

Electronic Circuit Basics With Tinkercad

By Alex Reyes



Digital Maestro Magazine
digitalmaestro.org

Table of Contents

Table of Contents.....	2
Electronic Circuit Basics with TinkerCAD	3
Closed Circuit with an LED	6
The TinkerCad Pushbutton	12
An LED Circuit with A Push Button.....	14
Troubleshooting the Switch.....	15
The Switch Solution	16
Using A Slide Switch in Circuits.....	17
Circuits and Resistors	22
Links to resources.....	28
Project Properties.....	29
A Circuit with An LED and Resistor	31
Circuits with A Potentiometer.....	36
Potentiometer Resistance Value	38
Student Activities	40
Circuits with Capacitors	41
Charging and Discharging the Capacitor.....	44
Student Activity:	45
Circuits with Arduino.....	46
Blinking LED.....	51

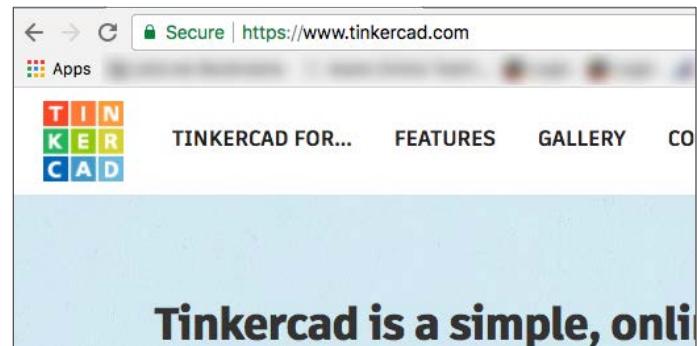
Electronic Circuit Basics with TinkerCAD

TinkerCAD is a free online service for creating basic 3D shapes and developing digital prototypes of electronic components. These prototypes include basic circuits with LED lights, buzzers, switches, and even light sensors.

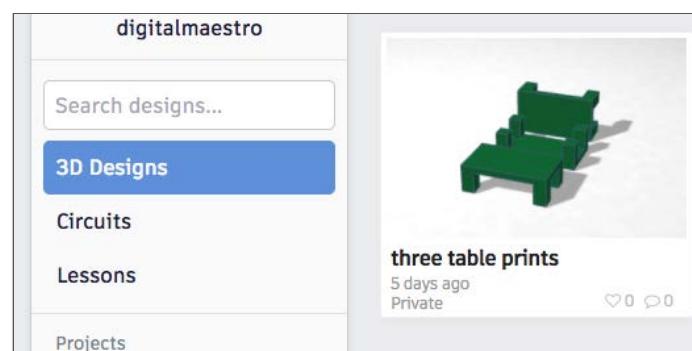
These prototypes can include a microprocessor as part of the design. Microprocessors are the simplest form of computer that can be programmed. They can be programmed to manipulate electronic components like LED lights and buzzers. Microprocessors can be programmed to gather information from sensors and interpret that information. They are used in a variety of devices all around us. They are in microwaves, refrigerators, cars, computers and many other electronic devices.

The process used in TinkerCAD is often used for rapid prototyping. Prototyping is a process where we can develop components in a flexible manner than can be quickly updated and modified to test a variety of options when developing a project or product. We will use this process of prototyping to learn how to create basic electronic circuits.

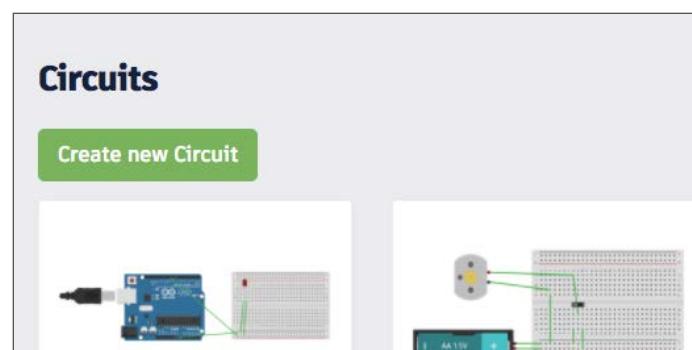
Go over to <https://tinkercad.com> and create a free account. Tinkercad integrates with Social Media services like Facebook. It also Integrates with services like Microsoft and Google. Students can use their district accounts to log into Tinkercad if your district uses active directory accounts with Google or Microsoft.



Go to the left side of the page after logging in and click the circuits menu option.

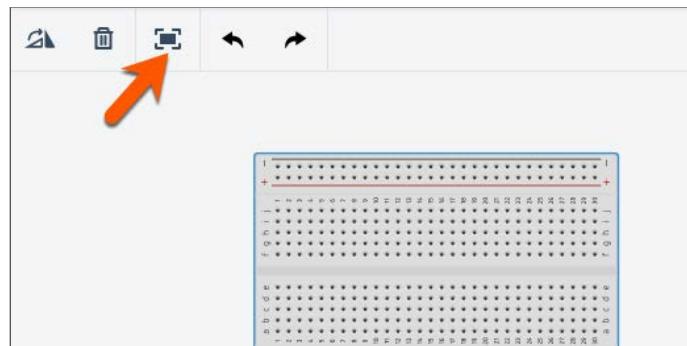
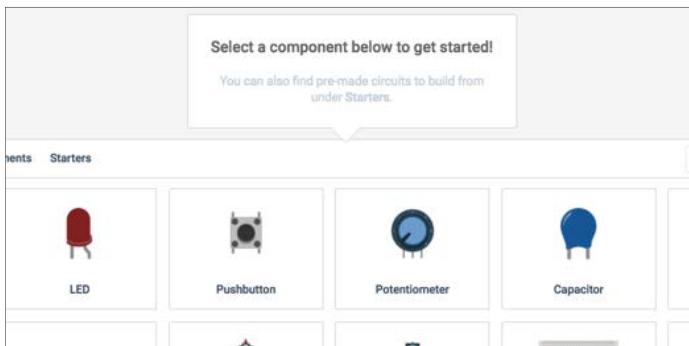


Click the create button to build a new circuit.

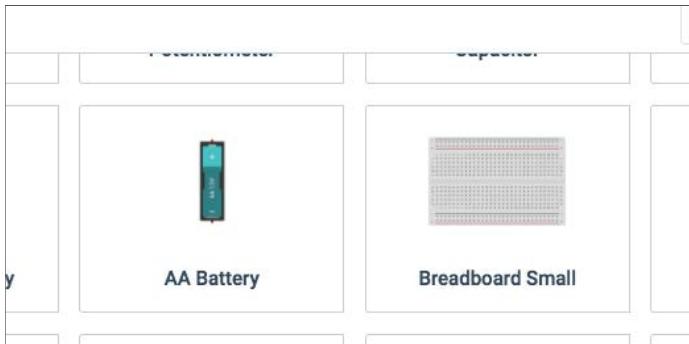


A circuit project is created and opened. A circuit project includes a variety of electronic components. Electronic components include LEDs, buttons, resistors, and a power source. The components we can use are displayed in a panel at the bottom of the page.

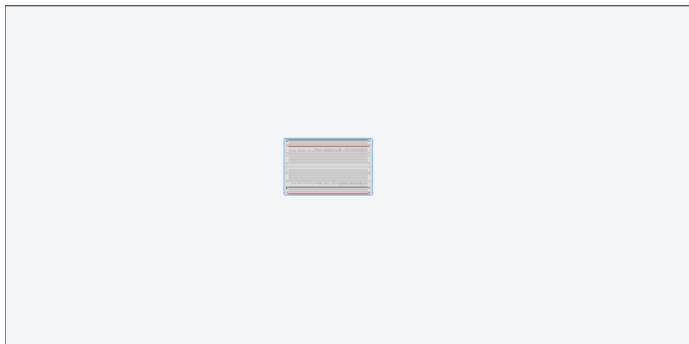
Electric Circuits with Tinker CAD and STEM Integration



Components are commonly assembled using a Breadboard. A Breadboard is a piece of plastic that has several holes. These holes are used to hold different components. Find the Breadboard component and click on it once.

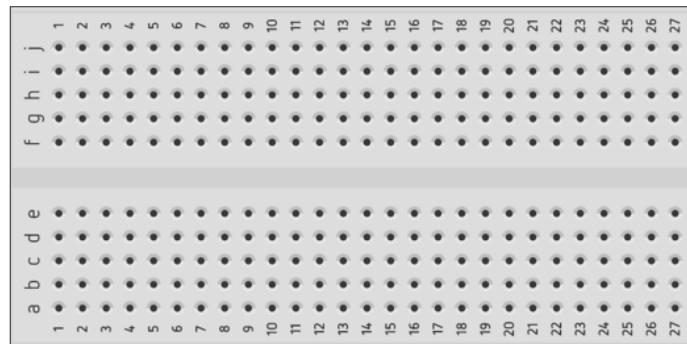


The Breadboard will be selected and temporarily attached to the mouse pointer. Move the mouse pointer onto the workspace and click the mouse button to place the Breadboard onto the workspace.

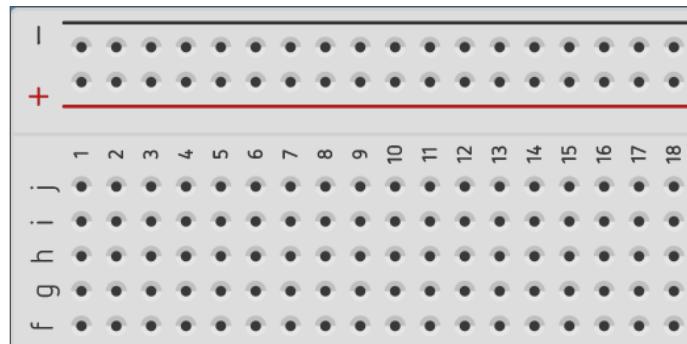


The Breadboard might appear too small or too large in the work area. Click on the zoom to fit button so the Breadboard is centered and magnified.

The Breadboard has a grid of thirty by ten holes in the main area. The rows are numbered 1 to 30 and the columns are labeled with the letters A through J. The columns A through E are separated from columns F through J by a piece of plastic.

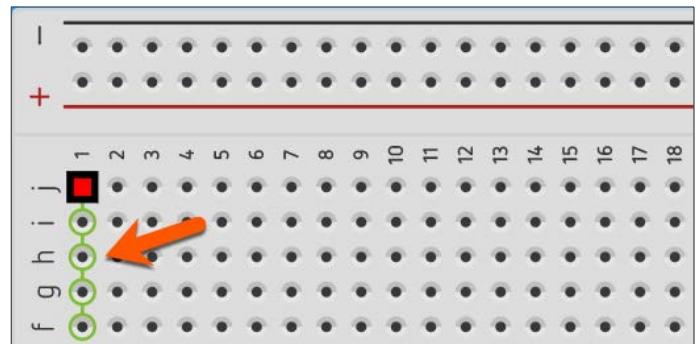


The edges of the board contain two columns with the same number of rows. These columns have negative and positive symbols. These columns and holes are used for the power supply. The components in the center part of the board will tap into these columns to draw electric current.



Electric Circuits with Tinker CAD and STEM Integration

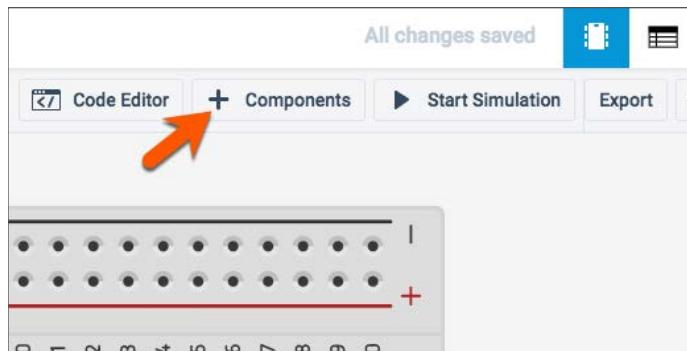
Move your mouse pointer over one of the holes in the center part of the board. The hole beneath will be identified with a red square and a black border. The other holes will be identified with green circles. These green circles indicate that each hole in the row is connected to each other hole in the row. There is a wire linking any connection within the same row. We will use this linking to help develop connect components.



Let's jump in and build our first circuit.

Closed Circuit with an LED

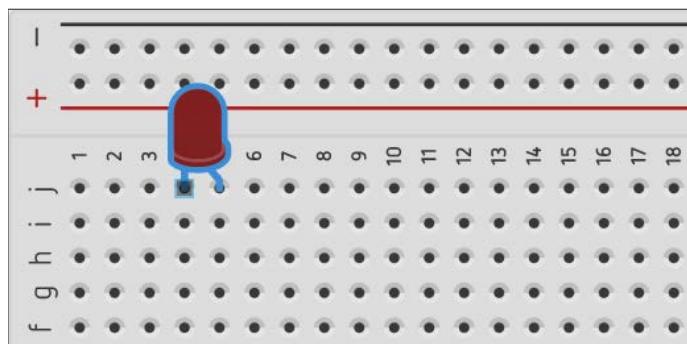
Our first project will create a basic closed circuit with a light. Our light will be supplied by a Light Emitting Diode, LED. Click the Components button to open the components drawer if it isn't already open.



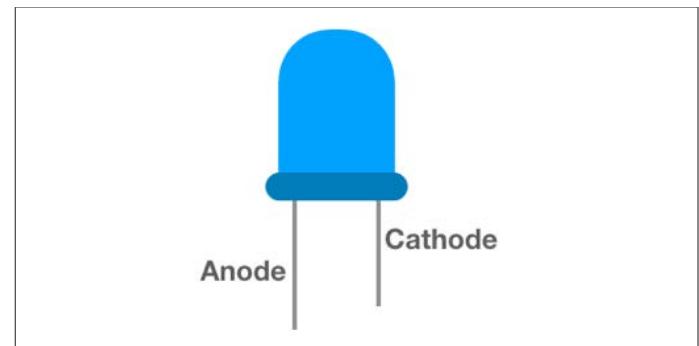
Click the LED component. It will attach itself to the mouse pointer.



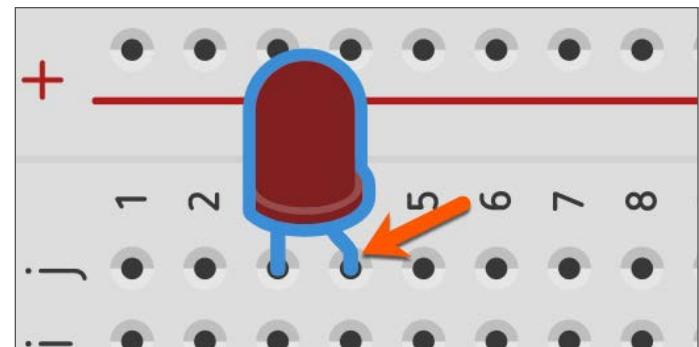
Place the LED onto the board so that each wire of the LED is in a hole. The wire coming from a component is often called a lead. It is pronounced like the word lead in **leader** and not the soft metal lead.



An LED has connections that make it different from a regular light bulb. An LED has one connection called a Cathode and another called an Anode. The Anode lead is usually longer than the cathode. This distinction is important because the anode must be connected to the positive end of an electric circuit. Current flows in only one direction through an LED.

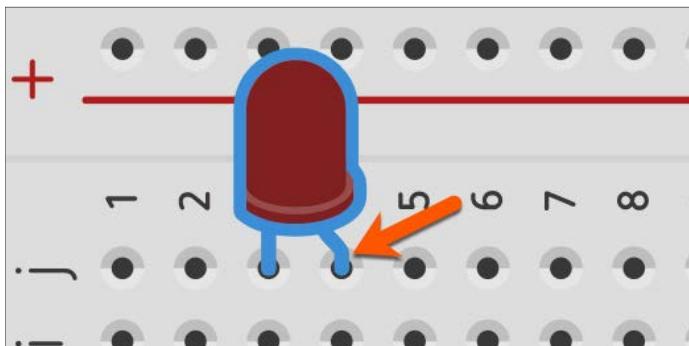


The LED anode in our circuit is identified by a bent lead. This is where the positive current must connect.

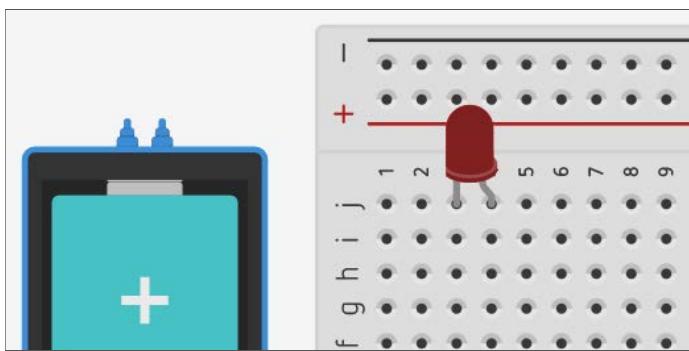


We need a power source to power the LED. Open the components panel and find a battery. We will use the AA battery with 1.5 volts for our first circuit.

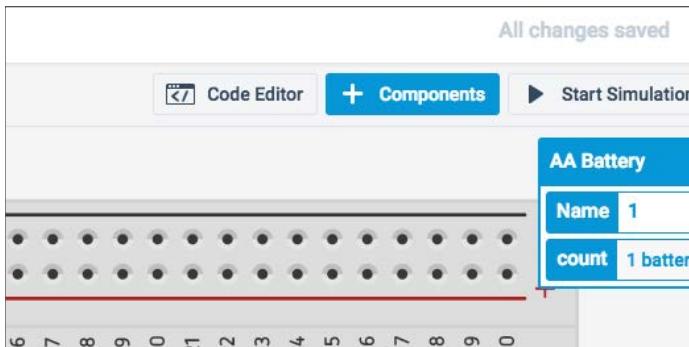
Electric Circuits with Tinker CAD and STEM Integration



Place the battery along the left side of the Breadboard.

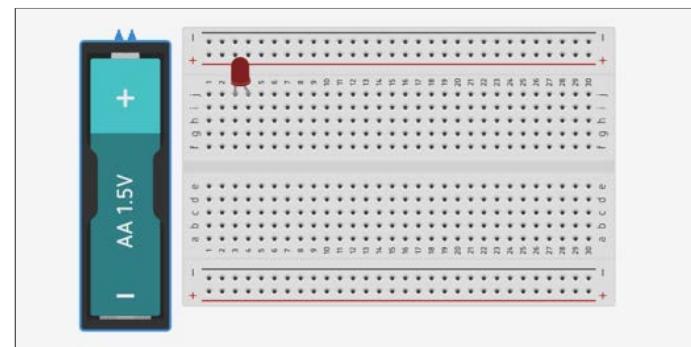


Click the active components button to close the components panel.



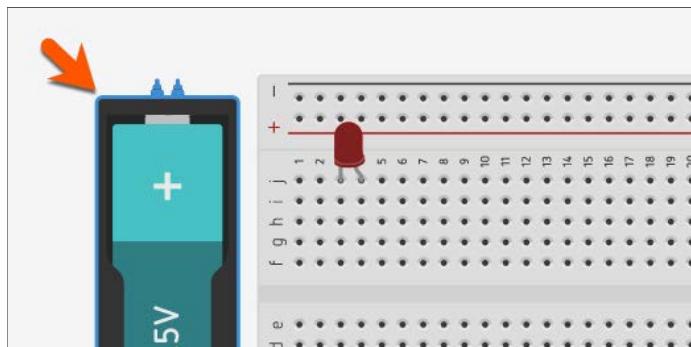
We can zoom in and out of our project using the scroll wheel on our mouse or touch gestures on a track pad, but that can be difficult. It might be easier to use these short cut key combinations. If you are on a Windows or Chromebook computer, hold the Control key and press the Plus key to zoom in or the minus key to zoom out. On a Mac, hold the Command key and use the Plus to zoom in or the minus to zoom out. *Tap the plus or minus key a couple of times.*

Zoom out of the project so you can see the battery alongside the Breadboard.



The terminals for the battery are at right angles to the Breadboard connectors. Battery terminals are the negative and positive connections. The terminals in this example, are pointing up.

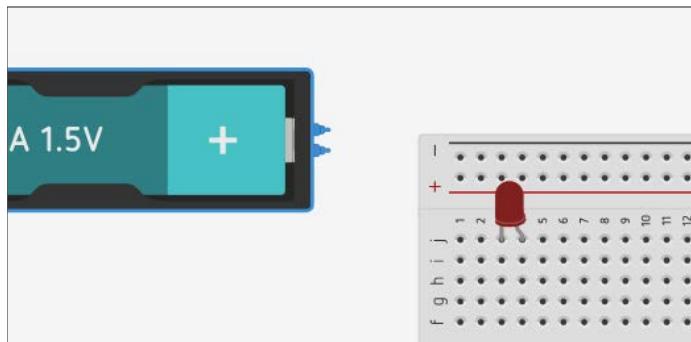
I find it useful for the terminals to be directly across from the connectors. We can rotate the battery so the connections are easier. Click the battery once to make sure it is selected. A blue border appears around selected components.



Click the rotate button in the button bar. This will rotate the component clockwise in small increments of about 30 degrees each. Click the button three times to rotate the battery 90-degrees clockwise.

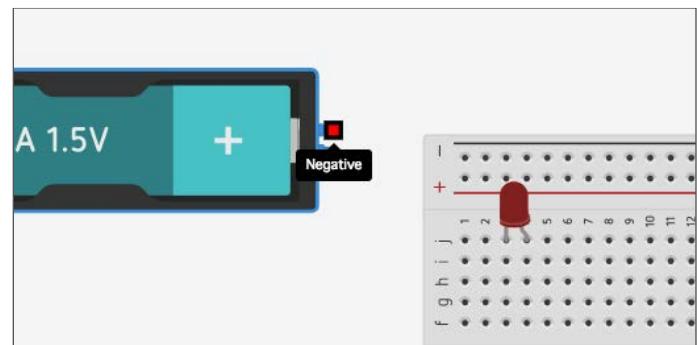


The terminals should be facing the Breadboard.

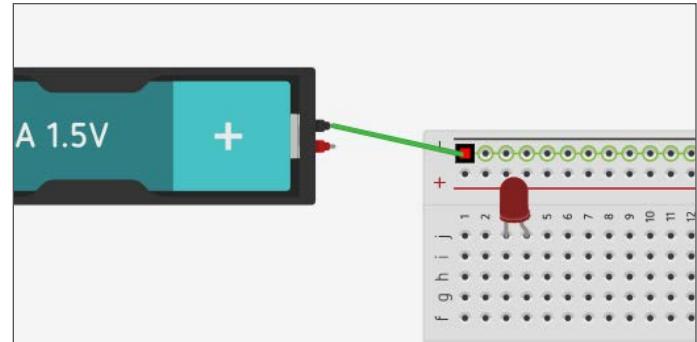


We need to connect the battery to the Breadboard so it supplies voltage to our LED. We connect components to one another when they are not on the same row with Lead wires or Jumper wires. These are wires coated with a plastic and exposed end that can be used to jump from one component to another. This is one reason they are called jumper wires.

Move your mouse pointer over the top terminal. A square will appear over the terminal and a label will appear identifying the terminal as the negative battery terminal.

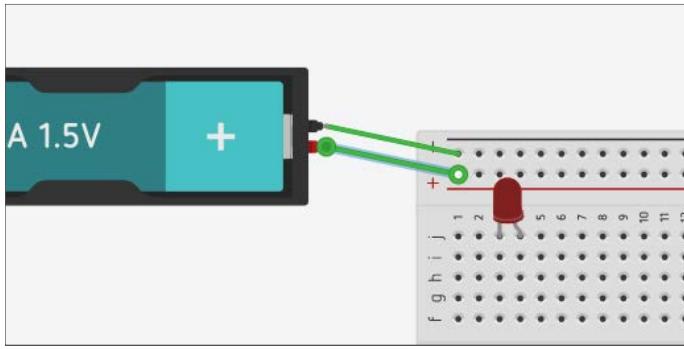


Click once on the terminal and move the mouse pointer to the first hole in the negative column. Click once on the hole to complete the connection. This is how we create jumper wire connections. All the other holes in the negative column are identified with a green circle. This means that all these holes are joined and can be used to connect the negative end of wires or components.



