

KidzCanCode

IGNITE, INSPIRE, INNOVATE

www.kidzcancode.com

Parallel Circuits – Resistor + LED + LED

KIDZCANCODE

TREVOR WARREN

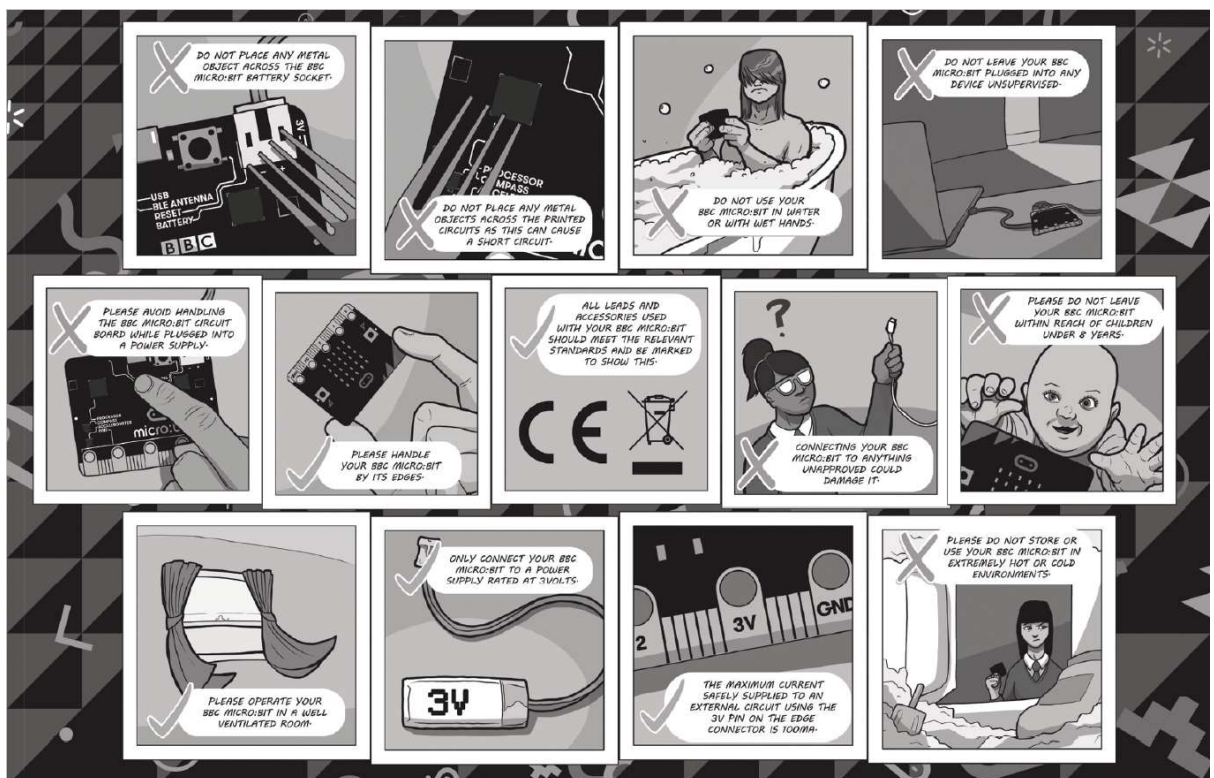
Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED

1. Safety Warnings

THE MICRO:BIT IS AN EXPOSED BOARD, TO BE USED WITH CARE PLEASE RETAIN THIS INFORMATION FOR FUTURE REFERENCE. You can read the detailed document at - <http://microbit.org/guide/safety-advice/>

a. General Safety Warnings

Using the BBC micro:bit is easy to use but is designed to have all the electrical parts on display. This does mean there's a small risk that the parts can be damaged and even overheat with a risk of injury but a little bit of care and caution will ensure you and your micro:bit will stay fit and healthy.

