EK307 Lab 0. Getting started with the EK307 lab kit

Your name______________________________ Your collaborators name_________________________

Lab objectives:

- Experimenting safely
- Get to know what is in your kit
- Using an Arduino as a power supply
- Make a simple Volt and Ohm meter
- Using the multimeter to measure voltage and resistance
- introduction to the solderless breadboard
- Resistors
- Troubleshooting

Deliverables:

- Demonstrate your LED circuit (figure 6) to a TA to get credit for building
- Turn in this completed worksheet to Blackboard or Gradescope (we will specify which to use)

Safety first:

Experimenting with electronics can expose you to live circuitry, sharp objects, bright lights (visible and invisible), chemicals, and high temperatures. There are small parts in the kit that can be accidently ingested.

We designed the BU EK307 lab kit with safety in mind. The voltages are low. The currents are low. Meaning their product, the power, is also low. The components are ROHS compliant.

General rules for electronics experimenting:

- Do not use it or leave your electronics kit unattended if there are people who cannot understand and abide by these rules in your work area. Especially young children.
- Do not set up on a surface you eat on. If you must, put a protective material like a tablecloth or cardboard on the surface.
- Do not eat and work at the same time.
- Always wash your hands when you leave the work area.
- Do not stare into powered LEDs for long periods of time.
- Do not plug your kit into the AC mains voltage (the wall outlets or any other high voltage source) except through a known good, reputable brand, unaltered, and safe USB charger.
- Be aware that overheating electronic components can get very hot.
- Exposed pins, especially the ones on the Arduino and OP amps are sharp. Be careful when pressing on them. Make sure they don't fall on the floor for someone to accidently step on with bare feet.

- Always turn off, unplug, and de energize any experimental electronics when you leave the work area.
- When practical, turn off, unplug, and de energize any experimental electronics when you are making changes to the circuits and making measurements.
- Do not work with this kit or electricity if your body may be well electrically grounded. Instances may be but are not limited to; if you are soaking wet, excessively sweaty, a large portion of your skin is touching bare earth or grounded metal, touching metal plumbing fixtures, in the rain, etc.
- Never work on high voltage circuits alone. Your kit is low voltage, but we want to stress this point.
- If you have any medical condition or medical device that may put you at risk while using this kit, contact us to seek an alternative safer way to work and learn.

Unboxing your kit:

Note that your kit is packed in plastic bags. Some have metal deposited on the inside. They are called anti ESD bags. ESD stands for electrostatic discharge. The bags are slightly conductive and this prevents them from developing static charge on the inside of the bag.

Electrostatic charges build up in many places. When you walk across a carpet in a low humidity climate you can generate a static charge that can be discharged to conductive items. This is a common occurrence in Boston winters. Put on your jacket over a wool sweater and you may end up with 50,000 volts of potential at your fingertips. The discharge is enough to give you a jolt! Field effect transistors (FETs) are particularly susceptible to being damaged by ESD. In your kit the Arduino and the Op Amps contain FETs. If you discharge a charged finger on the pins of these devices, they may be damaged.

The OP Amps only cost 0.25 USD but they are very valuable if you burnt out your only one! The other hidden cost of ESD damage is sometimes it *partially* damages the component. Then its behavior is unpredictable. This situation is a troubleshooting nightmare. It may cost you and your engineering team countless hours trying to debug a design only to find out one logic gate out of the few million in a common microcontroller was damaged. Moral of the story try to minimize ESD on your bench. Luckily the Arduino is fairly tolerant of ESD thanks to the design so we should not get into too much trouble. If you were sending your circuit into space or using it in a medical device, I would recommend storing it in an ESD bag when you are not using it.

Arduino as a power supply:

In this lab we will use the Arduino as a power supply. **For the labs where the Arduino is only functioning as a power supply it is recommended to power the Arduino from a phone charger, not your laptop or some other expensive peripheral.** The reason is because if something gets short circuited on the Arduino it may stress the USB on the expensive thing. There is a polyfuse (a resettable fuse that will open circuit in the event of an overcurrent) in the Arduino that will protect the expensive thing but why push it.

1. Unbox the Arduino and place it in the breadboard as shown in figure 1. Put it on one side of the breadboard so that you have space to build circuits on the other side. We often refer to this space colloquially as ‘real estate’. Think of the Arduino as the house and the breadboard as the

land. You want to be able to get a swimming pool and tennis court in the backyard…. Plus you don't want to block the USB port, think of the USB as the front door.

2. The pins on the Arduino, if they are not bent, will perfectly align with the holes in the breadboard. Once you have them aligned you will have to push down with a decent amount of force to seat the Arduino on the breadboard. Try to avoid pushing on the USB connector. They tend to break off the circuit board easily. The Arduino is kind of pointy and sharp, you can use a few layers of cardboard to protect your fingers. When the black plastic pin frame on the bottom of the Arduino is touching the white breadboard you are good. Watch this video to learn how

