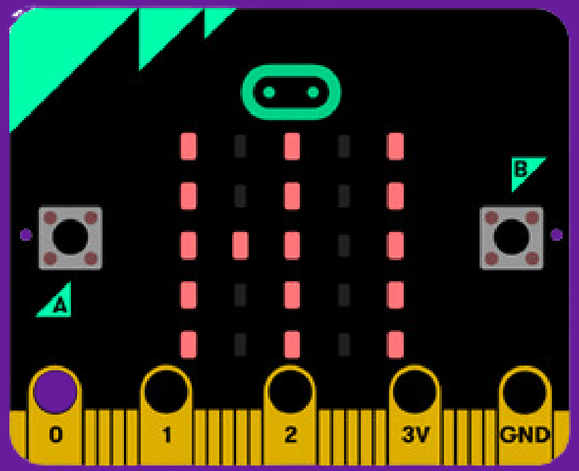
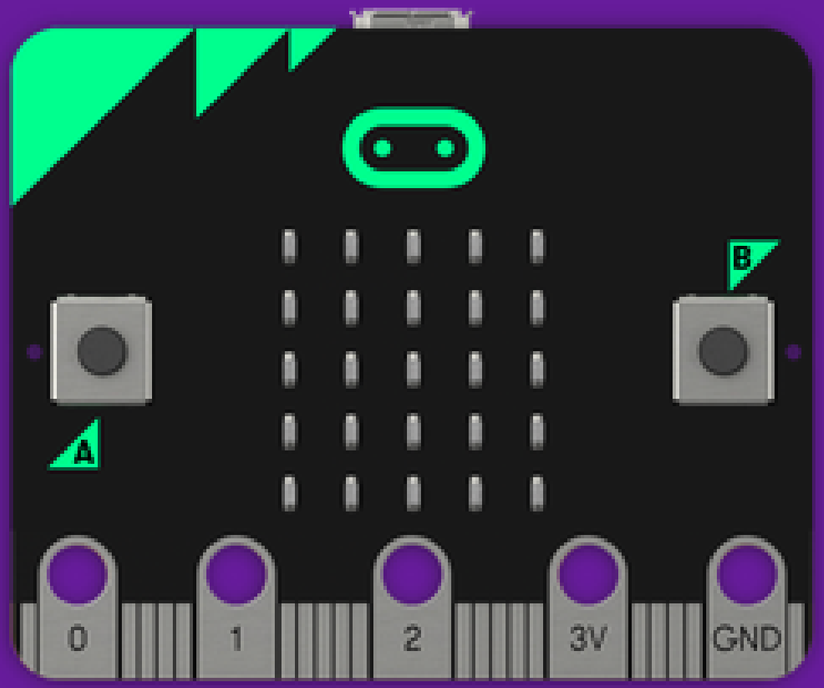


Micro:Bit 101

Explore Possibilities with Micro:Bit V2

@CoraEdTech

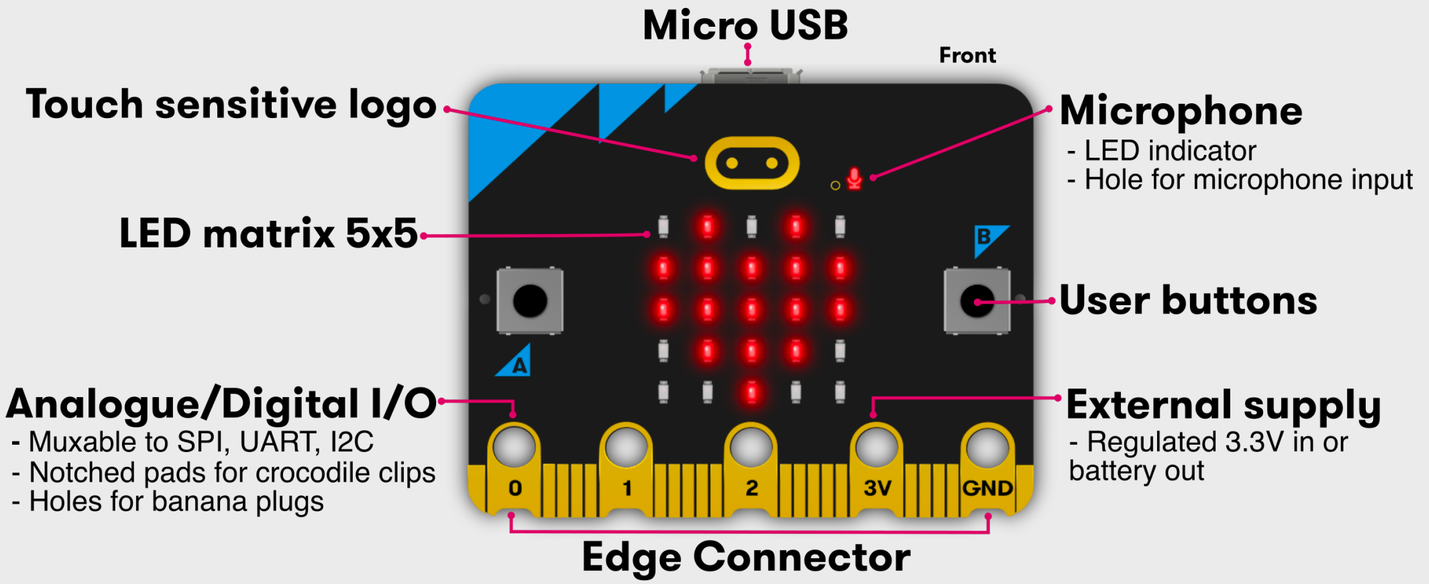


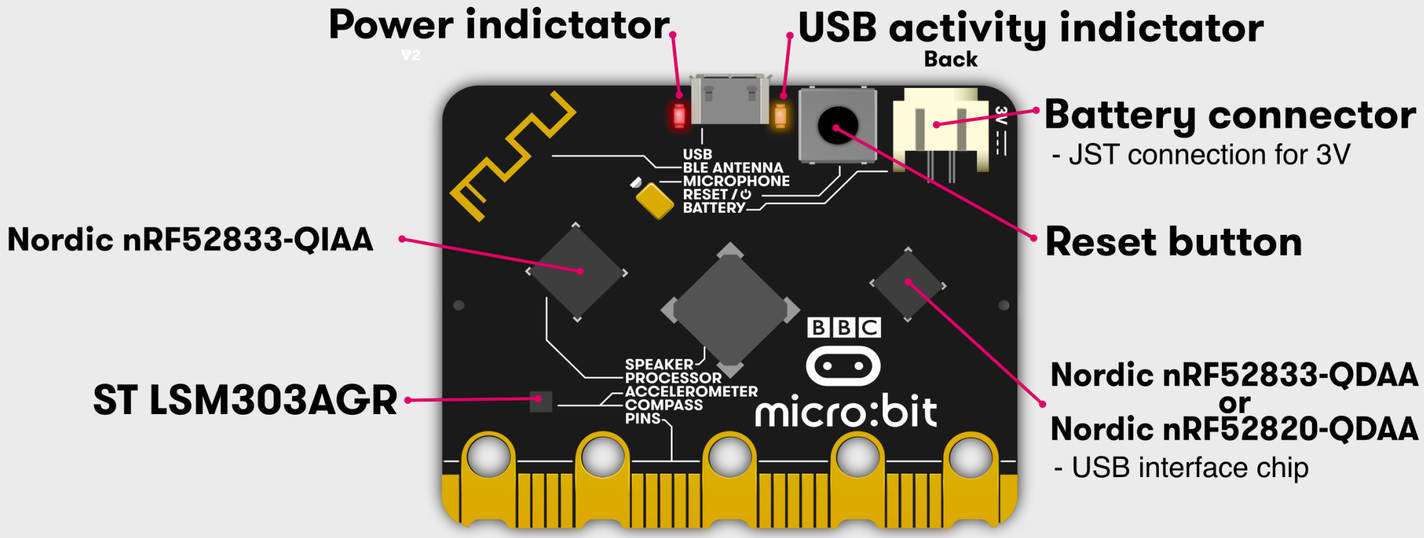
What is Micro:Bit?