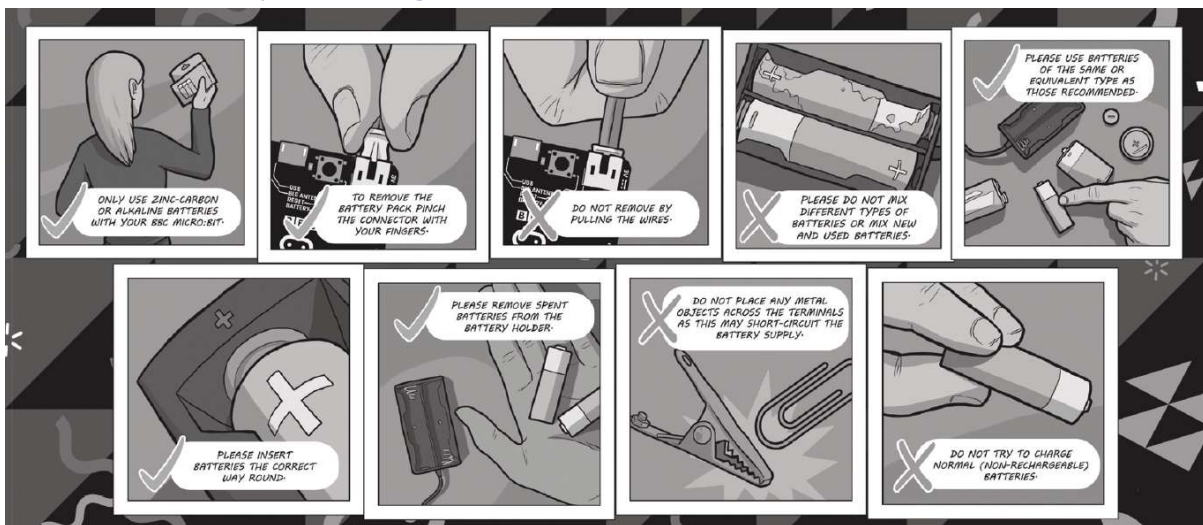


1. Always keep your BBC micro:bit in the anti-static bag when not in use. It's good practice for students to earth themselves before handling it.
2. Please handle your BBC micro:bit by its edges. This minimises the risk of damage through an electrostatic discharge.
3. Please use the battery pack and the USB lead provided to power your micro:bit. Do not use portable battery chargers or USB charging ports (often marked with a lightning bolt or 'SS'), to power your micro:bit. Using these may damage your micro:bit and stop it working properly.
4. Please avoid handling the BBC micro:bit circuit board while plugged into a power supply.
5. All peripherals (for example: USB cable, battery holder, sensors) used with your BBC micro:bit should comply with the relevant standards and should be marked accordingly.
6. Connecting your BBC micro:bit to any unapproved peripherals could damage your BBC micro:bit
7. Please do not attempt to keep using faulty micro:bits. If a school-issued micro:bit develops a fault, contact the vendor immediately.

Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED

8. The maximum current safely supplied to an external circuit using the 3V pin on the edge connector is 100mA. Please make sure this limit is not exceeded.
9. Please do not store or use your BBC micro:bit in extremely hot or cold environments.
10. Do not place any metal objects across the printed circuits on the board as this can cause a short circuit damaging your BBC micro:bit. This can cause risk of burn or fire.
11. Do not use your BBC micro:bit in water or with wet hands.
12. Do not leave your BBC micro:bit plugged into a computer or any other device unsupervised.
13. Please do not leave your BBC micro:bit within reach of children under 8 years of age.
14. Please operate your BBC micro:bit in a well ventilated room To remove the battery pack, pinch the connector with your fingers. Do not remove by pulling the wires.