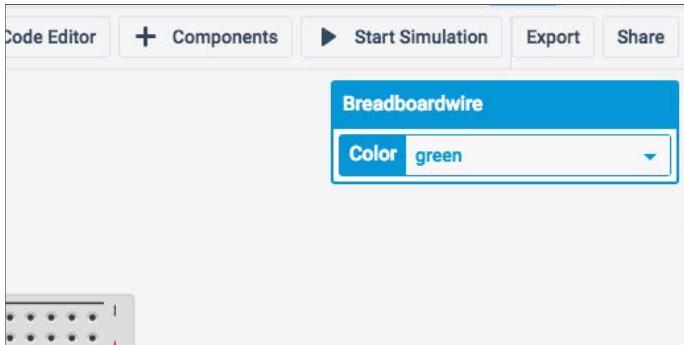
Electric Circuits with Tinker CAD and STEM Integration

Repeat the process with the positive terminal and place the jumper wire onto the positive hole on the breadboard. When a connection is completed the ends of the wire will be identified with circles.

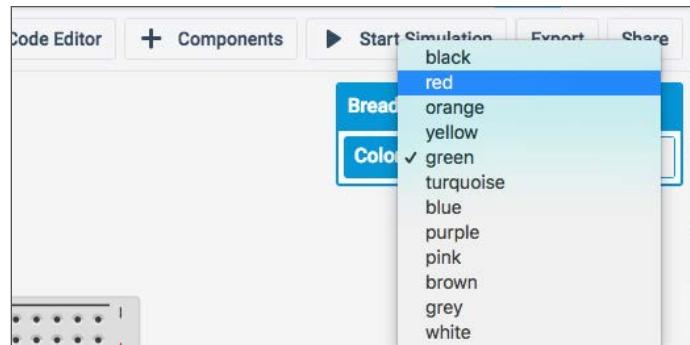


This is a simple circuit with only a few wires but circuits can get complicated very quickly with many wires jumping from one location to another. It is usually a good idea to identify these wires with colors. There are some standard colors that are used in electronics. Red is usually used for positive connections and black for negative connections in DC circuits. Sometimes green is used.

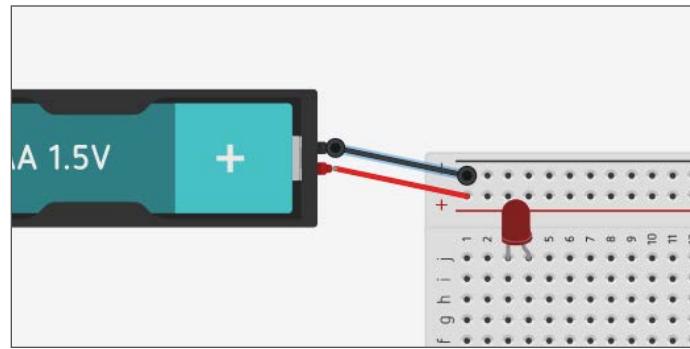
Make sure the positive terminal wire is still selected. A Breadboard wire configuration panel is located on the right side of the page. Click the color selector.



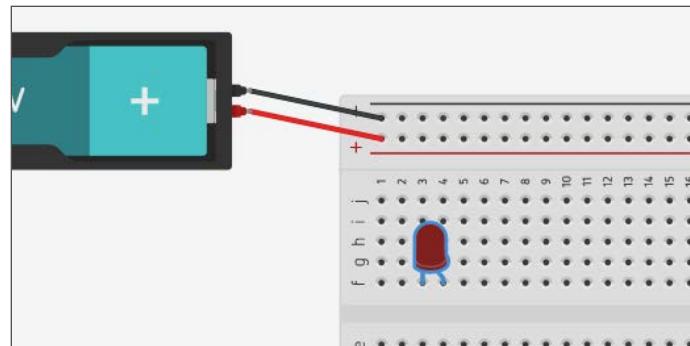
Select the red color.



Repeat the process for the negative terminal wire and change the wire color to black.

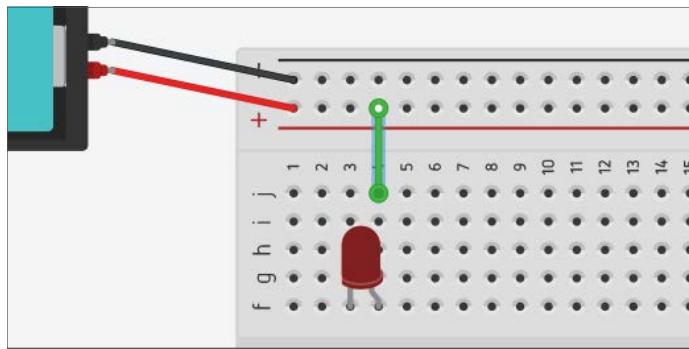


One of the benefits of using a Breadboard is that components can be easily moved around to form new connections or to make room for other components. Click and drag the LED to column f.

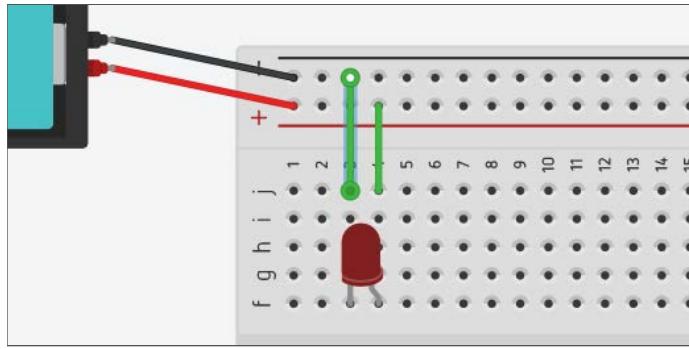


Electric Circuits with Tinker CAD and STEM Integration

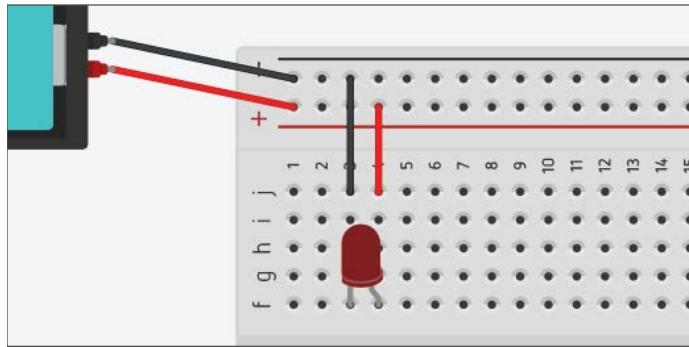
The LED will light up when we have a closed circuit. This circuit isn't closed yet. We need two wires to close the circuit. One wire for the positive and one for negative. The positive current must flow through the anode in our LED. Click on Row 4 Column J to begin a jumper wire. Connect the other end of the wire to the positive column.



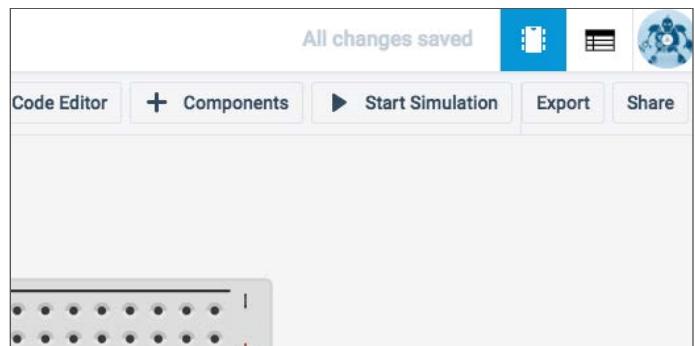
Connect another wire from Row 3 Column J to the negative terminal column.



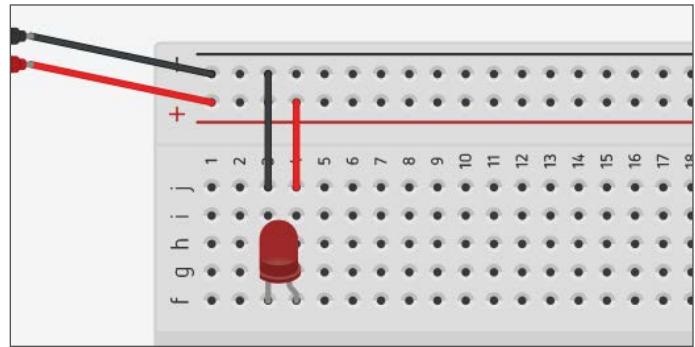
Our circuit is now complete. We should change the color or our wires so they match the polarities.



In the simulation, nothing happens until we run the simulation. Click the Start Simulation button.



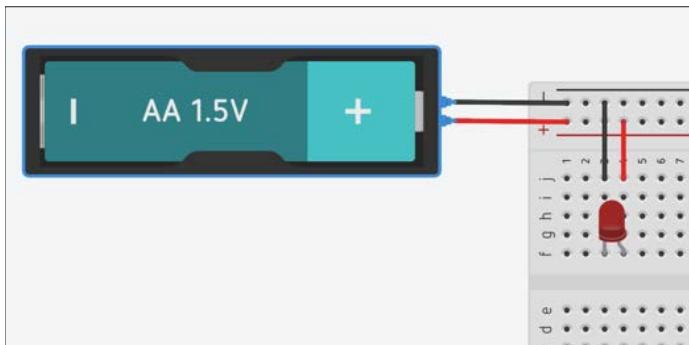
The LED on the Breadboard will change to a lighter color to simulate that the LED has turned on.



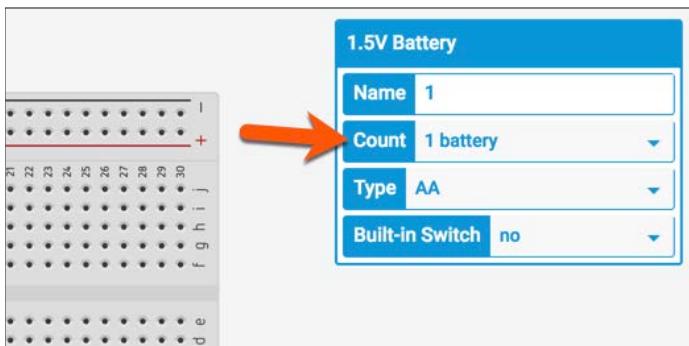
The LED doesn't appear all that bright. That isn't because of the program but because of the voltage being supplied to the LED. The single battery we used in the circuit is supplying a low voltage. In this example, we want a low voltage because LEDs have a limit. Too much voltage and the LED will blow out and stop working. This is where a simulator is very useful. In a real-world application, we wouldn't want to burn out our LED.

Electric Circuits with Tinker CAD and STEM Integration

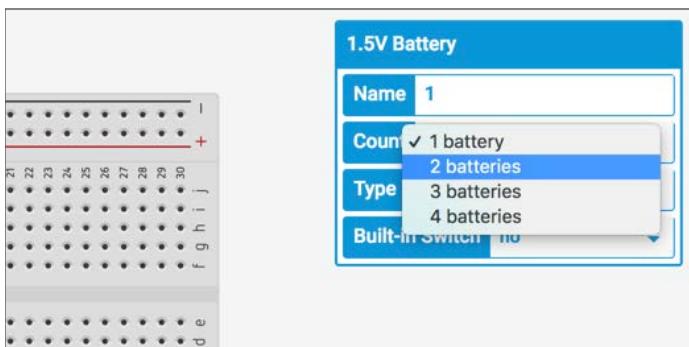
Click the battery once.



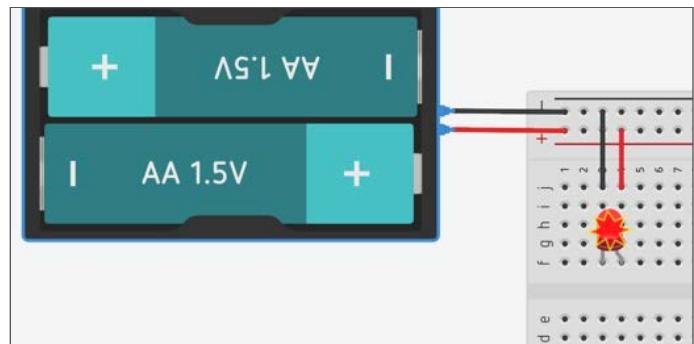
Go to the battery configuration panel and click the battery count selector.



Select two batteries.



Start the simulation and observe what happens to the LED. The star over the LED indicates that the LED blew out. In the physical world, this would mean that our LED is no longer good and needs to be replaced. LEDs are not very expensive but if you start blowing out several at a time it can get expensive. In a future lesson, we will learn how to use resistors so the LED doesn't get damaged.



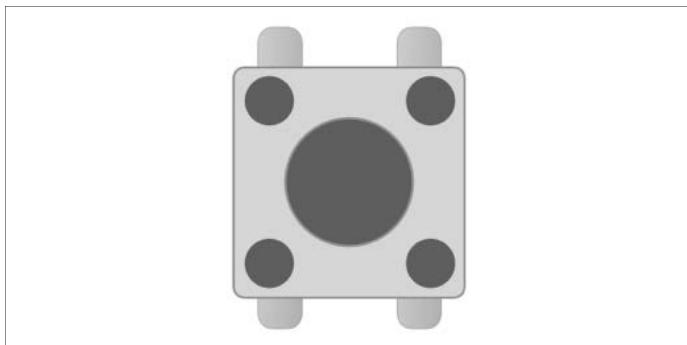
Click the battery component and return the number of batteries to one.

In this lesson, we created a circuit and used the Simulate button to turn the LED ON or OFF. In the physical world, we use switches and in the next lesson we will use a switch to turn the LED ON and OFF.

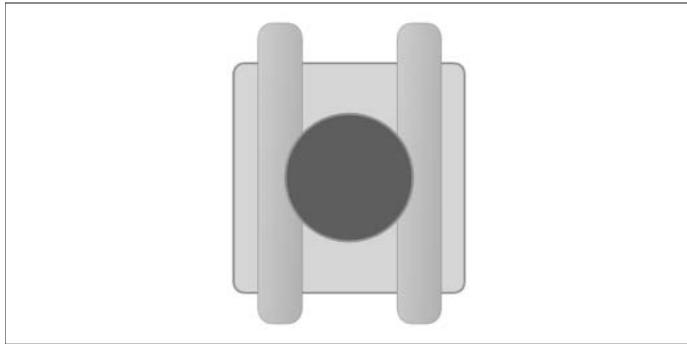
The TinkerCad Pushbutton

In this lesson, we will be using a pushbutton to complete a circuit with an LED. Before creating the circuit with the pushbutton, let's take a look at how the pushbutton works. The pushbutton is a temporary connection. Clicking the mouse button on your computer over the pushbutton will simulate pushing a button and closing a circuit. The mechanism for doing this is not visible so I have created illustrations to help explain.

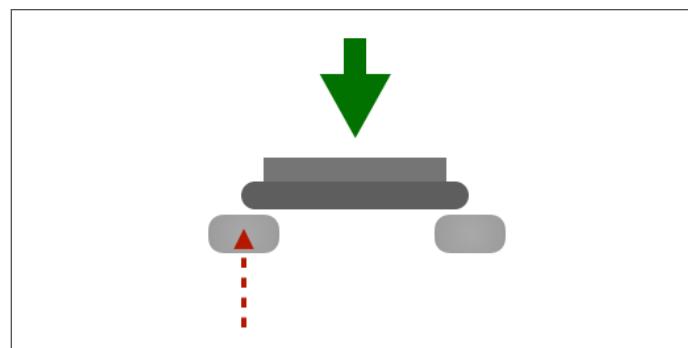
The button has one large circle in the center that is used to close the circuit. There are four connectors on the pushbutton. There is one connector on each corner of the button.



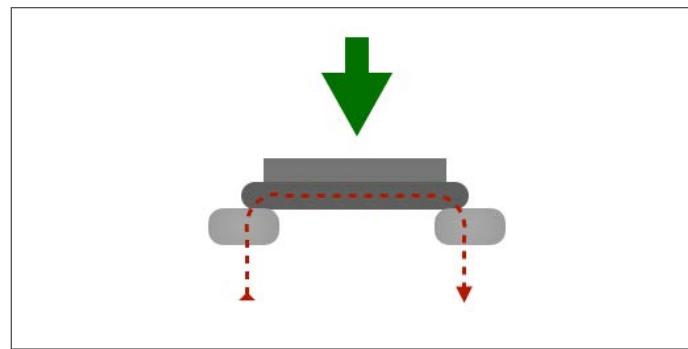
Looking inside the button just under the cover, we see that the left and right connectors are really two wires. One wire on each side of the button.



Looking at the button from the side we see that the button floats above the two wires with a spring. The spring isn't shown in the image. When the button is above the wires as shown in the illustration, the electric current cannot flow through the circuit. The current is shown in the image uses a red dashed arrow pointing up.

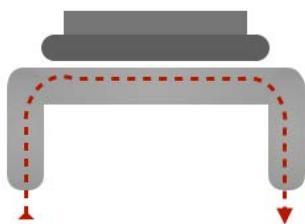


When the button is pressed, the circuit closes. The arrow with the dashed lines in the image represents current flowing from one end of the switch to the other.

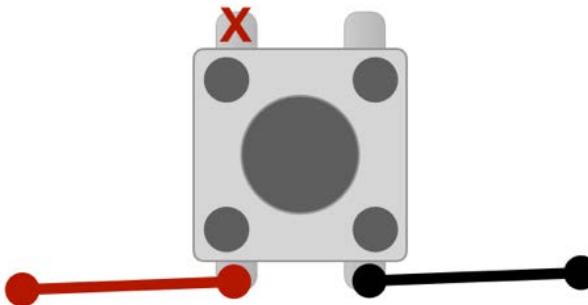


If we connect a circuit from one side of the switch to the same side of the switch on the other end, then the wire within the switch will allow current to flow through when the button is not pressed.

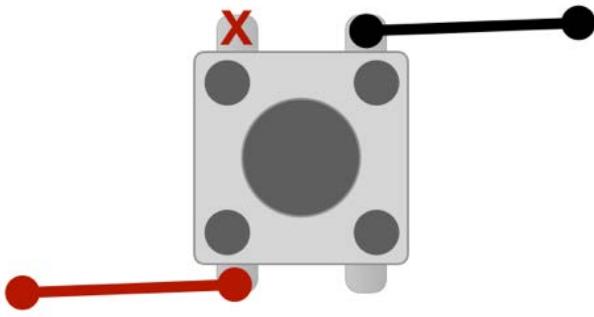
Electric Circuits with Tinker CAD and STEM Integration



Connections can also be made at the opposite end from the other connection as shown in the image below.

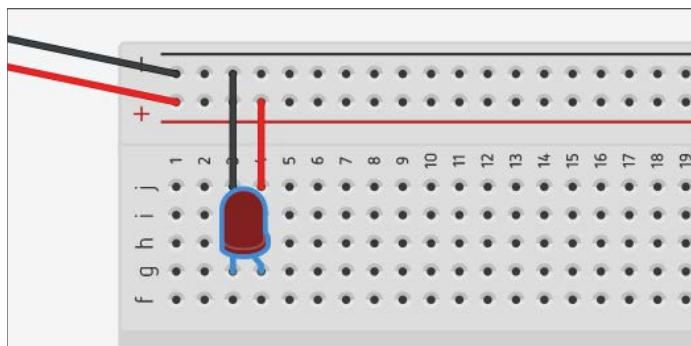


Connecting to the proper ends of the button is important to make sure the circuit works as intended.

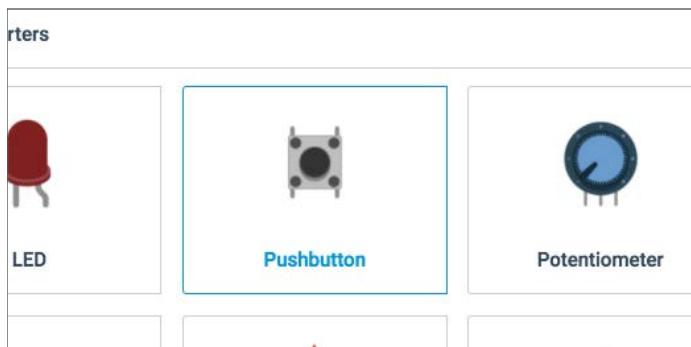


An LED Circuit with A Push Button

Circuits usually have a button that is used to open and close the circuit. We will be using the same circuit constructed in the previous lesson and add a button.

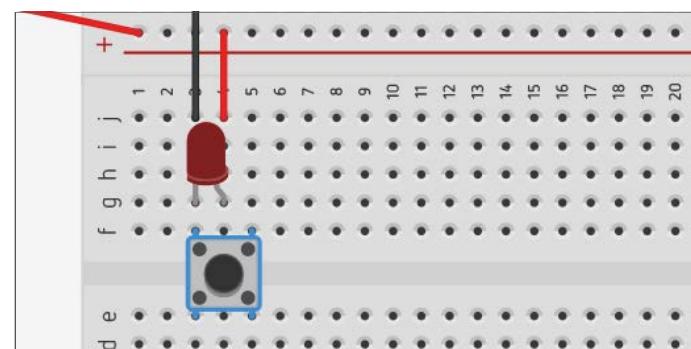


Open the components panel and find the Pushbutton component.

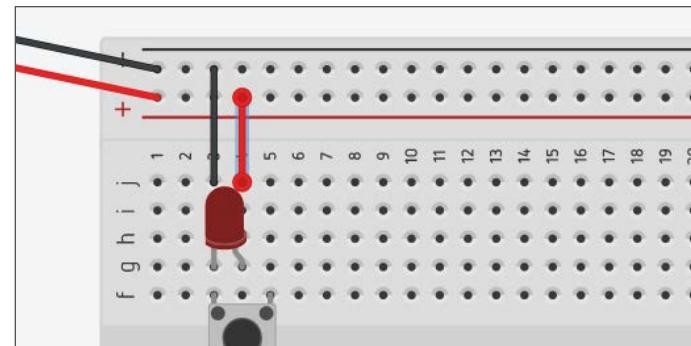


Place the Pushbutton component so that one end is on column F and the other end is on column E. In this switch, the diagonal terminals represent the opposite ends of the switch. This means that the current will flow from E5 to F3, as represented in the diagram below.

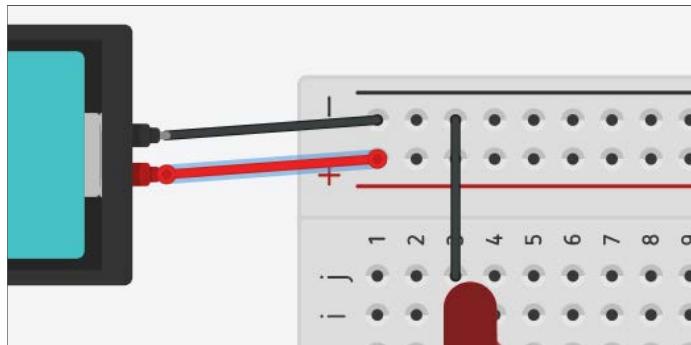
The closed circuit will include the button. Placing the button where one end of the button's connector is on f3 forms part of our closed circuit. Remember that everything in a numbered row is connected. This means that the button is connected to one end of the LED, and the jumper wire running from j3 is connected to the negative rail. We need to move the jumper wire from the positive terminal so it connects to the other side of the button. This will complete the circuit.



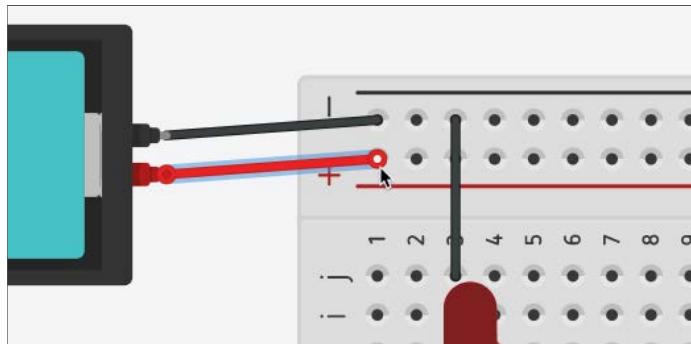
Click once on the positive terminal jump wire. This will select the wire so we can remove it from the circuit. We don't need the wire here and it's a good idea to keep things organized and free of extra wiring. Press the delete key on your keyboard after selecting the wire.