The micro:bit is a tiny computer that makes coding tangible and promotes digital creativity.

You can program instructions to tell it what to do by using the web-based editor MakeCode, and download codes as .hex files - a version of your codes that your micro:bit can read.

Getting To Know Micro:Bit V2





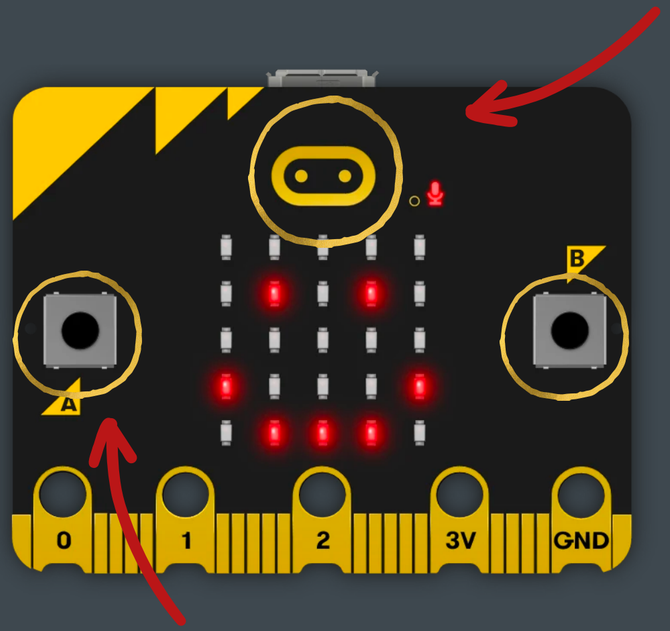
@CoraEdTech

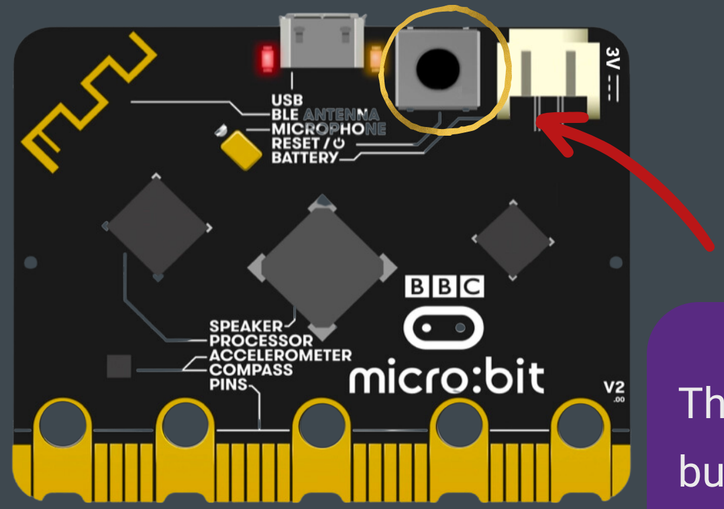
Getting To Know Micro:Bit V2

Buttons

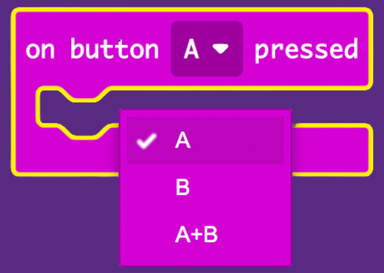
The logo (V2 only) on the micro:bit can be configured to know when they are being touched or pressed.







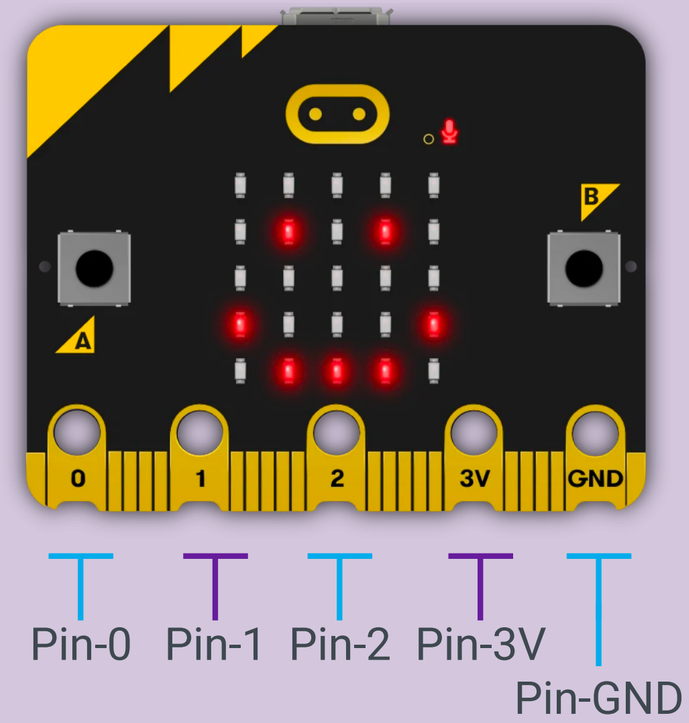
Front buttons A and B are sensors on Micro:Bit. Like other triggers, your code can start with pressing buttons.



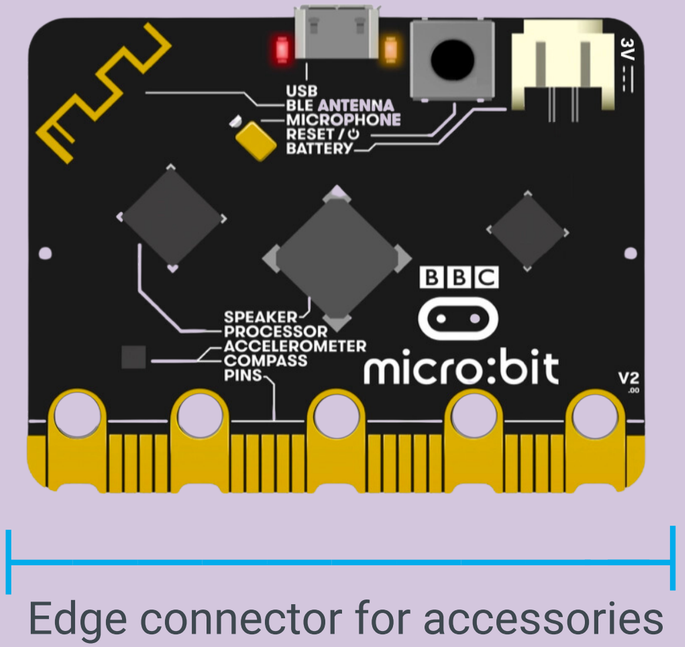
The micro:bit has a reset/power button located on the back.

Getting To Know Micro:Bit V2

Pins



On the bottom edge of your BBC micro:bit there are 25 gold strips, called pins. These pins allow you to really get creative. You can create circuits, connect external things like buzzers and motors and make your own fun projects.