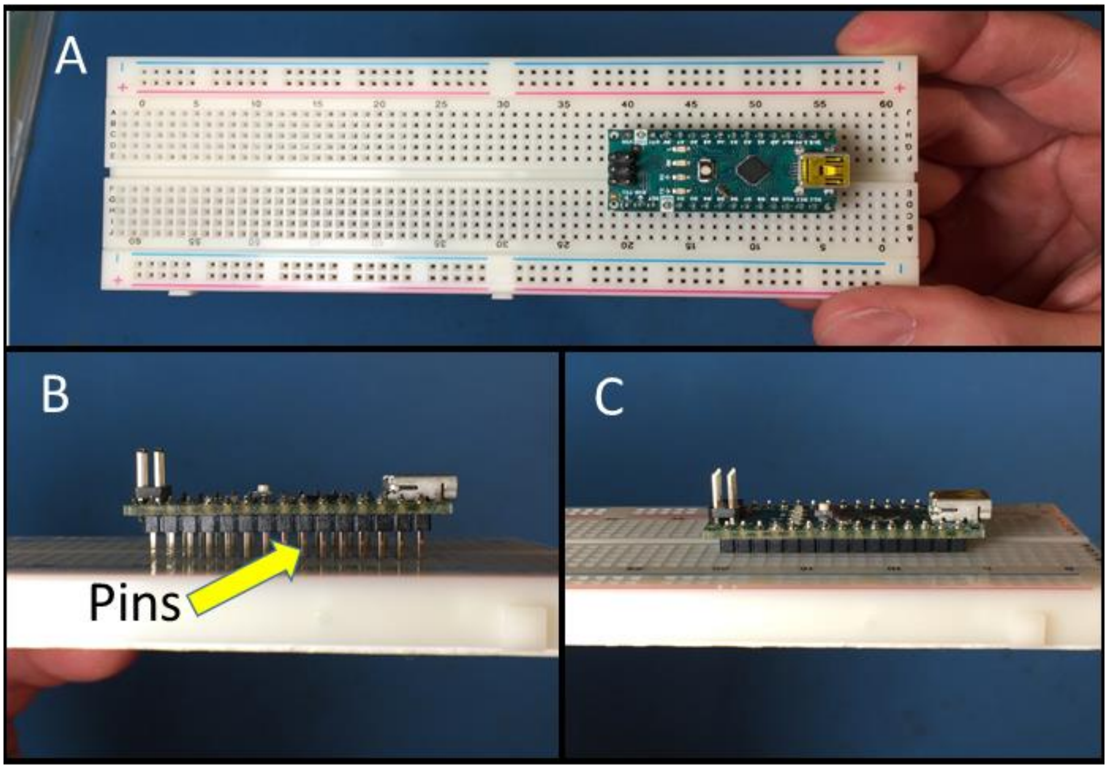


Figure 1: Panel A showing Arduino position on breadboard to give us good use of the real estate on the board. Panel B: Arduino is not seated properly on the SBB. Panel C: Arduino is seated properly on the SBB. The pins are fully engaged in the bread board terminals.

the breadboard makes electrical connections: https://youtu.be/_ukXSwEe1h0

When you are testing out something for the first time it is a good practice to test it in as small increments as practical. So, now is a good time to see if the Arduino powers up. If it does an LED or two on its PCB will glow.

3. Plug the USB mini cable into the Arduino and then plug the other end into a phone charger. The green LED labelled 'ON' should light up. If it is lit that means…. the LED works and is getting sufficient power from the USB. We *assume* the actual microcontroller works but you can't be 100% sure! The ON LED does not rely on the microcontroller to function. Research what your tests and indicators actually do. This is a good thought process for an engineer to consider when building and or testing anything.

Measuring Voltage:

If you are not familiar with handheld digital multimeters, please watch this video introduction: https://youtu.be/rZhUaPLmB6I It demonstrates how to measure voltage and resistance with a meter like the one available to you in the lab kits. If you have a different multimeter, that is ok. Most multimeters follow a similar principle of operation. We will use the multimeter often in the labs. Engineers are resourceful and can make things from parts and that is exactly what we will do in this lab: Make a multimeter from an Arduino.

Make the Arduino Multimeter:

Your Arduino has a built-in analog to digital converter (ADC). This is a peripheral circuit inside the microcontroller chip itself that can measure voltage on an input pin and convert it to a number in software. The ADC is the main component in a digital voltmeter. Our Arduino based meter lacks the finish and many features that are included in a handheld voltmeter instrument. The computer connected to the Arduino will serve as the user interface and display.

There are a few limitations that we can discuss as to why you would not want to rely on your Arduino for routine voltage measurements.

- Safety: The circuitry is exposed and can be touched.
- Ground referenced: The Ground terminal of your Arduino meter is connected to circuit ground through your computer. You cannot measure voltage unless the potential on each test lead is within 0 to 5Volts of this ground.
- Precision: The ADC in the Arduino is much less precise than a multimeter ADC.
- Range: The range of voltages we can measure is limited to -5 to 5 Volts
- Cost: You can purchase a decent multimeter for $20. Would you want to use a $20 Arduino and an expensive computer to do the same thing? (There may be a good reason, we will find out in later labs.)

4. Download and install the Arduino IDE on your computer. It can be found here: https://www.arduino.cc/en/Main/Software
5. Download the Zmeter software from Github. Github is a cloud GIT repository for source code. I find it very useful to organize projects that are shared. https://github.com/EkZosuls/Zmeter
6. Connect your Arduino to your computer with the USB cable
7. In the IDE Tools menu configure the board to be Arduino Nano.
8. Select the proper Port to connect to the Arduino
9. Compile and burn the Zmeter software to the Arduino.
   Copy, clone. Or fork the library or simply copy and paste the code from the `zmeter.ino` file into a blank Arduino project. Once you have the source code, compile and upload it to your Arduino.
10. Plug a red jumper into the column on the breadboard that is connected to the Arduino A0 pin. Plug a black jumper into the column on the breadboard that is connected to the Arduino A1 pin.