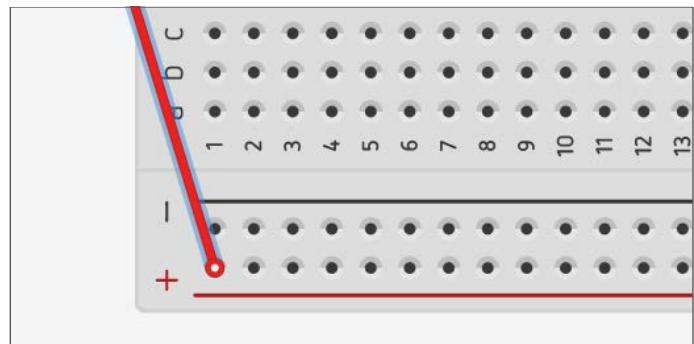
We need the positive connection to be made on the other side of the button. The breadboard has positive and negative rails on both ends. We will reconnect the positive end of our battery pack to the other end of the breadboard. Click the positive terminal wire.



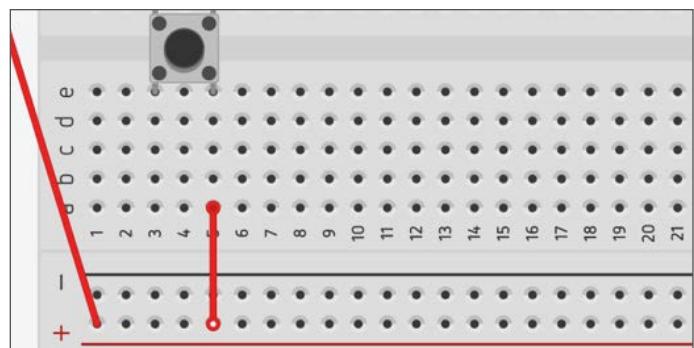
Move the mouse pointer to the end of the wire that is connected to the breadboard. The connection point of the wire will highlight and a white dot will appear where the wire connects to the breadboard. Click on this white dot to detach and drag the wire to another location.



Click and drag the end of the wire to the other side of the breadboard and connect it to the positive side of the rail.



Connect a jumper wire from the positive rail to the row that matches the opposite side of the button.



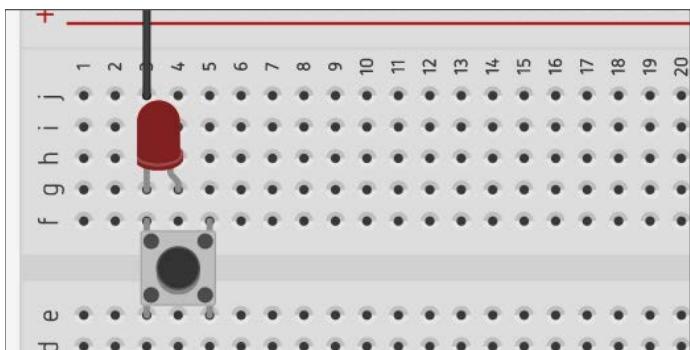
Troubleshooting the Switch

I don't like to solve problems for my audience during presentations and I won't solve the problem for you here either. Run the simulation and press the button. Does the LED turn on? What do you think is the reason for the LED not working as we intended? I'll give you a hint. The anode on the LED must be connected to the positive connection of the electric current flow.

I really did make this mistake and it took me a couple of minutes to realize what I had done. I was going to start all over again but I realized this is a good teachable moment. You are likely to make this mistake yourself and leaving it in seemed like a great teaching opportunity.

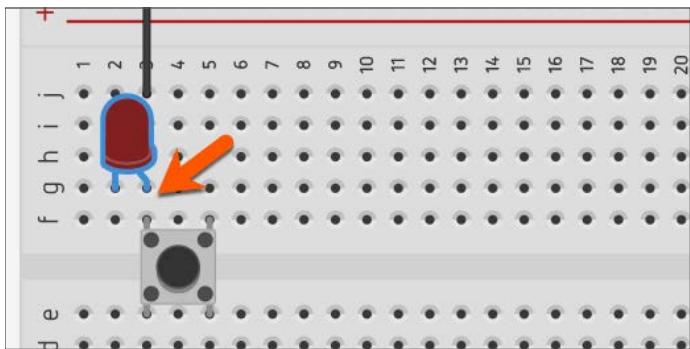
Stop the simulation before proceeding to the solution. We cannot move components while the simulation is running. Just like you should not move components when electric current is flowing through them in a real circuit.

Teachers might want to relate this to a real-world situation. For example, turning off the electricity to a section of their home if they ever intend on working on electric circuits and discuss the dangers of electrocution.

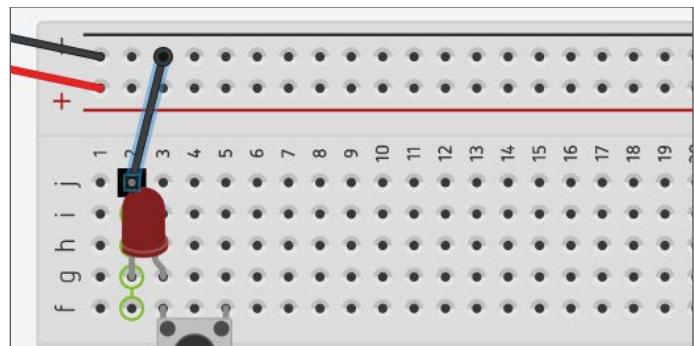


The Switch Solution

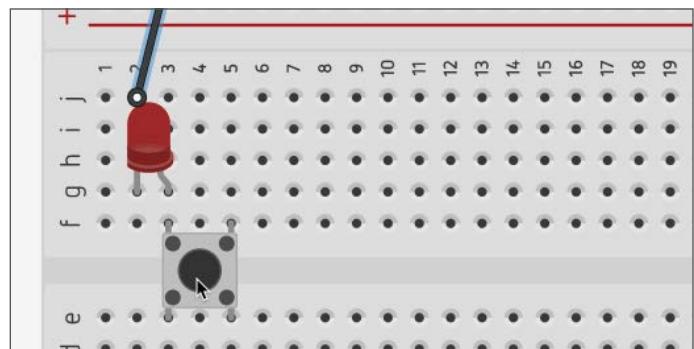
Move the LED so the anode is connected to the push button terminal. This means moving the LED one row up to row 2.



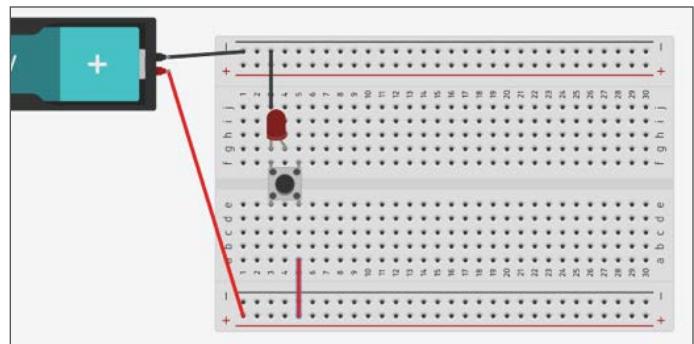
After moving the LED, the negative jumper wire is no longer connected to the cathode end of the LED. We don't need to delete the jumper wire. Existing jumper wires can be moved. Click once on the jumper wire and move one end so it connects the cathode of the LED.



Run the simulation and press the push button with the mouse pointer. The LED will remain lit while the button is pressed. Release the button and the LED will turn OFF.



This is the completed circuit.

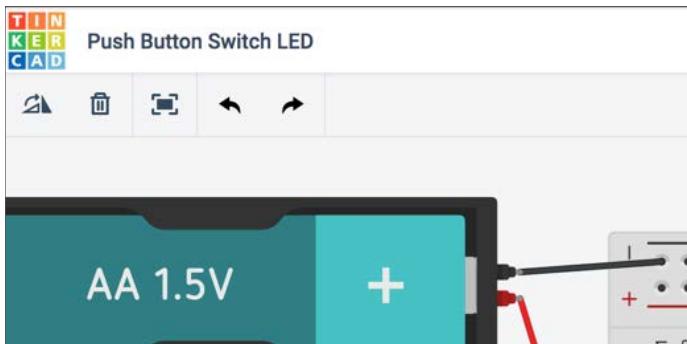


Buttons are a common way of controlling the flow of electricity through a circuit. In future lessons, we will explore other switch options that are controlled by physical switches and computers.

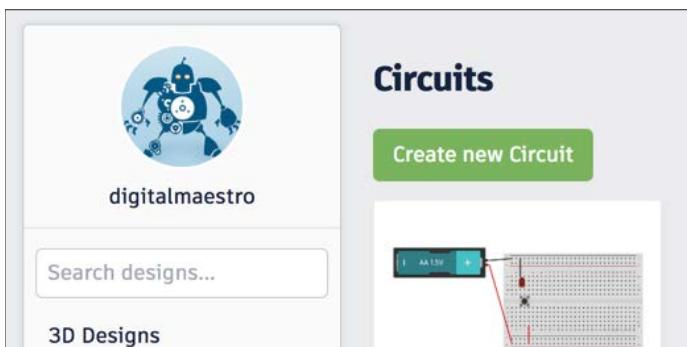
Using A Slide Switch in Circuits

In the previous lesson, we used a Pushbutton to pass voltage to the LED. The Pushbutton is a temporary button. When the button is released, so is the connection that closes the circuit. A Slide Switch is more permanent because it will continue to maintain either an open or close circuit without the user having to keep a finger on the switch.

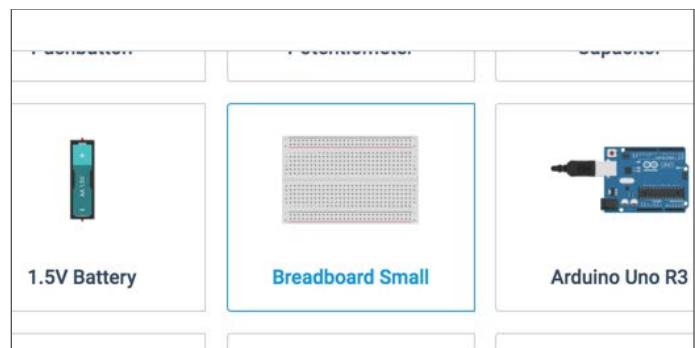
To create a new project for this circuit we need to return to the main TinkerCad page. Click the TinkerCad icon located in the top left side of the web page.



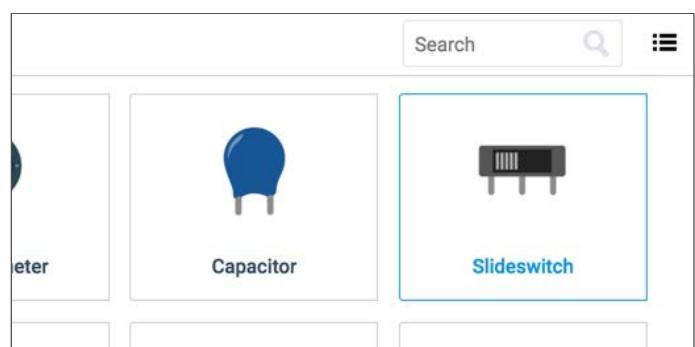
Create a new circuit.



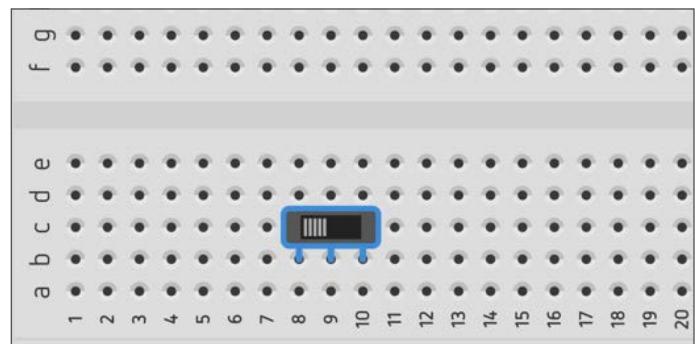
Find the small breadboard component in the components panel. Click on it once and place it on the workspace.



Find the Slideswitch component and place it on the breadboard.

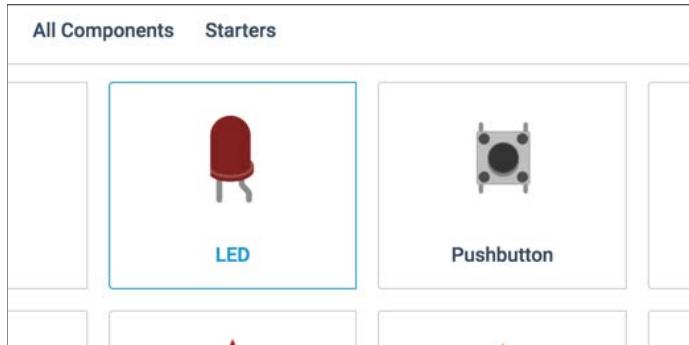


Place the switch in the bottom half of the breadboard. Leave at least one column above and below the switch for jumper wires.

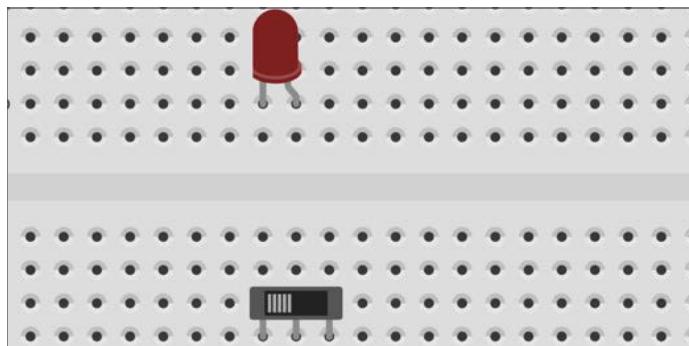


Electric Circuits with Tinker CAD and STEM Integration

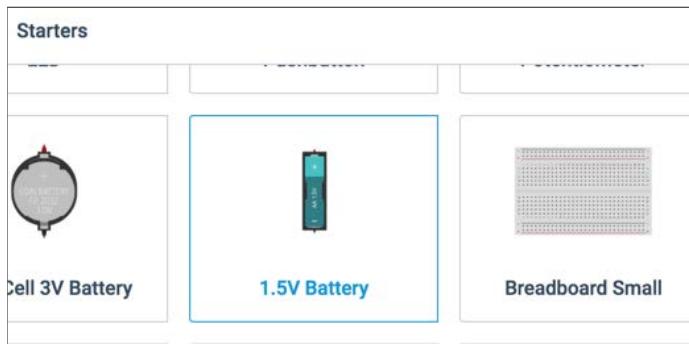
Get an LED to place on the breadboard.



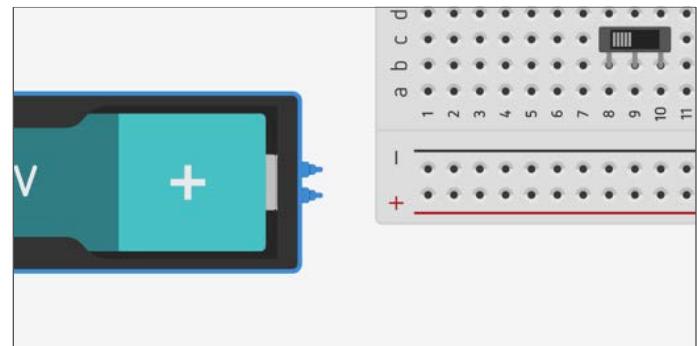
Place the LED in the upper half of the breadboard and leave space for jumper wires.



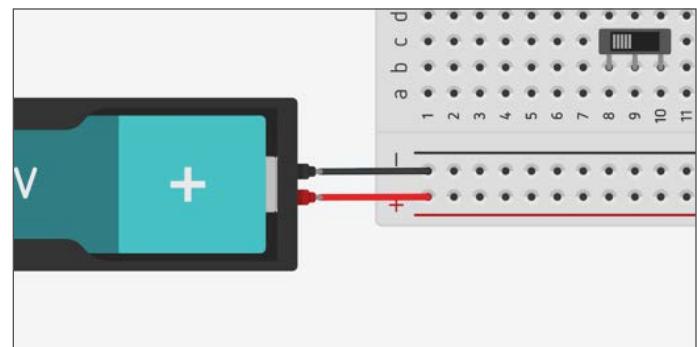
Get a 1.5-volt battery and place it near the lower half of the breadboard on the left side.



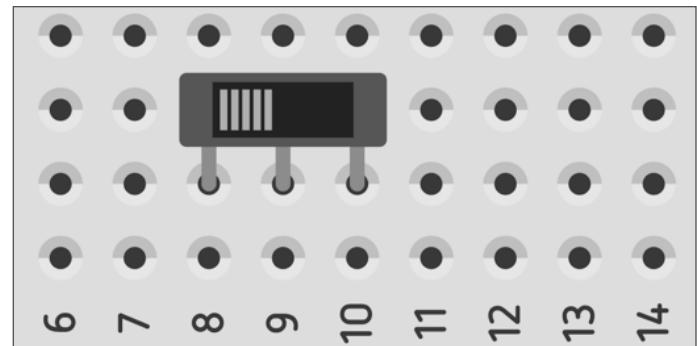
Rotate the battery so the terminals are facing the breadboard near the negative and positive rails.



Connect the negative and positive battery terminals to the corresponding holes on the breadboard. Don't forget to change the jumper wire colors.

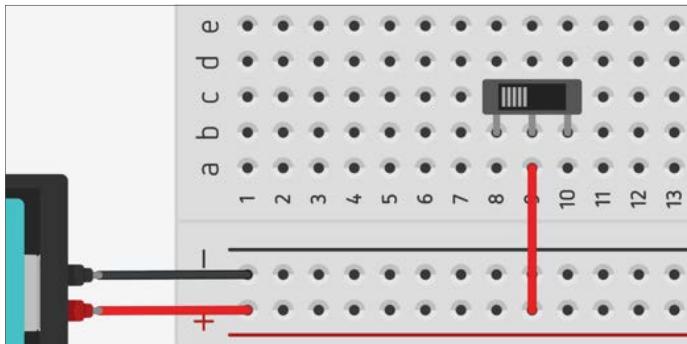


Let's talk a little about the switch. The switch has three wires that connect to the breadboard. Only two of the wires are used to create a closed circuit. The center wire is always part of the circuit. The left or right wire coming from the switch is used to complete the circuit. The wires coming from the switch are called leads. The part of the switch that slides has lines or ridges like a real switch.

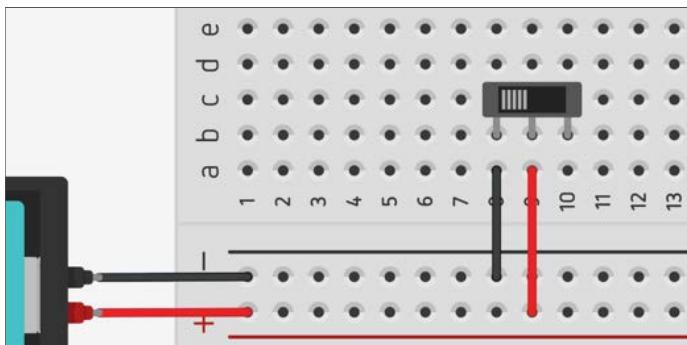


Electric Circuits with Tinker CAD and STEM Integration

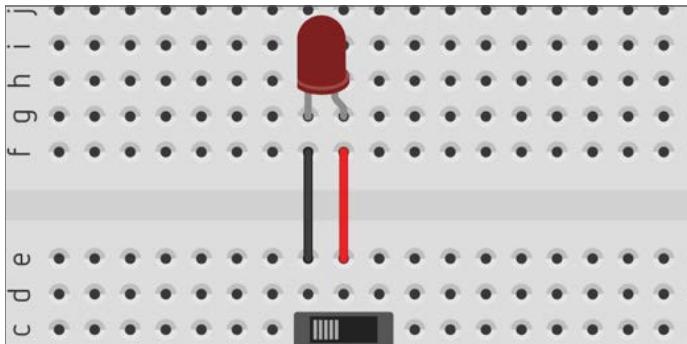
Connect a jumper wire from the positive rail to the center lead on the switch.



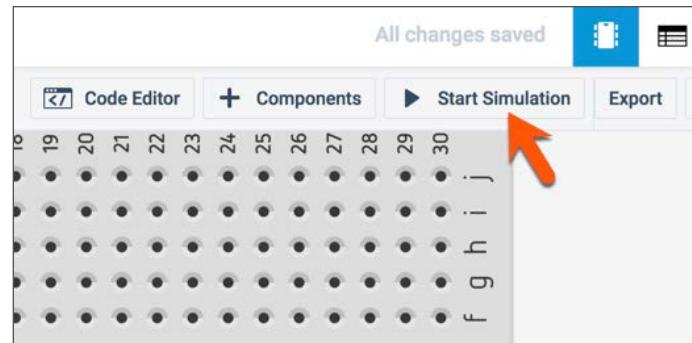
Connect a jumper wire from the negative rail to the left lead on the switch.



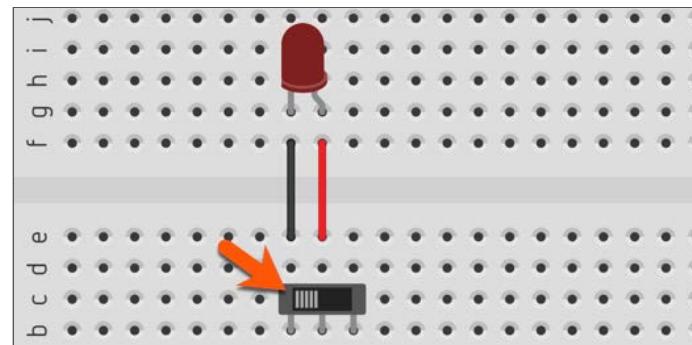
Connect jumper wires to connect the positive row of the button to the row that connects the LED Anode. Remember that the Anode of an LED must be connected to the positive wire. Connect the Cathode side of the LED to the negative row of the switch.



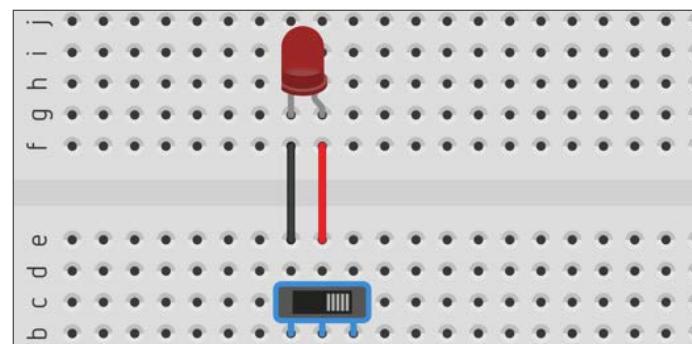
Click the “Start Simulation” button.



The LED is not lit because the switch is in the OFF position. Click the switch once to move it to the ON position.

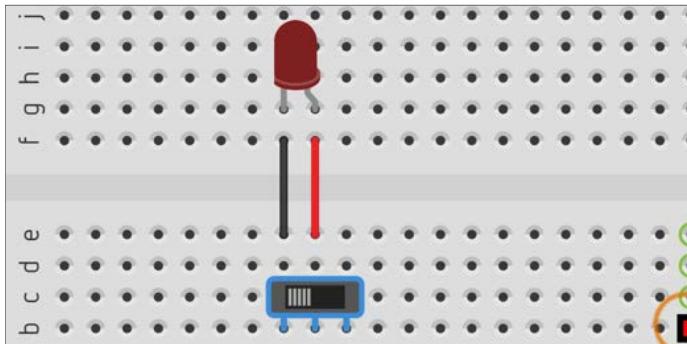


The LED will light and remain lit until we click the switch again to move the Slideswitch or stop the simulation. Click the switch to open the circuit and turn the LED OFF.

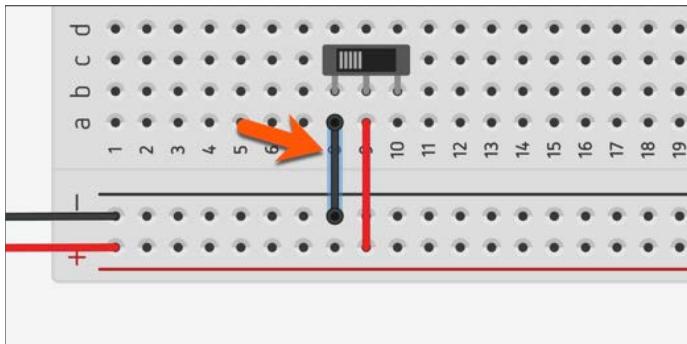


Electric Circuits with Tinker CAD and STEM Integration

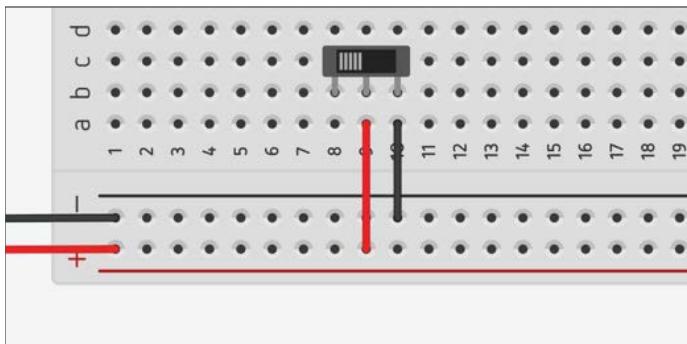
Stop the simulation so we can rewire the switch.



