Logic Gates Lab – Instructions

Overview

In this lab, you will create a simple NOT gate using a transistor and resistor (Part 1). Then, you will test the operation of pre-built integrated circuit (IC) chips with NOT, AND, OR gates, and XOR gates inside of them (Parts 2-5).

Preparation

1. Get your lab kit that you purchased from the bookstore.
2. You should have already reviewed the "connection diagram" in each of these *datasheets*:
    - 74LS04 Hex Inverter Gates
    - 74LS08 Quad 2-Input AND Gates
    - 74LS32 Quad 2-Input OR Gates
    - 74LS86 Quad 2-Input Exclusive-OR Gates
3. (Optional) If you are doing the lab on your own, you may follow along with one of these videos:
    - TA Walkthrough (better audio)
    - Instructor Walkthrough (more details)

Required Parts

- Breadboard (1)
- Red LED (1)
- 220 Ohm resistor (2) – color bands: red, red, black, black, brown
- Male-to-male jumper wires (20)
- NPN Transistor [PN2222] (1)
- Arduino Uno (1) and USB cord (1)
- The 4 Logic Gate Chips:
    - 74LS04 NOT Gate Integrated Circuit (1)
    - 74LS08 AND Gate Integrated Circuit (1)
    - 74LS32 OR Gate Integrated Circuit (1)
    - 74LS86 XOR Gate Integrated Circuit (1)

Note: The 4 chips listed above are not listed in the parts list because they are added to the kit by the bookstore, but they should still be in your kit. They may come in a dark Ziploc bag, pressed into foam. Ask your instructor if you need help finding them or need to borrow any parts. You may also buy the chips separately using this link: https://www.byuistore.com/Class-Kit-ECEN-101-106-Logic-Gate-Chips.

Background: Logic Gate Chips

The **74LS04 NOT chip** implements the NOT function. The output (F) is the opposite of the input (A).

A	F
0	1
1	0

The **74LS08 AND chip** implements the AND function. The output is HIGH only if *all* the inputs are HIGH.

A	B	A·B
0	0	0
0	1	0
1	0	0
1	1	1

The **74LS32 OR chip** implements the OR function. The output is HIGH if *any* of the inputs are HIGH.

A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

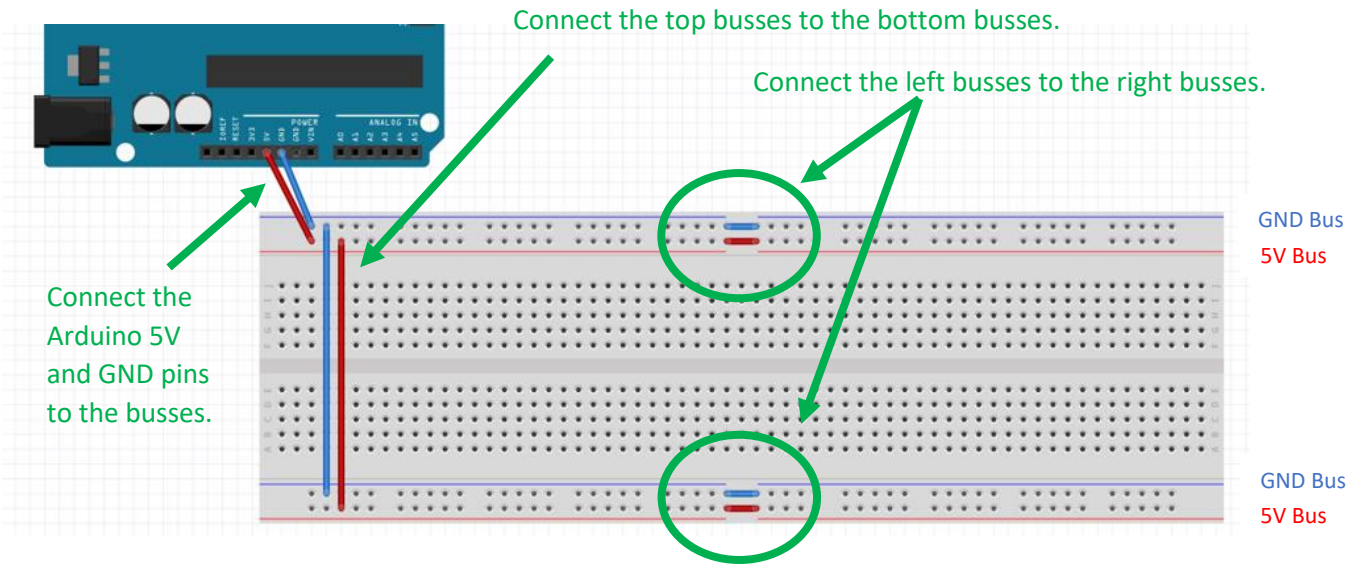
The **74LS86 XOR chip** implements the Exclusive OR (XOR) function. The output is HIGH when an *odd* number of inputs are HIGH.

A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

Provide Power to the Breadboard

Use your Arduino to supply power to your breadboard by connecting the busses:

1. Connect GND on your Arduino to a hole on a Ground Bus on your breadboard.
2. Connect that Ground Bus to the other Ground Busses on your breadboard.
3. Connect 5V on your Arduino to a hole on a High Voltage Bus on your breadboard.
4. Connect that High Voltage Bus to the other High Voltage Busses on your breadboard.



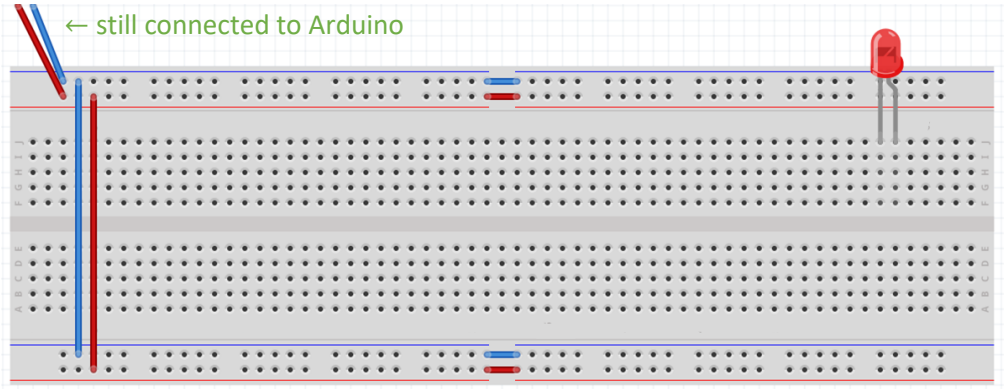
Cut the power by disconnecting your Arduino from your laptop's USB port. Whenever you are wiring a circuit, keep the Arduino disconnected. When you are ready to test a circuit, supply power by connecting the Arduino to your laptop via the USB cord, then disconnect it again after you finish testing.

Tip: Be sure that your breadboard isn't upside down.

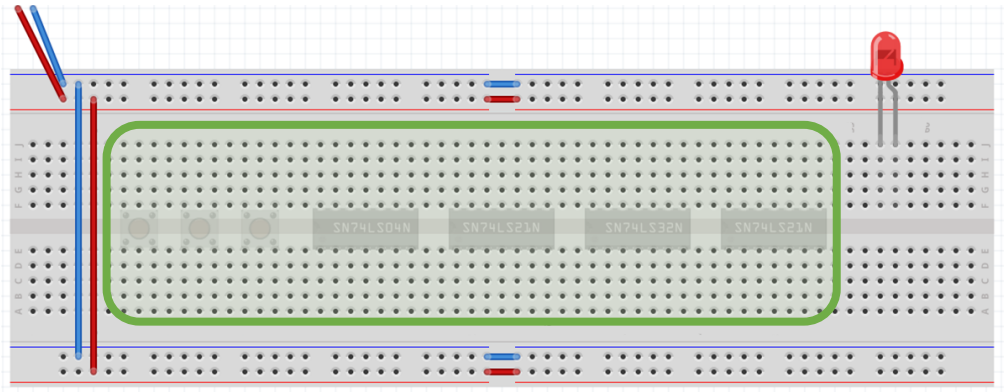
Testing the LED

The LED will be the "output" of your circuits. The negative side of the LED will always be connected to ground (low voltage). The positive side of the LED will have a high (True) or low voltage (False) depending on if we want to turn on or off the LED. **LEDs require resistors to not burn out.**