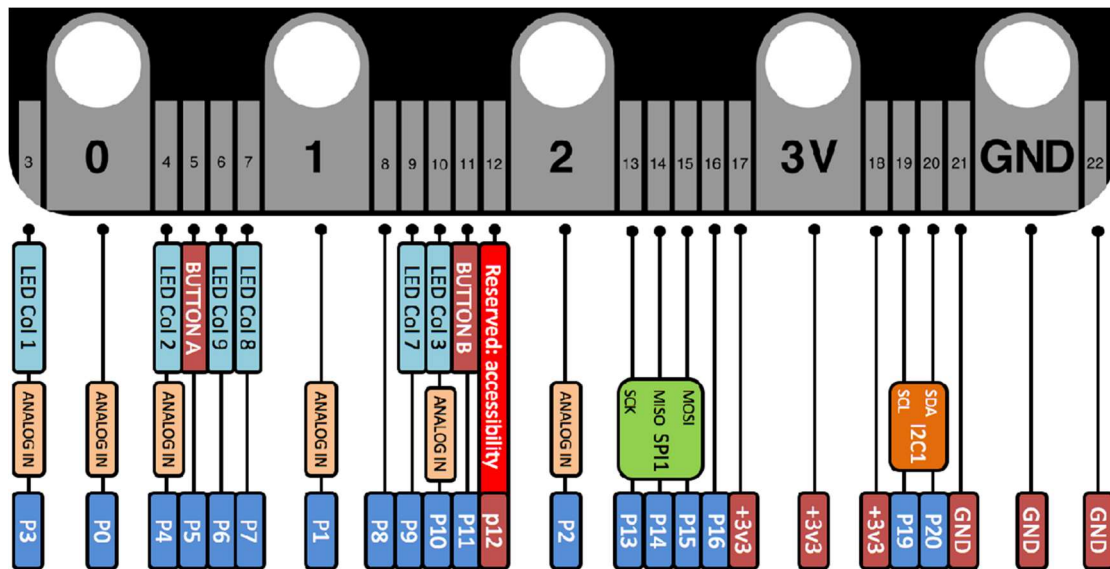
b. Battery Warnings



1. Do not try to charge normal (non-rechargeable) batteries
2. Please do not mix different types of batteries or mix new and used batteries.
3. Please use batteries of the same or equivalent type as those recommended.
4. Please insert batteries the correct way round (with the correct polarity).
5. Please remove spent batteries from the battery holder.
6. Do not short-circuit the battery supply terminals, for example by placing a metal object across the terminals.
7. Only use Zinc or Alkaline batteries with your BBC micro:bit.
8. Please do not use rechargeable batteries

2. BBC micro:bit – A Short Introduction

The BBC micro:bit has 25 external connections on the edge connector of the board, which we refer to as ‘pins’. The edge connector is the grey area on the right side of the figure above. There are five large pins that are also connected to holes in the board labelled: 0, 1, 2, 3V, and GND. And along the same edge, there are 20 small pins that you can use when plugging the BBC micro:bit into an edge connector.



Large pins - You can easily attach crocodile clips or 4mm banana plugs to the five large pins.

The first three, labelled 0, 1 and 2 are flexible and can be used for many different things - which means they are often called ‘general purpose input and output’ (shortened to GPIO). These three pins also have the ability to read analogue voltages using something called an analogue-to-digital converter (ADC). They all have the same function:

- **0:** GPIO (general purpose digital input and output) with analogue to digital convertor (ADC).
- **1:** GPIO with ADC
- **2:** GPIO with ADC

The other two large pins (3V and GND) are very different! The pins labelled 3V and GND relate to the power supply of the board, and they should NEVER be connected together.

If the BBC micro:bit is powered by USB or a battery, then you can use the 3V pin as a *power output* to power peripherals with.

3V: 3 volt *power output* or *power input*. (1) *power output*: If the BBC micro:bit is powered by USB or a battery, then you can use the 3V pin as a power output to power peripherals with; (2) *power input*: If the BBC micro:bit is not being powered by USB or battery, you can use the 3V pin as a power input to power the BBC micro:bit

GND: attaches to ground in order to complete a circuit (required when using the 3V pin)

If you hold the ‘GND’ pin with one hand, you can program the BBC microbit to detect yourself touching the 0,1 or 2 pins with your other hand, giving you three more buttons to experiment with (you just used your body to complete an electrical circuit).



Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED

Small pins - There are 20 small pins numbered sequentially from 3-22 (these pins are not labeled on the BBC micro:bit, however, they are labelled in the picture above).

Unlike the three large pins that are dedicated to being used for external connections, some of the small pins are shared with other components on the BBC micro:bit board. For example, pin 3 is shared with some of the LEDs on the screen of the BBC micro:bit, so if you are using the screen to scroll messages, you can't use this pin as well. See below for details of the other pins on the micro:bit.

- **pin 3:** GPIO shared with LED Col 1 of the LED screen; can be used for ADC and digital I/O when the LED screen is turned off.
- **pin 4:** GPIO shared with LED Col 2 of the LED screen; can be used for ADC and digital I/O when the LED screen is turned off.
- **pin 5:** GPIO shared with Button A. This lets you trigger or detect a button "A" click externally. This pin has a pull-up resistor, which means that by default it is at voltage of 3V. To replace button A on the BBC micro:bit with an external button, connect one end of the external button to pin 5 and the other end to GND. When the button is pressed, the voltage on pin 5 is pulled down to 0, which generates a button click event.
- **pin 6:** GPIO shared with LED Col 9 of the LED screen; can be used for digital I/O when the LED screen is turned off.
- **pin 7:** GPIO shared with LED Col 8 of the LED screen; can be used for digital I/O when the LED screen is turned off.
- **pin 8:** Dedicated GPIO, for sending and sensing digital signals.
- **pin 9:** GPIO shared with LED Col 7 of the LED screen; can be used for digital I/O when the LED screen is turned off.
- **pin 10:** GPIO shared with LED Col 3 of the LED screen; can be used for ADC and digital I/O when the LED screen is turned off.
- **pin 11:** GPIO shared with Button B. This lets you trigger or detect a button "B" click externally.
- **pin 12:** Dedicated GPIO, for sending and sensing digital signals.
- **pin 13:** GPIO that is conventionally used for the serial clock (SCK) signal of the 3-wire Serial Peripheral Interface (SPI) bus.
- **pin 14:** GPIO that is conventionally used for the Master In Slave Out (MISO) signal of the SPI bus.
- **pin 15:** GPIO that is conventionally used for the Master Out Slave In (MOSI) signal of the SPI bus.
- **pin 16:** Dedicated GPIO (conventionally also used for SPI 'Chip Select' function).
- **pins 17 and 18:** these pins are wired to the 3V supply, like the large '3V' pad.
- **pins 19 and 20:** implement the clock signal (SCL) and data line (SDA) of the I2C bus communication protocol. With I2C, several devices can be connected on the same bus and send/read messages to and from the CPU. Internally, the accelerometer and the compass are connected to i2c.
- **pins 21 and 22:** these pins are wired to the GND pin and serve no other function