Getting To Know Micro:Bit V2

LEDs

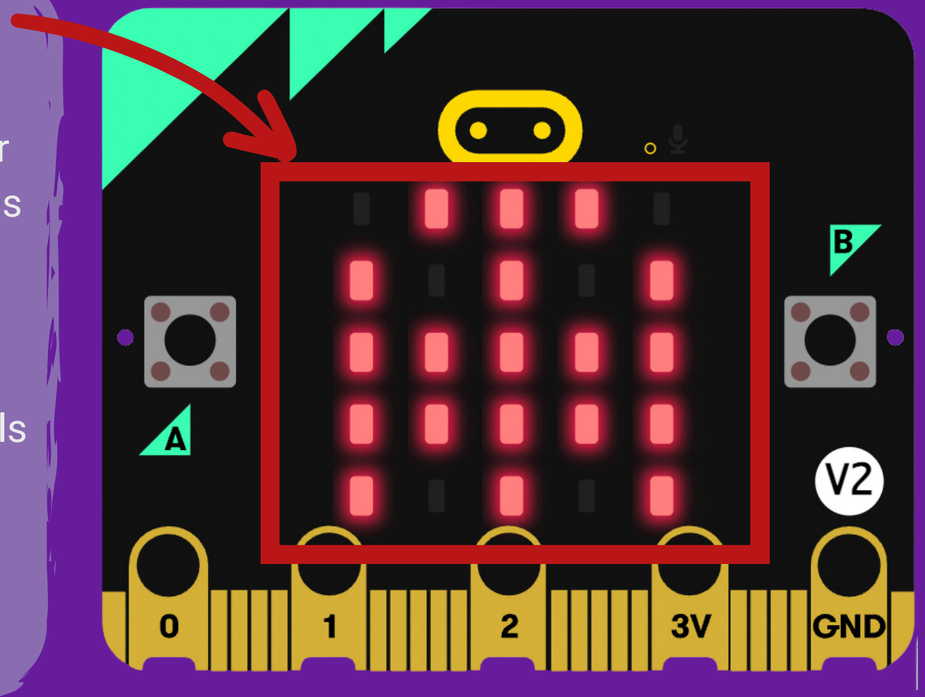
LEDs

An LED, or light-emitting diode is an output device that gives off light. Your BBC micro:bit has a display of 25 LEDs for you to program.

Light sensor

Micro:Bit uses LEDs to sense the levels of light and lets you program your micro:bit as a light sensor.





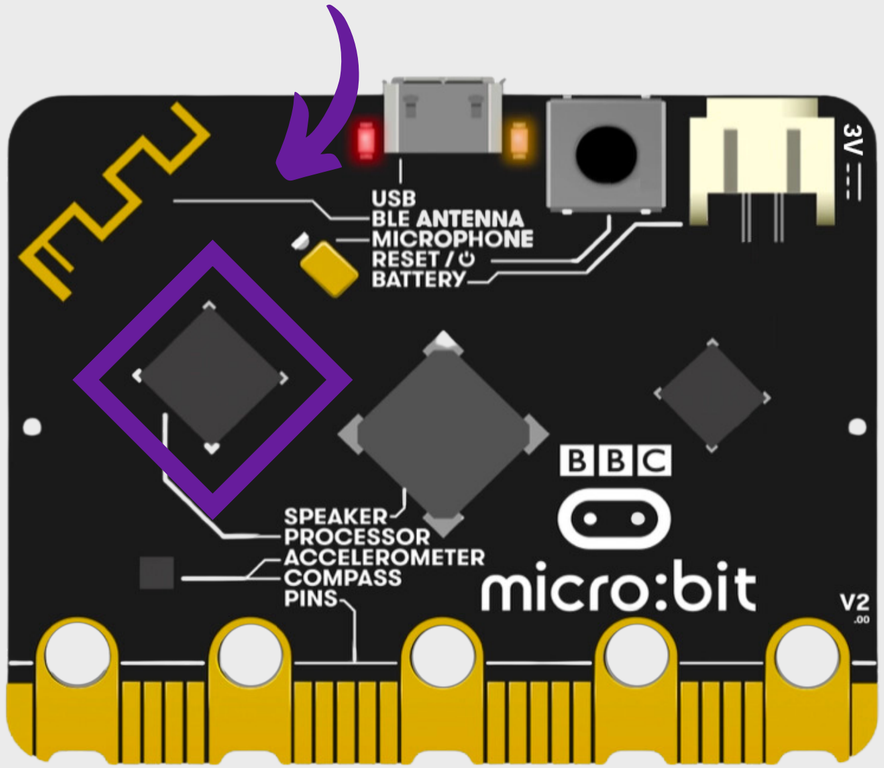
Output

LEDs on Micro:Bit V1 is the output. You can use them to express information within 5*5 LED grids.



Getting To Know Micro:Bit V2

Processor



What is it?

A processor is sometimes called the **‘brains’ of a computer** and each Micro:Bit has a micro processor inside it.
It’s an essential part of your micro:bit as it runs the programs you write.

Temperature sensor

`temperature (°C)`

A temperature sensor is an input device that measures temperature. A Micro:Bit has **a temperature sensor inside the processor** which can give you an approximation of the air temperature.

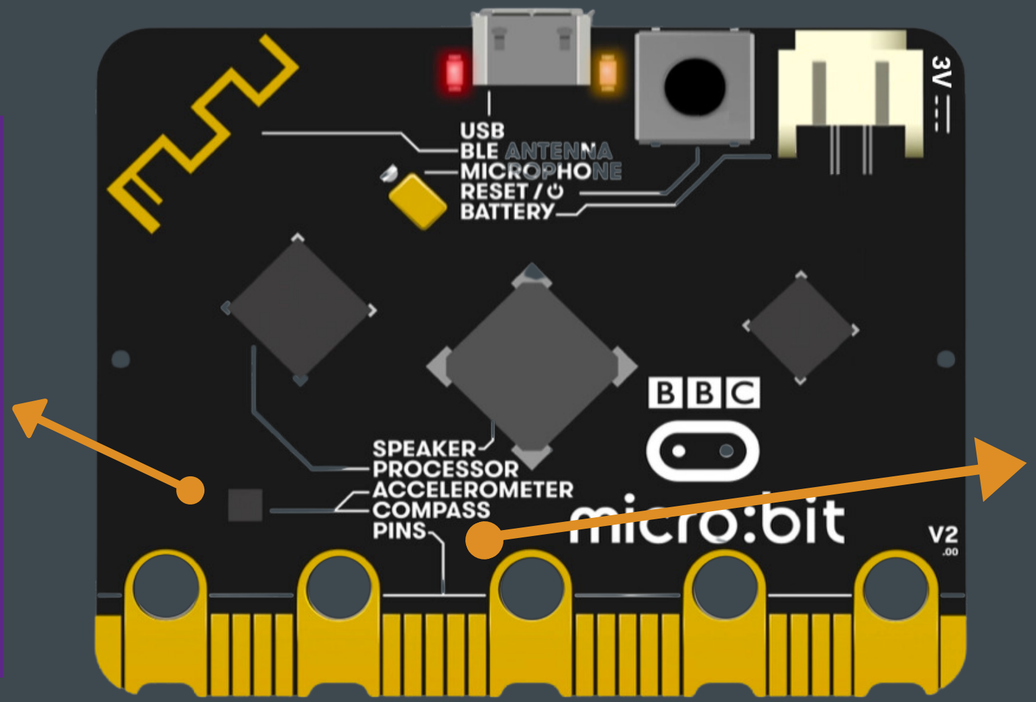
Getting To Know Micro:Bit V2

Accelerometer & Compass

Compass

A digital compass is an input sensor that detects magnetic fields.
Each Micro:bit has an inbuilt compass that can detect the direction in which it is facing.