It will look like figure 2. The black wire is the negative terminal, the red wire is the positive terminal. (It is ok to use different color wires, just keep track of the polarity)

11. Open the serial monitor in the IDE.
12. Use the loose ends of the jumpers as probes. Measure the voltage between the Arduino 3.3V pin and GND. Write your result here:___3.3433 V___
13. Measure the voltage between the Arduino 5V pin and GND. What do you get?___4.9922 V___
14. Measure the 3.3V terminal again. This time reverse the red and black wire connections. What value did you get?___-3.418 V___ How is it different than the first measurement of the 3.3V pin?_Its approximately the same but negative, since we switched the terminals

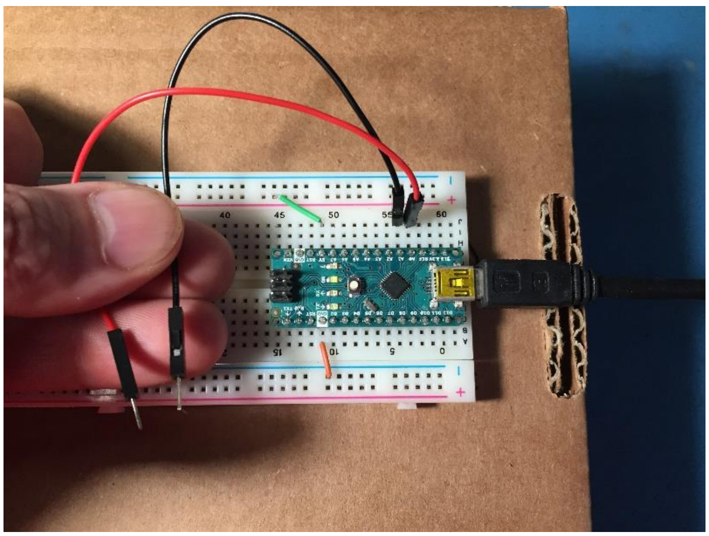


Figure 2: The Arduino voltmeter setup. The Red wire plugs into the A0 pin row. The black wire plugs into the A1 pin row. The red lead is the positive, black the negative or reference.

15. Now move the red or V test lead to the Arduino GND pin and probe the 5V with the black (reference or COM) lead. What happens?
_we get a value of -4.9922, exactly the negative value of the previous 5V measurement.
16. Obtain two AA batteries from the lab component bench (near the windows to the hallway).
17. Measure the voltage across the two batteries in each of the four arrangements shown in figure 3 and write them in the table below.

Battery arrangement	Measured voltage
A	2.116 V
B	-0.098 V
C	-2.1054 V
D	-0.0239 V

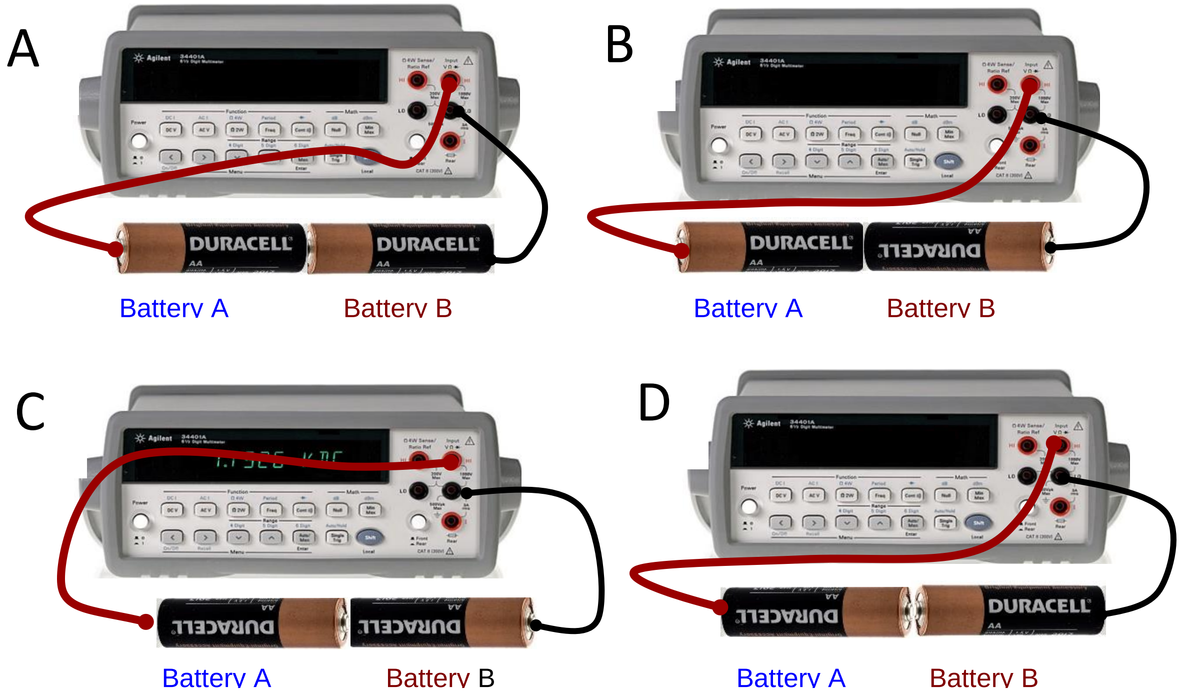


Figure 3: Measuring series connected battery voltage. The four arrangements to measure are labelled A-D. Note your measured values in the table. You can use the Arduino Z meter you made, the bench meter in the pictures, or your handheld multimeter to make the meaurements.

Explore the solderless breadboard (SBB):

1. If you didn't already, watch our video introduction to the solderless breadboard https://youtu.be/_ukXSwEe1h0 .
2. The SBB forms a mechanical substrate for your circuit. Yes, there is quite a bit of mechanical engineering in building circuits. If the circuit isn't stable it will quickly deteriorate. It also allows us to electrically connect components together. We will 'bring out' the power from the Arduino to the SBB. We will do this by using the solid jumpers to connect the power pins of the Arduino to the SBB rails.
3. Use two solid preformed jumper wires to bring the power to the rails. What we are doing is connecting the GND to one rail and the 5V to another. The color of the wire can be of your

choice. Note the wires are color coded according to length. Figure 4 shows the connections I made. There is some art and individual preference to how to choose the wires and where to plug things. If you are having trouble getting the wires to plug into the breadboard get a pair of small pliers. The other option is to use the alligator test leads from the kit as pliers.