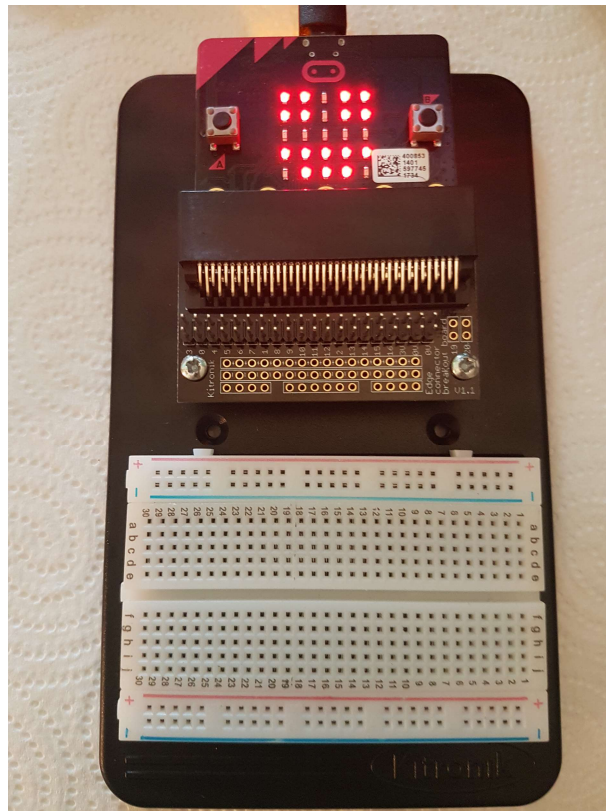
Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED

3. Pre-requisites

If you need assistance with the assembly of the micro:bit, Edge connector breakout board, mounting board and the breadboard please speak to one of the volunteers on duty.

To be able to perform this tutorial you will need the following components –

1. Parts required –
 - a. 1 x BBC Micro:bit
 - b. 1 x Mounting Plate
 - c. 1 x Edge connector breakout board
 - d. 1 x Bread board
 - e. 2 x LED
 - f. 1 x 2.2K (2200) Ohm Resistor
 - g. 3 x Male – Female Dupont wires
 - h. 3 x Male – Male Dupont wires
2. Assembly required –
 - a. Bread board mounted on top of the mounting plate
 - b. BBC Micro:bit inserted into the Edge Connector breakout board



Before proceeding please check your setup and confirm that all the required parts are configured as demonstrated in the above picture.

4. Learning Objectives

The objectives of this tutorial are to explore and learn the basic concepts of electricity, circuits, parallel circuits and elements that make up a parallel circuit. This tutorial builds upon concepts introduced in previous tutorials so please make sure you have covered the previous tutorials before you dive into this one.

Explore the following concepts through making –

1. What is a parallel circuit
2. Current flowing through a parallel circuit
3. How does the flow of current in a parallel circuit differ from that in a series circuit
4. What are passive and active components in a parallel circuit
5. Connecting elements in a circuit in parallel
6. How does the voltage drop across the LED's in a parallel circuit (shown below) differ from the voltage drop across two LEDs connected the series circuit.

5. Exploring components

If you haven't been introduced to the following components it's worth reading through this section –

1. Resistors
2. LED's

Let's briefly look at what resistors and LED's are all about before we dive into actually making the circuit.

a. What Are Resistors

"What is a resistor?" the student asked.

"It's a component that resists the flow of current" I said.

