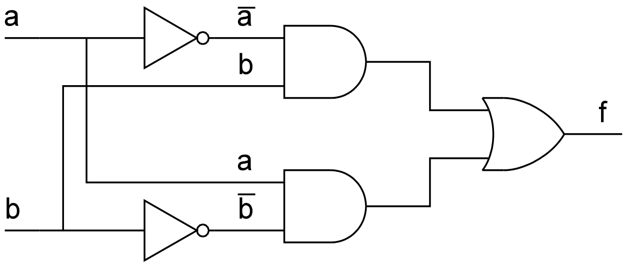
ENGR-354
HW # 10

The goal of this homework is to become familiar with the process of implementing a logic function using standard integrated circuit logic components (chips).

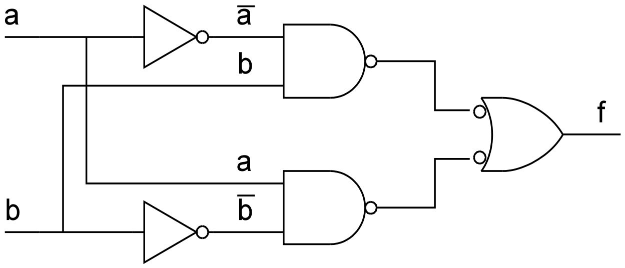
The task for this homework is to implement (that means construct) the circuit defined by this boolean relationship: f = a’b or ab’
You likely recognize this as an Exclusive OR (XOR) function. (While there is an integrated circuit in your collection that has XOR gates in it, for this assignment we will not use that part)

Here are the steps to follow when creating and implementing a logic circuit:

1) Draw a logic diagram. Use AND and OR symbols to show the desired logic:

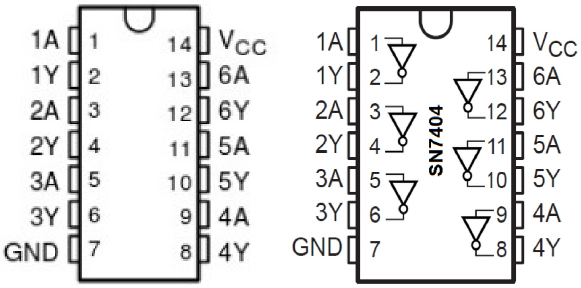


2) But, we don’t have AND and OR gates in our kit. We do have NAND and NOR gates. So in this problem convert AND and OR gates to NAND . Put a bubble on the output of the AND gates. Then put a bubble on the other end of the associated wires. Note that the logic doesn’t change assertion level if there is a bubble on each end of the wire.

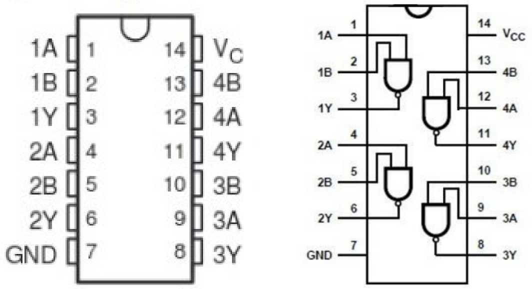


3) Next, have a look at the component data sheet or pin-out summary sheet that is posted on the class web page to find pin numbers to use for making connections. The inverter is a 74HC04 and the NAND gate is a 74HC00. There are 6 invertors in one 14-pin package and 4 NAND gates in one 14-pin package:

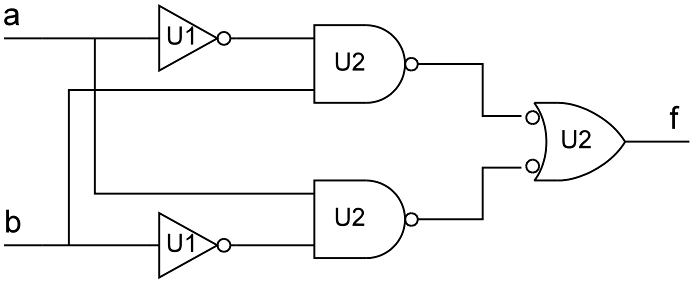
74HC04 & 74HC14
Six inverters



74HC00
Quad 2-input NAND

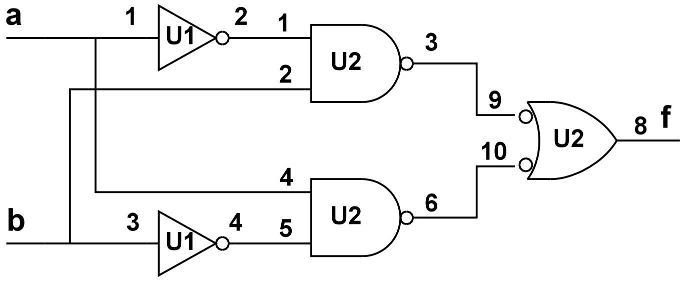


4) More than one or two integrated circuit (IC) packages often are needed to implement a circuit. It is helpful to annotate each gate on the diagram with a package reference number to assist in keeping track of which IC is which on your breadboard. By convention, the character U followed by a number is used:



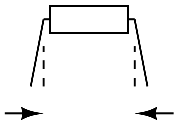
5) Step 5 may seem like extra work, but trust me, it will save you time while wiring up the circuit and make it much easier to debug a circuit if it doesn't work properly after you assemble it.

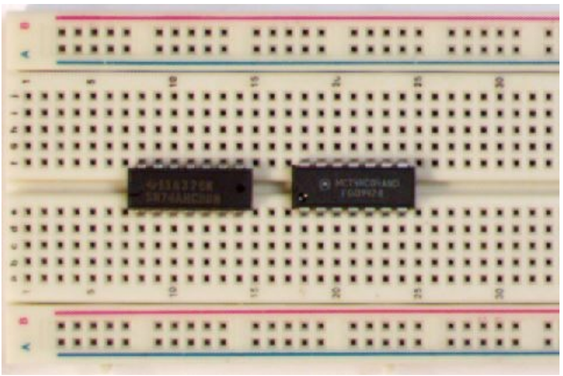
Write pin numbers on each input and output of every gate in your circuit diagram following the pinout diagrams in step 3 above:



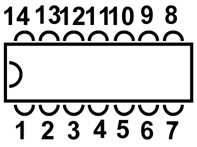
6) Find the needed parts and insert them into the breadboard. Sometimes the leads of an IC are spread apart wider than the hole spacing of the breadboard. To allow easy insertion bend the leads to be more parallel and match the breadboard spacing.

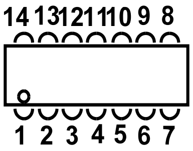
End view of an IC





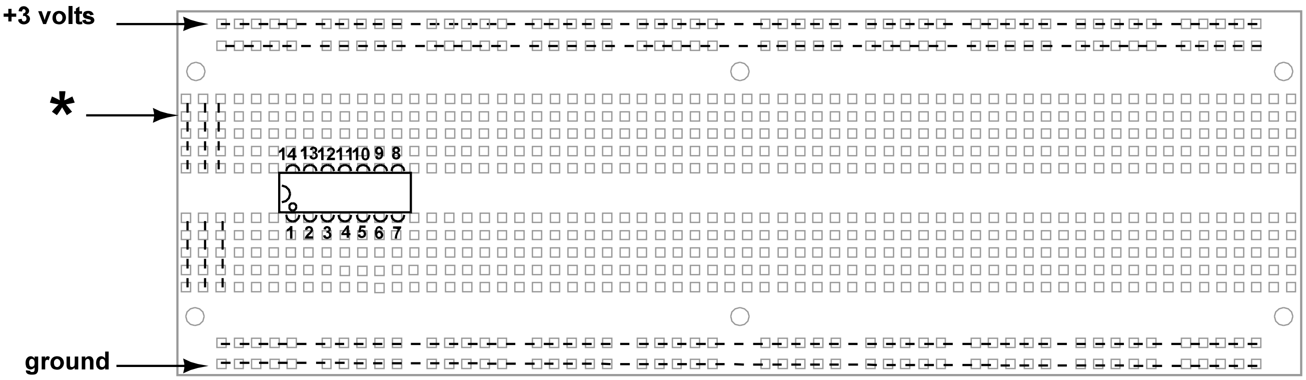
Pin one of an IC is typically on the end that has a small U cutout or a dot by it:





When positioned as shown below, rows of holes at the top of the bread board and rows at the bottom connect together for the full length of the bread board. Use those for hooking up power and ground. We refer to those as power busses.

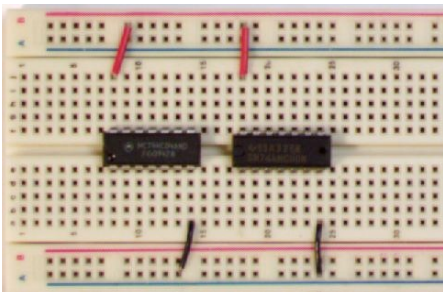
Vertically there are 5 holes in a column that are connected together. Use those to make connections to the IC pins.