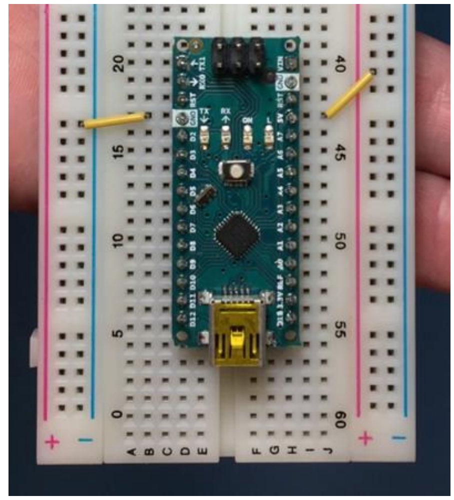


Figure 4: How my breadboard looks when I connect the Nano 5V and GND to the rails. Yours can look different as long as it brings power to the rails.

Let’s make a boring circuit. We will connect a resistor across the power rails on the breadboard. Figure 5 is a schematic and layout of our creation. Note, the circuit is not boring if you have infrared vision!

4. Pull the USB cable out of the Arduino to turn it off.
5. Grab a resistor from your bag of parts that has a value greater than 200 Ohms and less than 2.2 kilo Ohms (2.2k). You can find the resistor values by looking at the packing list and cross referencing to the part numbers on the bags. The description column indicates the resistance. For example, a 220R is 220 Ohms. A 1k is 1000 Ohms or 1 Kilo Ohm. You can verify its resistance with the multimeter if you have one. You can also decode the resistor values by observing the color codes on the resistor. Here is a video: https://www.youtube.com/watch?v=UINnKXSdIJw&feature=share Avoid resistors that are less than 100 Ohms. They will put a large draw on the Arduino power supply and may cause it to behave erratically.

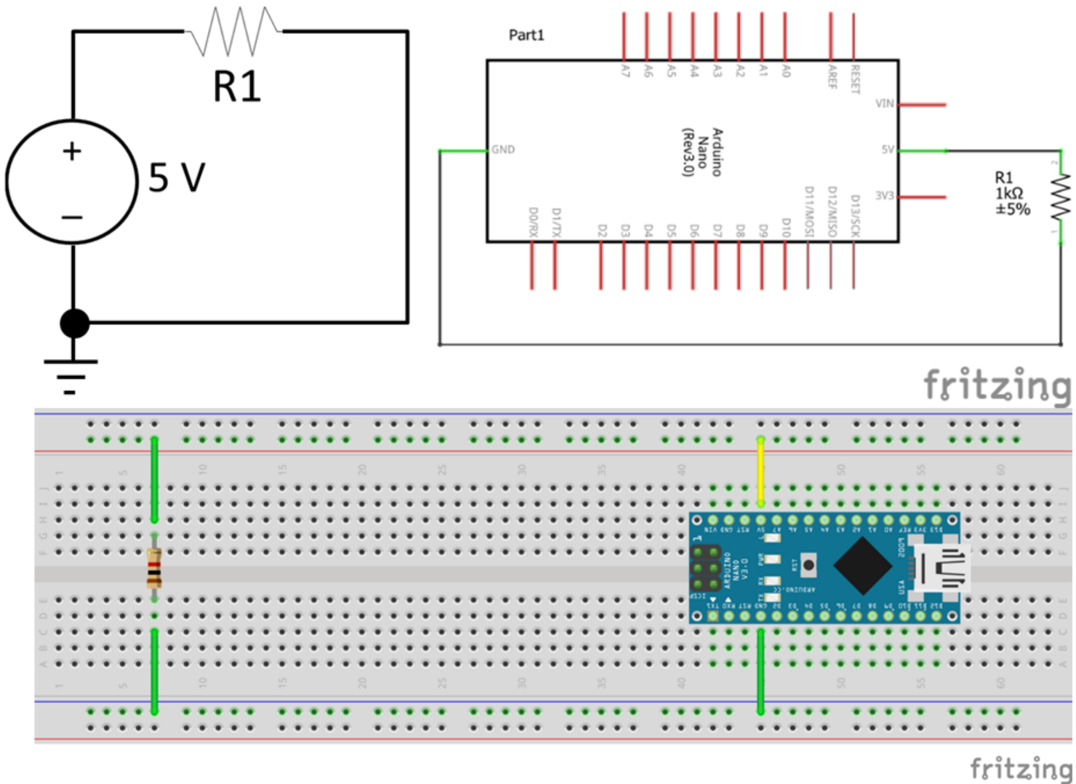


Figure 5: The boring circuit. The upper pane is the schematic of what we want to make, a 5V DC source connected to a resistor. The upper right pane is the schematic implementation with the Nano as the power supply. The bottom is the breadboard layout.

6. Insert the resistor into the breadboard. Note its value here: 1000 OMHS
   Make sure the resistor leads are in non-connected holes. It can be placed closer to the Arduino especially if you are using the Zmeter with its short leads.
7. Use the solid jumper wires to connect one lead of the resistor to the GND rail and the other lead to the 5V rail. Figure 5 has a breadboard example. Although, your layout can be different.
8. Plug the Arduino into a USB power supply. Immediately observe if the green ON LED is lit. If not unplug the USB and go to step 16 to troubleshoot.
9. Now measure and note the voltage across the resistor. You can do this with the Zmeter or a multimeter. ____4.9922 V____. Is it what you expect? ____Sort of____
10. If the resistor voltage is not what you expect, or the Arduino ON LED does not turn on, it's time to be a detective.
    a. If the LED lights but the resistor voltage is zero; use the multimeter to measure the voltage at the Arduino pins, at the jumpers going to the rails, at the jumpers going to the resistors. When you stop seeing 5V on your meter you found the problem. It is an open circuit (See examples in the appendix). Meaning somewhere the electrical conductor that is the wire is not connected.
    b. If the LED is not lit it is likely that you have a short circuit in the wiring. This means that there is a wire connecting 5V to GND. This causes an excessive amount of current to

flow and may open circuit (See examples in the appendix). the Arduino fuse or damage it.

c. The lab staff can help you with wiring issues during the lab sessions.