"Hmm.. I don't get it. What does it do to my circuit?" the student asked.

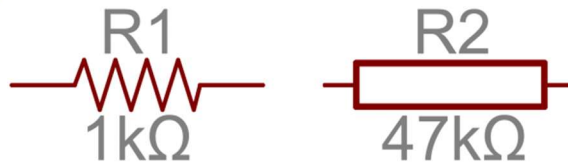
"Well, it doesn't actually do anything actively, all a resistor does is resists the flow of current through a given circuit" I said. Have a look at what the resistor might look like.

Resistors are electronic components which have a specific, never-changing electrical resistance. The resistor's resistance limits the flow of electrons through a circuit. They are passive components, meaning they only consume power (and can't generate it). Resistors are usually added to circuits where they complement active components like op-amps, microcontrollers, and other integrated circuits. Commonly resistors are used to limit current, divide voltages, and pull-up I/O lines.



The schematic symbol used for resistors in circuits is –

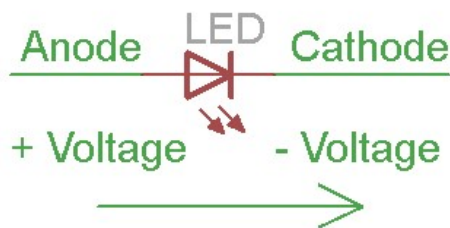
Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED



b. What Are LED's

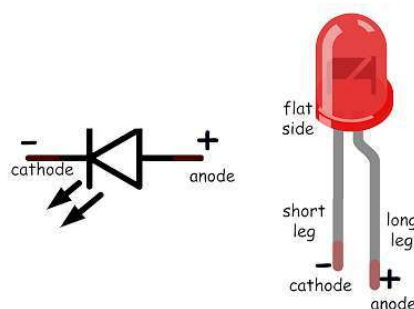
LEDs are so common, they come in dozens of different shapes and sizes. The LEDs you are most likely to use are the through hole LEDs with two legs. There are lots of LEDs that are small and hard to solder but these are easy to use with a breadboard because they have long wires we can stick in. The clear or clearish bulb is what protects the light emitter (that's where the magic happens). In fact, the first two letters of LED stand for Light Emitting.

A really nice thing about LEDs is that they are very simple. Unlike some chips that have dozens of pins with names and special uses, LEDs have only two wires. One wire is the anode (positive) and another is the cathode (negative). The two wires have different names because LEDs only work in one direction and we need to keep track of which pin is which. One goes to the positive voltage and the other goes to the negative voltage. Electronic parts that only work in 'one direction' like this are called Diodes, that's what the last letter of LED stands for.



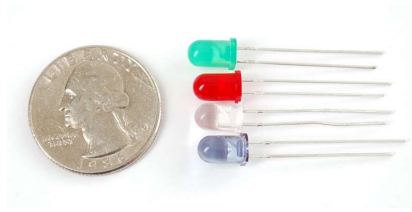
- The longer lead goes to the more-positive voltage
- Current goes in one direction, from the anode (positive) to the cathode (negative)
- LEDs that are 'backwards' won't work - but they won't break either

The schematic representation for LED's in circuits is as follows –



It's all a little confusing - we often have to think about which is which. So to make it easy, there's only one thing you need to remember and that's the LED won't light up if you put it in backwards. If you're ever having LED problems where they are not lighting, just flip it around. It's very hard to damage an LED by putting it in backwards so don't be scared if you do

Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED



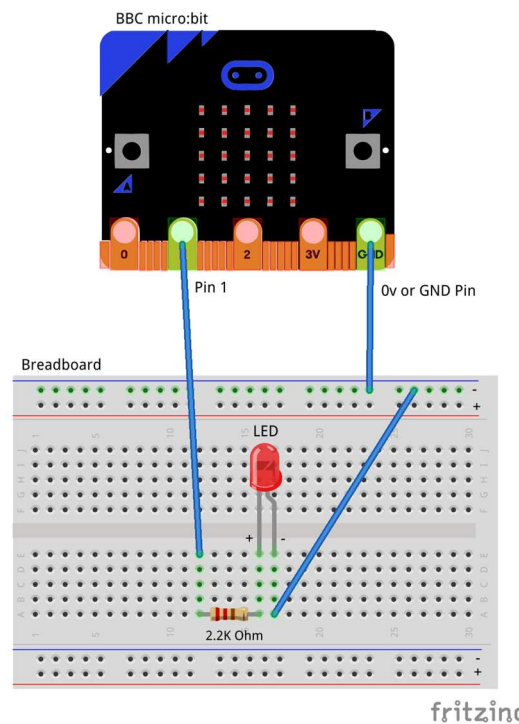
One of the best things about modern LEDs is all the colors they come in. It used to be that LEDs were only red or maybe yellow and orange, which is why early electronics from the 70s and 80s only had red LEDs. The color emitted from an LED has to do with what type of material they are made of. So red, for example, is made with Gallium Arsenide. Since then, scientists have experimented with many other materials and figured out how to make other colors such as green and blue, as well as violet and white.