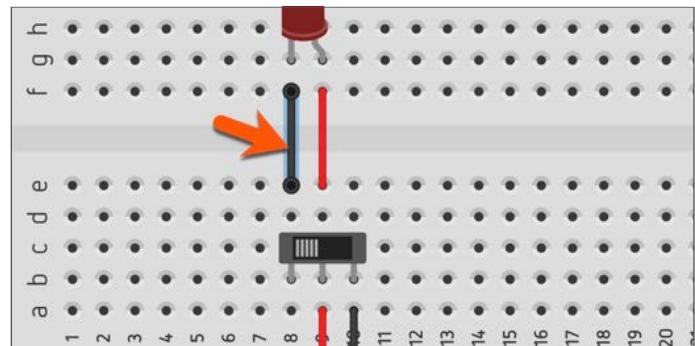
Click once on the jumper wire coming from the negative rail. Press the Delete button on your keyboard to delete the wire.



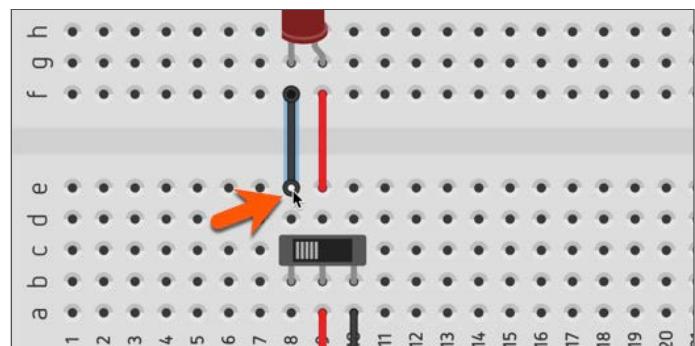
Connect the right side of the switch to the negative rail with a jumper wire.



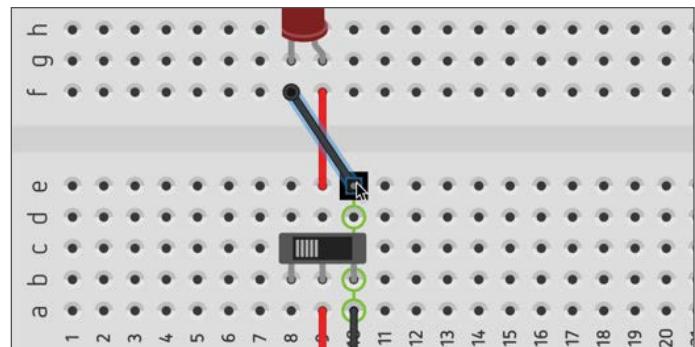
The Cathode side of the LED needs to be connected to the negative side of the switch to close the circuit when the switch is flipped. Click once on the jumper wire for the LED Cathode.



Move your mouse pointer over the bottom end of the wire. The end of the wire will change to a white dot. Click and drag this dot to move this end of the wire to a new location.

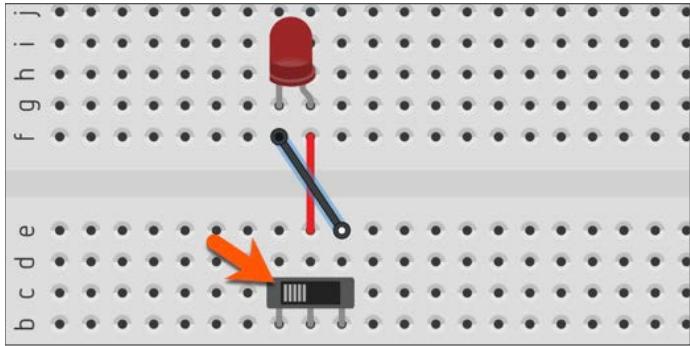


Connect the wire to the row that connects the right side of the switch lead.

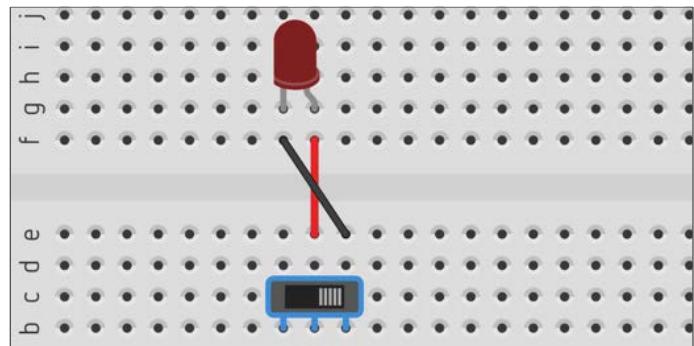


Electric Circuits with Tinker CAD and STEM Integration

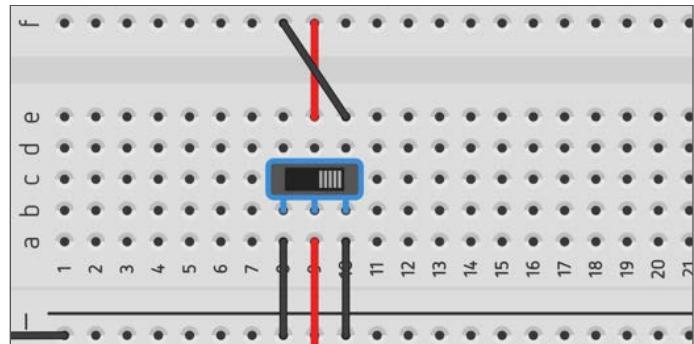
Press the “Start Simulation” button. Notice the position of the switch in our circuit and the status of the LED light. Changing the side that the wires connect to the Switch also changes the side that the Switch must be moved in order for the circuit to be closed. Click the Switch once to open the circuit.



The circuit is open and the light is not receiving current or voltage.



We can use one side or the other of the Switch to complete a circuit. We cannot use both sides of this switch at the same time. Try connecting a jumper wire to both sides of the switch and see what happens.



You will find that the LED will not light at all.

Circuits and Resistors

In the previous lessons, we created a circuit that lit an LED. In one of the lessons we added two batteries to the circuit and the LED was damaged. The LED was damaged because it received too much voltage. LEDs are low voltage or low wattage components and don't need much electricity to get them to emit light. This is one of the reasons they are so useful for lighting homes, businesses, and even cities. Their low power consumption makes them ideal when conserving energy. Power sources like batteries are designed to produce large voltages and generate current to power energy hungry components. In the past, one of these energy hungry components was the Incandescent Light Bulb, which requires more voltage than an LED.

When constructing circuits, we have a variety of components that require different voltages and current. We will learn how resistors are used to restrict the flow of current to electrical components. Electrical current is measured in Amps. In this lesson, we will learn about Ohm's Law and how we use it to understand what is going on when we use resistors in a circuit.

Ohm's Law states that we can calculate the amount of voltage in a circuit by multiplying the electrical current by the resistance in the circuit. In the formula, we use the letter "I" for Current. The letter "I" is used in honor of the man who formulated Ampère's force law by which we measure current and made electric motors possible from electro magnets. André-Marie Ampère, who was French, used it in a French phrase, "intensité de courant, (current intensity)". So, the letter "I" actually stands for Intensity.

$$\text{Voltage} = \text{Current} \times \text{Resistance}$$
$$V = I \times R$$

The formula can be manipulated to find the current in a circuit. To find the current in a circuit we divide the Voltage by the Resistance.

$$\text{Current} = \frac{\text{Voltage}}{\text{Resistance}}$$
$$I = V / R$$

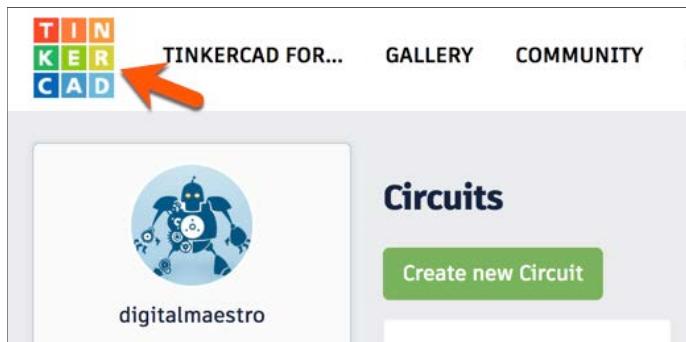
We can use the formula to find the resistance in a circuit. To solve for resistance, we divide voltage by the current.

Electric Circuits with Tinker CAD and STEM Integration

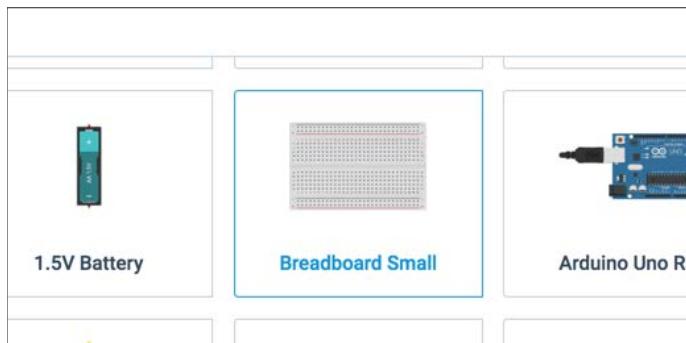
Teaching: Student should work out how we arrived at these formulas. It is a good idea to introduce or revisit formulas and how we can manipulate them to solve for different solutions.

$$\text{Resistance} = \frac{\text{Voltage}}{\text{Current}}$$
$$R = V / I$$

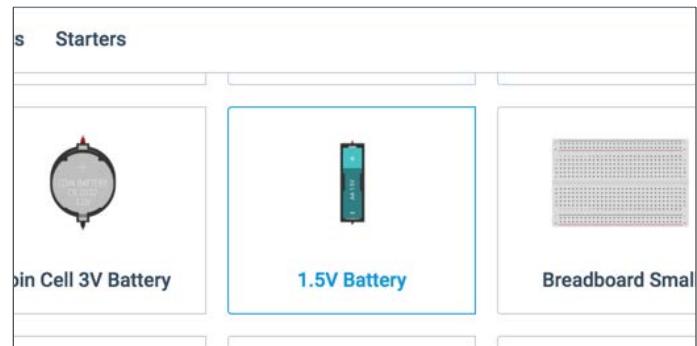
Create a new circuit. Return to the main Tinkercad Circuit page by clicking the TinkerCad logo. Click the “Create new Circuit” button.



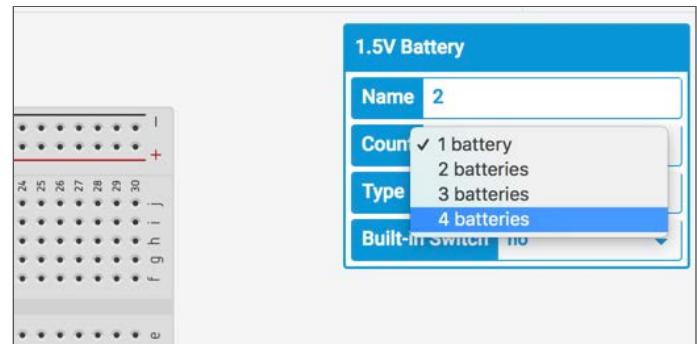
Find the small Breadboard component and place it on the canvas.



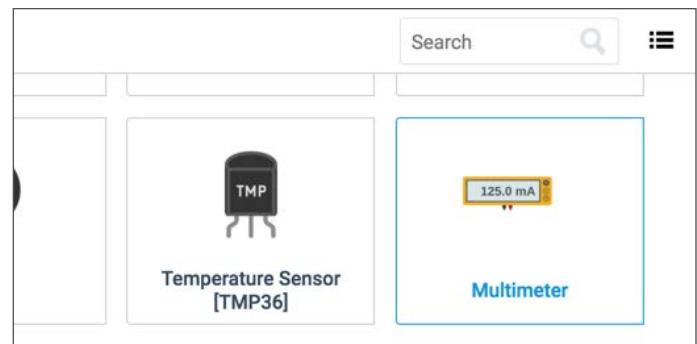
Find the battery component and place it on the canvas to the left of the breadboard.



Go to the battery options panel and change the battery count from one to four.

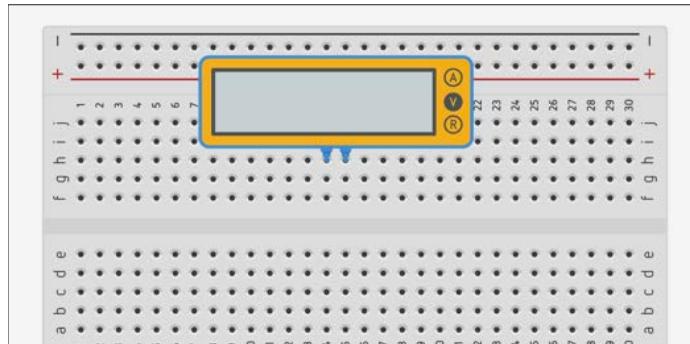


Scroll to the bottom of the components list and find the Multi-meter component.

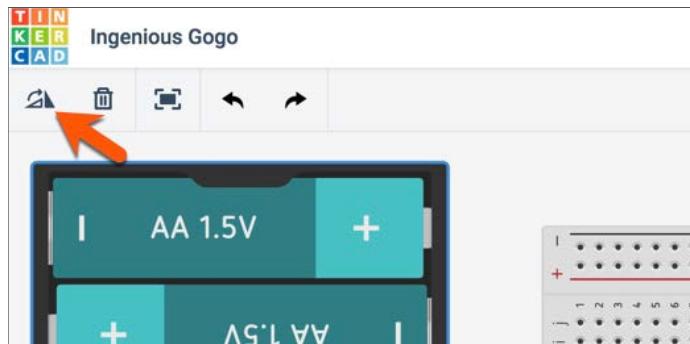


Electric Circuits with Tinker CAD and STEM Integration

Place the Multi-meter near the center of the breadboard in the upper half. Make sure to leave columns f and g available so we can make jumper wire connections.



Click on the battery pack so it is selected and click the rotate button so the terminals on the battery pack are facing the breadboard.



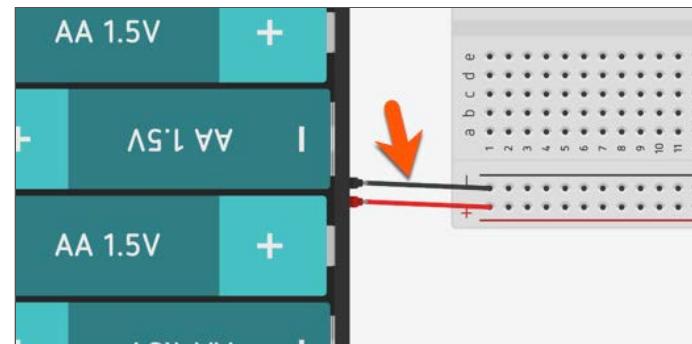
Move the battery pack so the terminals are near the positive and negative rails at the bottom of the breadboard.



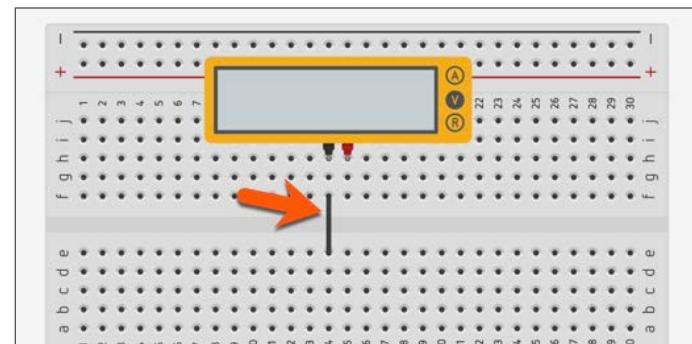
Connect jumper wires from the negative battery terminal to the negative rail on the breadboard and connect a jumper wire from the positive battery terminal to the positive rail on the breadboard.



Change the color of the wires so they are black for negative and red for positive.

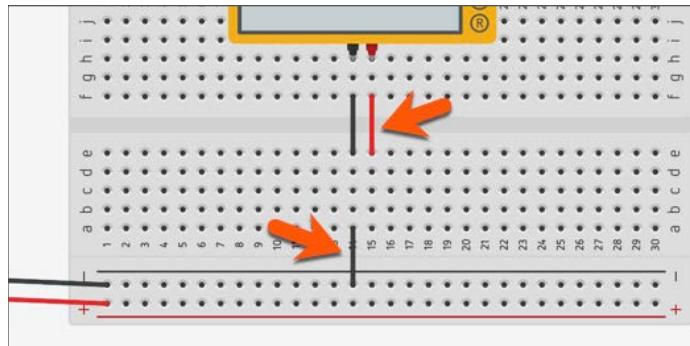


Add a jumper wire that connects the negative end of the Multi-meter across the gap to the other side of the breadboard.

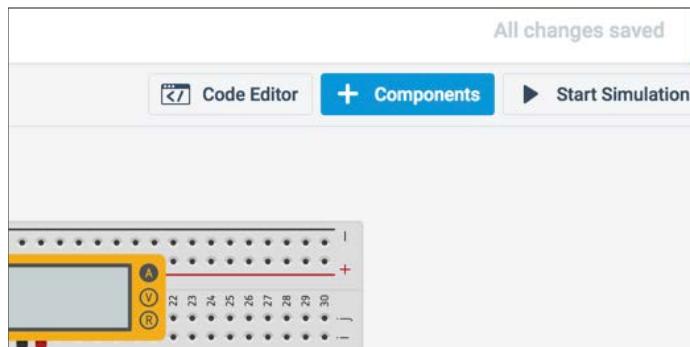


Electric Circuits with Tinker CAD and STEM Integration

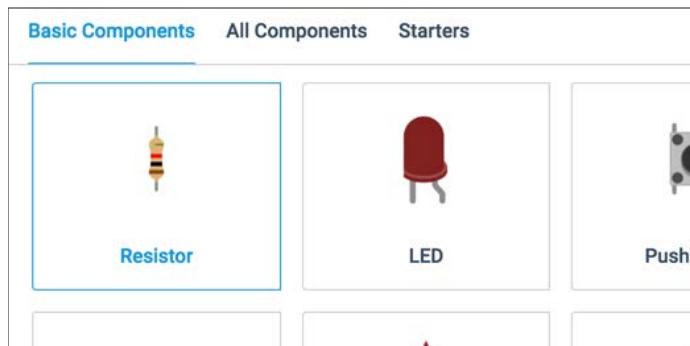
Connect two more jumper wires. One wire will connect the positive terminal of the Multi-meter to the lower half of the bread board. The other jumper wire will connect the negative rail to the multi-meter along one of the rows. In the image below that would be along row 14.



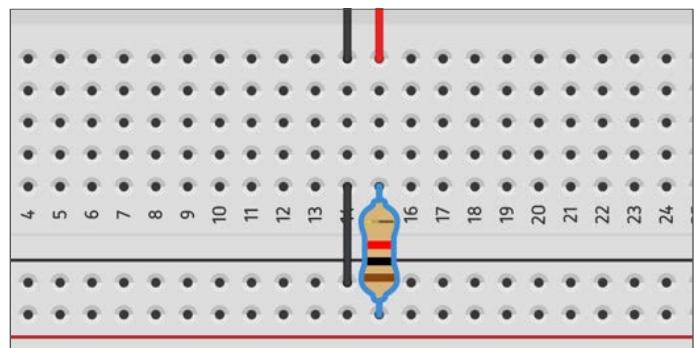
Click the Components button to open the components panel.



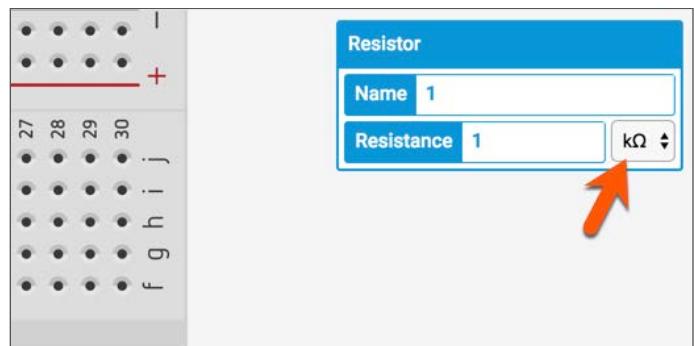
Find the resistor.



Place the resistor on the breadboard so that one end connects to the positive rail and the other end is opposite the positive terminal for the Multi-meter.



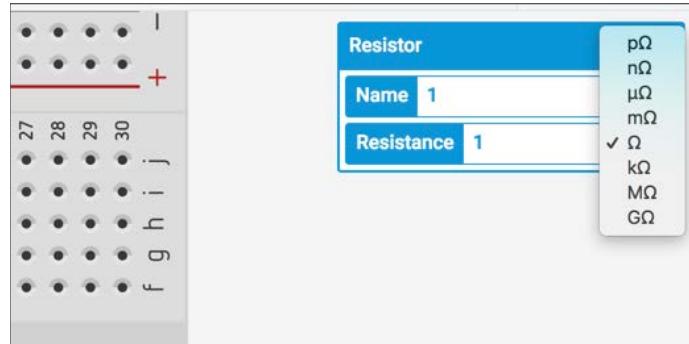
We will be using this setup to learn about current flow across a resistor. While the resistor is still selected we can see that the resistance of this resistor is measured at 1 Kilo-Ohm. The symbol next to the letter k is the Greek letter Omega(Ω). The Omega symbol is used because the “Om” in Omega sounds similar to Ohm. One Kilo-Ohm is 1000 Ohms.



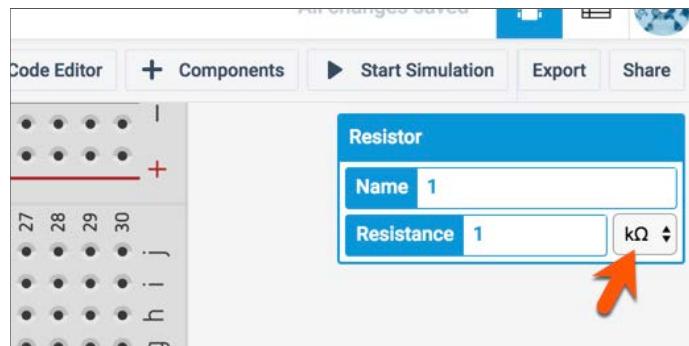
Click the symbol selector to see other measurement symbols for Ohms. There is a total of eight. We won't be using most of these measurement symbols. I want to point out a few that will be used in this and other chapters.

Electric Circuits with Tinker CAD and STEM Integration

The Omega symbol by itself represents one Ohm. The symbol above the Omega with the lower case “m” before it is a measurement in milli-ohms. The Symbol before milli-ohms with the “u” before it is micro-ohms. In our circuits, we will be using Ohms, milli-ohms and kilo-ohms.



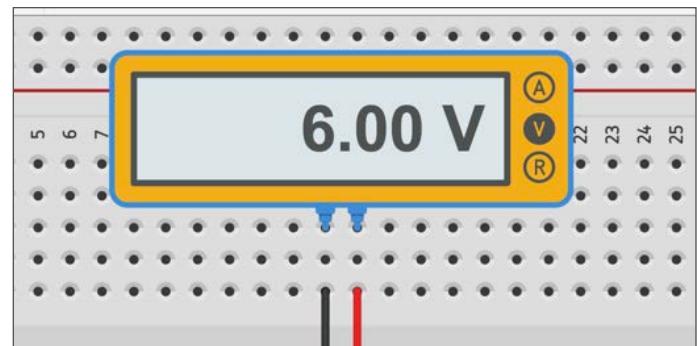
Make sure the resistance unit of measure is set to kilo-ohms and click the Start Simulation button.



The meter measures that there is 5.99 milliamps flowing through the circuit. The Multimeter has three measurement options. The “A” is selected on the meter. This means it is set to measure Amps or Current.