11. Once you measure the value of voltage you expect then calculate and write down the current through the resistor. You know the equation I = V/R = 4.9922/1000 = 0,0049922 A = 4.9922 mA

Now we will add an LED (light emitting diode) to the circuit. We will put it in series with the resistor. See figure 6. The resistor acts as a current limiter to protect the diode and Arduino from excessive current. We will learn more about this in the next lab.

When you put an element in series with an existing circuit, such as the boring circuit, you need to break into the loop of the boring circuit. Imagine the boring circuit is a loop of rope or string. If you want to make the loop larger in diameter you have to cut it or untie it and add in more material and join it at the loose ends to make a larger loop. That is exactly what we will do here. We will break the loop at one of the boring resistor leads and attach an LED with its two terminals to the loose ends. The caveat is the LED needs to be put in the correct direction for current to flow. It is polarity sensitive. In fact, a diode is like a one-way valve, current can only flow in one direction. It flows from the anode to the cathode lead.

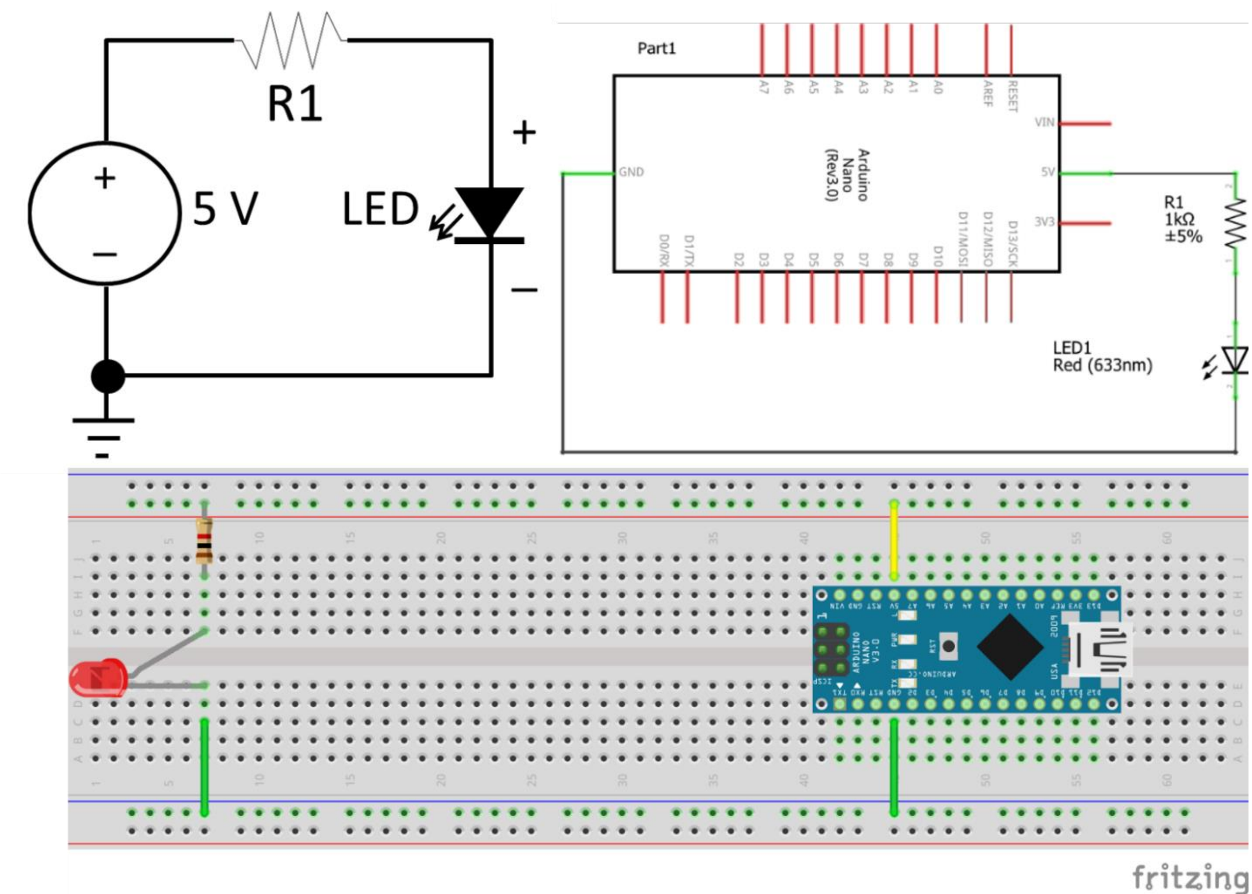


Figure 6: The LED circuit.

12. Find the red LEDs in your kit (figure 7). Note the LEDs in the BU kit have narrow beams and are very bright when they are running at full power. **Don’t stare into the beam for long periods of time.** The

kit red LEDs are strung together on a piece of cardboard tape. The tape is part of a neat technology: On a robotic assembly line, thousands of LEDs are on a reel of this tape. The robot uses wheels and motors to advance the tape, then a cutter cuts the LED off the tape and a robotic hand places it in a circuit board. It's called a pick and place machine. Today you are the pick and place machine. Here is a pro tip: Ideally you will cut the LED free of the tape. You can unpeel it, but the tape residue will gum up your breadboard and make its electrical connections unreliable. The best tool is a pair of sharp wire cutters, if you don't have wire cutters, a pair of scissors or nail clippers that you don't care too much about will work. Note that scissors or nail clippers will dull quickly if subject to this so don't expect to cut hair with them again!