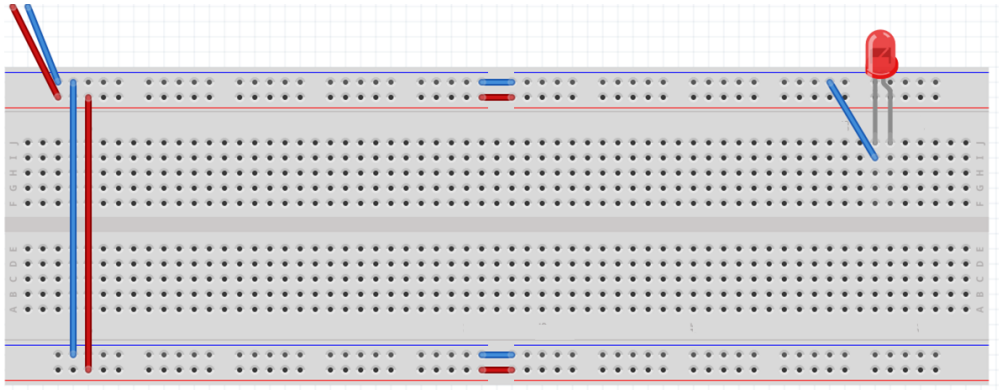
1. Find a red LED. Note that there is a long leg and a short leg. The long leg is the positive side, and the short leg is the negative side. Place the LED into the breadboard with each leg in a different column and with the long leg to the right. The "bent" leg in the diagram below is the long leg.



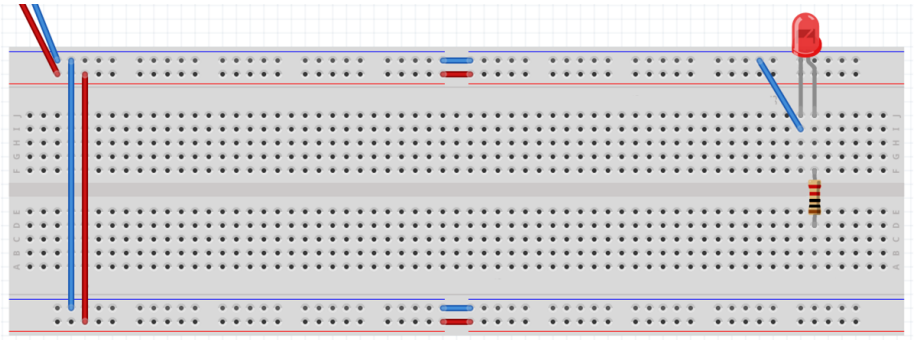
Tip: Leave room in the center of the breadboard for the chips and for buttons you will add in the next lab, shown by the green area. *Don't add anything to that area yet.* Use the visual below to know about how much space to reserve as you place the LED on the far-right side.



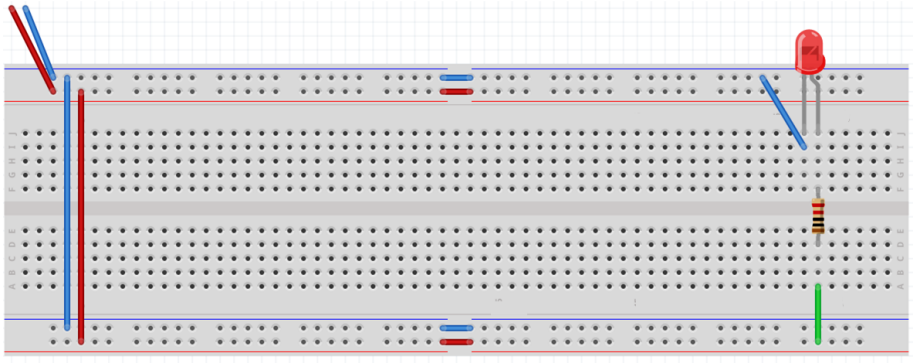
2. Connect the negative side of the LED to a low voltage (GND) with a wire from the GND bus to the same column as the negative side of the LED.



3. Find your 220 Ohm resistors. The color bands are red, red, black, black, brown (or reversed).
4. Plug a 220 Ohm resistor into the breadboard across the middle gap in the same column as the right leg (positive side) of the LED.



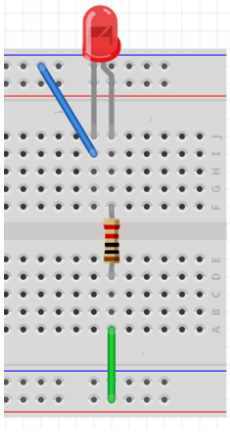
5. Connect the other end of the resistor to a high voltage to turn on your LED by plugging one end of a wire to the same column as the resistor on the bottom half of the breadboard and the other end to a High Voltage bus. (See the green wire below.) This sends a "True" signal to the LED.
6. Turn on power by connecting your Arduino's USB cable to your laptop. The LED should be on.



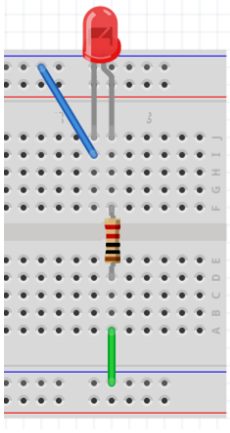
<u>If your LED doesn't turn on:</u>

- Make sure the Arduino is connected to your laptop.
- Check that the power busses are connected across the gaps.
- Make sure the LED is not backwards (long leg is positive – on the right).
- Check the resistor colors.
- Ask the instructor for help.

7. Move the wire you just added back and forth between the Ground Bus (False) and the High Voltage Bus (True). The light should turn off (False) when connected to the Ground Bus, and the light should turn on (True) when connected to the High Voltage Bus.



HIGH (True) - Light should be ON

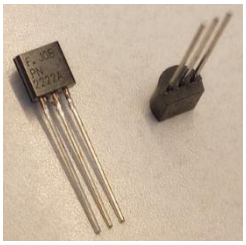


LOW (False) - Light should be OFF

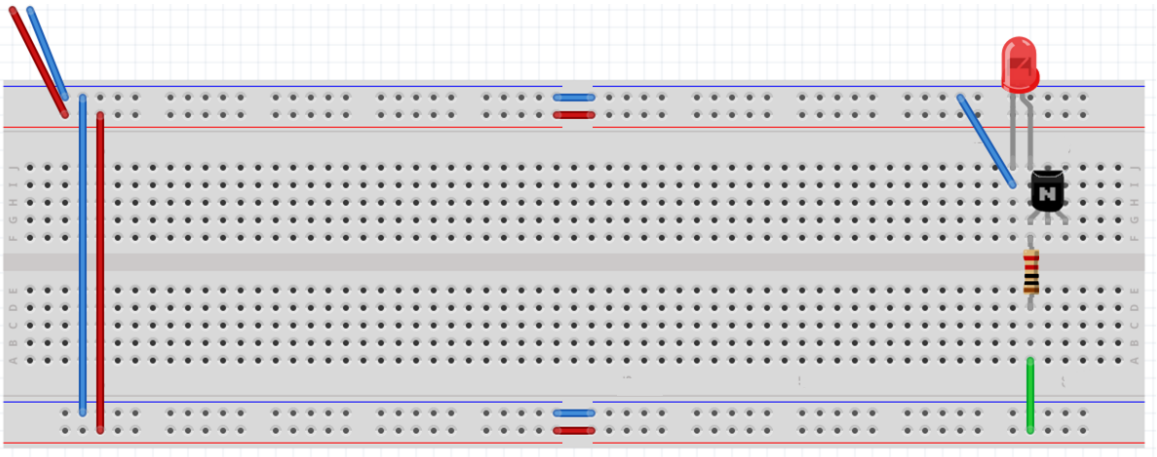
8. Move the green input wire back to the High Voltage Bus (True).
9. Disconnect power (unplug the USB).