6. Activity

Before we get started, let's make sure that all the required components are configured as required –

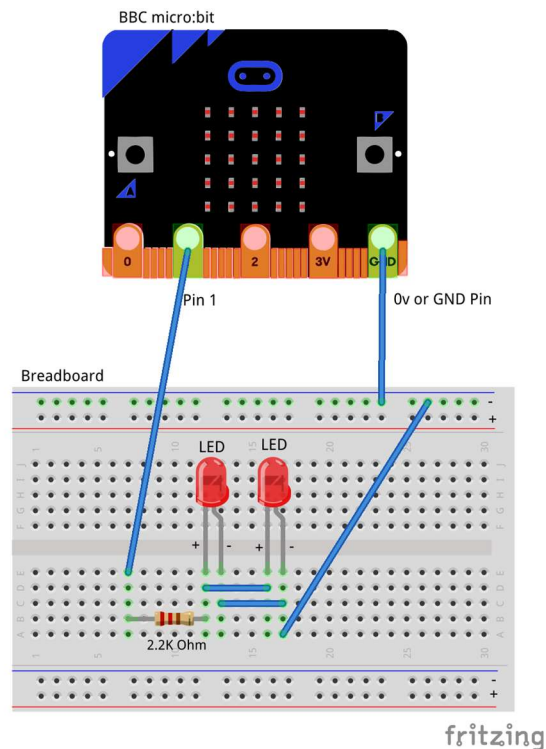
a. Step 1 - Build

Let's now connect up the relevant components on our micro:bit.



You will realize that the diagram above is similar to the one you have seen in the previous tutorial. The first step in this tutorial involves putting together a series circuit with all the elements i.e. LED, Resistor and micro:bit which you've used in the previous tutorial. In the next step we will then extend this circuit to include elements in parallel.

Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED



In step 2 you connect up the second LED in parallel to the first LED with a slight modification to the circuit. The main difference between the first circuit and the second is the addition of the second LED in parallel which causes the current to split between both the LED's equally. The voltage drop however across both the LED's are now the same because they are in parallel. The total voltage drop across the circuit is the sum of the voltage across the resistor and the LED's i.e. $V_{total} = V_{resistor} + V_{led}$

However, in our previous tutorial where we build the series circuit which included two LEDs and two Resistors, the voltage drop across the circuit is a combination of, Voltage across LED1 + Voltage across LED2 + Voltage across resistor1 + Voltage across resistor 2. The equation can be written as $V_{total} = V_{led1} + V_{led2} + V_{resistor1} + V_{resistor2}$

The connections should be made as follows:

1. Connect the Male – Female Dupont wire from the Ground (0v) pin to the ground rail on the breadboard.
2. Connect a Male – Female Dupont wire from Pin1 pin on the micro:bit (using the Edge Connector Breakout board) to 2.2K Ohm resistor on the breadboard
3. Connect the 2.2K resistor to the Anode (Positive, Longer leg) of the LED on the breadboard
4. Connect the Anode (Positive, Longer leg) of the first LED to the Anode (Positive, Longer leg) of the second LED.
5. Connect the Cathode (Negative, Shorter Leg) of the first LED to the Cathode (Negative, Shorter Leg) of the second LED.
6. Connect a Male – Male DuPont wire from the Cathode (Negative, Shorter leg) of the second LED to the ground rail (0v) on the breadboard.

Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED

The flow of the current is now from Pin1 to the resistor, splitting into two across the two LEDs and then back to ground. Therefore the current through each of the LED's should in theory be half of the current that flows through the resistor. However, the voltage across the LED should be the same for both the LED's since they are in parallel.

Self-Assessment

- I have the breadboard mounted onto the mounting plate.
- I have the micro:bit plugged into the Edge connector breakout board.
- I understand each of the components in the circuit i.e. LED, resistor, Male – Female Dupont wires
- I have each of the elements in the circuit wired up using the breadboard
- I have checked the connections on the circuit and have not powered it on.
- I understand the use of the 2.2K resistor in the circuit
- I understand the difference between 3 Volt (3V) and Ground (0 Volt of 0V)
- I understand how to use the male – male and male – female dupont cables to create connections using the breadboard
- I understand what the positive / 3v and ground / 0v rails (top and bottom) of the breadboard do and how they are connected
- I understand how the central rows in the breadboard are lined up together and how they are connected.