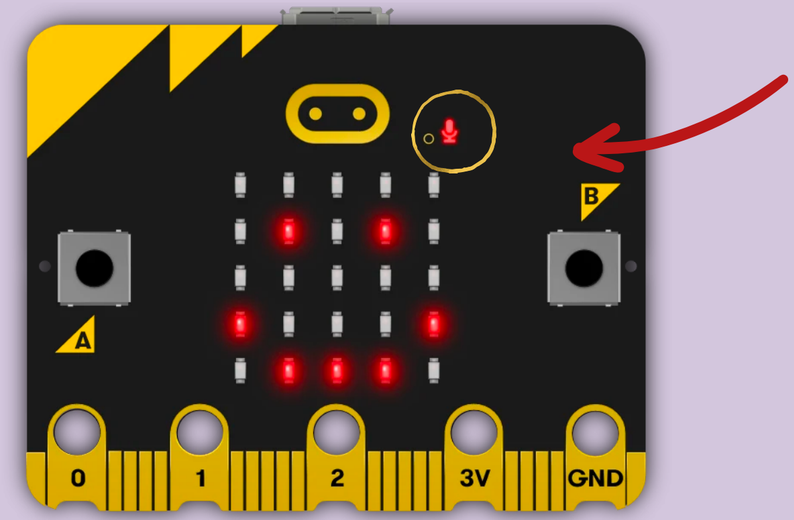
```
compass heading (°)
```



Accelerometer

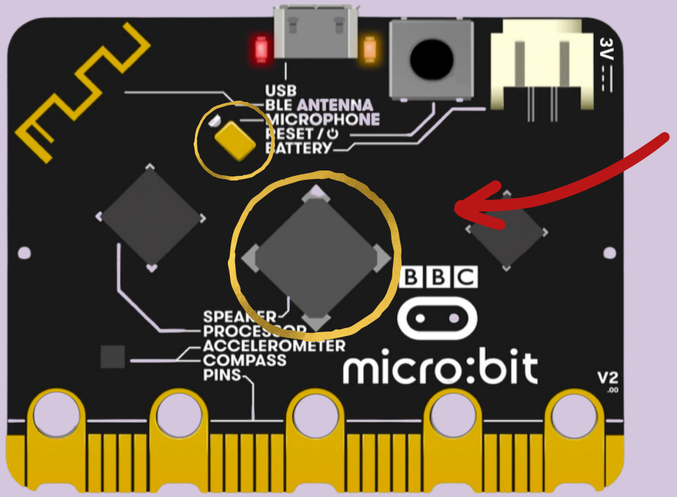
An accelerometer is a motion sensor that measures movement.
The accelerometer in each Micro:bit detects when you tilt it left to right, backwards and forwards and up and down.

```
acceleration (mg) x ▾
```



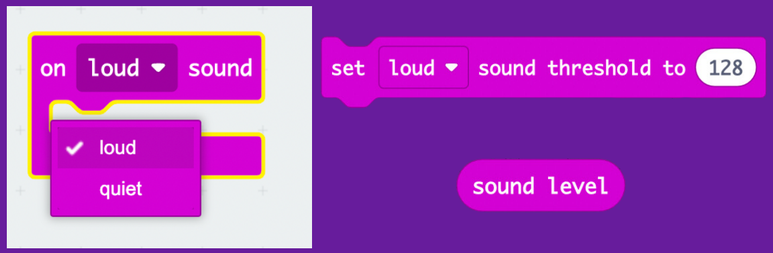
Microphone Input and LED Indicator

It can be used to respond to sound. The microphone input is located on the front of the board alongside a microphone activity LED, which is lit when the microphone is in use.



Speaker and Microphone

The Micro:Bit V2 includes a speaker and a microphone. It is capable of detecting sound and its level, while also generating sound through a buzzer.



Sound

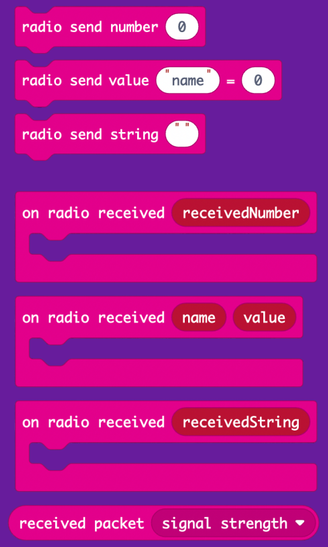
Getting To Know Micro:Bit V2

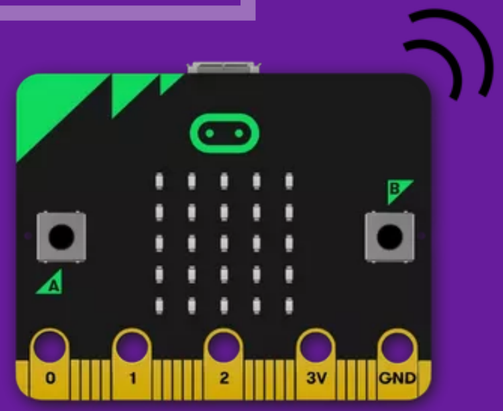
Getting To Know Micro:Bit V2

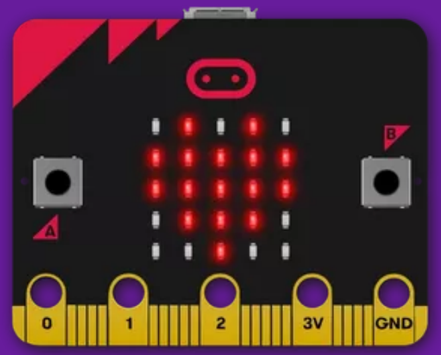
Communications

Radio

Radio is a way of sending and receiving messages and micro:bit can use radio waves to communicate with each other.

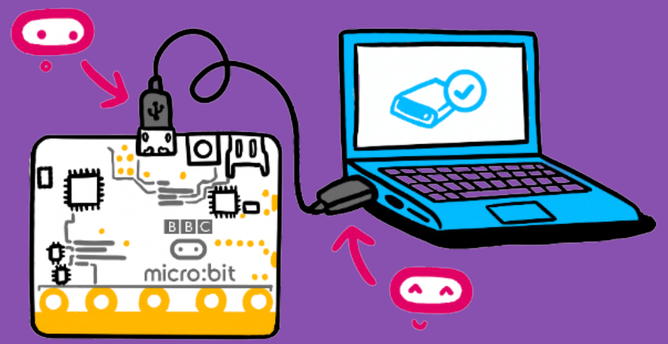






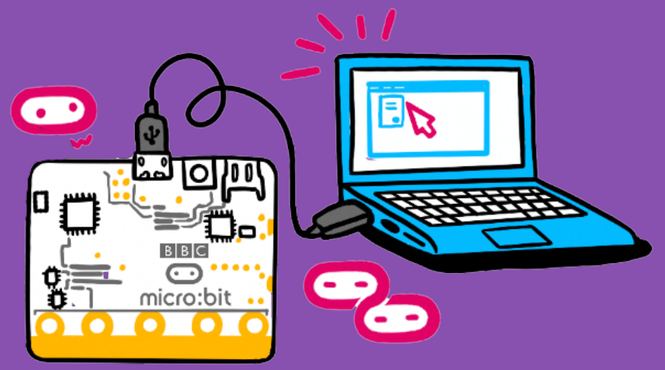
Wireless Bluetooth

The Bluetooth function provides a fully qualified Bluetooth low-energy stack. This allows the micro:bit to communicate with a wide range of Bluetooth devices, including smartphones and tablets.



Step 1

Make sure your micro:bit is connected to your computer with a USB cable.

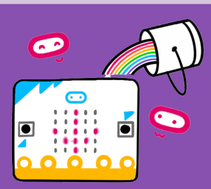


Step 2

Pair your micro:bit to the computer by selecting 'BBC micro:bit CMSIS-DAP' or 'DAPLink CMSIS-DAP' from the popup that appears after you press the 'Next' button below.

Connection

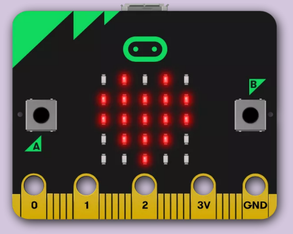
Getting To Know Micro:Bit V2



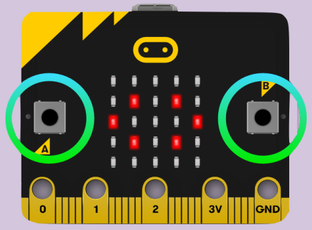
Troubleshooting: Make sure your micro:bit firmware is version 0249 or above.

Sensor Practice

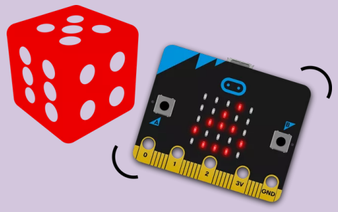
Learn from hands-on activities



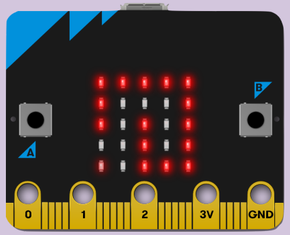
Beating Heart



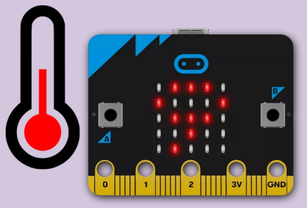
Press It!