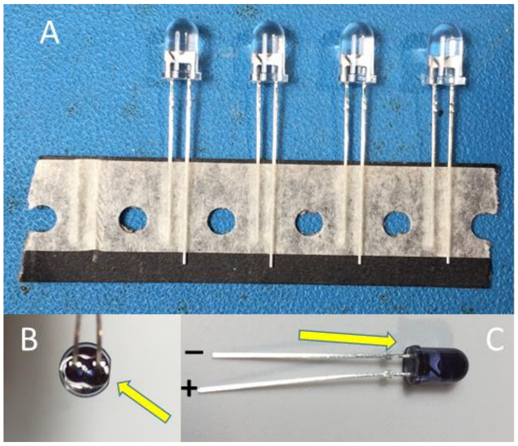


Figure 7: LEDs. Panel A shows the RED LEDs connected to their tape. Panel B shows the notch (arrow) in the annulus. The pin closest to the notch is the negative terminal.

13. Disconnect your Arduino USB cable.
14. Open the loop on your boring circuit and insert the LED in series with the circuit to close the loop (see figure 6).
15. Plug in the USB cable, did the RED LED light up? Is the Arduino ON LED still lit? If the Arduino ON LED is not lit unplug the USB cable immediately and check to see if you have a short circuit.
16. If the RED LED lit up, you are lucky or know what you are doing. If not, there are a few reasons it will not light up.
    a. The LED is in backwards.
    b. The resistor is so large that little current is flowing and there is little if any visible photon generation.
    c. Something short circuited.
    d. The loop is not complete.
    e. Something is burnt out
    f. It might be lit but it is too bright in your workspace to see it.

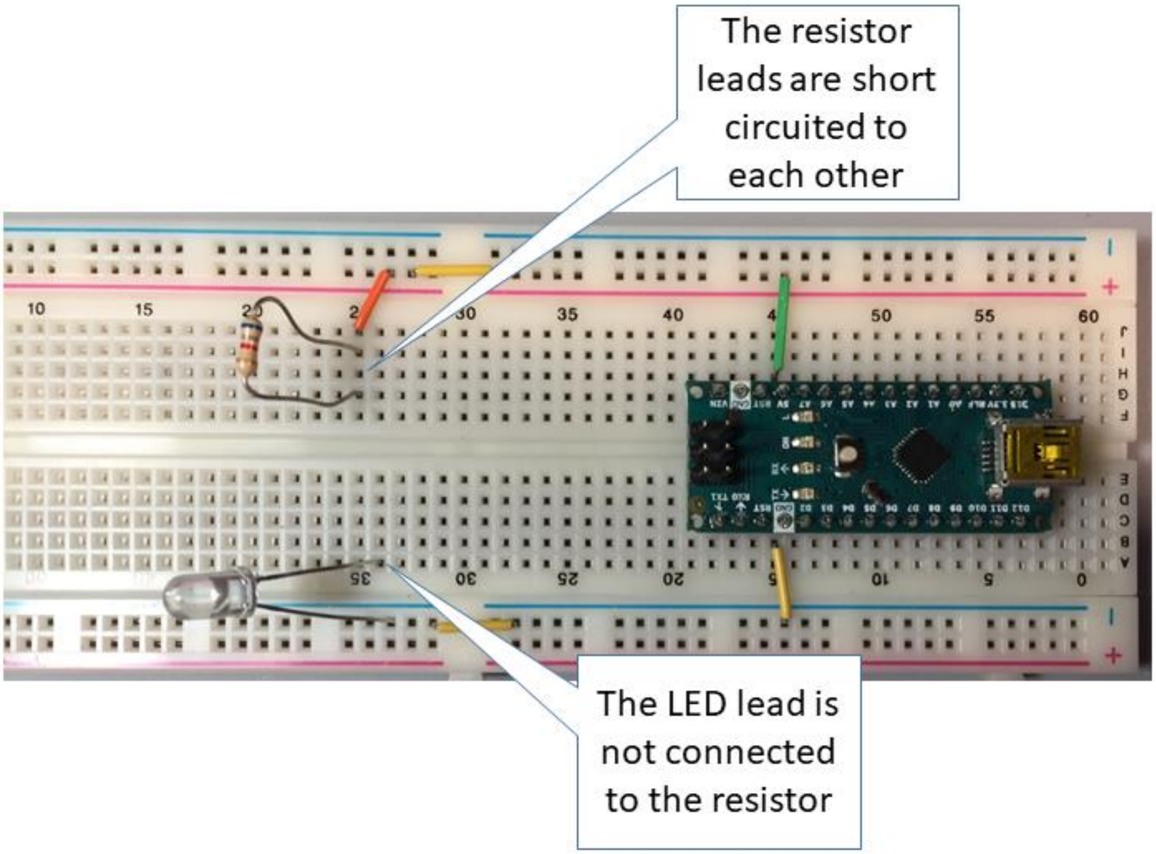
You could try to fix the circuit by trying to randomly change the arrangement or resistor or maybe the LED. Eventually it will work. A better way is to measure voltages and make corrections based on circuit theory principles.