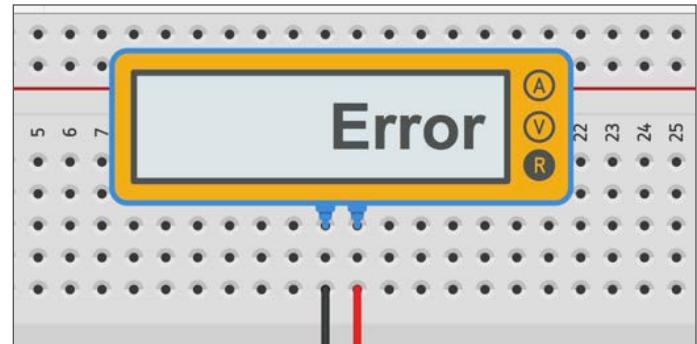


Click on the letter “V” on the meter. The meter indicates that there is 6 volts going through the circuit.



Click on the letter “R” on the meter. The meter displays an error message. We are getting this error because the connection to measure resistance across a resistor is different from the one we are currently using.

The error message doesn't mean much in this example because we know that the resistance value is 1 kilo-ohm. We know this because we set the value on the resistor configuration panel.



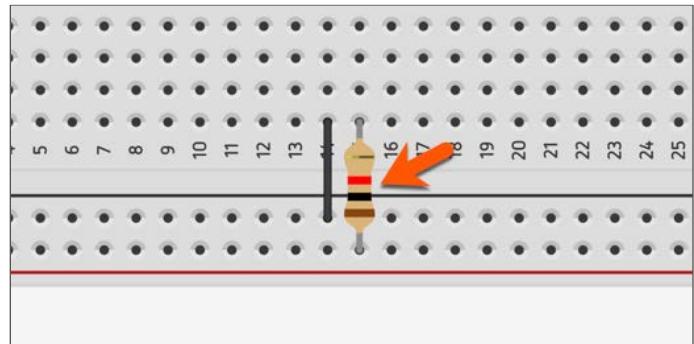
There are other ways of knowing the resistance of a resistor. Each resistor comes with a set of color bands. These color bands tell us the resistance value of the resistor. To know the value, we need to know what the color bands mean. In this example, we know that the colors brown, black, and red designate a resistor with 1 kilo-ohm.

There is another color that is as close to the other colors. This color is used to inform us of the tolerance of the resistor. Resistors are made out of a variety of materials. These materials include carbon and metal. The materials are not perfect and the amount of resistance in these materials can vary. A gold band informs us that the resistor has a tolerance of plus or minus 5 percent. This means that the resistance can be five-percent more or less than the value designated by the color bands.

This color band is not so important when we are working with TinkerCad because in a simulation the values are mathematically precise. In real world applications, this tolerance may be very important. It all depends on the circuit or component we are designing.

Teachable moment: Discuss the difference between models and real-world applications.

To understand the meaning of the colors of these bands we need to have access to a table that lists the hundreds of possible resistors. There are web pages that have downloadable tables. There are web pages that allow us to select the color bands and it provides the necessary information. There are also apps that can be used on mobile devices.



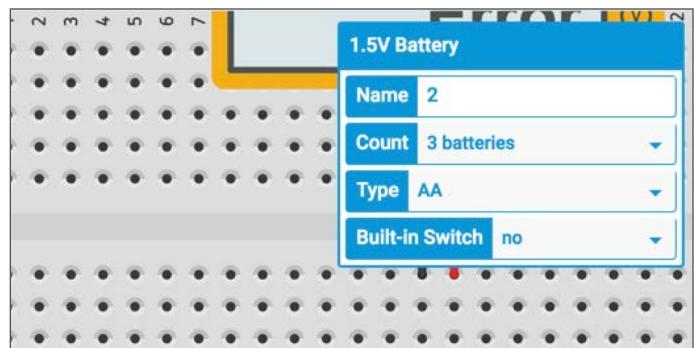
Another way to find the resistance value of the resistor is to use Ohms Law. To find the resistance we divide voltage by the current. The meter provides this information for us. The voltage is 6 and the current is 5.99 amps. Dividing 6 by 5.99 comes very close to 1, and 1 kilo-ohm is the value of our resistor.

$$R = V / I$$

$$R = 6 / 5.99$$

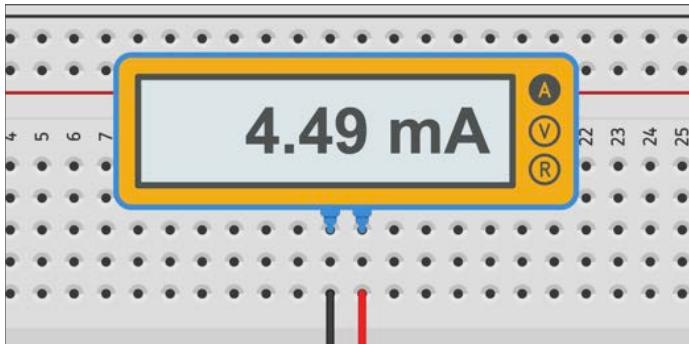
$$R = 1 \text{ kilo-ohm}$$

This process works even if we change the value of the voltage. Select the battery pack and change the number of batteries from 4 to 3.



Electric Circuits with Tinker CAD and STEM Integration

Click the letter “A” on the meter to read the value for the amount of current going through the meter. The image below shows we have a current of 4.49 mili-amps.

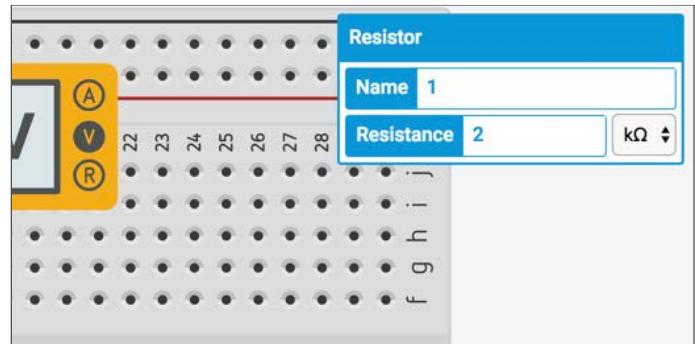


Click on the letter “V” on the meter to read the voltage. The voltage is 4.50. Dividing these two numbers will give us 1 kilo-ohm.

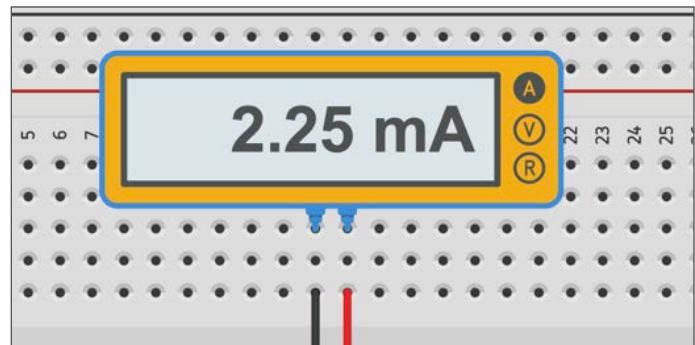
Teachable Moment: This example is very simple, try changing the value of the resistor to other values like 1.5 or 1.75 and verify these values with the meter and Ohm's Law.



The value of the resistor can be changed in the configuration panel. Click once on the resistor and change the value from 1 to 2 kilo-ohms.



Click on the letter “A” in the meter to measure the current. The current measures 2.25 milli-amps. Dividing the voltage, 4.5 volts, by 2.25 amps gives us 2 kilo-ohms resistance.



We will be using resistors in most of our circuits. In the next lesson, we will use the same circuit to power an LED with a resistor.

Save this project and call it Circuits and Resistors. We will return to this project in the future to test the current, voltage and resistance across other components.

Return to the TinkerCad main page to create a new project

Links to resources

[Link to online Resistor calculator](#)

[Link to online resistor color code calculator](#)

[Ohms Law calculator](#)

[Ampère's force law](#)

[André-Marie Ampère](#)

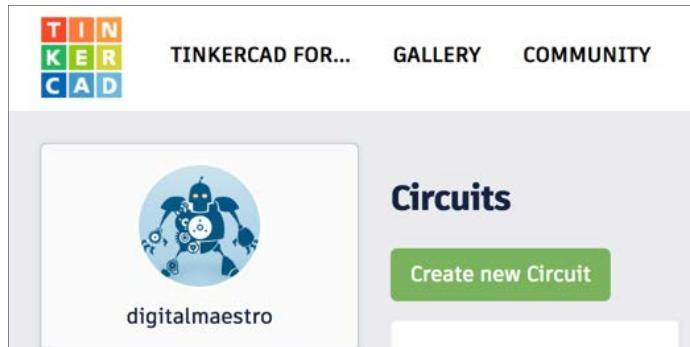
[Electric Current](#)

[Ohm's Law](#)

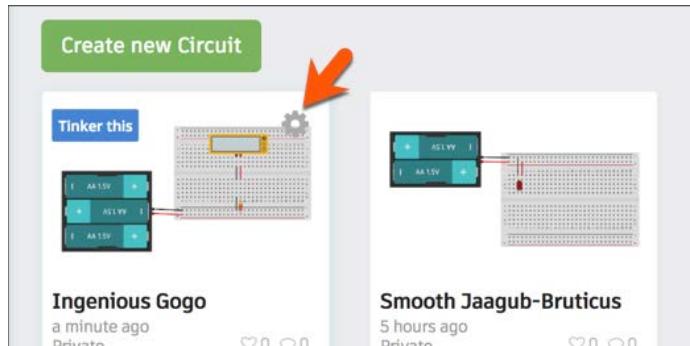
Project Properties

It is a good idea to keep track of our projects in TinkerCad. Throughout these lessons we will be creating a variety of projects and many of them will look very similar. The preview image in the main page doesn't always help. This is why it is important to provide a descriptive name to each project.

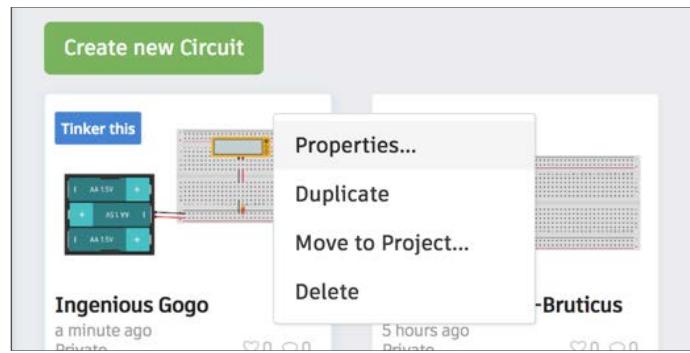
Go to the TinkerCad circuit main page by clicking the TinkerCad logo at the top of the page.



Move your mouse cursor over the last project we created. A button that reads "Tinker This" will appear. Clicking this button will open the project and we can begin working on the project. On the right corner of the preview box is a gear icon. Click once on this icon.

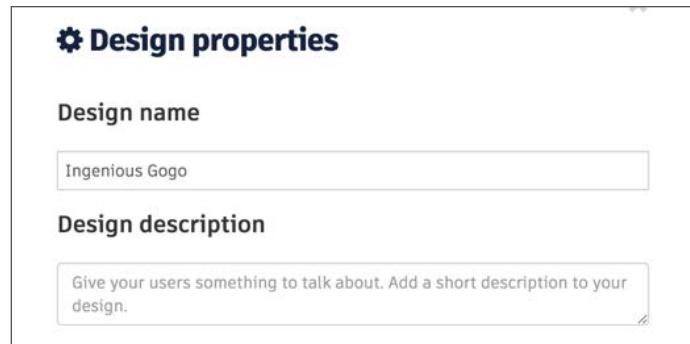


A menu will open with several options. The menu options include Properties, Duplicate, Move to Project..., and Delete. Click the Properties option.



A properties configuration box will open. In this box, we can change the name of the project. We can also provide a description for the design. Providing a description here is very important as designs get more elaborate. A good description can distinguish one of your designs from another.

Teacher Tip: Students should use the Design Description box to provide at least a one paragraph description of their circuit design. The description should include the purpose for the design.



In the properties box, we can choose to include Tags. Tags in TinkerCad are like categories. These categories can be useful when organizing and searching projects. The other options include privacy settings. Projects are private, meaning they can only be viewed by the creator unless made public. When they are public then anyone can view the project.

Electric Circuits with Tinker CAD and STEM Integration

The license option provides several license options that fall under the Creative Commons license. One includes the option to make the project part of the public domain.

Teacher Tip: This is a good opportunity to have students research and write their thoughts about copyright. You might have students write a paragraph about copyright, one on public domain, and one on Creative Commons. As an extension, you might have students debate the merits of each form of copyright. Have students write which license they will choose for their projects and why they made that choice. Think about why only these two licenses are available when sharing.

Tags (5 maximum)

Enter tag(s) here separated by commas. Press Enter to add a tag. ex. tag1,tag2

Privacy

private Not publicly listed, visible only to you

License

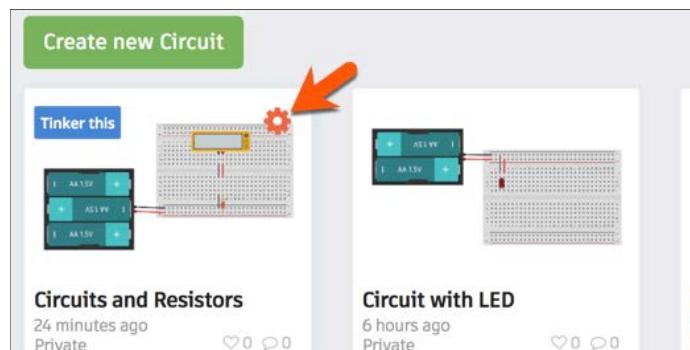
Public Domain

Click on the button to save changes to the project properties.

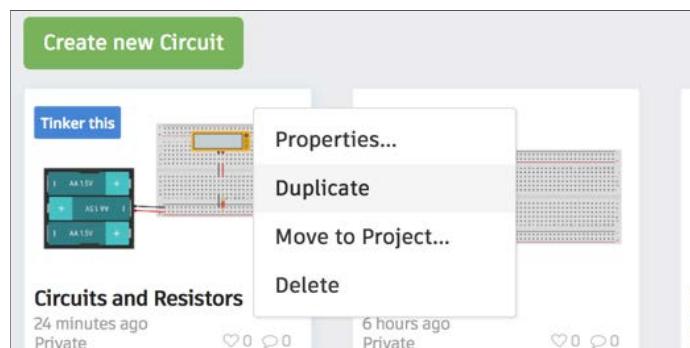
A Circuit with An LED and Resistor

We know that an LED will stop working and can be damaged if we apply too much voltage. In the previous chapter, we learned that resistors are used to limit the flow of current through a circuit. Resistors like these are often called “limiting resistors”. We will use the same circuit from the previous lesson to explore how a resistor works in an LED circuit to assure the LED doesn't get damaged.

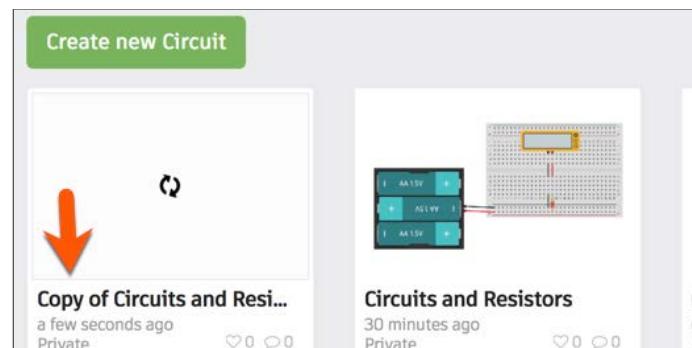
Hover your mouse over the preview of the previous project and click on the gear icon.



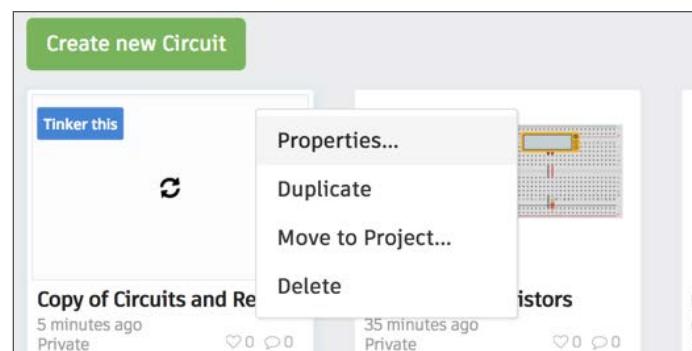
Select the option to duplicate the design.



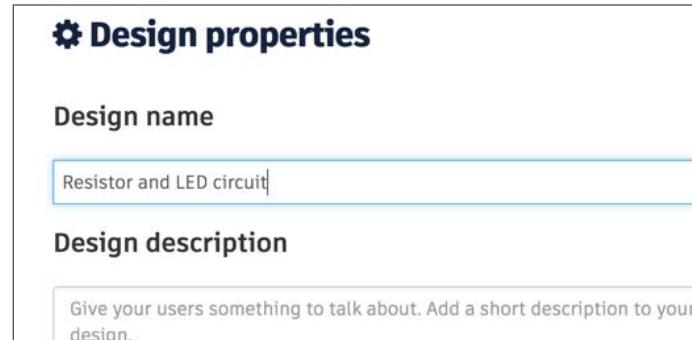
A duplicate of the project will be created and placed in the TinkerCad main page. The word “Copy” will be attached to the beginning of the project.



Move your mouse over the project and click the gear icon. Click the properties option when the menu appears.

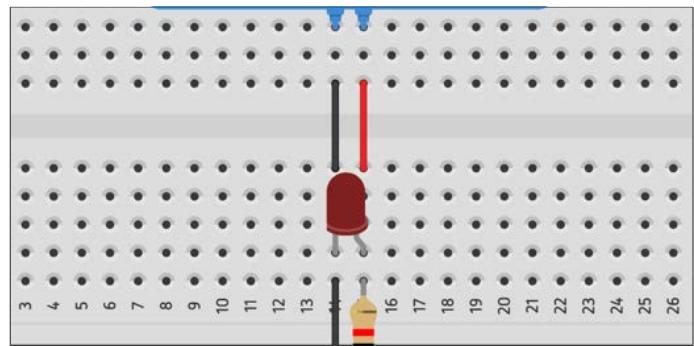
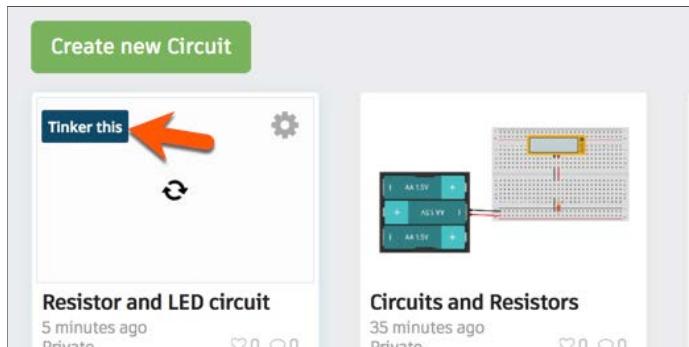


Name the design “Resistor and LED circuit”. Click the Save Changes button.

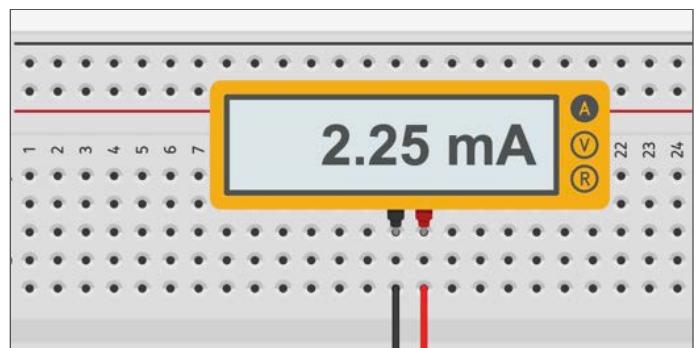


Electric Circuits with Tinker CAD and STEM Integration

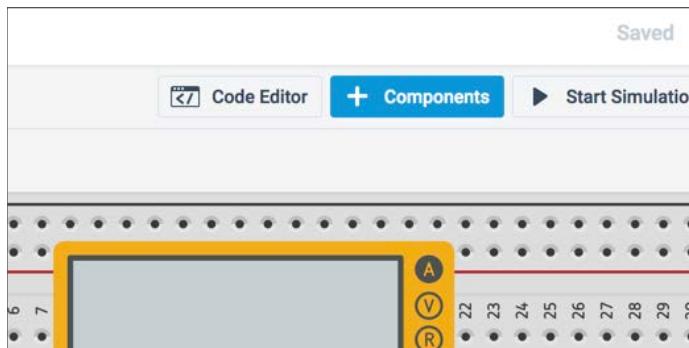
Move your mouse over the design preview box and click the button that reads “Tinker This”.



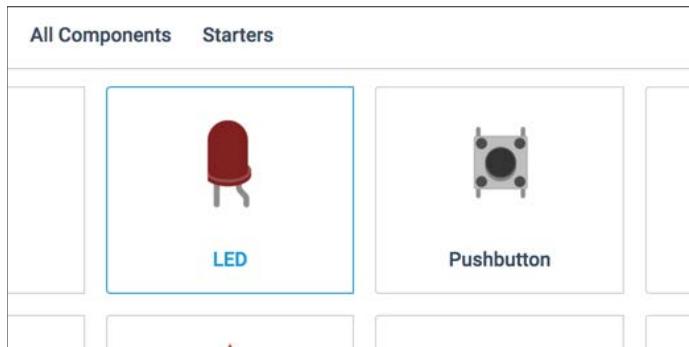
Click the Start Simulation button. The meter will read 2.25 mA(milliamps). This is the same current we measured in the last lesson with the same project. The project is using a 2 kilo-ohm resistor.



Click the Components button when the design opens.

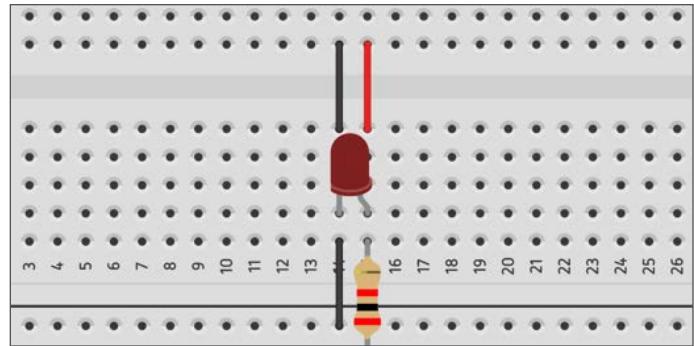


Find the LED component and click it once.

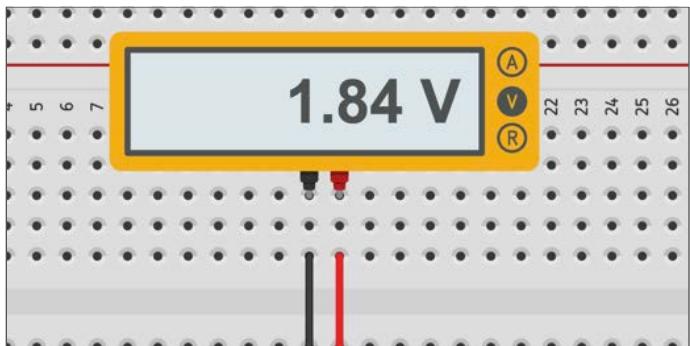


Place the LED above the resistor so the Anode of the LED is in the same row as the resistor. Remember that the Anode must be connected to the positive voltage. The Anode is the one with the bent wire.

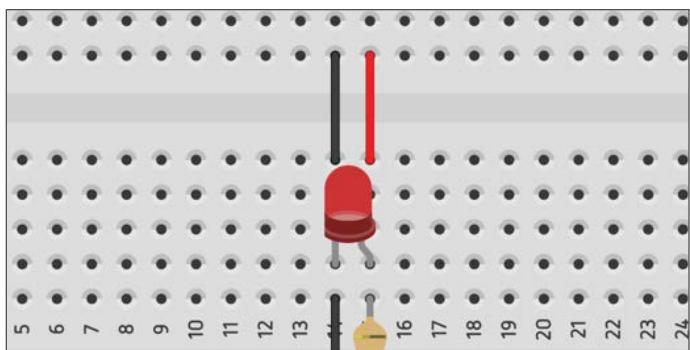
Take a look at the LED in the circuit. The LED isn't lit. We didn't make a mistake this time. This time we introduced a Multi-meter into the circuit. A multi-meter usually doesn't interfere with the voltage and current flowing through the circuit. In this simulator, the meter is shorting out the circuit and the LED doesn't light.