*** groups of 5 vertical holes connect together but not to adjacent groups**

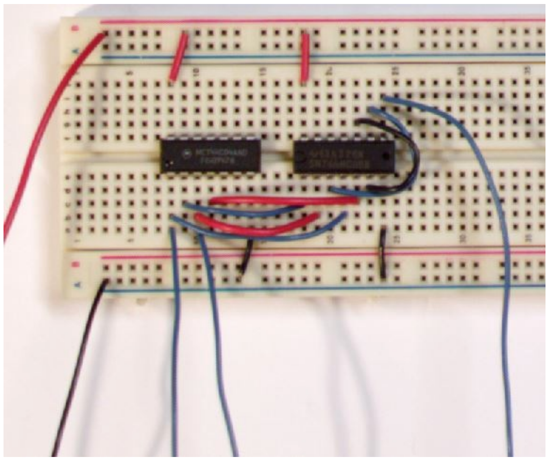
horizontal rows connect together for the full length of the breadboard

7) Connect power and ground of each IC to the appropriate power bus. I recommend using short wires that lie flat against the breadboard for the power connections. For electronic circuits, red wire is typically used for positive power and black wire for ground connections.



The wire stripper has holes matched to wire size. The supplied wire is 22 gauge, so use the 22 hole.

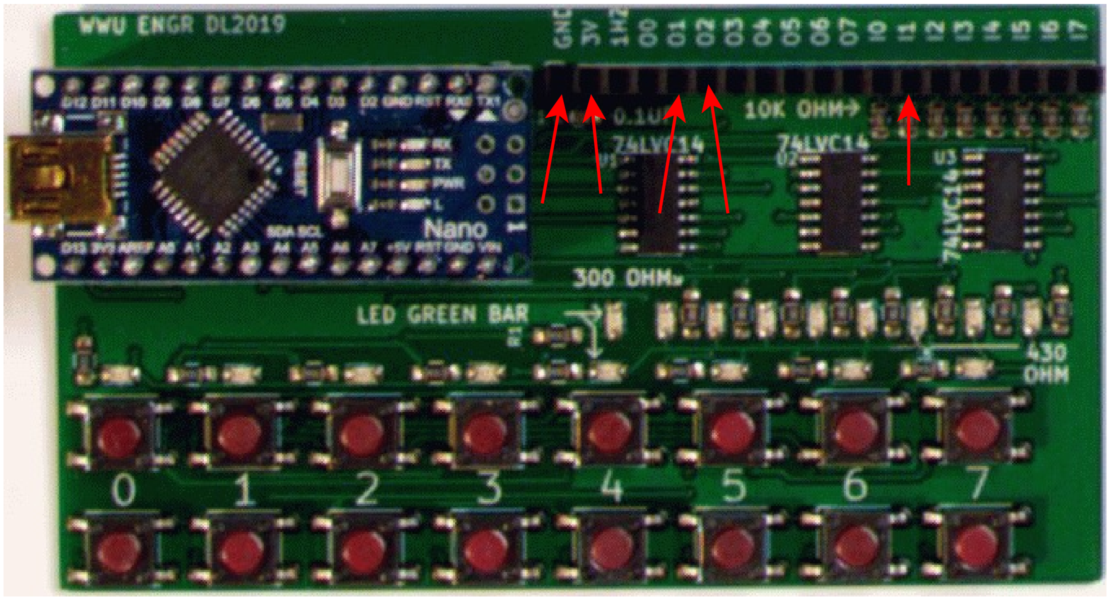
8) Then make connections to complete the circuit. Use wires a few inches long to connect power, input switch signals, and output signals going to LEDs on the logic interface board.



Making connections between pins of the chips using short wires tends to make trouble shooting easier than when long loopy wires are used. Wiring can become a tangled jungle and hence hard to debug when longer wires are used.

9) Connect wires to the logic interface board.

Five connections to the logic interface board are needed. Assume that switch 1 and 2 will be used to create the a and b logic signals for input to the XOR function and that LED1 will be used to display function f. From left to right the red arrows point to where wires should be plugged in to connect: ground, +3 volt power, switch 1, switch 2, and LED 1 respectively.



Power for the logic board is obtained with a USB cable connected between the USB port on the Arduino Nano and the supplied wall wart (or any computer USB port).

To turn in as homework # 10:

Submit two things to the D2L dropbox. 1) a statement declaring success or not of making the circuit work per the function definition along with any comments you wish to make 2) a photo of your completed circuit on the breadboard.