This now completes your circuit.

PLEASE DO NOT POWER ON YOUR micro:bit UNLESS YOU HAVE HAD YOUR CIRCUIT VALIDATED BY A FACILITATOR/MENTOR/Volunteer.

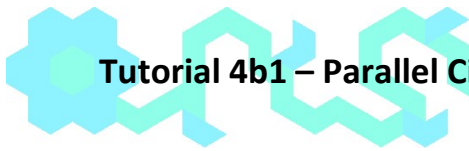
b. Step 2 - Explore

Discuss the following concepts with the learning facilitator

1. What is a series circuit? Which components in this circuit are in series?
2. What is a parallel circuit? Which components in this circuit are in parallel?
3. What are the paths for current flowing through this parallel circuit?
4. Which are passive and active components in a parallel circuit?
5. How does the voltage drop across the LED's in a parallel circuit (shown below) differ from the voltage drop across two LEDs connected the series circuit.

Self-Assessment

- I can explain what a circuit means, what are components connected in series and components connected in parallel.
- I can explain the differences between active and passive components in circuits.
- I understand how components can be connected in series.
- I understand how components can be connected in parallel.
- I understand how the current flow changes in parallel circuits as compared to series circuits.
- I understand how the voltage drop across the LED's in the parallel circuit differs from the voltage drop across the LED's in the series circuit.



Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED

c. Step 3 – Validate

Please call a facilitator to check your circuit before you power it on. It is important that you get a learning facilitator/mentor/volunteer to check the connections on your circuit before powering it on.

Incorrect connections might result in a dead (bricked, burnt, short-circuited, etc.) Micro:bit requiring you to purchase another Micro:bit for \$30 AUD.

d. Step 4 – Power On

Awesome, you are almost at the point where you can power on your circuit and watch the magic un-fold. Please call a facilitator to check your circuit before you power it on. It is important that you get a learning facilitator to check the connections on your circuit before powering it on.

Once the learning facilitator has verified your circuit you should now proceed and power on the Micro:bit. If you have not had your learning facilitator verify the circuit PLEASE GO BACK To step 3 and have one of the learning facilitators verify your connections.

Powering on the Micro:bit requires that you connect the USB power cord for your Micro:bit (plugged into the Micro:bit board) into your laptop.

Incorrect connections might result in a dead (bricked, burnt, short-circuited, etc.) Micro:bit requiring you to purchase another Micro:bit for \$30 AUD.

Self-Assessment

- I am able to wire the series circuit with all the elements shown
- I understand the flow of current through a series circuit
- I understand the difference between 3 Volt (3V) and Ground (0 Volt of 0V)