Part 1: Building a Transistor NOT Gate

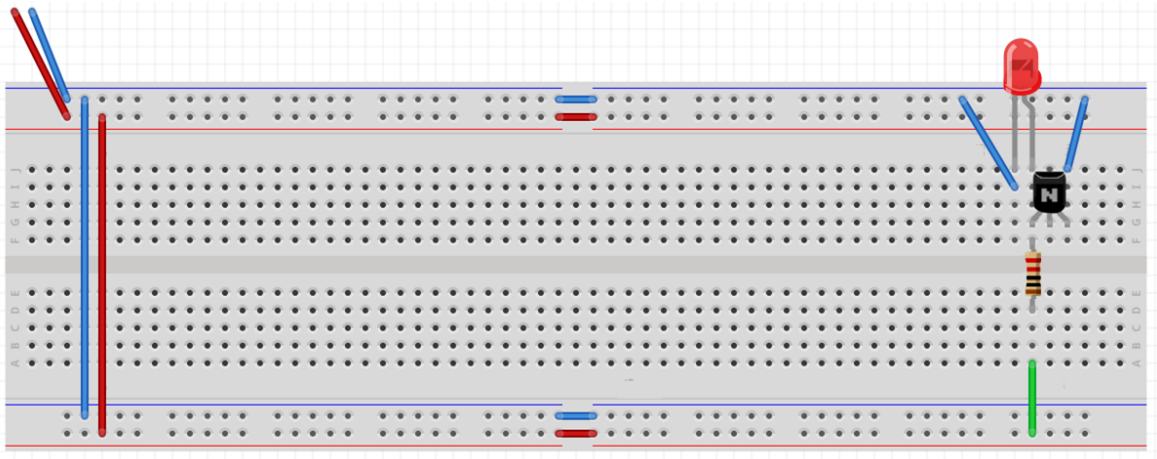
You will now create a NOT gate using a transistor and another resistor.



1. Find an NPN transistor. They are black with three legs. They may be labeled “PN2222” or “PN 2222A.”
2. With the flat side of the transistor *away* from you, plug the left leg into the same column as the long leg of the LED with the other two legs being in the two columns to the right of that. (See image below.)

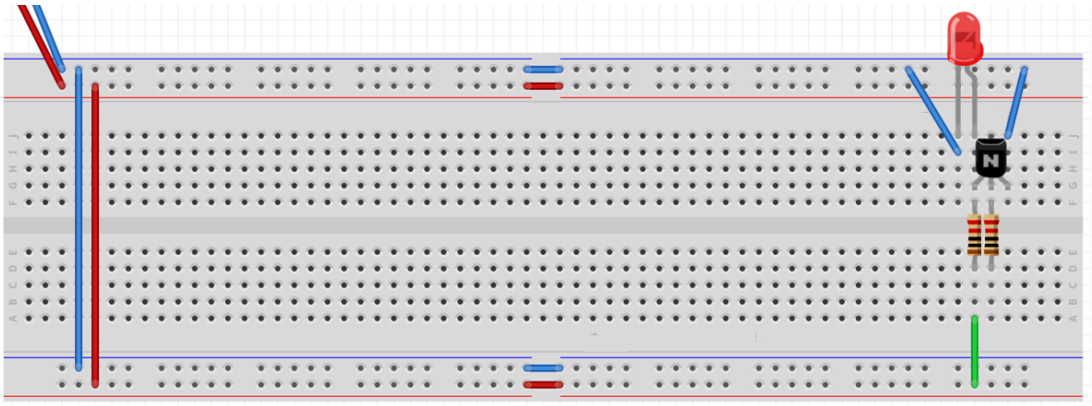


3. Connect the right leg of the transistor to Ground by plugging one end of a wire into the same column as the right leg and the other end into the Ground Bus.

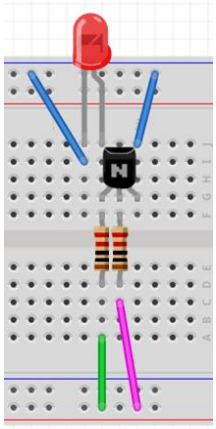


Why are you connecting the transistor this way? The transistor acts as a switch. When the transistor is open, the LED will remain connected to a HIGH voltage through the existing resistor and green wire (attached to the *left* pin of the transistor). When the transistor is closed, the LED will connect to a LOW voltage through the transistor with the wire you just added (connected to the *right* pin of the transistor). Now, you will add a resistor and wire to the *middle* pin of the transistor, which will control whether the transistor is open or closed.

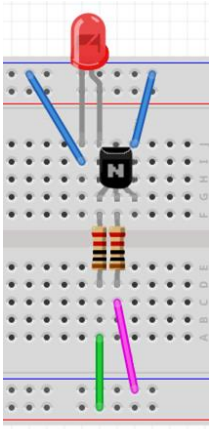
4. Add another 220-ohm resistor to the same column as the *middle* transistor leg, across the gap.



5. Using a wire, connect the middle-leg resistor (the resistor on the right) to a HIGH voltage. This will be the input to your NOT gate (shown in magenta below).
6. Power on the circuit by connecting the USB cord to your laptop. **The LED should be off.**
7. Switch the input wire (in magenta below) back and forth between high and low voltage. When the input is HIGH, the output LED should be LOW (off). When the input is LOW, the output LED should be HIGH (on).



Input (magenta wire) HIGH
Light should be OFF (LOW)



Input (magenta wire) LOW
Light should be ON (HIGH)

Congratulations! You have built a NOT gate using a transistor and resistors.

Record – Truth Table: In your report, enter the values you observed with your LED into the truth table for Part 1. "A" is your input (magenta wire). For the row where "A" is 0, that means your magenta wire was connected to Ground Bus. For the row where "A" is 1, that means your magenta wire was connected to the High Voltage Bus. The "Actual Output" is the LED. It should match the truth table of a NOT gate.

Part 1: Transistor NOT Gate

Please enter in the values you observed after building a **NOT** gate with a transistor:

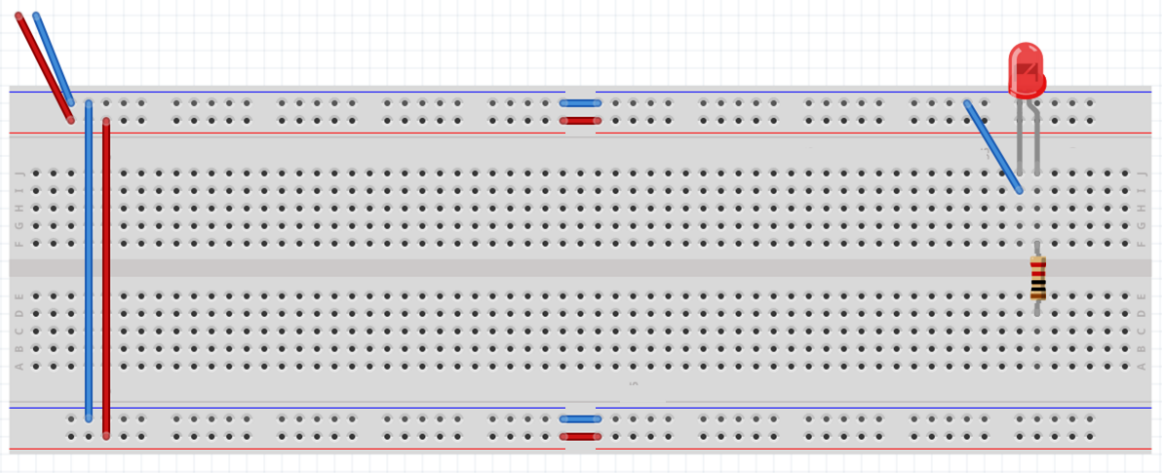
A	Actual Output
0	
1	

Enter 1 for the light being on (true). Enter 0 for the light being off (false).

Record – Photo: Take a photo of the NOT gate you built for Part 1. You will upload this to I-Learn.

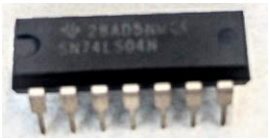
For the remaining portions of this lab, you will be using pre-built logic gates which have the transistors inside of *integrated circuit* (IC) chips.

8. Remove everything except the parts shown below:



Inserting your Logic Gate Integrated Circuit (IC) Chips

Your lab kit should come with a set of logic gate IC chips in foam in a dark bag. The image below shows an example of one of those chips. Each logic gate chip has 14 metal legs called pins – 7 on each side. Don't confuse them with the other two black chips in your kit that look similar but have 16 pins. As you remove them from the foam, be careful not to bend the legs.



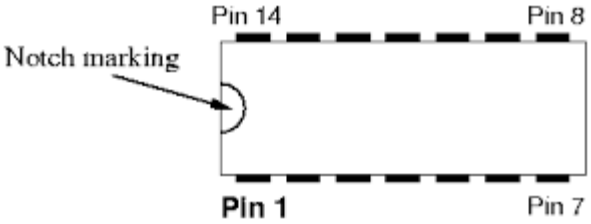
Pay attention to the labels on the chips. You should have *4 different types* of these chips:

- the 74LS04 (with six NOT gates inside),
- the 74LS08 (with four AND gates inside),
- the 74LS32 (with four OR gates inside),
- the 74LS86 (with four XOR gates inside).

Note: These chips are usually labeled with additional letters. For example, you may see it labeled as 74LS04 or SN74LS04. You may ignore the letters – the numbers are the important part of the label.