Click on the letter “V” in the meter to measure the voltage. The meter will read that the voltage going through the circuit is 1.88 volts. The circuit is powered by three 1.5 volt batteries in a battery pack. This provides a total of 4.5 volts to the circuit. The meter is telling us that we have dropped 2.62 volts across the circuit. To get this measurement in dropped voltage all we need to do is subtract the supplied voltage of 4.5 from the metered voltage of 1.84.



Take a look at the LED. The LED is lit because we opened a switch in the meter that allowed voltage to pass through.



The resistor prevented damage to the LED. In real applications, the use of resistors in an LED circuit can either shorten or lengthen their lives. The goal is to lengthen the life of any LED in a circuit while maintaining LED brightness. There is a way to figure out the optimal resistor to use for an LED. There is a formula to help us determine the resistor that will prevent damage to the LED. This formula will not give us the resistor that will provide the brightest LED display nor the dimmest, but it will give us a place to start and we can adjust the resistance from there.

Teachable Moment: Ask students how a voltage or current drop at the beginning of a circuit might affect other components in a circuit. This re-enforces the concept of cause and effect for younger students. It also re-enforces the understanding of how cause and effect are important in a closed system. This can be related to the plant cycle, life cycle and environmental cycle.

To determine the resistor, we need to know the voltage being applied to the LED and the tolerances for the LED. These tolerances are usually supplied when LEDs are purchased. TinkerCad does not provide this information in a direct way but we can get their tolerances from standards used in real LEDs. We know the voltage applied is 4.5 volts.

Calculating the resistor, we need requires the LED voltage tolerance and the current tolerance. Most LEDs have a voltage tolerance of 2.62 volts. This just so happens to be the voltage drop in our circuit. The forward voltage is how much voltage the LED needs to light. Voltage tolerances in LED information sheets usually provides a range of voltages known as forward voltage. We also need to know the forward current. This is the current tolerance for the LED, or how much current the LED can tolerate before it is damaged. This is also the manufacturer's forward current recommendation. Most LEDs have a forward current of 20mA to 30mA. The forward current for this LED is 20mA.

Teachable Moment: Ask students why we are using the terms forward voltage and forward current. Does this relate to the path taken by electricity through a circuit? Does this relate to the anode and cathode orientation of the LED?

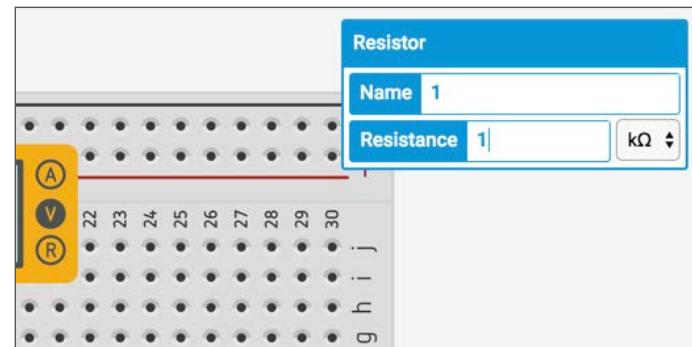
With this information, we can calculate the resistor value needed. To determine the resistor value, we subtract the voltage supply and the forward voltage requirement. This value is then divided by the forward current. The formula is shown in the image below.

$$R = \frac{V_s - V_f}{I_f}$$

In this example, we will subtract 2.62 from 4.5 volts to get 1.9 volts. This answer is then divided by 20mA to get ".095". This gives us the answer for a resistor of 95 Ohms. Multiply the answer by 1,000 to get the Ohms value. The resistor requirement is close to 1 Kilo-Ohm.

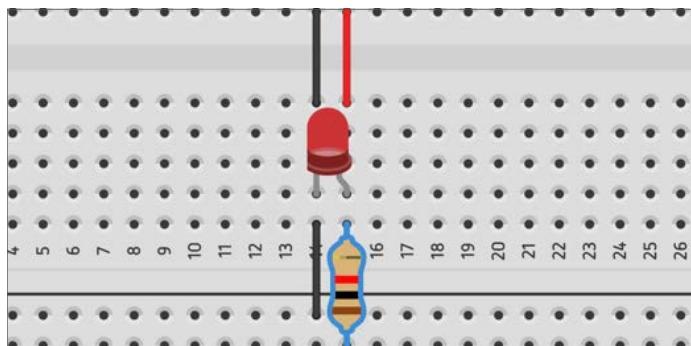
$$R = \frac{4.5v_s - 2.62v_f}{20mA_f}$$

Click once on the resistor and change the value from 2 to 1.



Electric Circuits with Tinker CAD and STEM Integration

The LED on the breadboard is lit without exceeding the voltage tolerance.



The image below was taken from a kit of LEDs I purchased. The LEDs come in a variety of colors and each has a voltage tolerance range. The yellow LED in the image has a voltage tolerance between 1.8 to 2.2 volts. The White LED has a tolerance between 2.8 and 3.2 volts. Different LEDs have different voltage tolerances. Another way to think of voltage tolerances is to think of these voltages as the minimum and maximum voltages required to light the LED properly. We should stay away from the maximum voltage so the LED lasts longer.

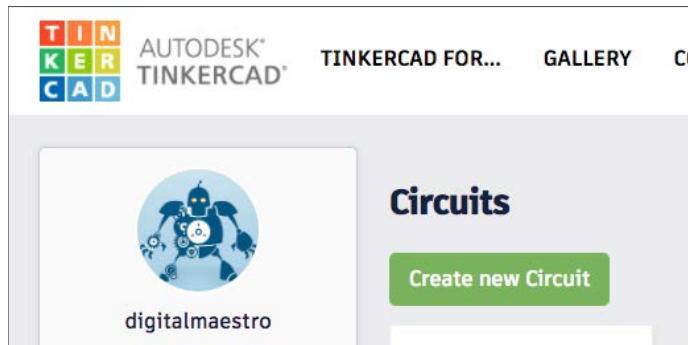
1.8-2.2V	2.8-3.2V	2.8-3.2V
Yellow 5mm 1.8-2.2V	White 5mm 2.8-3.2V	Blue 5mm 2.8-3.2V
		Blue

Circuits with A Potentiometer

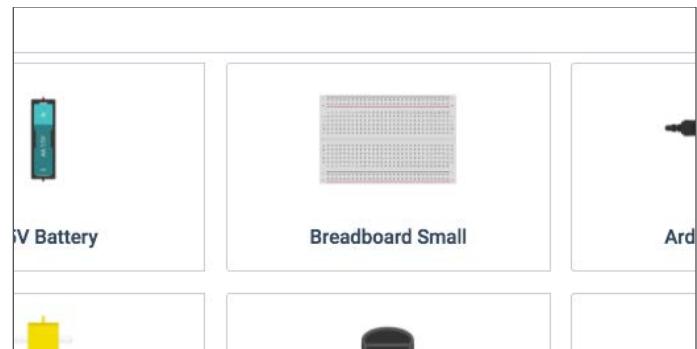
A potentiometer is a dial with three connectors. The dial is used to vary the amount of voltage that flows through the potentiometer and sent to a circuit or component. Potentiometers are used as volume controls on audio equipment and usually don't handle a lot of voltage.

Inside the potentiometer is a resistive material where one end of the circuit connects. The amount of resistance applied to the current changes as the dial is turned. We will use a potentiometer to control the amount of voltage applied to an LED. We will use the potentiometer instead of a resistor, but we can also use it with a resistor to prevent the LED from being damaged if the potentiometer cannot limit enough of the voltage being supplied to the LED.

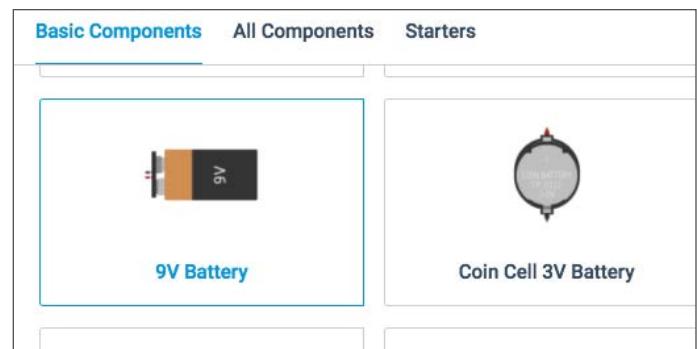
Click the TinkerCad icon to return to the main page if you are not already there. Create a new project by clicking the “Create New Circuit” button.



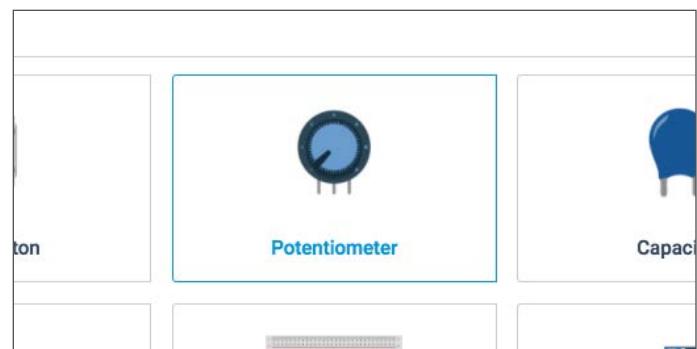
Place a small breadboard onto the work area.



Find the 9 Volt battery in the components panel and place it to the right of the breadboard.

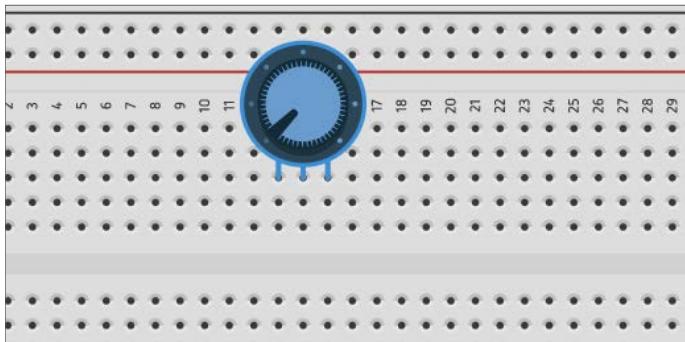


Find the potentiometer in the components panel and place it on the top half of the breadboard.

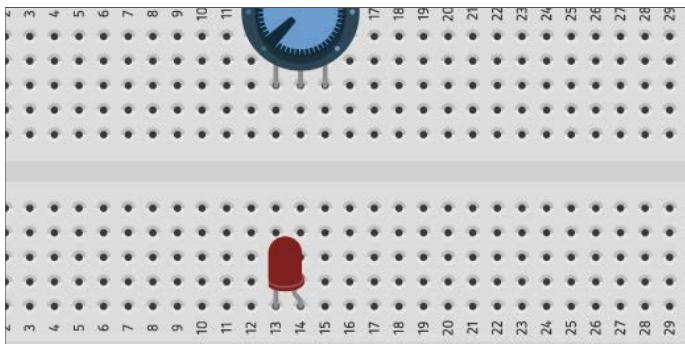


Electric Circuits with Tinker CAD and STEM Integration

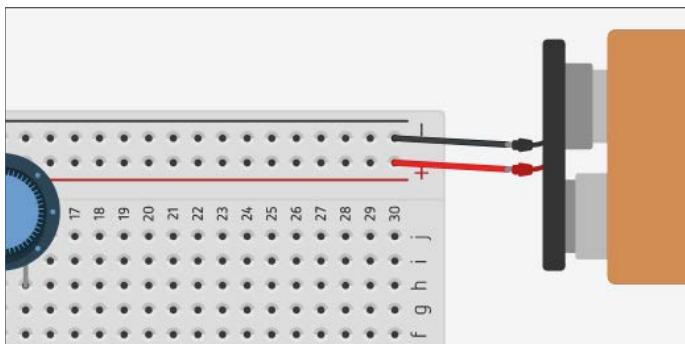
Make sure to leave one or two columns below the potentiometer for jumper wires.



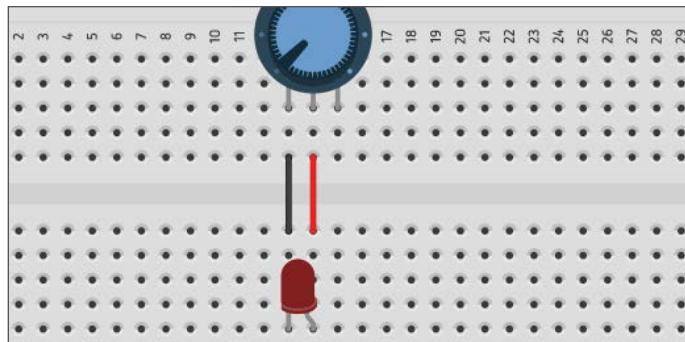
Get an LED and place it in the lower half of the breadboard. Align the Anode of the LED with the center connector of the potentiometer. This will make things easier when connecting jumper wires.



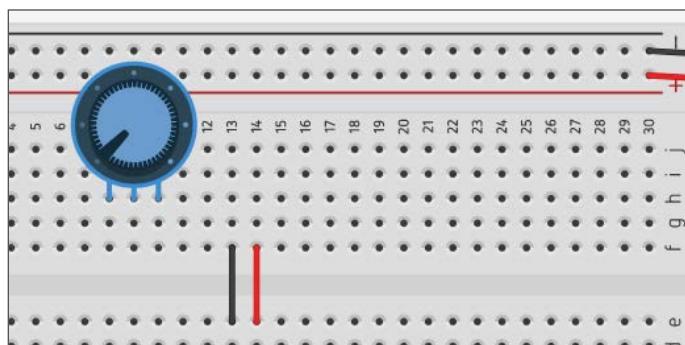
Find the 9-volt battery in the components panel and place the battery near the upper half of the breadboard and connect the battery terminals to the breadboard positive and negative rails.



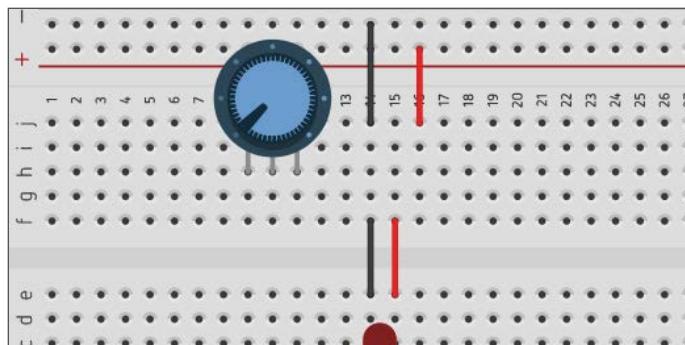
Connect jumper wires from the center and left potentiometer to the LED. The center connector on the potentiometer will be used for the positive flow.



The potentiometer is large and connecting the jumper wires from the railing to the potentiometer rows is difficult. Move the potentiometer module to the left or right so the connector rows are visible.

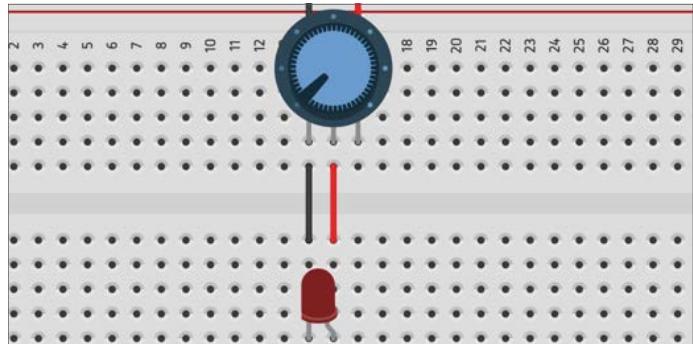


Connect jumper wires from the negative and positive rails to the matching wires that connect to the left and right ends of the potentiometer. Move the potentiometer back into place.

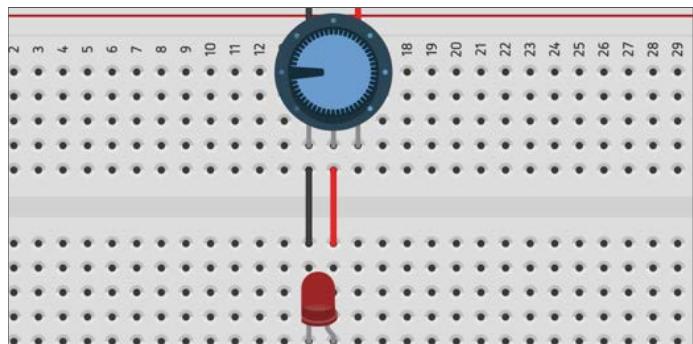


Electric Circuits with Tinker CAD and STEM Integration

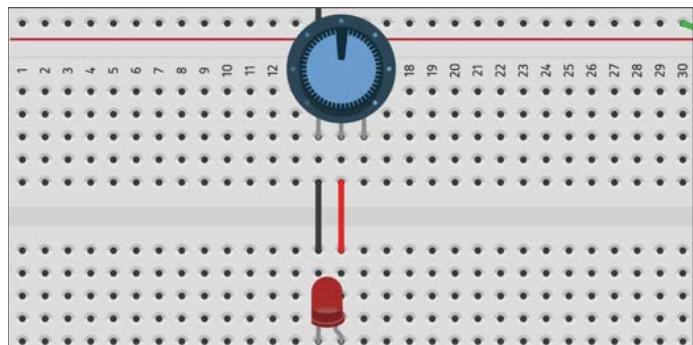
Click the “Start Simulation” button. The dial on the potentiometer is all the way to the left and no voltage is being sent to the LED. Move the dial slowly to the right until the LED lights.



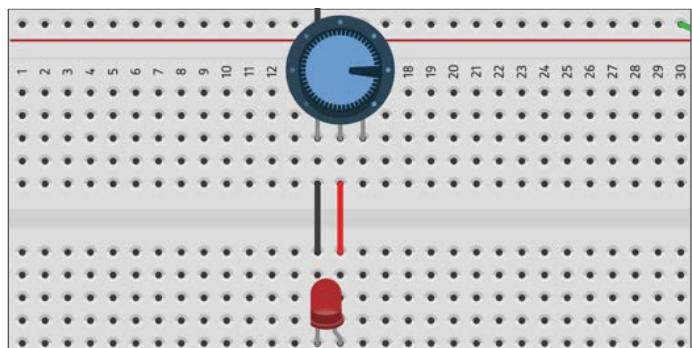
The LED will light a little after the dial passes the first marker on the dial.



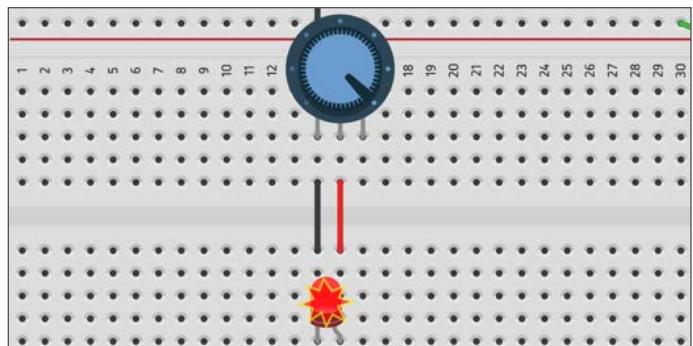
Turn the dial half way and the LED might look a little brighter.



Turn the dial until it reaches the second to the last mark on the dial. It might look a little brighter.

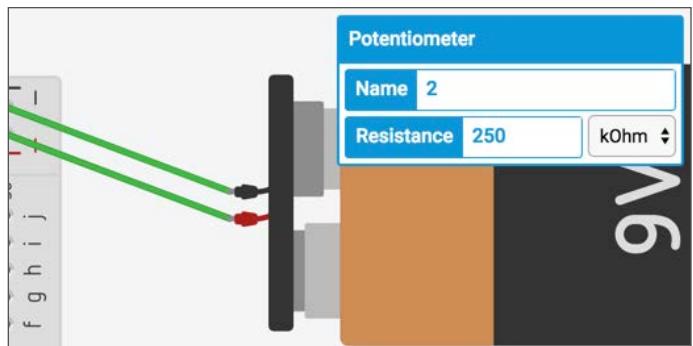


Turn the dial all the way. This time we send too much voltage and the LED has been damaged.



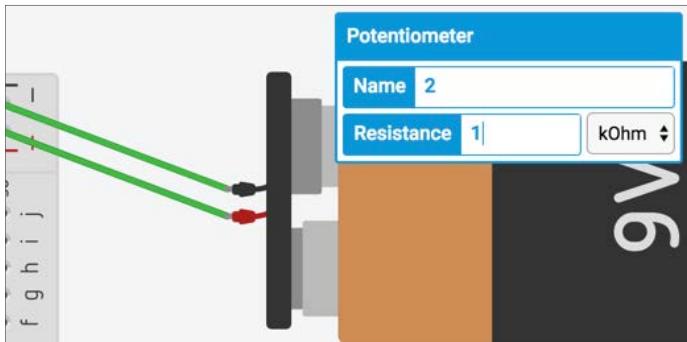
Potentiometer Resistance Value

The potentiometer has a level of resistance like the resistor we used in an earlier lesson. Stop the simulation and click once on the potentiometer. A configuration box for the potentiometer will open. The maximum resistance for the potentiometer is 250 kilo-ohms. That is a lot. We used a 1 kilo-ohm resistor with the push button and LED.

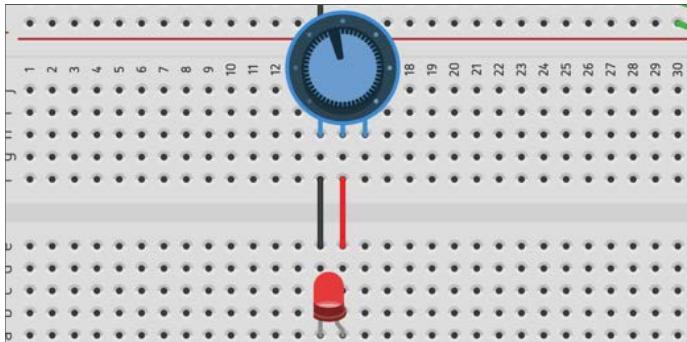


Electric Circuits with Tinker CAD and STEM Integration

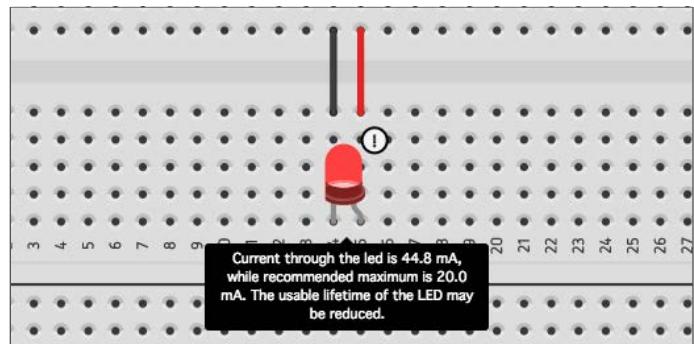
Change the resistance value to 1 kilo-ohms and start the simulation again.



Turn the dial gradually to the left and watch as the light dims.

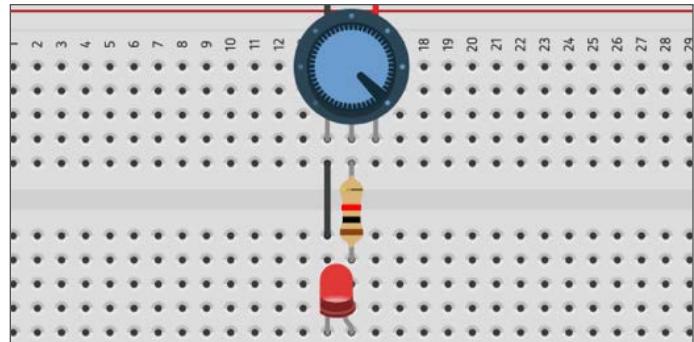


Turn the dial back to the right until an exclamation mark appears to the right of the LED. Move the mouse pointer over the LED. An information box will display the warming information related the LED. The message is informing us that the current through the LED is currently 44.8 mA (milliamps) and that the maximum recommended current is 20 mA. The maximum recommended current for the LED is useful when we are looking for a resistor to limit the amount of voltage send to the LED. We figured this out in a previous lesson.