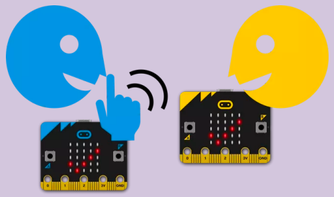
Roll the Dice



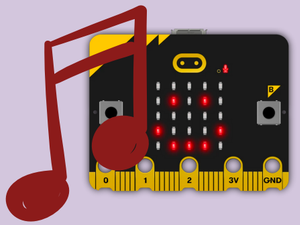
Floating Sand



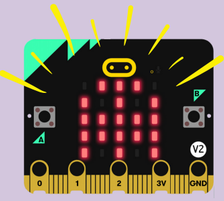
Thermometer



Talk to Me



Play Lullaby



Don't Touch Me!

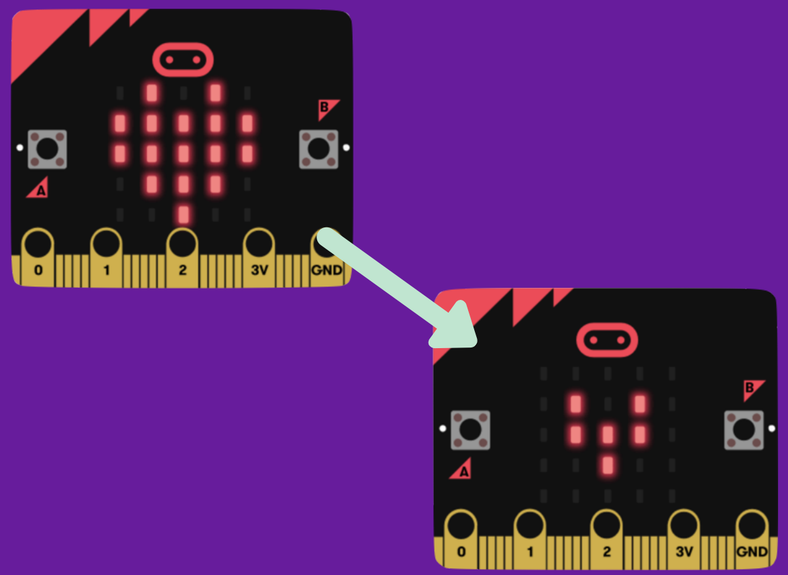
@CoraEdTech

❤ Beating Heart

Use two LED images to make a beating heart.

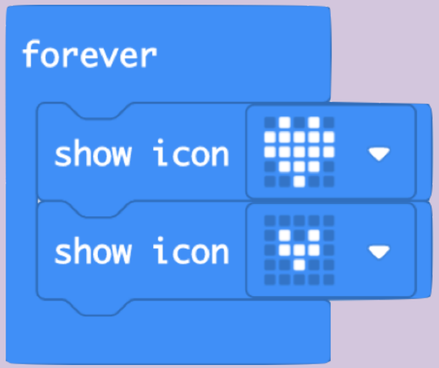
Requirements:

Programme your Micro:Bit to make LEDs show big heart and turn to the small heart - make it look like a beating heart.



Solutions:

Use "show icon"/"show LEDs" to program different icons on LEDs.
You can use "forever" block to make a loop.



Challenge:

Can you try other different icons? Maybe animals, squares or circles?
Or you can create your own icons by lightening LEDs.



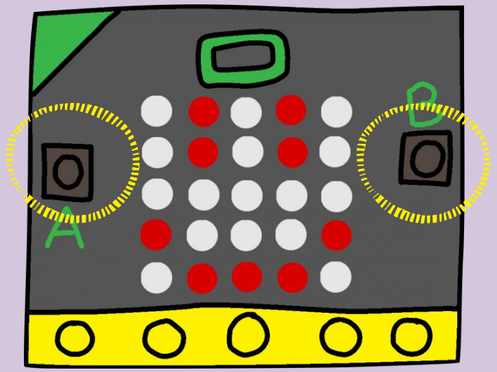
@CoraEdTech

Press It!

A game that using buttons as triggers.

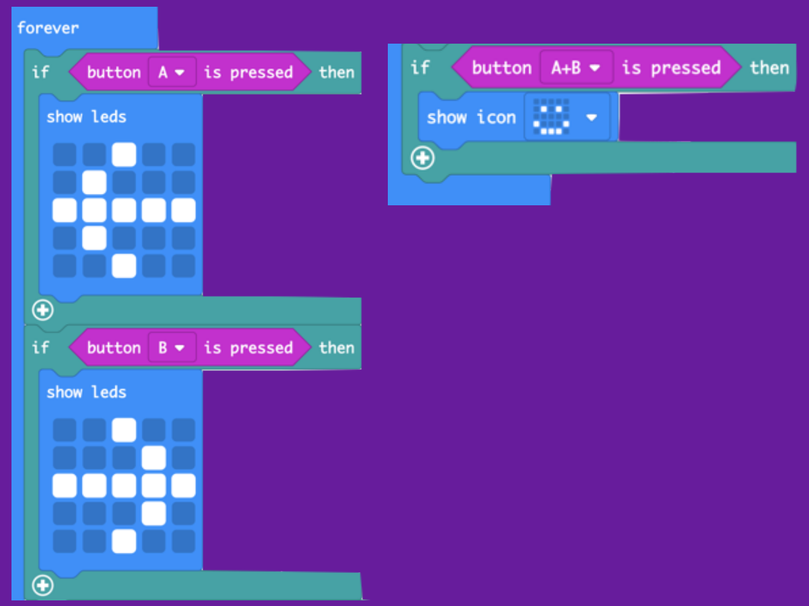
Requirements:

Two players press buttons A/B at the same time.
If button A is pressed first, LEDs will show an arrow point to A or vice verse.
If two players press at the same time, LEDs will show a smily face.



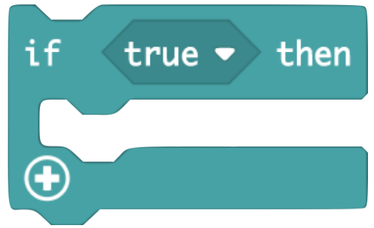
Solutions:

Use conditional block "if... then..." and input block "when button is pressed" to insert conditions in your programming.



Challenge:

Can you try to use buttons to make different animations? For instance, you can press A to make a beating heart; press B to make a shrink square.

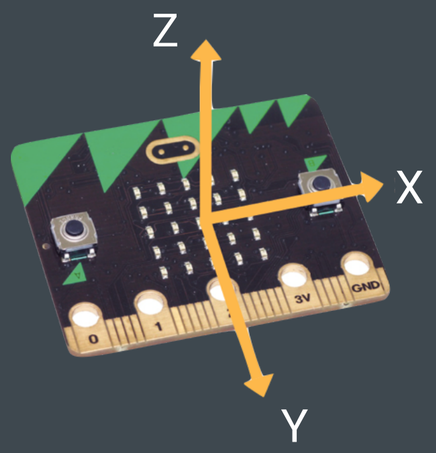


Floating Sand

Use accelerometer to measure the movement of Micro:Bit.