(If your kit has 8 chips instead of 4, it just means you have a backup of each.)

Important: The orientation matters. If you place a chip backwards, it can ruin the chip. Each chip has a semicircle "notch" on one side to orient the chip correctly. The notch should always be on the *left*.



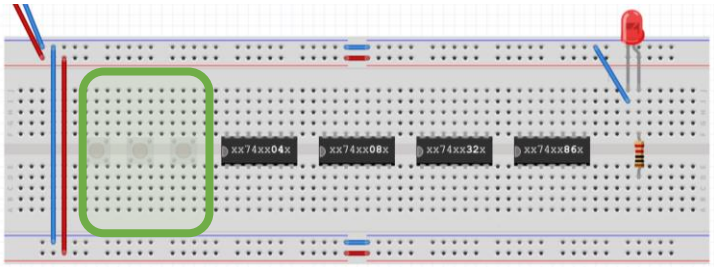
Please read through this whole page of instructions *before* attempting to insert any chips.

Note on removing chips: Once placed into the breadboard, the chips can be difficult to remove without bending or breaking the legs. Once inserted, it is recommended that you leave the chips in the board for the remainder of this lab *and the next lab*, so do your best to place them correctly the first time. If you need to remove a chip, the best way is to use an IC puller tool, which are available in the labs on campus. If you are not on campus, pry the chip out very slowly and evenly – avoid having the chip come out of the board at an angle.

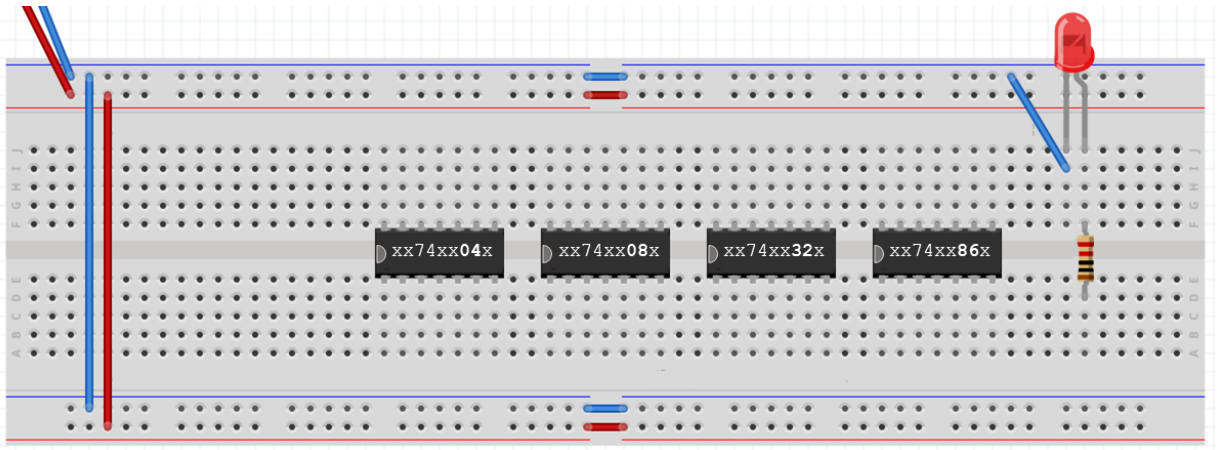
Note on replacing broken chips: If one of your chips breaks, check for a backup. If you don't have a backup, talk with your instructor. Your instructor may be able to provide you with a replacement or help you fix it. You may also purchase more chips: https://www.byuistore.com/Class-Kit-ECEN-101-106-Logic-Gate-Chips.

The chips are designed to straddle the middle gap of the breadboard with the 7 pins on top inserted into the row of holes just above the gap and the 7 pins on bottom inserted into the row of holes just below the gap. If you skip a row of holes, your circuit may not work correctly. The pins come spread out a little too far. To get the pins to align with the holes, you may have to first bend the pins slightly inward *carefully* until they are almost at right angles with the chip. To do this, place the chip sideways on a flat table. Slowly press down on the pins with your thumb (not too far!) until the pins are almost at right angles with the chip – not quite parallel with the table.

Leave room on your breadboard to the left of the chips to add buttons in the next lab. The area in green below shows you approximately how much space to leave to the left before you begin placing the chips.

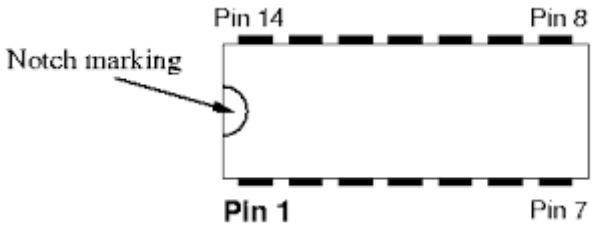


1. Find a 74LS04 chip, a 74LS08 chip, a 74LS32 chip, and a 74LS86 chip (one of each).
2. Place them in that order, left to right, into your breadboard, leaving 2 columns of space between chips. **The semicircle notch of each chip should be on the left.** Insert them now.

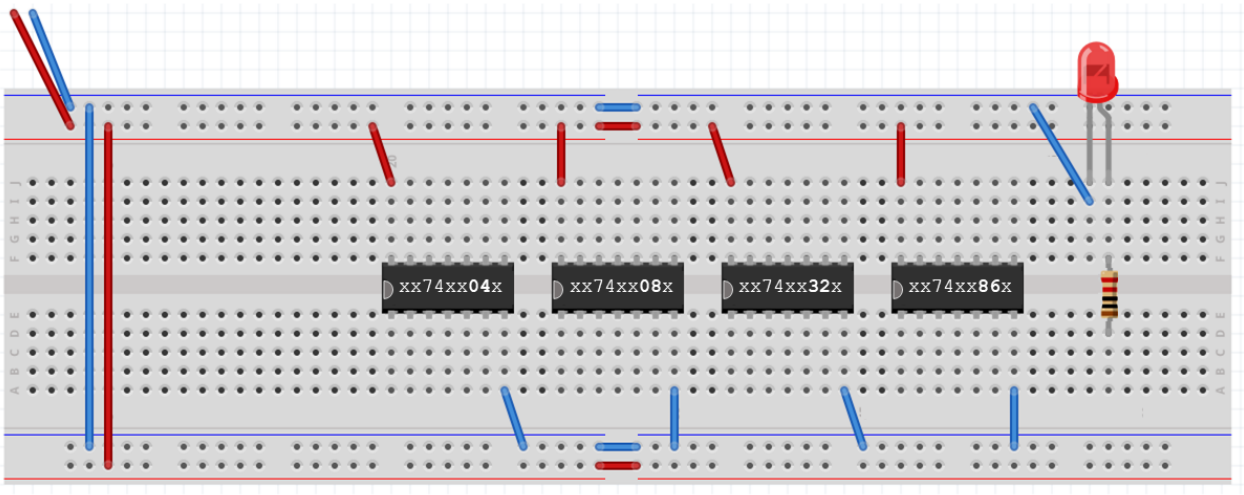


Provide a High Voltage and Low Voltage to Each Chip

Each chip requires a High voltage *and* a Low voltage. The pins are numbered 1 through 14 (see image). Pin 7 must be connected to Ground (Low), and Pin 14 must be connected to 5 Volts (High).



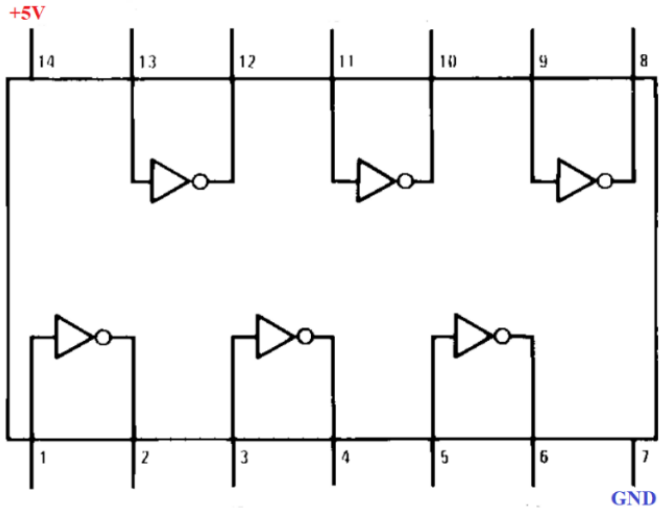
1. For each chip, add a wire from the same column as its Pin 14 to the High voltage bus (shown with the four new red wires below).
2. For each chip, add a wire from the same column as its Pin 7 to the Ground bus (shown with the four new blue wires below).