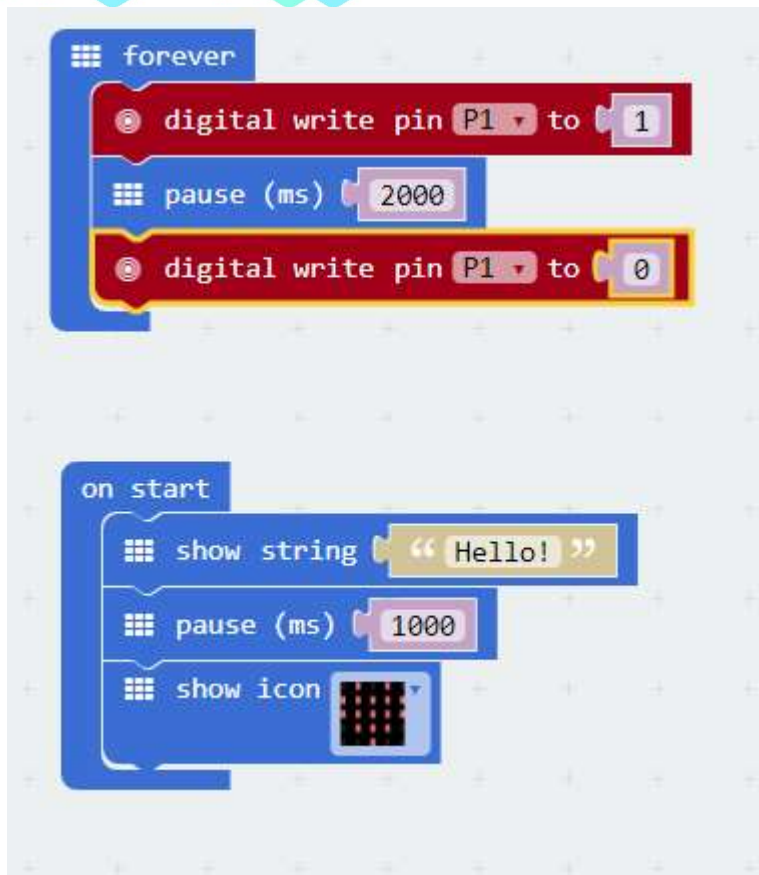
7. Let's write some code

Now that we have put together the circuit it's time to write some code and get the circuit working. So let's head over to the micro:bit block code editor page (<https://makecode.microbit.org/>) and get coding!!!

In the following section we will dive into the code you will put together for this tutorial –



Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED



On start code block -

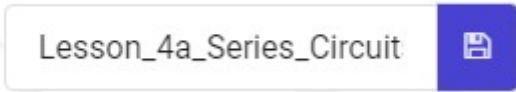
1. The on-start code block only runs once per session. You can make this block run again by hitting the re-set button on the reverse of the micro:bit.
2. The first line on the code block tells the micro:bit to display the string, "Hello". Feel free to customize this to whatever you would like to see.
3. The second line of code simply puts the micro:bit to sleep for 1000ms or 1s.
4. The third line of code tells the micro:bit to display the heart pattern by lighting LED's on the board in the shape of a heart. You should customize this and change the shape to what you would like to see.

Forever code block –

1. The forever code block on the bbc micro:bit is designed to run code literally for ever. So any code you place within this block will run for ever.
2. Our first statement in this block is a "digital write pin P1 to 1" which says, please turn on the LED by giving it a high (3v) signal.
3. Our second statement in this block simply tells the micro:bit to please go to sleep and do nothing for 2000ms or 2s
4. Our third statement in this block is a "digital write pin P1 to 0" which says, please turn off the LED by giving it a low (0v) signal.

8. Downloading Your Code To The micro:bit

Once you have completed the program, enter a name for your program using the option provided below i.e. in the text box adjacent to the small save button.



Lesson_4a_Series_Circuit

Now that you have given your program a name and saved it you can download it your micro:bit. But before we do that let's confirm what drive your micro:bit shows up as. On most machines the micro:bit will show up as an additional USB drive. So head over into windows explorer and confirm what drive name (D:, E:, F:, G:, etc.) the micro:bit shows up as. You need to absolutely be sure what drive the micro:bit shows up as. Once you've confirmed what drive the micro:bit shows up as on your machine you can select the right drive when downloading the code to the micro:bit. If in doubt please ask the volunteer/mentor/session facilitator helping out.



Download

To download the newly written code to the micro:bit, hit the download button shown above. You should now see a dialog box open up and you will be asked to save the file somewhere on your machine. Please choose the drive your micro:bit shows up as i.e. D: or E: or F: or whatever it shows up as on your machine. A sign of success is when you see the lights on the rear (orange) lighting up in quick succession suggesting that the code is being written to the micro:bit. On completion the micro:bit reboots and you should now see the code in action on the micro:bit.

If hitting the download button shown above does not open up a dialog box asking you to save to the micro:bit please save the file (you will have a <filename.hex> file) to your desktop. Then open up windows explorer and drag that file onto the drive which is your micro:bit. A sign of success is when you see the lights on the rear (orange) lighting up in quick succession suggesting that the code is being written to the micro:bit. On completion the micro:bit reboots and you should now see the code in action on the micro:bit.

Please feel free to customize the code blocks, have a play. Add your own custom code and re-download the code to the micro:bit. Give yourself a tap on the back, you've just completed your first circuit!!!!

Tutorial 4b1 – Parallel Circuits – Resistor + LED – LED

9. Challenges

Well done for completing the Series Circuit tutorial. There's a lot of ground we have covered in this tutorial so please feel free to make notes, come back to the tutorial at some point down the line and ask your learning facilitator any questions or doubts you might have on the concepts covered this far.

For those who want to stretch it a bit further -

1. What happens if you add another LED or more than a couple of LED's in series with the first one?
2. What happens if you add a lower value resistor in parallel with the first resistor? How is the flow of current impacted?
3. What happens if you add a higher value resistor combined with the lower value resistor in series? How is the flow of current impacted?
4. Can you write the code for the above circuit in python?