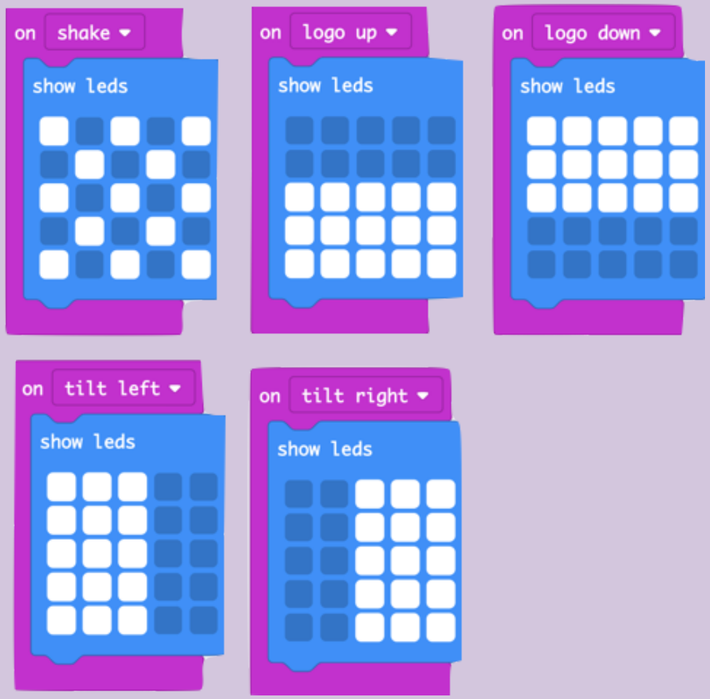
Requirements:

Shake Micro:Bit - LEDs lighten separately;
Tilt to the left - Left side of LEDs lighten;
Tilt to the right - Right side of LEDs lighten;
Tilt forwards - the upper side of LEDs lighten;
Tilt backwards - bottom of LEDs lighten.



Solutions:

Input blocks could help Micro:Bit sensor movement precisely.



Challenge:

Add more gestures to see what are differences among all triggering gestures.



@CoraEdTech

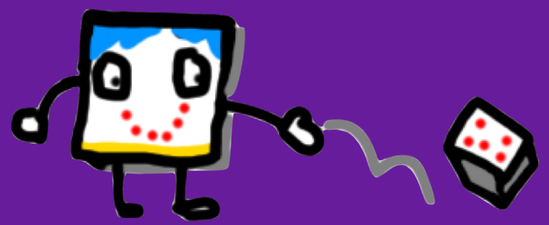


Roll Dice

Shake your Micro:Bit to show a random LED picture.

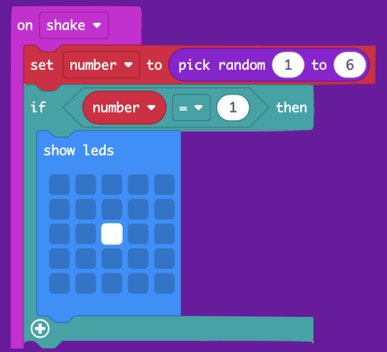
Requirements:

Shake your Micro:Bit - on the LEDs, it shows a random LED picture. It can be one bright LED one, or two, or three... Exact the same like dice we usually play!

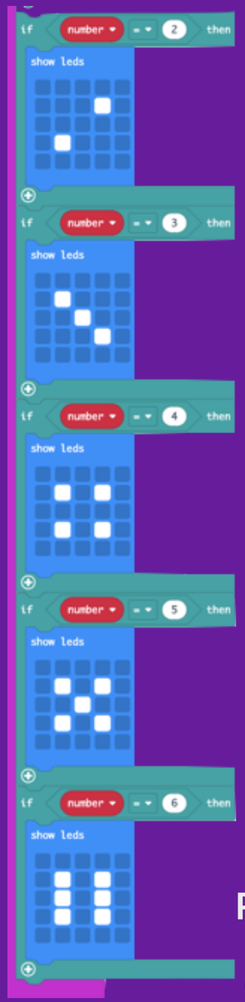


Solutions:

Use "on shake" block to trigger the Micro:Bit.
Set a variable called "number" and pick a random number between 1 to 6 (in math block list).
Set different LED pictures when the number comes as different results.



Picture 1



Picture 2

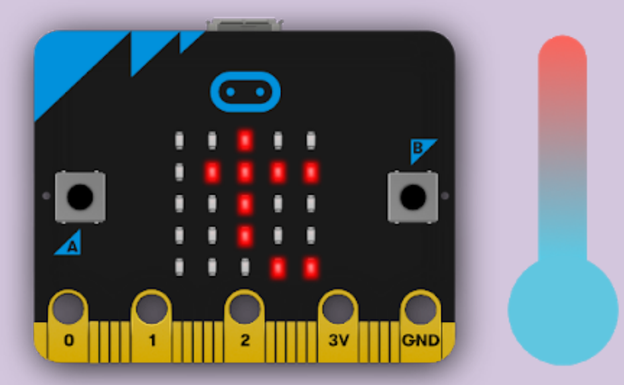
@CoraEdTech

Thermometer

Use temperature sensor to sense the temperature and make it shown on LEDs.

Requirements:

Use temperature sensor to sense the room temperature. The digit will show on LEDs.



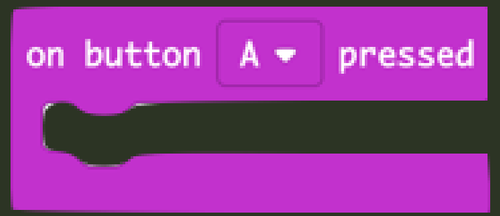
Solutions:

Use input block "temperature" to make your Micro:Bit sense the temperature.
"Show string" block can put the digit to LEDs.



Challenge:

Can you use buttons to make the Micro:Bit switch the number of temperature to other pictures; and switch it back?



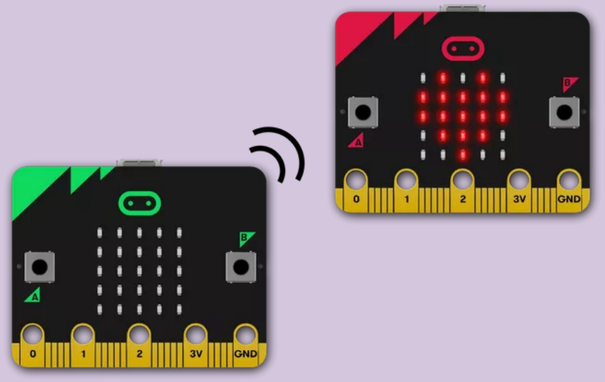
@CoraEdTech

Talk To Me

Use radio to communicate with each other - work in pairs.

Requirements:

When button A on Micro:Bit A is pressed, Micro:Bit B will receive the signal and show "A" on LEDs;
When button B is pressed, Micro:Bit B will receive another signal and show "B" on its LEDs.



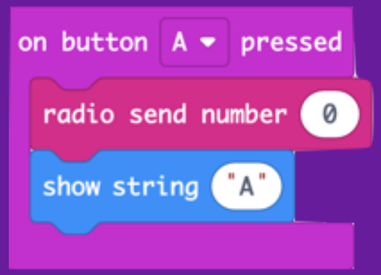
Solutions:

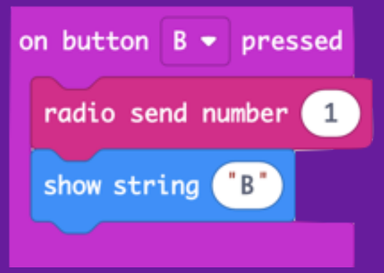
Use "radio set group" to make a program have the group ID you tell it for sending and receiving with radio.

Micro:Bit A:



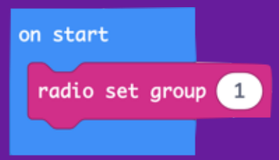
The radio group number should be the same for both Micro:Bits.

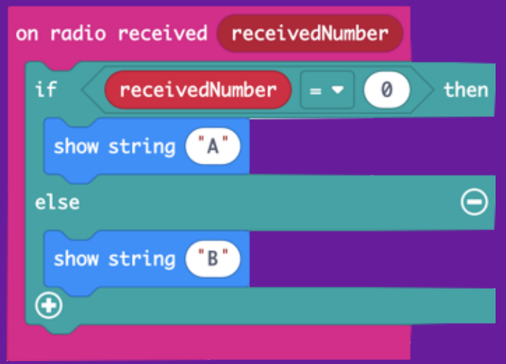




When it receives signal 0, Micro:Bit B will show string A; otherwise, it will show B.