If your breadboard looks like the image above, you are now ready to proceed to Part 2 on the next page.

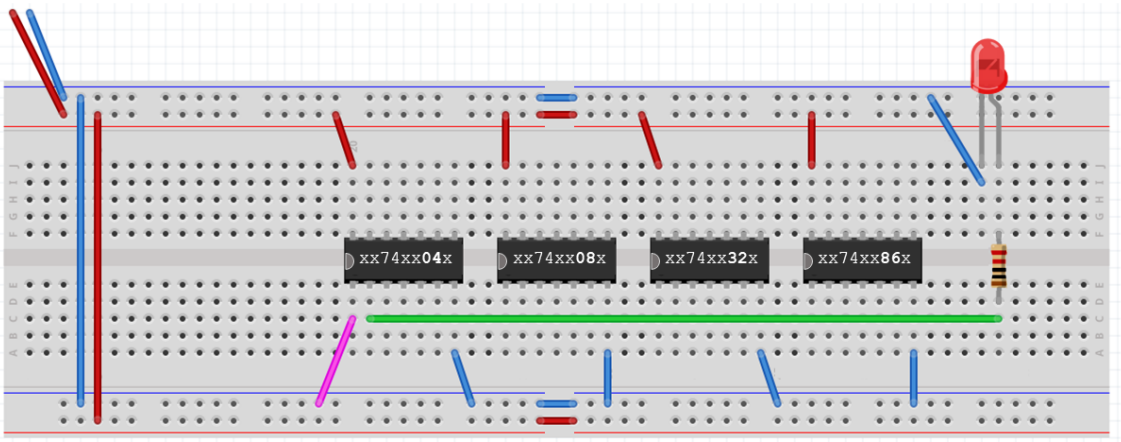
Part 2: 74LS04 NOT Gate

The chip on the left should be a 74LS04 chip, which includes six NOT gates inside. The diagram below shows which pins are connected as inputs or outputs to the different NOT gates.

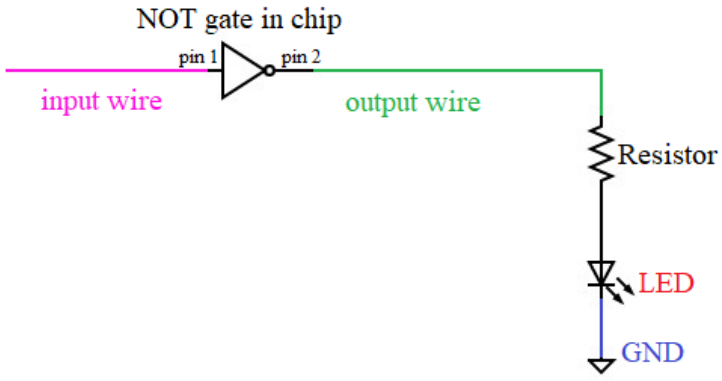


Use the NOT gate in the lower left corner, with Pin 1 as the input and Pin 2 as the output.

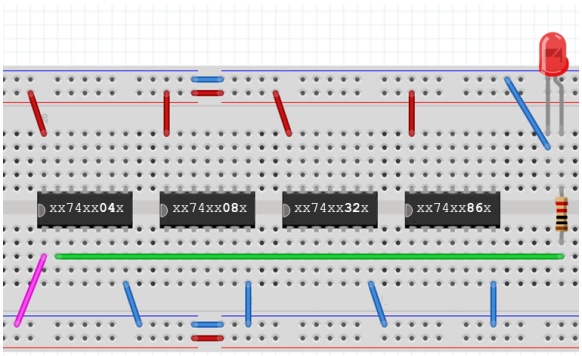
1. Connect the output of that NOT gate (**pin 2**) to the **resistor** of the LED (green wire below).
2. Connect the input of that NOT gate (**pin 1**) to **GND bus** (magenta wire below).



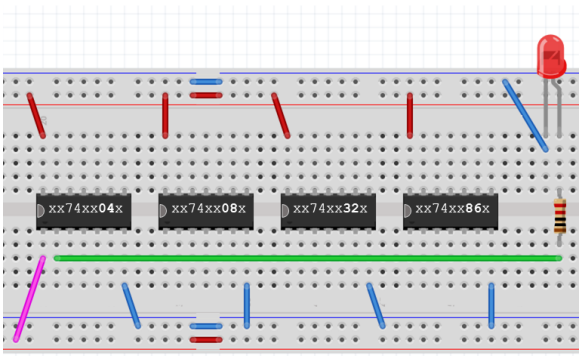
You have now connected a circuit that matches this schematic:



3. Power the breadboard by connecting the Arduino to your computer.
4. Move the input wire (magenta) back and forth between the Ground Bus (logic 0) and the High Voltage Bus (logic 1). Because this is a NOT gate, the LED should do the opposite of the input.



Input is LOW (logic 0).
Output should be HIGH (logic 1).

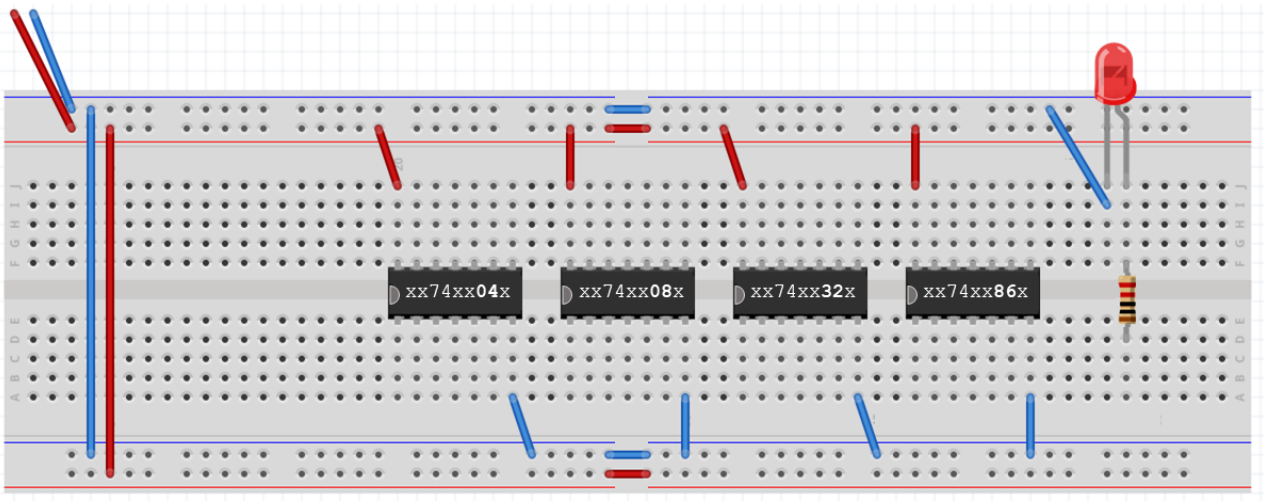


Input is High (logic 1).
Output should be LOW (logic 0).

If your circuit isn't correctly performing a **NOT** operation, double check the wiring and/or ask for help.

Record – Truth Table: In I-Learn fill out the truth table for Part 2, recording the values you observed.
Record – Take a Photo: Take a photo of your circuit in Part 2 to upload in I-Learn.

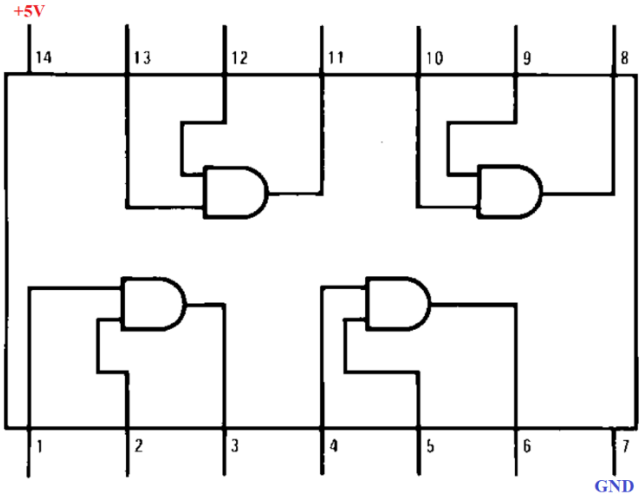
5. Disconnect power by unplugging your Arduino from your computer's USB port.
6. Remove the input wire and the output wire, returning your breadboard to the setup below.



You have now completed Part 2 of the lab. Continue onto the next page for Part 3.