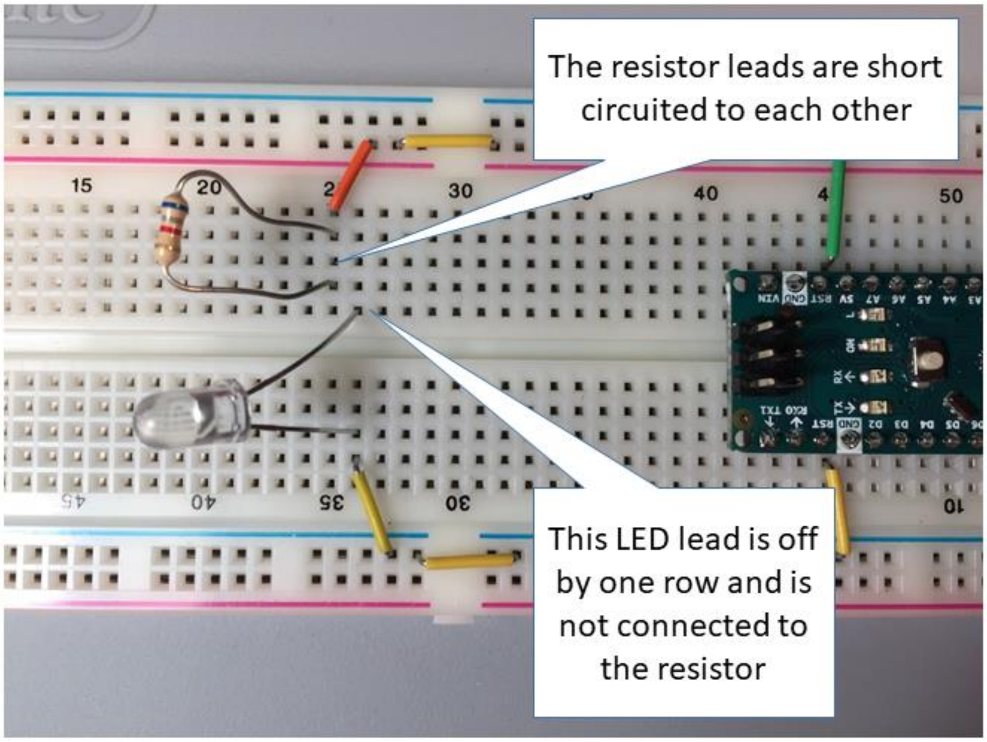
17. Measure the voltage across the LED and write it here ____________. It should be approximately 2 volts plus or minus a few tenths of a volt. Note this measurement is polarity sensitive. The black or COM wire from the meter should be touching the lead that is proximal to the notch in the annulus of the LED package (See figure 7). The red or V wire should be on the other LED lead.
    a. If its voltage is in range yet very dim or not lit, switch your resistor for one in the 220Ω to 1kΩ range.
    b. If the voltage is approximately 5 volts the LED might be in backwards, if you properly measured the voltage across the LED you would see a negative sign on the meter. Try to reverse the LED leads and measure it again. Does it light now? Did the voltage across it change? If it dropped but is not near 2V try to lower the resistance.
    c. If the voltage is zero you either have a short circuit or an open circuit. A common cause of a short is if both + and GND are somehow plugged into the same row on the SBB. A common cause of an open is if elements or wires that should be in series are not properly connected from the rails, to the rows, and to the components. Once again, your voltmeter can be used to trace the circuit and find where the voltage vanishes.

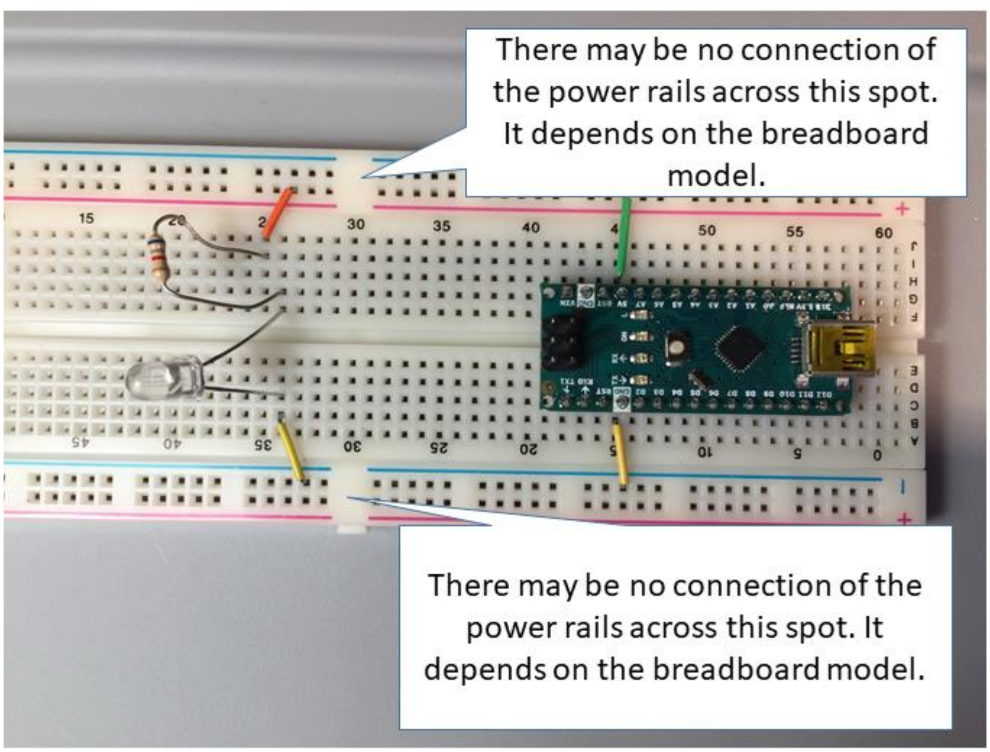
If you are still having trouble, we can help you during the lab sessions. There is quite a bit of new material here, we don't expect everyone to finish without help.

18. Measure the voltage across the resistor and calculate the current. Write it down here 3.0212V -> I = V/R = 3.0212/1000 = 3.0212 mA
19. Measure and note the voltage across the LED 1.9682 V.
20. Measure and note the voltage across the Arduino 5V supply 4.9922 V.
21. If everything is connected properly the LED and resistor voltages should sum (within a few millivolts) to the Arduino 5V supply voltage. Do they? 3.0212+1.9682 = 4.9894 V -> YES
22. If they do sum then you probably connected everything correctly and the circuit consists of a 5V source (from the Arduino), a resistor, and an LED. What is the current through each element?
    Since this is all in parallel, the current in the whole circuit will be the same as the one of anyindividual element such as the resistor, or 3.0212 mA
23. Take a photograph of your working LED circuit with your BU ID next to it.
24. Raise your hand and a TA will view a demonstration of your working circuit. This demonstration is a large component of your grade for this lab exercise. If they are satisfied with your circuit they will give you credit for this part of the lab.
25. Submit this completed lab workbook to Blackboard or Gradescope before the lab due date.
26. You are now done with this lab! Leave your circuit together we will use it in the next lab.
27. Always wash your hands after playing with electronics or doing any lab-based work.