Micro:Bit B:





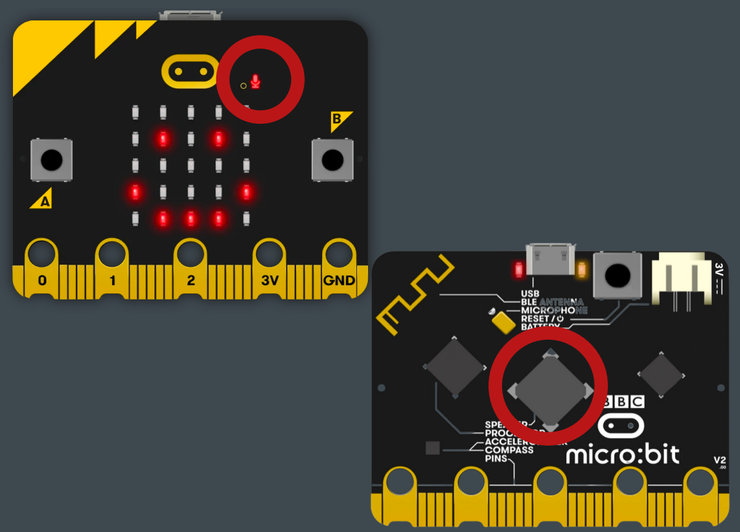
@CoraEdTech

Play Lullaby

Use sound detector and buzzer to play lullaby when it's quiet.

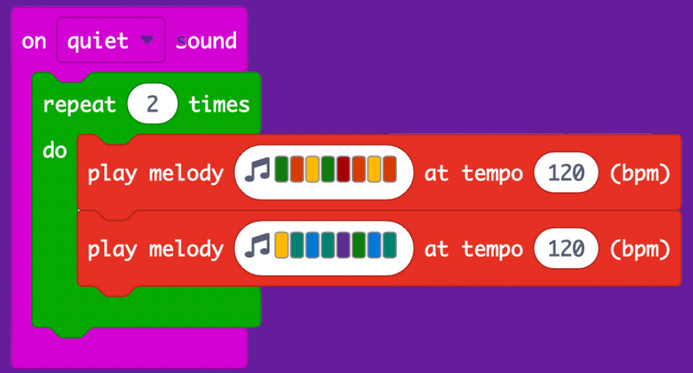
Requirements:

- Use the sound detector and buzzer to compose your own lullaby when the surrounding is quiet.



Solutions:

Use "on quiet" blocks to sense the surroundings and play music. (you can compose the music by yourself)



You can choose sound effect with this block:



Challenges:

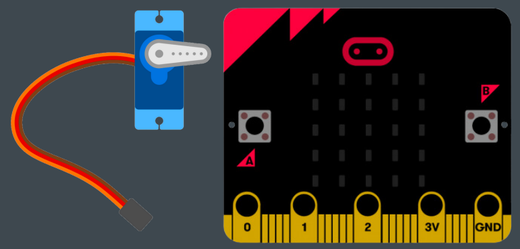
- Use buttons to control music playing.
- Show players the name of the music.
- Compose your own music.



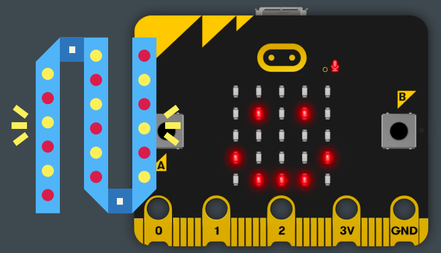
@CoraEdTech

Using Add-on Components with Micro:Bit

We'll dive in Micro:Bit by using add-on components.



Make It Move



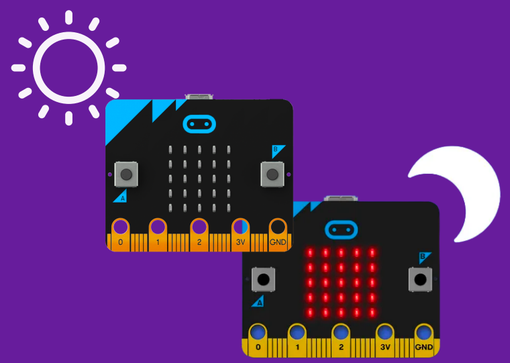
Glow with Flow

Make It Move

Attach servo to Micro:Bit to make the movement as an output.

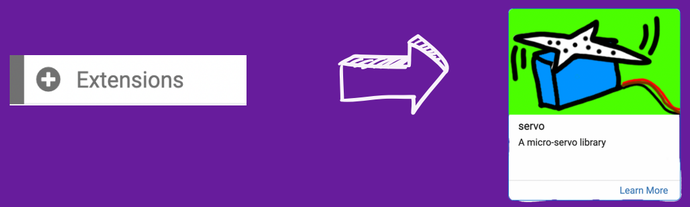
Requirements:

Make the servo move when the environment is bright. Cover your Micro:Bit to mimic the nighttime, so the servo will 'go to bed' and stop moving.

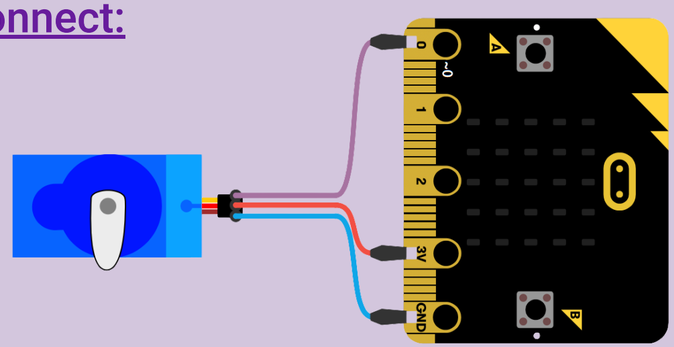


Add Extension:

The servo blocks are not listed on the default categories. You need to go to Extensions and add the servo blocks.



Connect:



Solutions:

In this solution, we use the conditional block to set the brightness condition to trigger the servo.



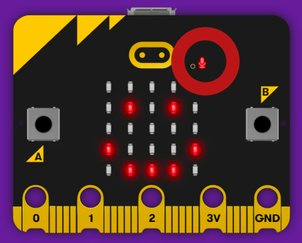
Use pause blocks to give servo time to move around.

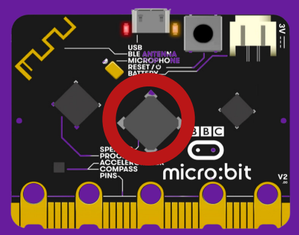
Glow With Flow

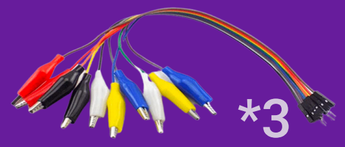
Use the sound sensor to control the Neopixel.

Requirements:

You need a Micro:Bit V2, a neopixel RGB LED strip, and three alligator clips to male jumper wires.