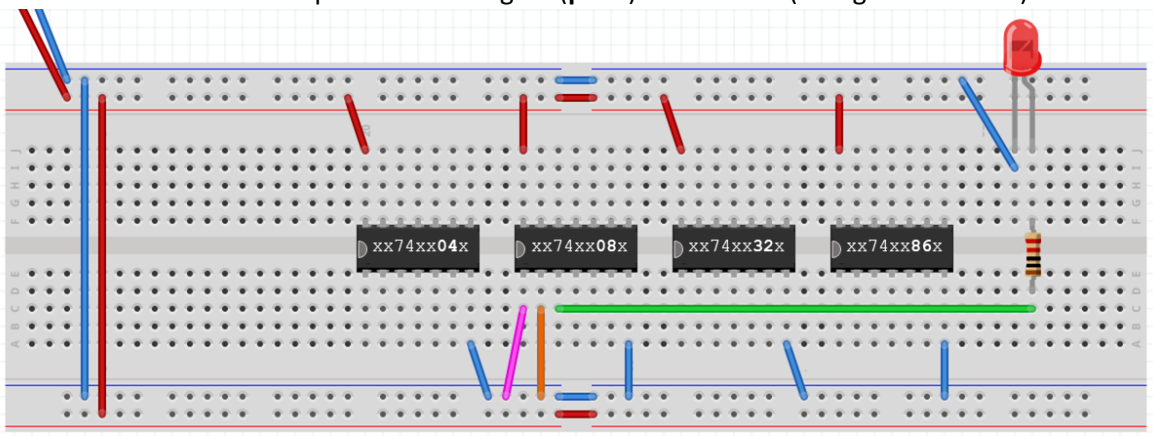
Part 3: 74LS08 AND Gate

The next chip to the right should be a 74LS08 chip, which includes four AND gates inside. The diagram below shows which pins are connected as inputs or outputs to the different AND gates.

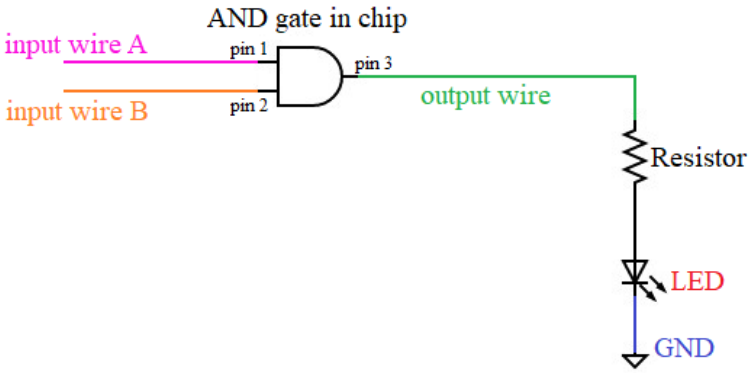


Use the AND gate in the lower left corner of the chip, with Pin 1 as the first input, Pin 2 as the second input, and Pin 3 as the output.

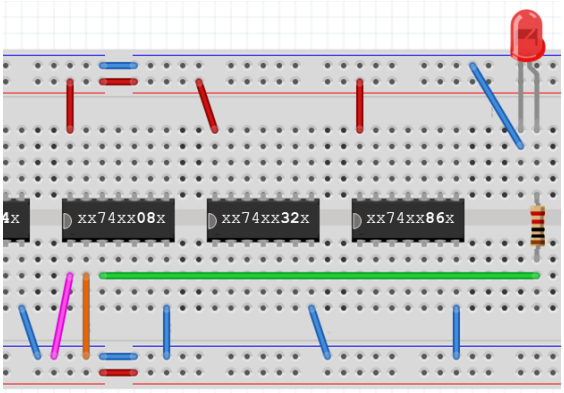
1. Connect the output of the AND gate (**pin 3**) to the **resistor** of the LED (green wire below).
2. Connect the first input of the AND gate (**pin 1**) to **GND bus** (magenta wire below).
3. Connect the second input of the AND gate (**pin 2**) to **GND bus** (orange wire below).



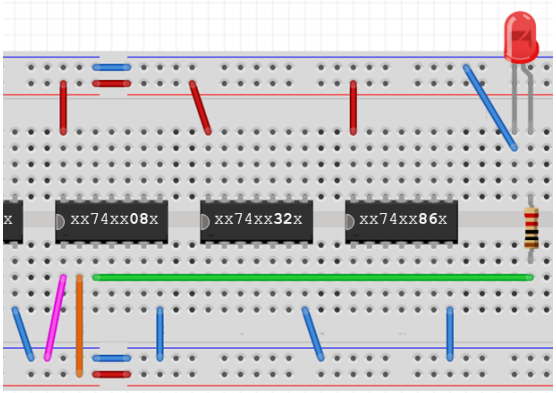
You have now connected a circuit that matches this schematic:



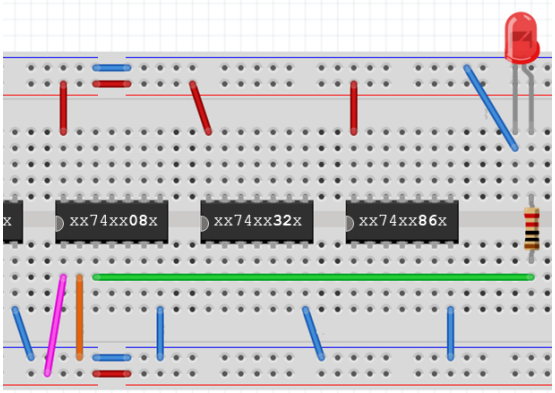
4. Power the breadboard by connecting the Arduino board to your computer.
5. Move the two input wires back and forth between the Ground Bus (logic 0) and the High Voltage Bus (logic 1) to cover all four possible combinations of inputs:



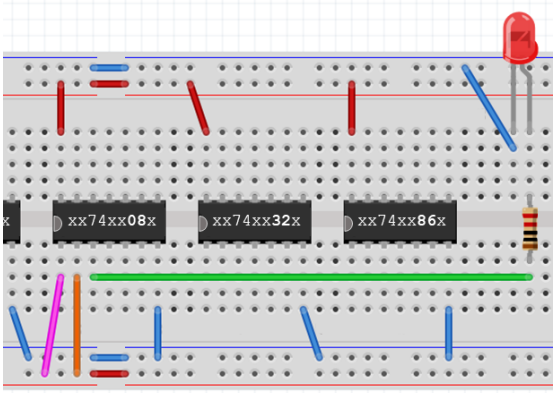
Input A is LOW. Input B is LOW.
Output should be LOW (LED off).



Input A is LOW. Input B is HIGH.
Output should be LOW (LED off).



Input A is HIGH. Input B is LOW.
Output should be LOW (LED off).

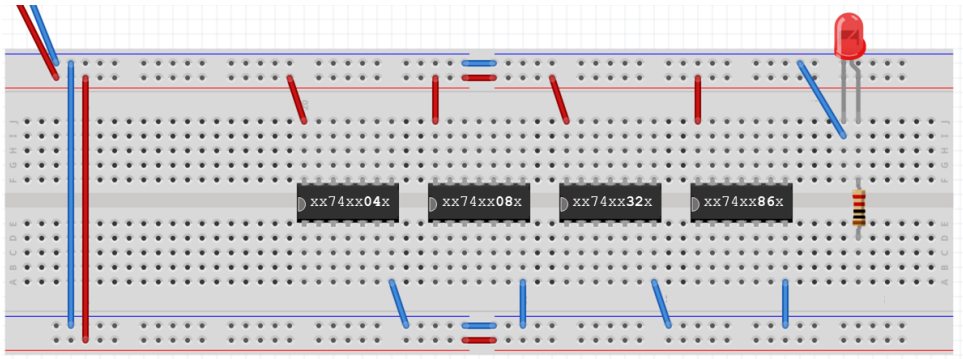


Input A is HIGH. Input B is HIGH.
Output should be HIGH (LED on).

If your circuit isn't correctly performing an **AND** operation, double check the wiring and/or ask for help.

Record – Truth Table: In I-Learn fill out the truth table for Part 3, recording the values you observed.
Record – Take a Photo: Take a photo of your circuit in Part 3 to upload in I-Learn.