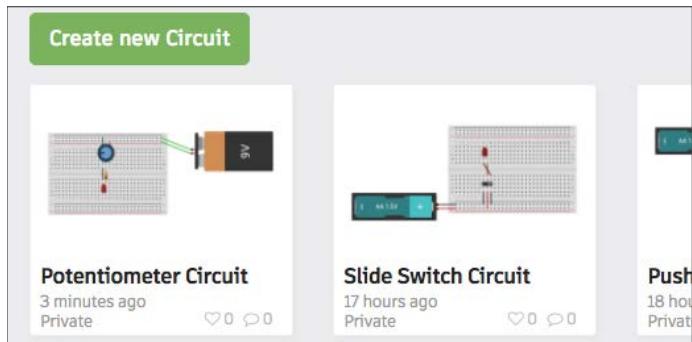
If we turn the dial all the way to the right we still run the risk of damaging the LED. We could raise the resistance of the potentiometer but this will reduce its efficiency and still not prevent the LED from being damaged. It is always a good idea to place a resistor between a power source and LED to prevent damage.

Stop the simulation and replace the jumper wire for the LED Anode with a 1 kilo-ohm resistor. Start the simulation again and turn the dial all the way to the right. The LED will gradually illuminate and it will not be damaged when the dial is all the way to the right. Placing a resistor to limit the amount of voltage being applied to an LED is known as a voltage limiting resistor.



Electric Circuits with Tinker CAD and STEM Integration

We are done with this project. Return to the main page and use the properties box to give this project a name. Call it Potentiometer Circuit.



Student Activities

1. Ask students to explain why a variable resistor might be better than a regular resistor.
2. Ask students to explain why we need to add a voltage limiting resistor to the circuit.
3. Ask students to think of various examples of potentiometers.

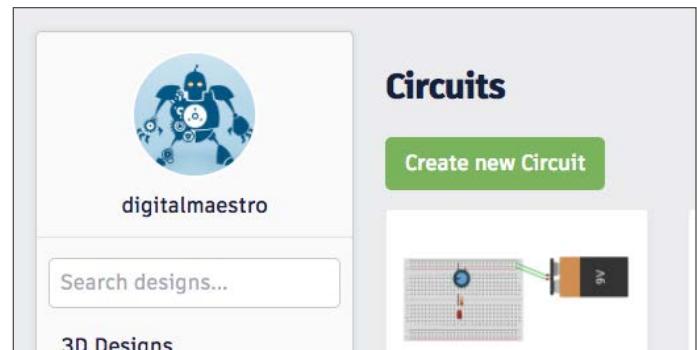
Circuits with Capacitors

A capacitor is a component that stores electrostatic energy. This is a form of potential energy. Capacitors contain two electrical conductors and are separated by an insulator. The conductors are thin films of metal. The insulators can be glass, ceramic, plastic, or air. When a capacitor is connected to a battery positive charges collect on one plate and negative charges collect on the other plate. When the capacitor is filled with an electric charge it will not allow current to flow through the capacitor. The electric charge can be released by the capacitor into a circuit. Capacitors store electrical energy as potential energy and release it into a circuit as voltage and current.

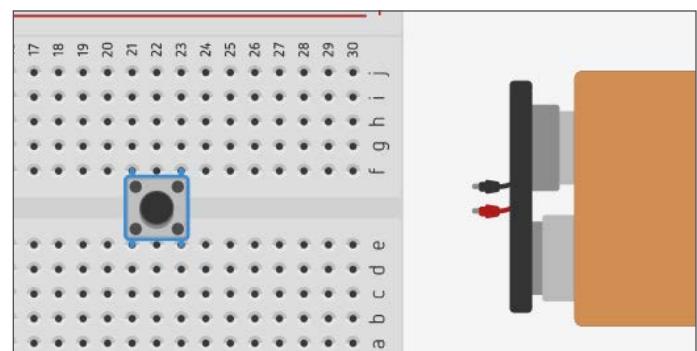
Teachable Moment: Take a look at the various materials used as insulators and think about which material might provide the best resistance. What makes a good resistor? Most capacitors use air for the capacitor because air is a much better resistor. Why is air a much better resistor? Think about the properties of air and the other resistor materials.

In this circuit, we will create a circuit that uses a capacitor to store the energy from a battery and then discharge that energy into an LED. To do this we will need two buttons to control the flow of voltage to the capacitor and to the LED. One button will allow us to send electricity to the capacitor while the other will prevent the electricity from going to the component until we are ready.

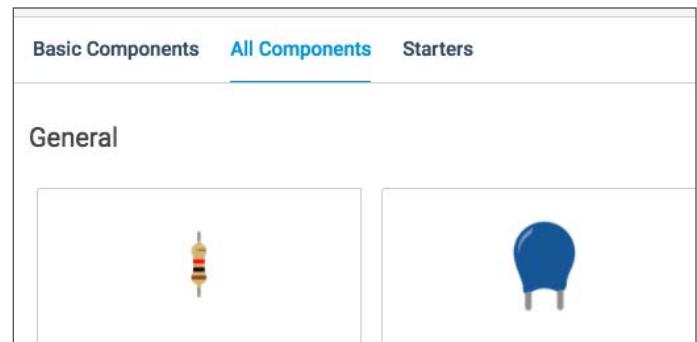
Create a new circuit project.



Place a small breadboard onto the work area. Place a 9-volt battery on the right side of the breadboard and place a push button switch to connect the top half of the breadboard with the lower half.



Click the All Components section to search for the capacitor. The capacitor we will use in this circuit is not part of the basic components.

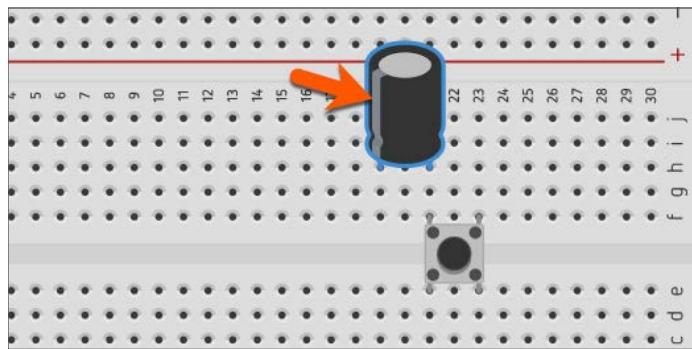


Electric Circuits with Tinker CAD and STEM Integration

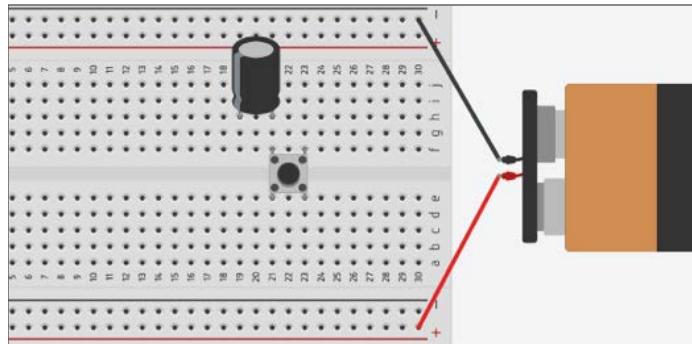
Look for the Polarized Capacitor.



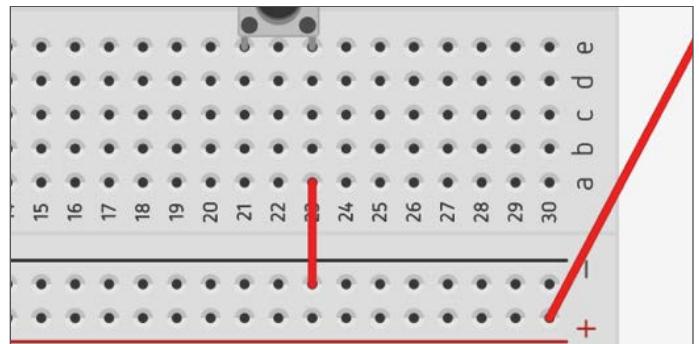
Place the Polarized capacitor in the top half of the breadboard. Leave three columns to connect jumper wires. This capacitor is polarized, so that means that it has a positive and a negative terminal. The negative terminal is the one that has a strip above the terminal wire. Align the positive side of the capacitor to the left side of the switch.



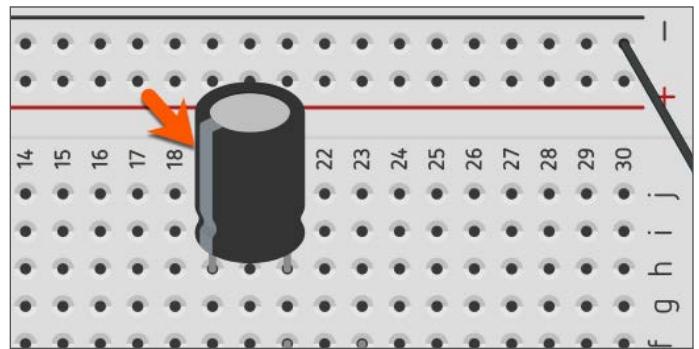
Connect the negative part of the battery to the top of the breadboard and the positive part of the battery to the bottom of the breadboard rails.



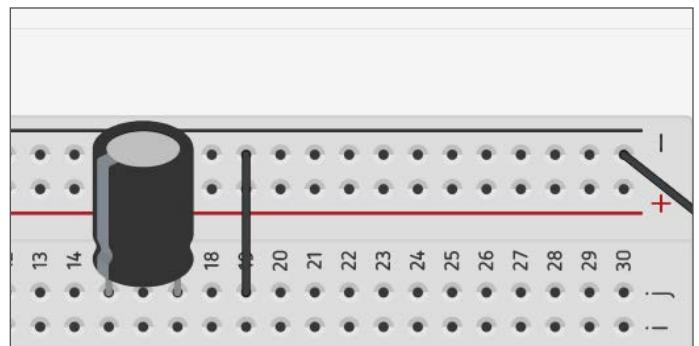
Connect a jumper wire from the positive rail to the push button. Remember, the opposite end of the switch is used to connect circuits.



We need to connect the negative connector of the capacitor to the negative rail but it is too large to see the connector row.

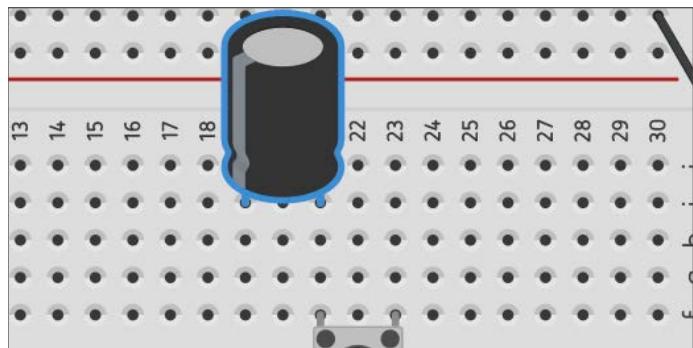


Move the capacitor to the left or right and connect a jumper wire to the row where the negative connector for the capacitor would be.

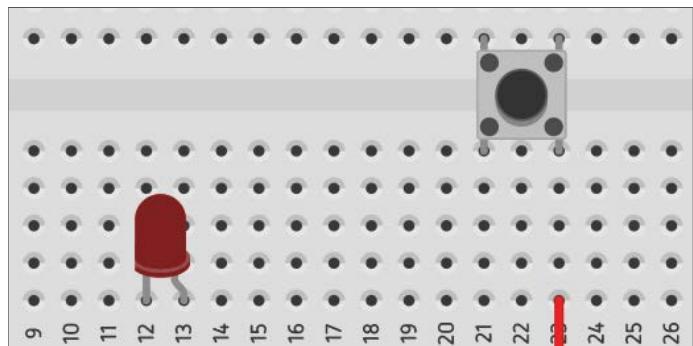


Move the capacitor back into place. Make sure to leave two columns between the switch and the capacitor for jumper wires.

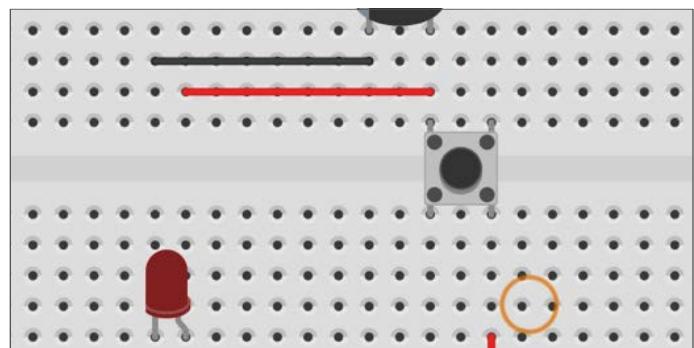
The capacitor is not connected into a circuit. Pressing the button will send current to the capacitor. The capacitor will collect this current until it is full. It is possible to overfill a capacitor and that would damage the capacitor and could be dangerous. Capacitors store lots of energy like a battery.



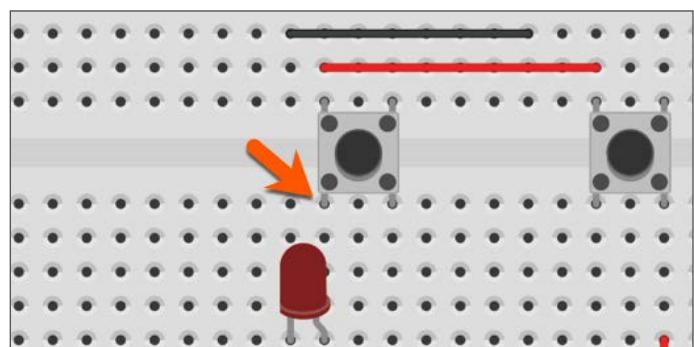
To discharge the capacitor, we need a component. We will use the trusty LED to help us see the capacitor's discharge. Place an LED in the lower half of the breadboard. You might need to go back to the Basic Components section to get the LED.



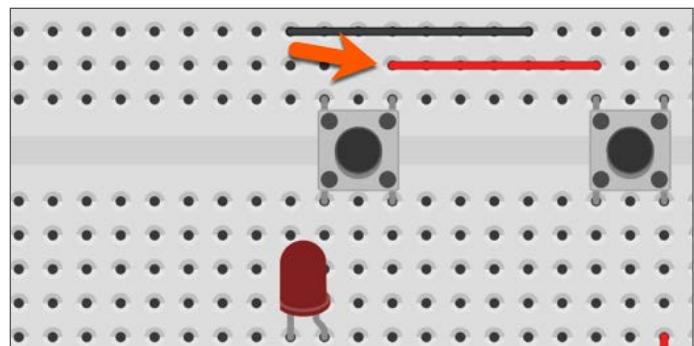
Connect jumper wires to other rows on the left side of the capacitor. We will use these rows to connect to the LED on the lower half of the breadboard with more jumper wires. Align the wires to the Anode and Cathode rows on the LED.



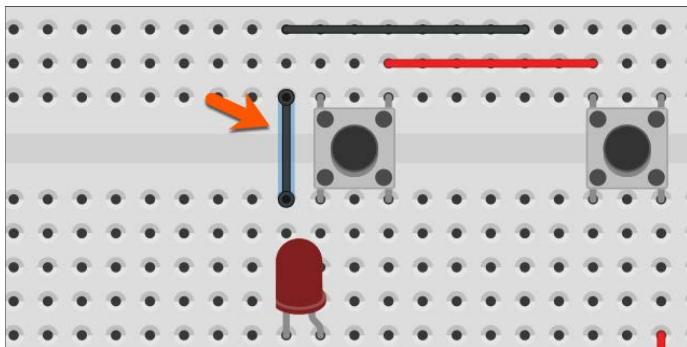
Go to the components panel and find another push button switch. Place the switch so that the left side of the switch aligns with the LED anode.



Shorten the positive jumper wire so it connects with the first push button connector.



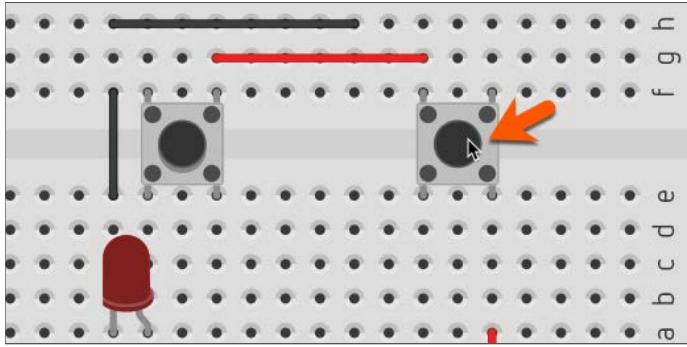
Use a jumper wire to connect the negative voltage from the capacitor to the cathode of the LED.



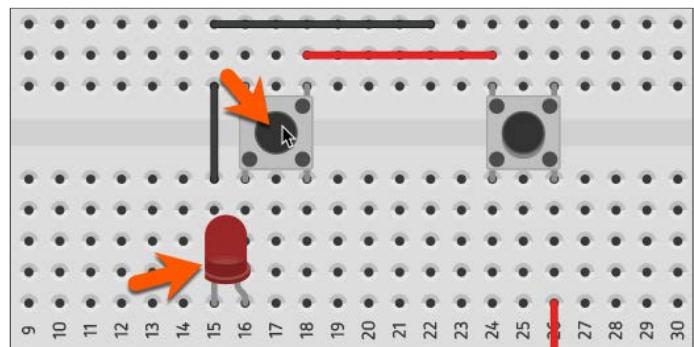
The circuit is complete. Start the simulation then click the button on the right. This part of the circuit sends electricity to the capacitor. The other button is not pressed so that part of the circuit is open and does not allow electricity to flow.

Charging and Discharging the Capacitor

Hold the button down for a few seconds. Two to three seconds would be good. This simulates voltage being passed to the capacitor. We don't actually see the capacitor charging. Capacitors charge very quickly so in a real-world capacitor the capacitor would be charged as soon as the button was pressed. The capacitor is now charged and will hold the electrostatic charge. Release the button to open the circuit to the capacitor.

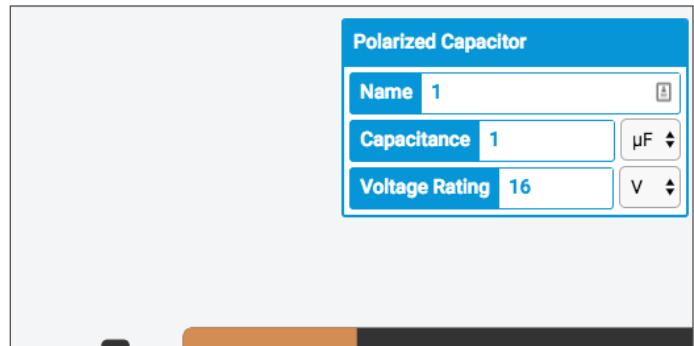


Press the other button and you will see the LED light for a few seconds and then dim away. The LED dims away because the electrostatic discharge from the capacitor drains away. If we could maintain the button pressed on the right side then the LED would remain lit. The capacitor will receive a continuous charge which it would then release to the LED. Capacitors are used to store large amounts of electricity to feed the needs of large components.



Capacitors are used with devices like amplifiers so that they can supply large voltage when needed. They are used to start equipment that requires large amounts of current.

Stop the simulation and click on the capacitor. Look at the capacitor information panel. The capacitor is rated to store 16 volts. Different capacitors are constructed to handle different voltages.

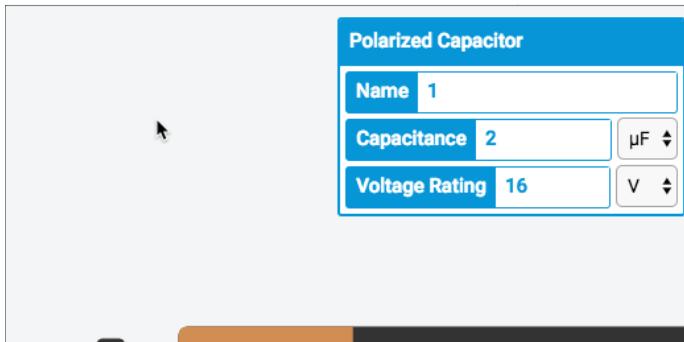


Electric Circuits with Tinker CAD and STEM Integration

Capacitors have a capacitance. Without getting very technical, this is amount of charge the capacitor can hold. The amount of capacitance is measured in Farads in honor of Michael Faraday. We won't go into the complexities related to calculating the capacitance of a capacitor. This information is provided on capacitors or on an information sheet when capacitors are purchased.

To demonstrate capacitance, change the capacitance to 2 in the capacitor configuration panel. The "u" in front of the letter "F" represents micro. A micro represents one millionth. This capacitor has a capacitance of 2 micro-farads or 2 millionths of a farad. This is a very tiny amount but more than enough for our circuit.

Start the simulation and charge the capacitor. Press the button to release the charge into the LED. Note that the LED remains lit longer. The LED will remain lit for about five seconds.



The image below is an example of an information sheet that comes when capacitors are ordered. The capacitor outlined in red has a capacitance of 46 micro-farads with a tolerance of 10 volts.

0.47uF 50V 5x11 20pcs	1uF 50V 5x11 20pcs	2.2uF 50V 4x7 30pcs	3.3uF 50V 4x7 30pcs	4.7uF 50V 4x7 30pcs	10uF 25 4x7 30pcs
22uF 25V 4x7 30pcs	33uF 16V 4x7 30pcs	47uF 10V 4x7 30pcs	47uF 25V 5x11 20pcs	47uF 50V 6x11 15pcs	100uF 16V 5x11 20pcs
220uF 25V 6x11	330uF 25V 8x12	470uF 10V 6x11	470uF 16V 8x12	680uF 16V 8x12	1000uF 16V 10x16

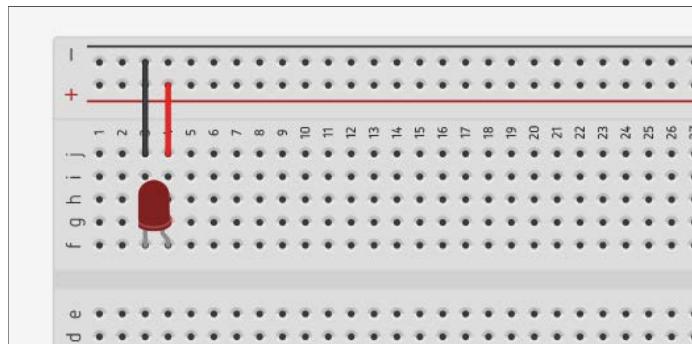
Student Activity:

1. Ask students to explain what is going on when the LED takes longer to turn OFF.
2. Have the students create a table with the headings Capacitance and Time. List the capacitance from 1 to 10 down the column. Have students change the capacitance to each value and record the time it took for the LED to turn off.
3. Have the students look at the data collected and ask them to determine if the time increase is consistent.
4. Is there a relationship between the capacitance and the time it takes for the capacitor to discharge and the LED to turn off?
5. Ask students to compare and contrast a capacitor and a battery.
6. When would a capacitor be more useful than a battery?
7. Ask the students to change the voltage rating from 16 to 3 and charge the capacitor. What happened? Why did it happen? Why didn't this happen when the voltage rating was at 16?

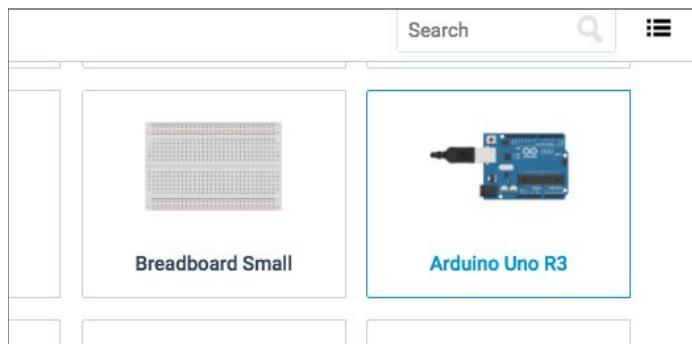
Circuits with Arduino

The Arduino board is a micro controller. A micro controller is a very simple computer that accepts basic code and translates that code into instructions that interact with the physical world. In this lesson, we will use a micro controller to take the place of a switch. The micro controller will do more than take the place of the switch. We will use the micro controller to augment what can be done with the circuit and a push button.

