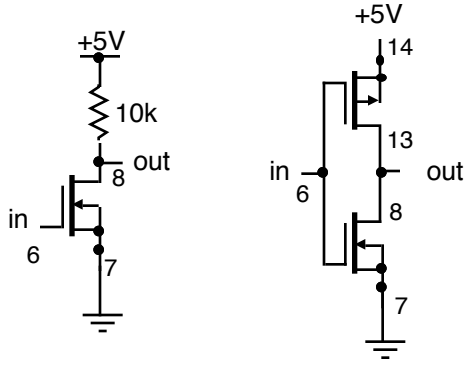


Fig 12.4

Now connect the right-hand inverter circuit. It uses a complementary MOSFET as an "active" pull-up. Also look at its output as high and low frequencies and compare with the passive pull-up.

The NAND gate shown in Fig 12.5 is simple. Make it. Test it.

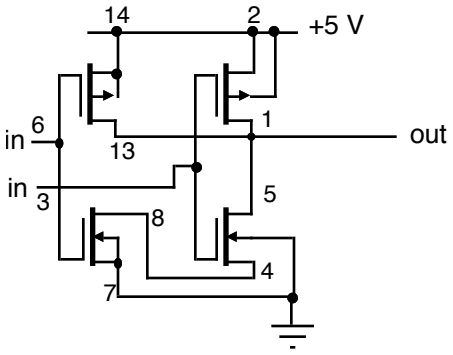


Fig. 12.5

3. CMOS Logic Chips

a) "Connect all Inputs[1]." When you use CMOS chips like the 74HC00 NAND gates it is important to tie all inputs of all the gates on the chip to a definite logic level. Otherwise the input logic level will be indeterminant. In order to convince yourself of this, connect the 74HC00. Pins 7 and 14 should be ground and +5 V respectively. You are using only one NAND gate so ground the other six inputs of the unused NAND gates. Tie one input of a NAND to HIGH and connect about 6 inches of wire to the other input. Leave the other end of that long wire dangling in air. Watch the output of the NAND gate as you wave your hand around near the long wire. Try touching your other to +5 V as you do this waving. What you see should convince you that you can't rely on the unconnected inputs of CMOS gates. Now replace the chip with a 74LS00 chip and notice the difference. (*Turn power off before changing chips.)*

Indeterminate inputs can also cause both transistors of the complementary pair to conduct and consequently draw a lot of power from the supply. Intermittent surges of load to the power supply can make glitches. You can test this excessive current consumption using the setup below. First connect all the NAND inputs to ground and verify the low power consumption on the meter's most sensitive scale. Then put the meter on the 150 mA scale. As you drive the inputs with a voltage intermediate between the LOW and HIGH levels appropriate for CMOS, the measured current should go up abnormally. Try this with the 74LS00 too.

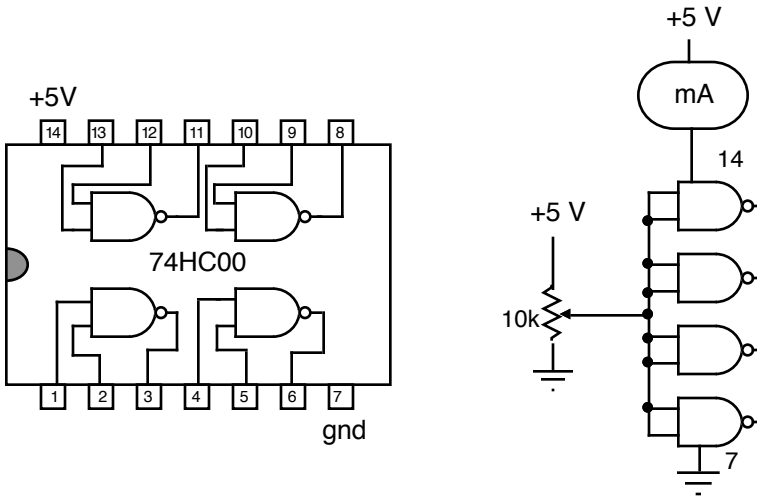


Fig 12.6

[1]You can remember this rule by recalling the famous movie entitled "Destroy all Monsters."

b) "NAND is All You Need[2]". NAND gates are very useful because all other logical functions can be built up from them. Because of DeMorgan's theorem we know that, "NAND is all you need." As an exercise design and build the AND function from NAND gates so that you can light one of the LEDs on you breadboard when both of the inputs are high. Next, build the OR function that lights an LED when one or both of the inputs is low. (I.e., OR with negative logic input and positive logic output—this is easy.) Verify they give the results you hope for.

Design, build and test an XOR circuit (exclusive OR function) using only NAND gates.

4. The Latch Flip-Flop

The flip-flop is an essential element in digital and computer circuits. Its ability to store information entered on its input after the input has been changed is useful for memories, registers and counters–almost every component of digital and microprocessor devices. The simplest flip-flop is the NAND latch shown below. There are three useful states: set, reset and "no change." Fig 12.7 shows the "no change" state with both inputs high. This is its quiescent resting position and the output could be either high or low depending on whether the previous state was set or reset. The set state is when is low and high The reset state is when is low and high. The fourth state, both inputs low, results in an indeterminant output when the inputs return to their quiescent state. If both and are brought low and then raised, the output retained by Q depends on which input goes high first. This condition is not very useful and should be avoided.

Build the NAND latch shown in Fig 12.7 and make its truth table. See if you can verify the indeterminancy of the fourth state. You can use either 74HC00 or 74LS00.

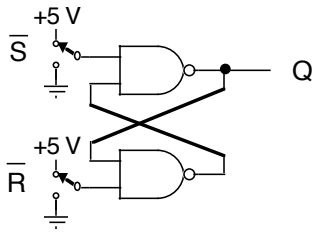


Fig. 12.7

The little bars above S and R indicate that the set or reset condition occurs when the respective input is grounded instead of at 5 V.

[2]It's rumoured that the Beatles once wrote a song with this title. Unfortunately, the title was changed at the last moment before release for marketing reasons.

Homework

Using only NAND gates, design the circuits for the logical operations (1) AND, (2) OR with inverse logic inputs, and (3) XOR for part 3(b).

Congratulations, you made it to **The End!**