EC410 Laboratory 5 – Fun With Logic Gates: Making Them

Background

In our last EC410 lab, we investigated the current-voltage characteristics of the MOSFET. The focus was on the transistor as a single, three-terminal circuit element. In this lab, we expand upon our MOSFET by using these devices to construct logic gates. While MOSFETs are sometimes used to implement analog circuits, such as operational amplifiers and sensors, their most prevalent use is, *by far*, as the basic cellular building blocks of digital logic chips such as microprocessors, memory arrays, digital signal processors, microcontrollers, and network communication hardware. To help illustrate this point, consider the comparative number of transistors within each of the following integrated circuits:

Integrated Circuit	Type of Device	No. of Transistors
CD4007	analog or digital	6
LM741	analog	22
Intel Pentium-4 Processor	digital	42,000,000
Intel Core 2 Duo Processor	digital	410,000,000

Note for “Out of Sequence Students”:

In the Spring and Fall 2007 EK307 labs, students (hopefully) learned how complex decision-making operations can be performed using a combination of AND, OR, NAND, NOR, and NOT logic gates. If you did not take EK307 at Boston University, or have forgotten the content the logic-gate lab, or completed an older version of the EK307 labs, you might want to review the logic gate lab. This reference handout may be found at http://www.bu.edu/eng/ec410. What follows in the next section is an abbreviated version of excerpts from the EK307 lab.

A Brief Review of Logic Gates

A logic gate is a binary device which interprets voltage values of 0 V and 5 V as representing, respectively, the binary digits **0** and **1**. Some battery powered devices use a 3-V level instead of 5 volts to help conserve power, but in this lab we’ll assume the standard logic **1** = 5 V .

In a 5-V system, any voltage that is smaller than the midpoint value of 2.5 V can, in principle, be interpreted as a logic **0**, and anything above 2.5 V as a logic **1**. In a well-designed logic system, however, a given voltage is held as close as possible to one of the defined values of 0 V and 5 V, except, of course, in transition from one value to another.

Truth Tables

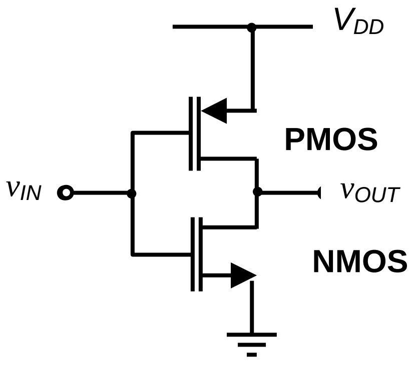
Truth tables are often used to describe the way in which logic gate functions. A truth table contains a complete set of all possible binary input combinations as well as their corresponding output values. The relationship between the input and output values depends of the specific type of gate. For your reference, the truth tables for five common gates types: AND, OR, NAND, NOR and NOT are provided below.

AND				OR				NOT				NAND				NOR			
Input		Output		Input		Output		Input		Output		Input		Output		Input		Output	
0	0		**0**	0	0		**0**	0			**0**	0	0		**1**	0	0		**1**
0	1		**0**	0	1		**1**	1			**1**	0	1		**1**	0	1		**0**
1	0		**0**	1	0		**1**					1	0		**1**	1	0		**0**
1	1		**1**	1	1		**1**					1	1		**0**	1	1		**0**

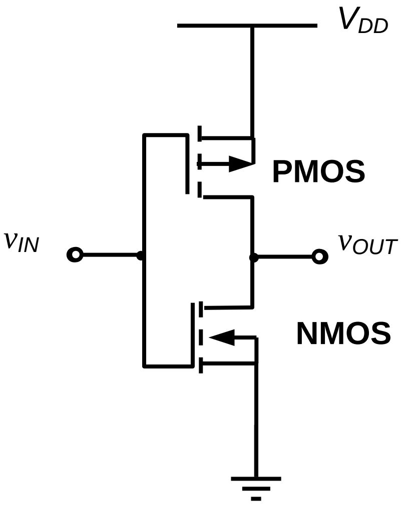
The CMOS Inverter

The principal building block of all modern gates and, by inference, most complex logic circuits, is the complementary metal–oxide–semiconductor (CMOS) inverter. The details of the CMOS inverter are discussed in your text; a brief summary is provided here.