Create a new circuit and place a breadboard onto the canvas. Place one LED on the Breadboard and place two jump wires. Make sure to connect the anode wire to the positive lead and the cathode wire to the negative lead.



Click the Components button and find the Arduino Uno R3 micro controller. You will find it next to the small breadboard.



Place the micro controller to the left of the Breadboard. There are several components that are part of the Arduino board. Let's look at a couple of these components.



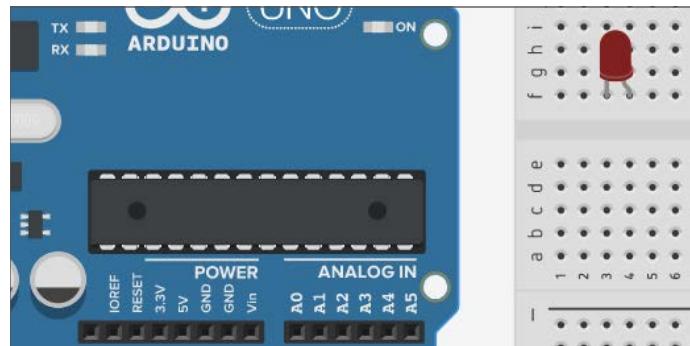
The holes along both sides of the board are called GPIO pins. This stands for General Purpose Input/Output. Each is a connector that can be linked to our Breadboard with a jumper wire. Most of these are marked with a number. These numbers are used to identify the pins in the code we will write to control the LED. Code we develop on the board can reference these pins as either input or output. There is one connector labeled GND. This is the ground connector or the negative terminal in our circuit. The GND is the same as the negative terminal on a battery. The other connectors marked with a number are the same as the positive terminal on a battery. An Arduino is different from a battery in that it does not produce electricity but passes along electricity from a power source. The Arduino can only output between 3 and 5 volts to a circuit. If we need to power a component that needs more than this voltage we will need to use a transistor or capacitor.

Electric Circuits with Tinker CAD and STEM Integration

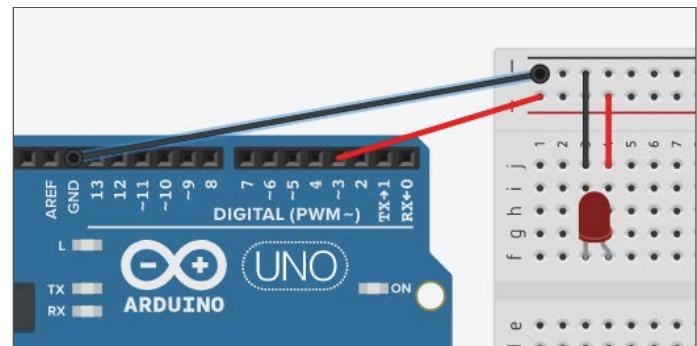
A physical Arduino board is connected to a five-volt power supply from a computer USB port or from a battery pack. The Arduino itself can supply the same 5 volts to our components. For some components, this is too many volts or more accurately too much current. In our example, the 5 volts will destroy the LED on the Breadboard. We will be adding a resistor to limit the amount of current going to the LED.



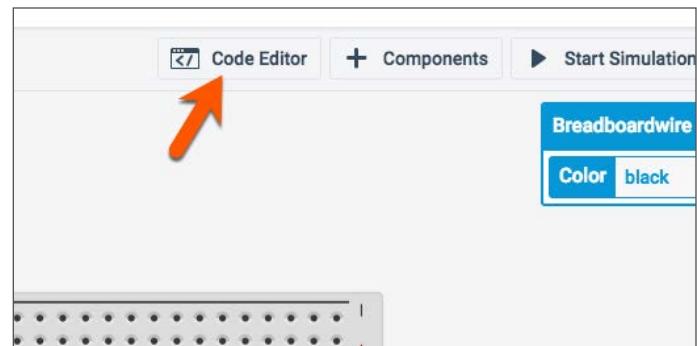
The solid rectangle on the board with connectors coming out of the sides is the actual micro controller. This is the component that handles all of our instructions and sends them to other components.



Connect a jumper wire from the GND connector to the negative column. Take another jumper wire and connect it from the number 3 connector to the positive column. There is nothing special about the number 3 in this example. We could have selected any of the other numbered connector on the board. I moved the board slightly so you could see the connections. I also color coded the wires.

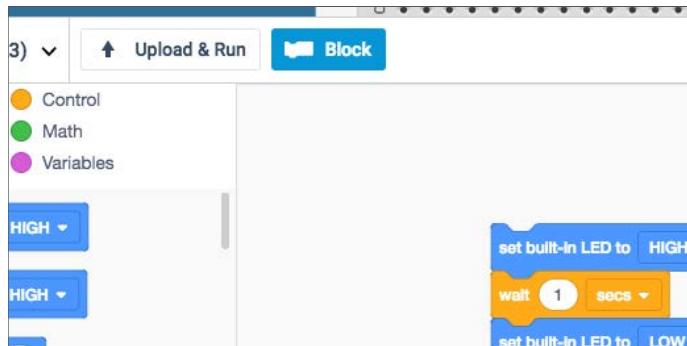


This isn't enough to turn on the LED. We need to do a few more things. Click on the Code Editor button.



Electric Circuits with Tinker CAD and STEM Integration

A coding panel will open at the bottom of the page. This coding panel uses puzzle blocks to develop code. This is very similar to blocks used in Code.org or Scratch. This the standard block of code included each time we place an Arduino board onto the work space. This block of code will cause the LED on the Arduino board to blink. But this is not the LED on our Breadboard. The Arduino has a small LED on the board itself. Click the Start Simulation button to see the LED blink on the board.



The blinking LED can be seen on the left side of the Arduino logo. Stop the simulation.



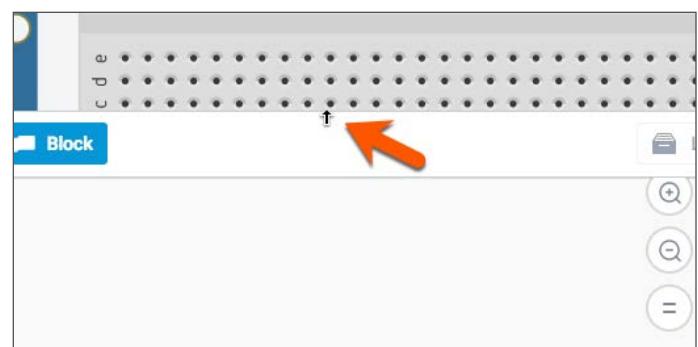
We don't need this code because we want to use the LED located on the Breadboard. Click and drag the code onto the trash can icon. Click the first code block and drag it to the trash can icon.

```
void setup()
{
    pinMode(13, C);
}

void loop()
{
    digitalWrite(HIGH);
    delay(1000);
    digitalWrite(LOW);
    delay(1000);
}
```

A screenshot of the Tinker CAD coding panel. It shows a portion of the code: 'void setup()' followed by 'pinMode(13, C);'. Below it is 'void loop()' followed by 'digitalWrite(HIGH);', 'delay(1000);', 'digitalWrite(LOW);', and 'delay(1000);'. A blue block labeled 'set built-in LED to HIGH' is highlighted with a yellow border.

We need a little more room to code. Move your mouse pointer to the top edge of the coding panel until you see the arrow change. Click and drag up to expand the coding panel.



The coding panel has different sections of code. We will be using code blocks in the Output section. Drag the set pin code onto the coding canvas.

```
set built-in LED to HIGH
set pin 0 to HIGH
set pin 3 to 0
rotate servo on pin 0 to 0 degrees
```

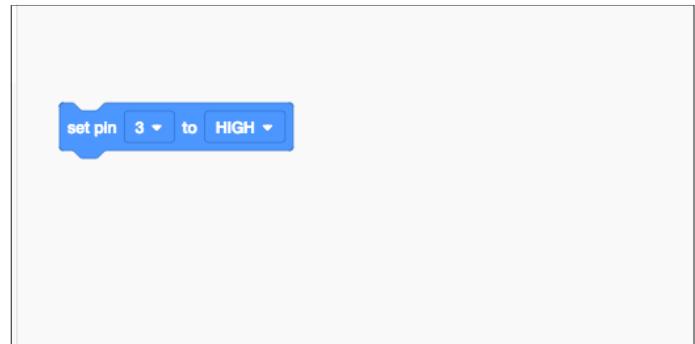
A screenshot of the Tinker CAD coding panel showing four code blocks: 'set built-in LED to HIGH', 'set pin 0 to HIGH', 'set pin 3 to 0', and 'rotate servo on pin 0 to 0 degrees'. The first block is highlighted with a yellow border.

Most code blocks have options that can be changed. This code block includes a pin number and a state. The PIN number references the connector we used to send current to the Breadboard. This is the positive jumper wire we connected earlier. We connected the wire to pin 3. The options in the code block are called arguments. The term argument comes from mathematics. The argument of a function is a specific input to the function. It is an independent variable like the pin in our code block. This code block has two arguments.



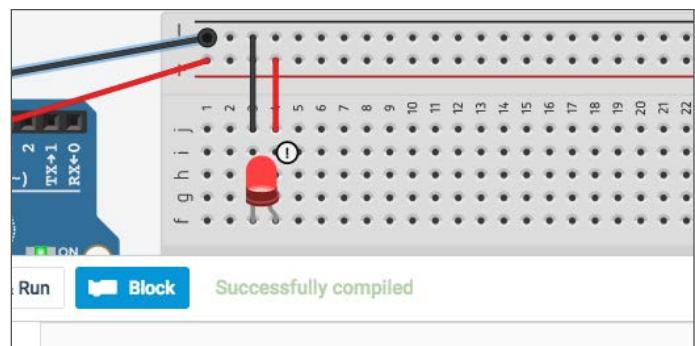
Change the pin argument to 3. The second argument has two states. A state has one of two options. A state can be on or off. In this code block, the state is set to high. A high state is the same as an ON state. The other option is a low state. This is the same as OFF. Computers read everything as either ON or OFF or ones and zeros. Binary numbers like one and zero are at the heart of all computers.

That's all we need to get started. Click the Start Simulation button. Resize the code block panel to see the LED.



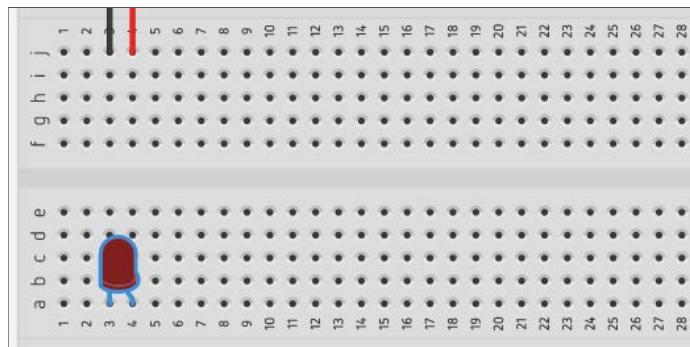
The LED will change color to indicate that it is on. There is an exclamation mark next to the LED. This exclamation mark is a warning. The current going through the LED is too high. In the simulation, we get a warning. In a physical board with a real LED the LED might be damaged and will not work again. This is why testing or prototyping on digital breadboards can be very useful. LEDs aren't expensive but expensive enough that you don't want to be burning them out regularly.

To avoid burning out LEDs we need to use resistors. Resistors restrict the flow of current to components. Every circuit includes voltage, resistance, and current. Current is the part of the equation that does all the work. Think of current like water flowing through a river or stream. Resistance is the width of the river or stream. Narrow streams have more resistance than wide streams.

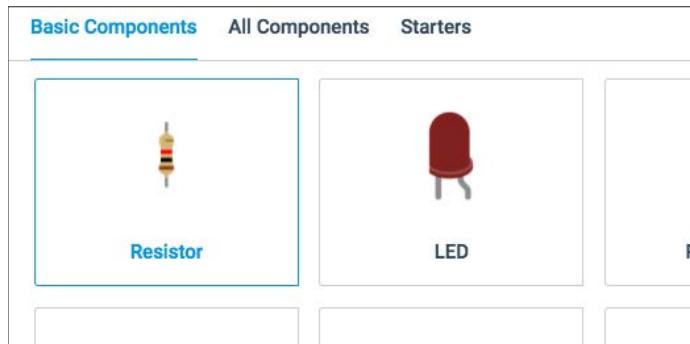


Electric Circuits with Tinker CAD and STEM Integration

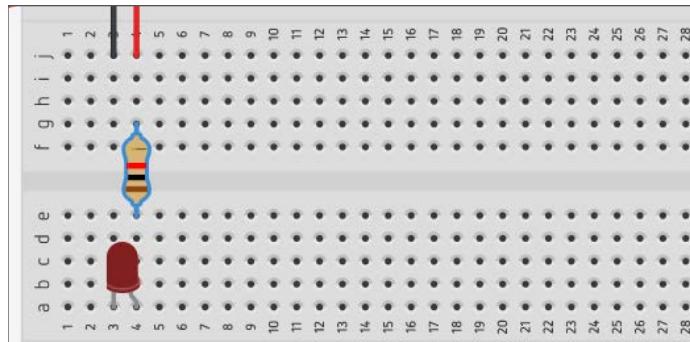
Stop the simulation and close the code editor. We need to make room of the resistor. Move the LED to the other side of the board and place it in column A. We need to leave room for the jumper wire and to see the connection in our illustration.



Open the components panel and find the resistor.

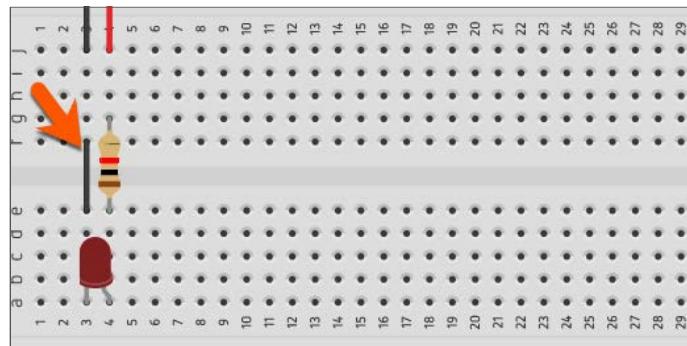


Place the resistor so it bridges the gap between the two halves of the board. Make sure the resistor is in the same row as the anode and the positive jumper wire.

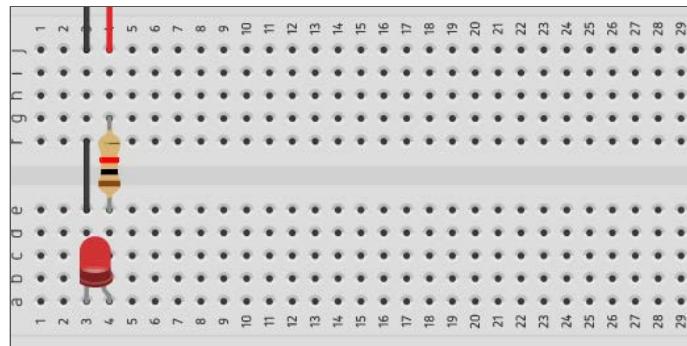


You might have already figured it out but I want to test your understanding anyway. Run the simulation. Why didn't the LED turn on? Hint, is there a complete circuit?

The row connections do not span across the board between E and F. We can either place a jumper wire from F to E or place a jumper wire directly from the negative terminal column to row E. I added another jumper wire to complete the circuit.



Run the simulation and the LED should light without a warning.

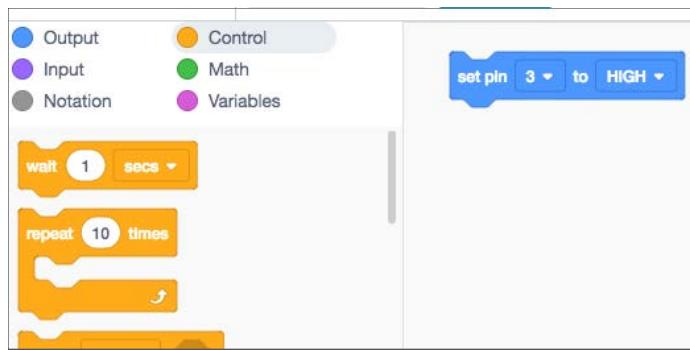


Ok Time for Another Troubleshooting Problem

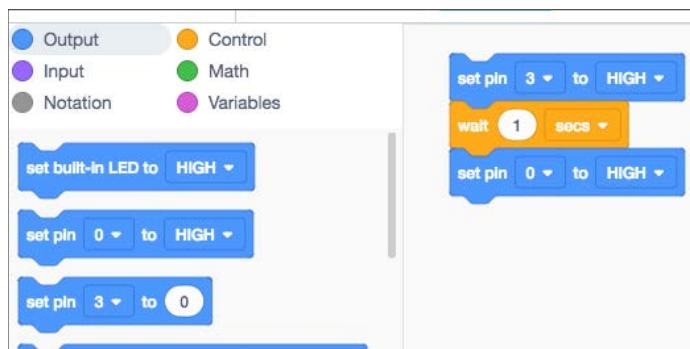
Blinking LED

The code we used in the previous example turned the Arduino into a glorified switch. We can do so much more with Arduino and in this lesson, we will take a closer look at what can be done with code and the same LED.

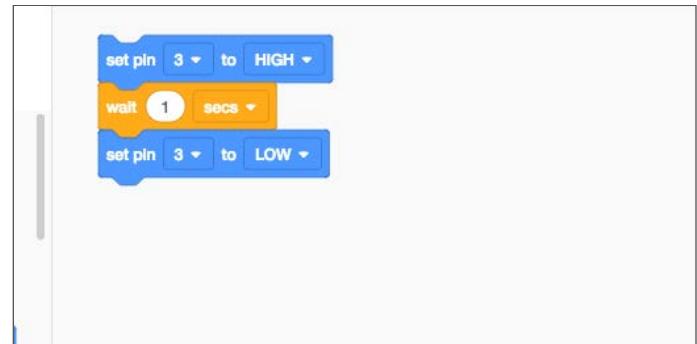
Click the code editor button to open the coding panel. Click the Control code block category and look for the wait code block.



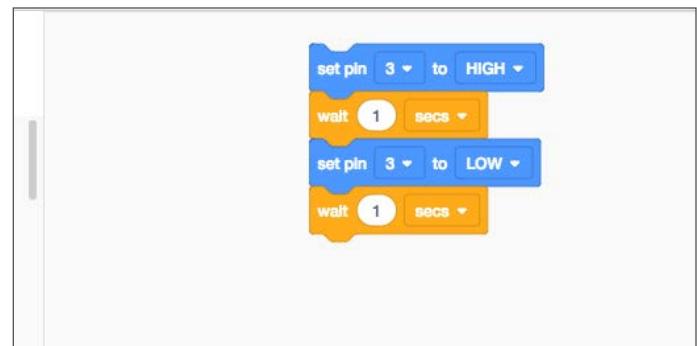
Place the wait code block below the set pin code block. The wait argument is set to one second. Leave the argument at this value. Go back to the Output code block category and place another set pin code block onto the canvas below the wait code block.



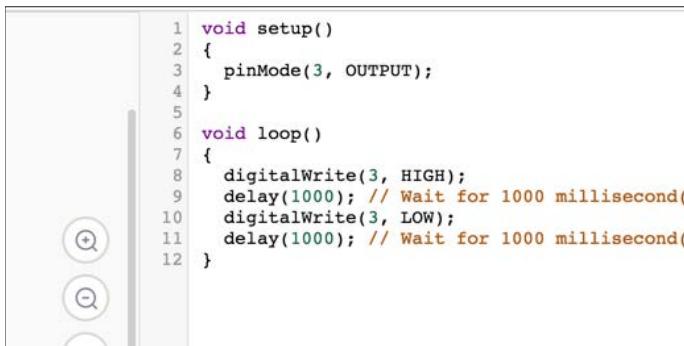
Set the pin argument value to 3 and the state to low. Low will turn the light OFF. Run the simulation. The LED will turn on and off but the OFF state is too short. Arduino is a very simple computer but it is still very fast and processes our instructions in fractions of a second. We need to instruct the code to slow things down so we have time to see the changes.



Go to the scripts panel and find a wait code block. Add a wait code block after the last pin code blocks. Leave the wait argument at the value of 1. Run the simulation again. This time the LED in the simulation will turn on and off repeatedly. The code we write does not include a loop function but the Arduino repeats the code anyway. Arduino is built to repeat coded instructions until it is turned OFF.



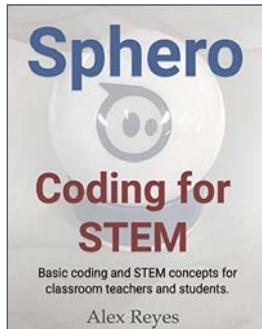
The code blocks we use are representations of written code. The written code is shown on the right side. The code has two main sections or functions. The void setup function is used to set pin 3 as the output pin for the instructions. The void loop function is where we write the main part of our code. The void loop repeats the code within the function until the simulation stops or until the power is turned off in a physical Arduino board.



```
1 void setup()
2 {
3     pinMode(3, OUTPUT);
4 }
5
6 void loop()
7 {
8     digitalWrite(3, HIGH);
9     delay(1000); // Wait for 1000 millisecond(
10    digitalWrite(3, LOW);
11    delay(1000); // Wait for 1000 millisecond(
12 }
```

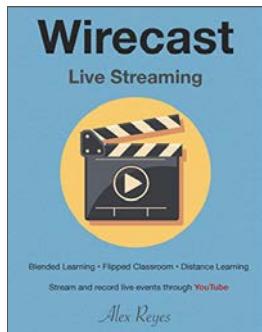
The void loop instructs the board to set the power to pin 3 to high or ON then wait one second. After one second the power to pin 3 is set to low or turned off and then wait one second. The instructions repeat all over again until we stop it by removing power from the Arduino board.

If you like this book then you might like these books too.



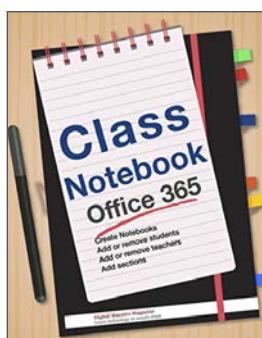
Sphero for STEM

Code Sphero and teach your students STEM concepts. Teach students to apply geometry skills like angles, shapes, and perimeter. Apply science skills of mass, motion, and acceleration.



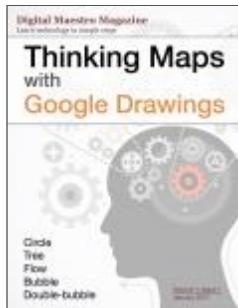
Wirecast Live Streaming

Use the free version of Wirecast to live stream events over YouTube. Create studio quality video that integrates a variety of media and different camera shots. Start your own television studio with very little and stream live content at your campus.



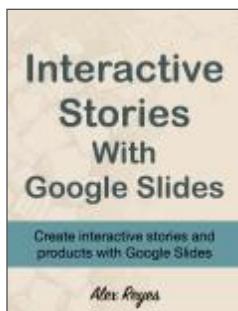
Microsoft Class Notebook

With a Microsoft Office 365 account available to school districts you can create classroom notebooks. Classroom notebooks are great for distributing and collecting a variety of written exercises from students. Class Notebook uses the free OneNote app which is available for a variety of mobile devices.



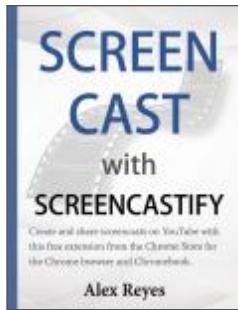
Thinking Maps with Google Drawings

Use Google Drawings to create online thinking maps. Thinking maps can be created by students or shared by teachers with students. Learn to create circle, bubble, flow, brace, and double-bubble thinking maps. Students will not only learn to use thinking maps but they will also learn the skills it takes to build the thinking maps by manipulating graphic objects in Google Drawing.



Interactive Stories with Google Slides

Google slides can be used to create interactive stories. In this book you will learn how to add interactive elements like buttons and links. You will also learn how to add voice over audio for students with reading difficulties. You will learn how to add animated Gifs to a presentation for something a little more interesting than simple slide in and slide out transitions.



Screencast with Screencastify

Screencastify is a free screen recording app for the Google browser and Chromebook. In this book you will learn how to use Screencastify to record web pages, computer desktops, and content captured with an external camera like a webcam. You will also learn how to upload your video to YouTube and share them with the world. Students can create video and share them with you and other students through Google Drive.