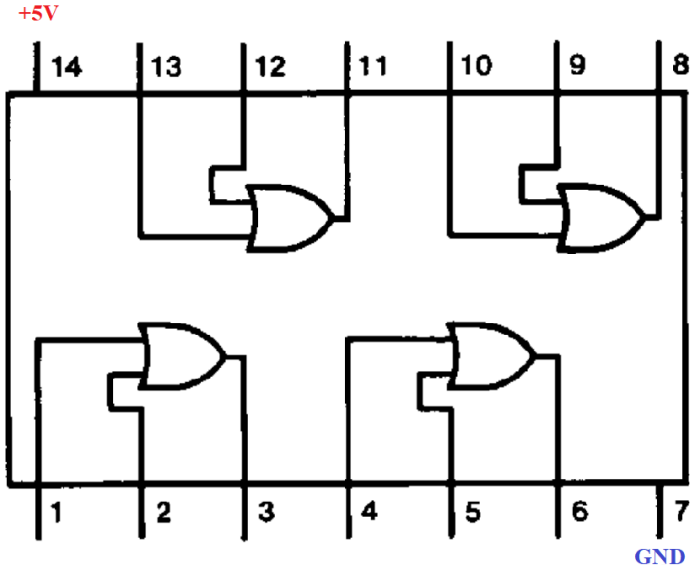
6. Disconnect power.
7. Remove the two input wires and the output wire, returning to the setup below.



You have now completed Part 3 of the lab. Continue onto the next page for Part 4.

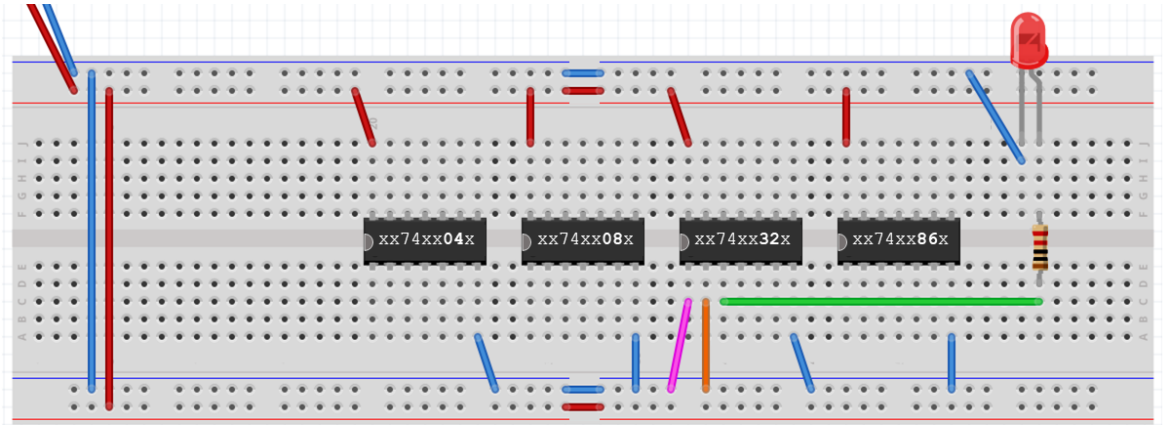
Part 4: 74LS32 OR Gate

The third chip should be a 74LS32 chip, which includes four OR gates inside. The diagram below shows which pins are connected as inputs or outputs to the different OR gates.

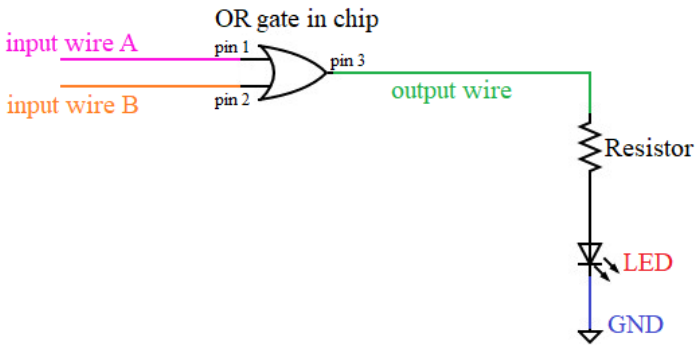


Use the OR gate in the lower left corner of the chip, with Pin 1 as the first input, Pin 2 as the second input, and Pin 3 as the output.

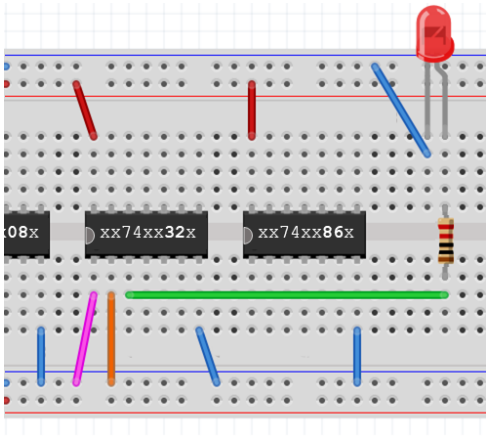
1. Connect the output of the OR gate (**pin 3**) to the **resistor** of the LED, (green wire below).
2. Connect the first input of the OR gate (**pin 1**) to **GND bus** (magenta wire below).
3. Connect the second input of the OR gate (**pin 2**) to **GND bus** (orange wire below).



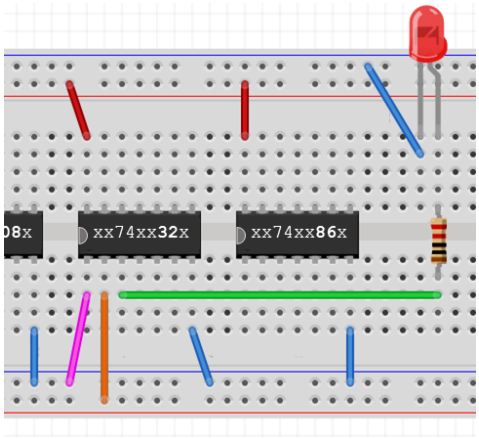
You have now connected a circuit that matches this schematic:



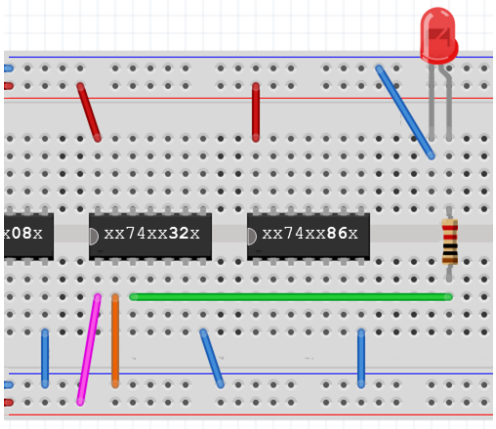
4. Connect power to the breadboard.
5. Move the two input wires back and forth between the Ground Bus (logic 0) and the High Voltage Bus (logic 1) to cover all four possible combinations of inputs:



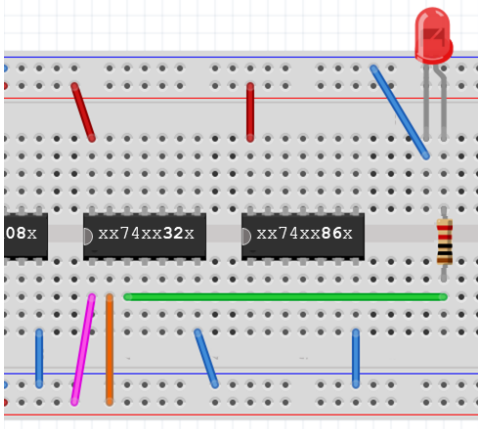
Input A is LOW. Input B is LOW.
Output should be LOW (LED off).



Input A is LOW. Input B is HIGH.
Output should be HIGH (LED on).



Input A is HIGH. Input B is LOW.
Output should be HIGH (LED on).

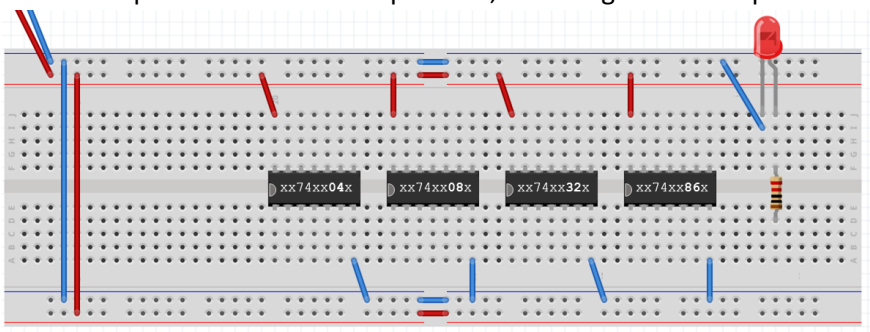


Input A is HIGH. Input B is HIGH.
Output should be HIGH (LED on).