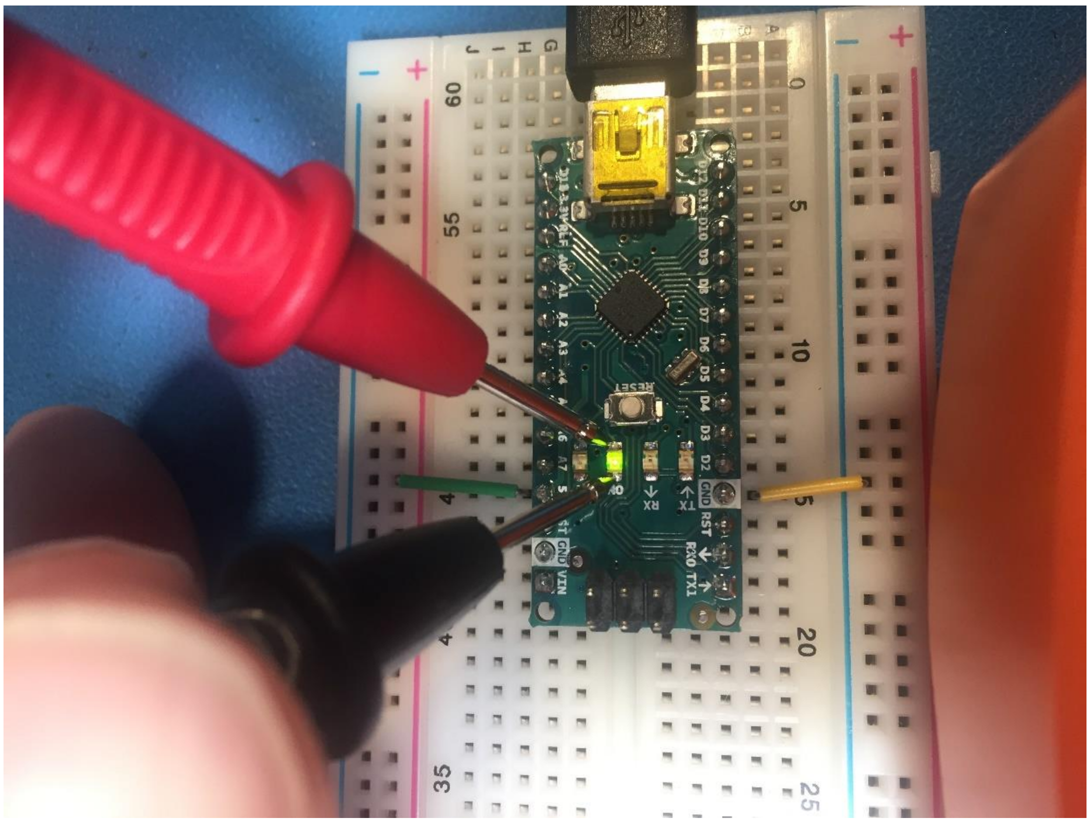
Appendix A: A few common breadboard wiring mistakes




The resistor leads are short circuited to each other
This LED lead is off by one row and is not connected to the resistor


There may be no connection of the power rails across this spot. It depends on the breadboard model.
There may be no connection of the power rails across this spot. It depends on the breadboard model.

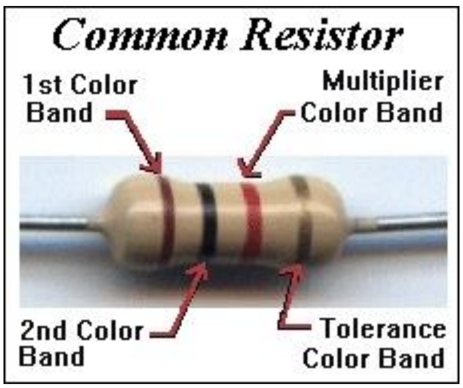
Appendix B: Measuring the voltage across the surface mount LED on the Arduino



Resistor Color Codes:

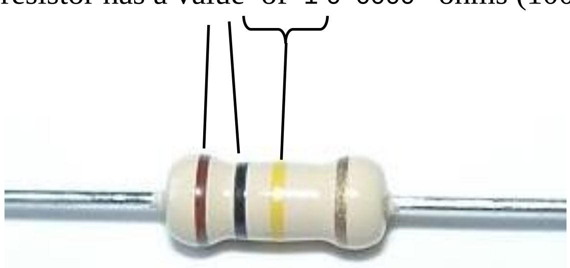
The value and percent tolerance of a common resistor are indicated by a series of colored bands, in which each color represents a digit:

0	1	2	3	4	5	6	7	8	9

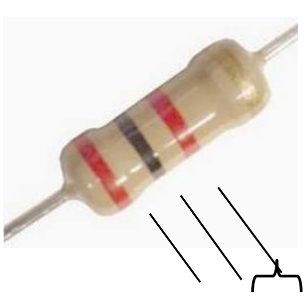


To determine the value of a given resistor, position the gold or silver "tolerance band" on the right-hand side, as shown t the left. Next convert the colored bands into three digits. The resistor value is equal to the first two digits, reading from left to right, followed by the number of zeros indicated by third "multiplier" band.

Thus, for example, for the resistor shown below, **Brown** = 1, **Black** = 0, and Yellow = 4, so the resistor has a value of 1 0 0000 ohms (100 kΩ).



Similarly, the resistor to the left has a value of Red = 2, **Black** = 0, and Red = 2, or 2 0 00 ohms (2 kΩ).



2 0 0 0