A CMOS inverter, shown below, consists of an NMOS device (n-type, enhancement-mode MOSFET) controlled by the inverter's input node, and an active PMOS (*p*-type, enhancement-mode MOSFET) functioning as the pull-up load. The PMOS device also has its gate connected to the inverter input. This configuration can be summarized by either of the following simple diagrams:



OR

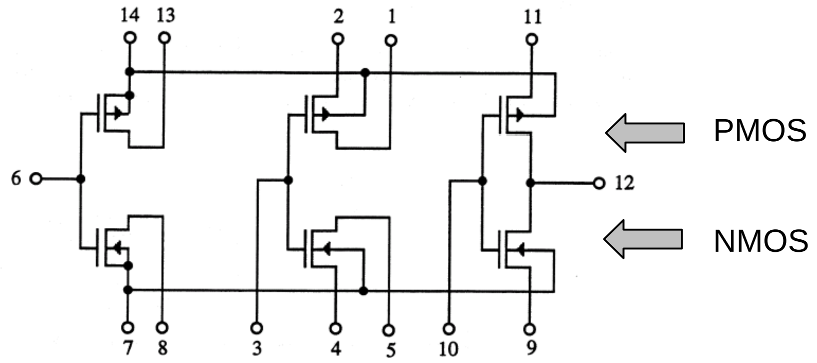


The gate-to-source voltages of the transistors are such that each one operates either in cutoff (i_D = 0), or in the deep-triode region ($v_{OUT} \approx 0$). In this way, the transistors emulate switches, rather than amplifiers, in that they are forced into either and open (off) or closed (on) state. When v_{IN} is high, for example (i.e., equal to 5 V), then M_1 will be "on", and M_2 will be forced into cutoff. Under these conditions, v_{OUT} will be zero, or logic **0**. Conversely, when v_{IN} is low (i.e., equal to 0 V), then M_1 will be in cutoff and M_2 will be "on". In this latter state, v_{OUT} will be forced to 5 V , or logic **1**).

An important feature of the CMOS inverter is that no current flow can flow into the v_{IN} terminal except the current needed to charge or discharge the gate-to-source capacitances of M_1 and $M_{2.}$ The latter will transpire during the transition of the inverter between its high and low states. More importantly, drain current can flow through both MOSFETs *only* during the transition between HI and LOW states. When the inverter has settled into *either* a HI or LOW output state, one of the two transistors will be in cutoff, so that no current will be extracted from the V_{DD} source. This feature makes the CMOS inverter a valuable building block for logic circuits incorporating thousands of transistors.

Laboratory Goal:	Measure the transfer characteristic and switching speed of a CMOS inverter; Design one or more logic gates using NMOS and PMOS devices.
Learning Objectives:	Properties and function of CMOS circuits.
Suggested Tools:	CD4007 waveform or function generator, multimeter, light-emitting diodes (LED), oscilloscope, op-amps.

Your lab kit contains at least one CD4007 Dual Complementary Pair/Inverter chip or its equivalent. This chip has the layout shown below. Note that the NMOS and PMOS transistors on the right in the diagram are already wired in the form of a CMOS inverter, while the other transistors share only their gate inputs.



Level 1:
Build a simple CMOS inverter, and display its input-output transfer characteristic (that is, a plot of v_{OUT} as a function of v_{IN}) over the range $0 < v_{\mathrm{IN}} < 5$ V. Your V_{DD} supply should be set to 5 V.

Level 2:
Use the devices on the CD4007 chip to design a two-input version of one of the following logic gates: AND, NAND, OR, NOR. Drive an LED from your gate's output to indicate the its HI and LOW output states. Verify the logic operation of your gate.

Level 3:
Attach a 0.001-μF load capacitance to the CMOS inverter of Level 1, and measure its dynamic response time. That is, for a nearly perfect 0- to 5-V step-function input, measure the time it takes the output to rise from $0.1V_{DD}$ to $0.9V_{DD}$ or, alternatively, to fall from $0.9V_{DD}$ to $0.1V_{DD}$. These delay durations are called, respectively, the *rise time* and *fall time* of the inverter. (The 10% and 90% levels are accepted in industry as he threshold levels for determining rise and fall times.) From your measurements of rise and fall time, estimate the K values of the NMOS and PMOS transistors in your inverter.