If your circuit isn't correctly performing an **OR** operation, double check the wiring and/or ask for help.

Record – Truth Table: In I-Learn fill out the truth table for Part 4, recording the values you observed.
Record – Take a Photo: Take a photo of your circuit in Part 4 to upload in I-Learn.

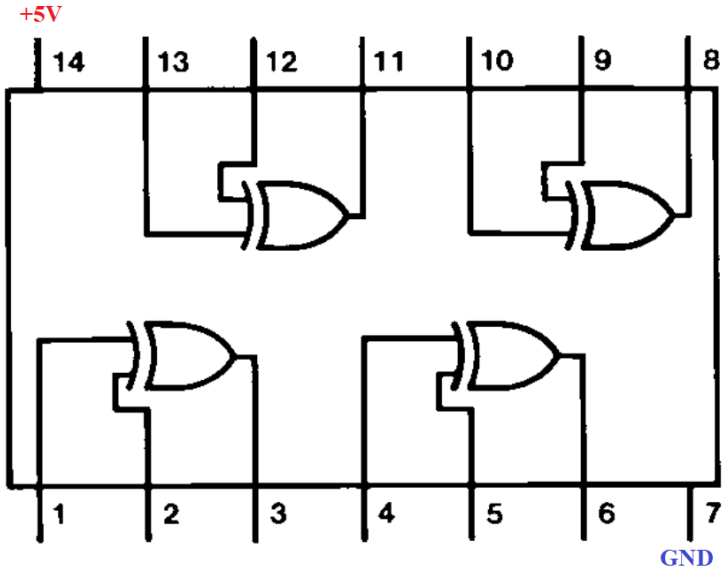
6. Disconnect power.
7. Remove the two input wires and the output wire, returning to the setup below.



You have now completed Part 4 of the lab. Continue onto the next page for Part 5.

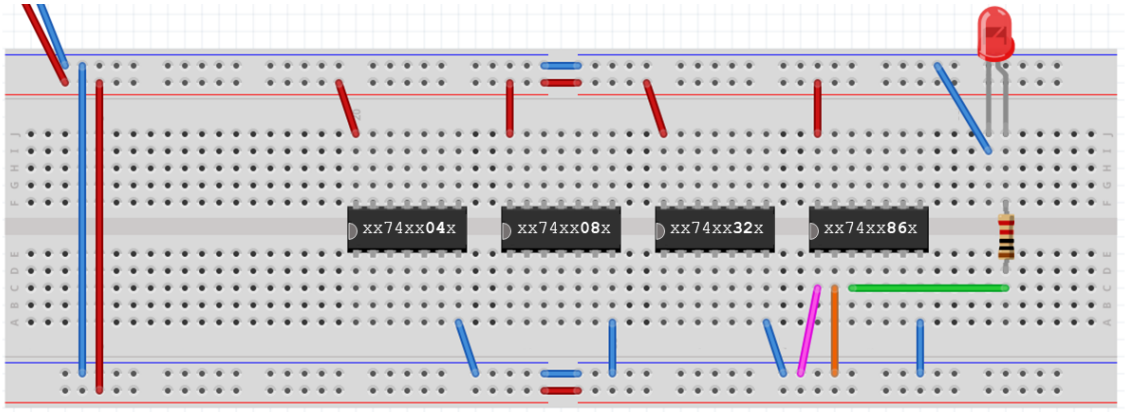
Part 5: 74LS86 XOR Gate

The last chip should be a 74LS86 chip, which includes four XOR gates inside. The diagram below shows which pins are connected as inputs or outputs to the different XOR gates.

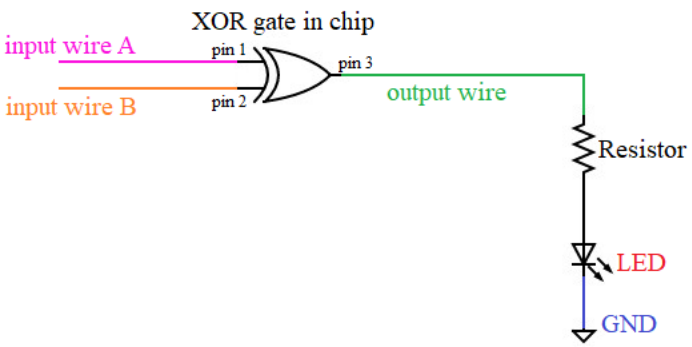


Use the XOR gate in the lower left corner of the chip, with Pin 1 as the first input, Pin 2 as the second input, and Pin 3 as the output.

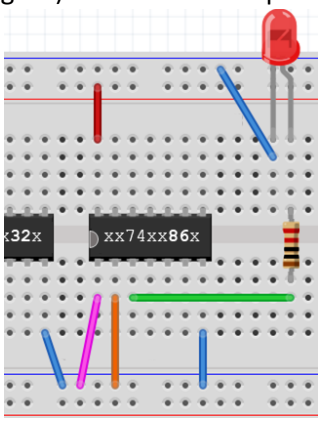
1. Connect the output of the XOR gate (**pin 3**) to the **resistor** of the LED, (green wire below).
2. Connect the first input of the XOR gate (**pin 1**) to **GND bus** (magenta wire below).
3. Connect the second input of the XOR gate (**pin 2**) to **GND bus** (orange wire below).



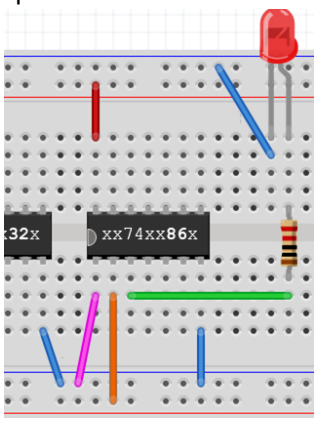
You have now connected a circuit that matches this schematic:



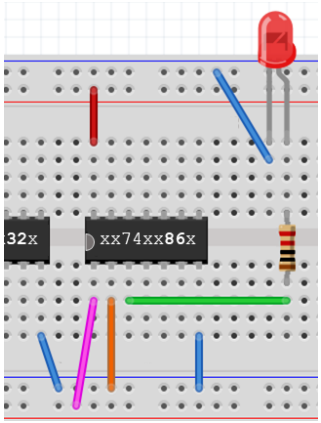
4. Connect power to the breadboard.
5. Move the two input wires back and forth between the Ground Bus (logic 0) and the High Voltage Bus (logic 1) to cover all four possible combinations of inputs:



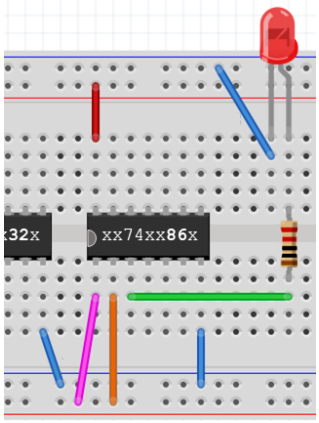
Input A is LOW. Input B is LOW.
Output should be LOW (LED off).



Input A is LOW. Input B is HIGH.
Output should be HIGH (LED on).



Input A is HIGH. Input B is LOW.
Output should be HIGH (LED on).



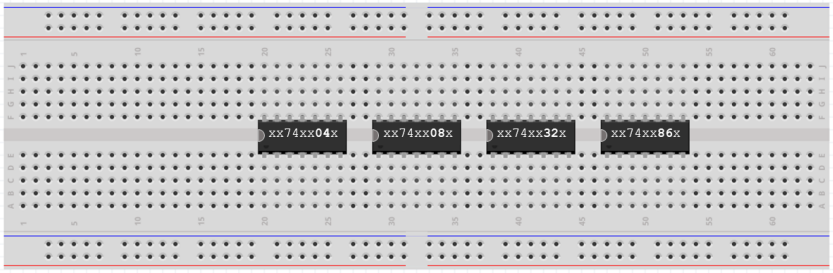
Input A is HIGH. Input B is HIGH.
Output should be LOW (LED off).

If your circuit isn't correctly performing an X**OR** operation, double check the wiring and/or ask for help.

Record – Truth Table: In I-Learn fill out the truth table for Part 5, recording the values you observed.
Record – Take a Photo: Take a photo of your circuit in Part 5 to upload in I-Learn.

6. Disconnect power.

Remove all wires, the resistor, and the LED. **Leave the four logic chips in the board for the next lab.**



Submit Your Laboratory Report

After completing this laboratory go to I-Learn, finish the Lab Report assignment for this lab.