Physics 334  
Notes for Lab 7 – Digital Gates  

Lab manual sections 13-0, 13-1, 13-2, and exercises below.  

Advice:  

- Use the fixed 5 volt supplies for your power in digital circuits, rather than the variable outputs.  
- Contrary to the advice given for analog circuits, set up the power busses along the sides of the breadboard, using one for +5 and one for 0 volts. Ground the 0 volt side. Put bypass caps, one 15μF tantalum cap and one 0.1μF mylar or ceramic cap across the breadboard power supply strips.  
- Connect all +5 circuit points with red wires and all ground points with black wires. This helps when you try to trouble shoot.  
- Power supply connections in digital circuits are almost never shown in the circuit diagram. As a way to remember that they are necessary, always make the power connections first, before you wire up the signal paths.  

Additional exercise 1. Wire up the simplified discrete logic gate shown below.  

![Logic Gate Diagram](image)

Apply +5 or 0 volts to each of the inputs, and make a table showing the resulting voltages that all combinations of inputs produce at the output. From this table, make a truth table (using 1’s and 0’s, or H’s and L’s) and deduce what sort of a gate this is: AND, OR, NAND, NOR, or XOR.  

Describe in your report how the circuit works. That is, explain what the transistor does in order to generate the kind of output you see, how much current flows in the base and collector legs of the transistor, and what you expect the threshold voltage to be. The threshold voltage is the voltage applied at the input which is necessary to cause the gate to go from its low to its high state (or vice versa).  

What happens if both inputs are left unconnected (open)?  

Now turn to Lab 13: Digital Gates  

13-0. Read through the preliminary material on pages 309–310, but do what is stated here in the notes, rather than what the manual suggests. Ignore the sections on “LED indicators” and “Switches”; they refer to a breadboard we do not use.
The SYNC output from the function generator is a square wave of 5V peak-to-peak amplitude. Verify this by looking at the SYNC output on the ’scope. Use DC coupling and see if the SYNC really varies between 0 and +5 volts.

Study the laminated sheet which describes the operation of the logic probe. Connect the power clips to the power supply, with +5V to red and 0V to black.

Run the SYNC signal to a place on your breadboard, and touch the tip of the probe to this signal. Use the PULSE setting and try a very low frequency setting (< 0.3Hz). Watch how the probe responds in correspondence with the signal seen on the scope. Turn up the frequency (slowly, to > 1kHz) and note what the probe does.

Try the MEM setting on the probe with very low frequency from the generator (≈ 0.1Hz) and compare what you see with the behavior you get with the PULSE setting.

13-1. In part (a) just connect the inputs with jumpers to either +5 or ground instead of using switches. Fill in the truth table using the logic probe for “Logic Levels” and your voltmeter for “Voltages”. Be sure to connect all CMOS inputs on the chip, as stated. Open CMOS inputs are very sensitive to noise and tend to float to the threshold voltage, which in turn causes the gates to draw lots of current, as the MOS transistors are biased partially on.

In part (b) Do “1-TTL” as stated. Do “2-CMOS, Floating input: effective logic level in,” as stated, but do not do the section on power consumption.

13-2. Use a TTL NAND chip (74LS00) for all parts (a, b and c) and instead of lighting an LED, use your logic probe or ’scope to see if you have a HIGH output. For part (c), the following gives a 4 gate realization of XOR. Verify to yourself that it should work as advertised.

Additional exercise 2: Multiplexer Circuit. For the rest of this lab follow these notes rather than the manual lab exercises.

Wire up the 74LS151 8-input multiplexer as shown below. Connect an LED with current-limiting 390Ω resistor to the \(\overline{Q}\) ("Q-bar") output as shown. Since LOW TTL outputs sink current better than HIGH outputs source current, we use \(\overline{Q}\) to run the LED via a pullup resistor.
Set up a binary number (address) on the \( A, B, C \) inputs and ground the \( \text{STROBE} \) (i.e., \( \text{ENABLE} \)) input. Momentarily bring each one of the \( D_0 \) to \( D_7 \) to ground until you see the LED turn on and off. The input which causes this response is the input selected by the address. Repeat this procedure for several different \( A,B,C \) addresses until you understand the chip’s function. Note: the LED lights up when \( \overline{Q} \) is LOW which means that \( Q \) is HIGH.

Add the inverter circuit shown below, and make the connections to the data input lines as indicated. Use one of the 74LS00 NAND gates wired as an inverter. Don’t forget the power supply connections! The addition of the gate and wiring turns the multiplexer into a “31-day machine”: the LED lights whenever a month number (Jan. = 0001, Feb. = 0010, Mar. = 0011, …) is applied to the address lines \( A_0 \ldots A_3 \).

Try the circuit out and draw a truth table (i.e, the states of \( A_0 \ldots A_3 \) vs. \( Q \)). In the table, group the the months in pairs according to the patterns in the 3 most significant bits, and note how for each pair the “31-ness” of the month depends on the least bit \( A_0 \). Compare this with the wiring of the inverter circuit and discuss how the circuit does indeed indicate which month has 31 days.