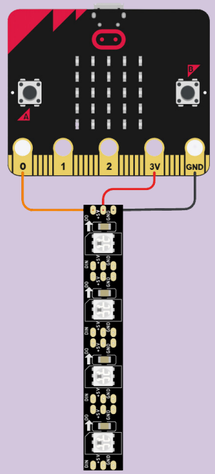




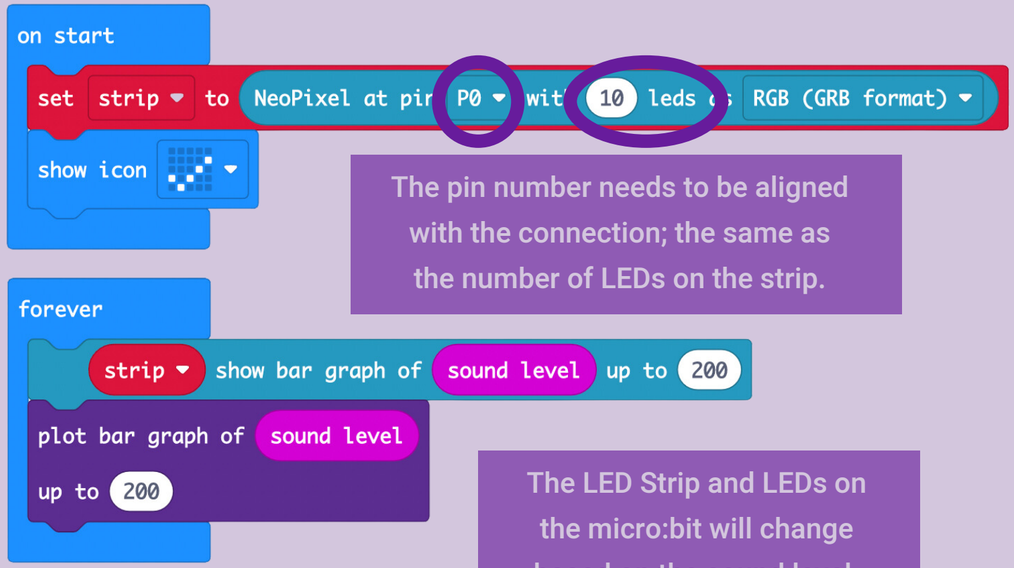


Solutions:

As Micro:Bit V2 has the sound sensor, we can use the sensor to control the Neopixel.



The NeoPixel strip is connected with Pin0.



The pin number needs to be aligned with the connection; the same as the number of LEDs on the strip.

The LED Strip and LEDs on the micro:bit will change based on the sound level.

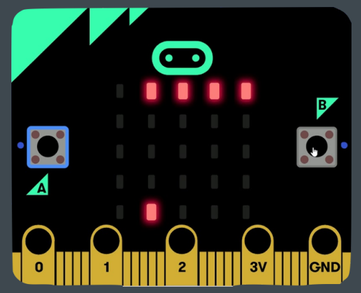
@CoraEdTech

Extra Activities:

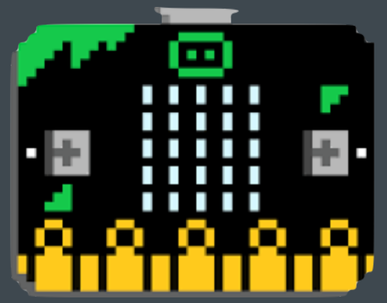


Turn Micro:Bit into Gameboy

Turn your Micro:Bit into gameboy - Let's code on it to make interesting games on this little chip!



Dodgeball



Shooting Game

Space Invaders

Program on Micro:Bit to create the fun Space Invaders game!

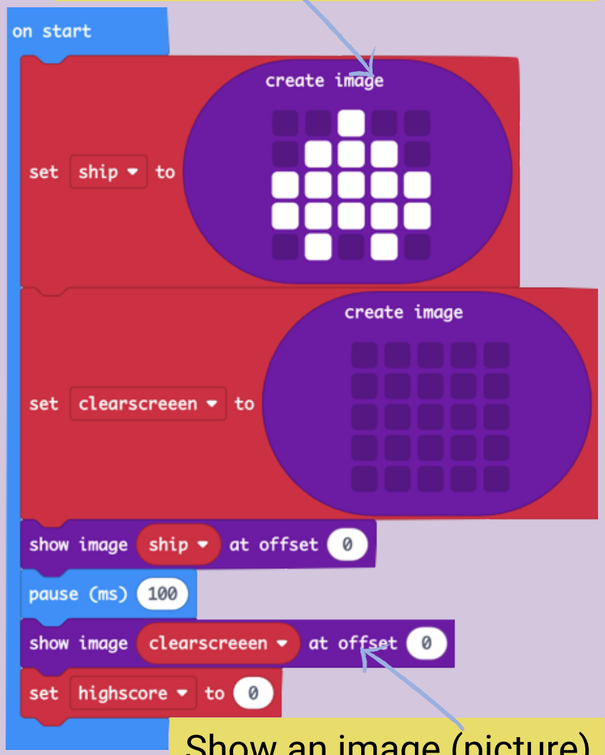
Requirements:

When you see the spaceship appears on the screen, press the button A to catch it. You have 5 opportunities to capture it. Let's see how brave you are at protecting your planet!

Variables:

- clearscreeen
- delay
- highscore
- onscreen
- score
- ship

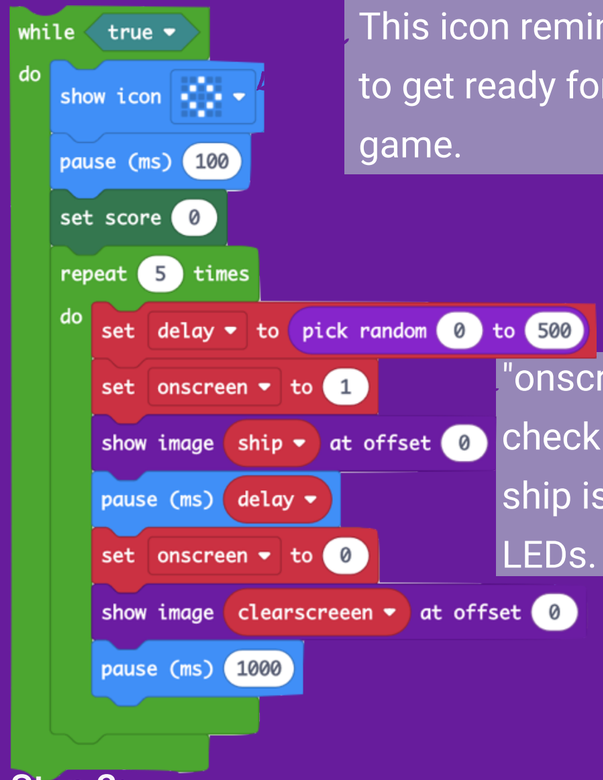
Setting a variable with an image is because we will use it for further coding.



Show an image (picture) on the LED screen.

Step 1:

Set up the game in the beginning, including show the game image.



This icon reminds players to get ready for the game.

"onscreen" is to check if your ship is shown on LEDs.

Step 2:

Set your spaceship to show on LEDs randomly within 0 to 500 milliseconds. Make it to repeat 5 times.

@CoraEdTech

Space Invaders

Program on Micro:Bit to create the fun Space Invaders game!

Step 3:

Each round you play will have a score - if the score is higher than previous rounds, you set yourself a higher goal.

If you couldn't reach to your previous high score, you have to restart it again.



Once you lose the game, the highscore will be set as 0; which means the game will start over again.

Onscreen is the block to sense your captures.



Step 4:

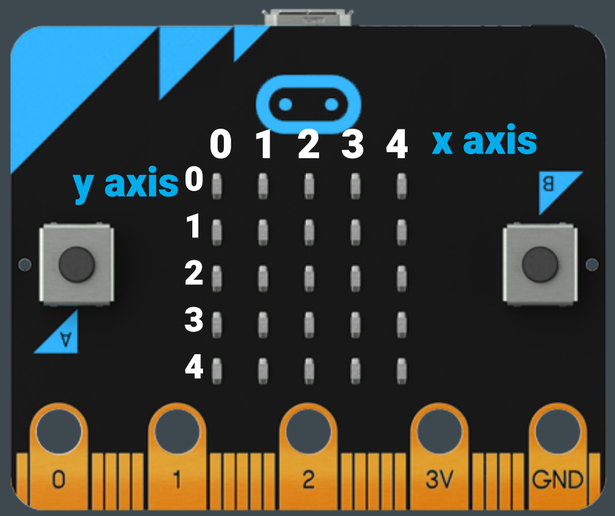
When button A is pressed, your Micro:Bit will detect if you capture the space ship. If so, you'll score one; otherwise, you'll get a sad face.

Shooting Game

Use bullets to shoot your goals - let's see how many scores you can get!

Requirements:

Press buttons A, B to control the movement of your sprite.
Press button A+B to shoot. The game last for 10 seconds.
You'll see your final score once the game finishes.



x&y axis

Variables:

- Bullet
- Enemy
- Sprite

Step 1:

This is the setup of this game. The game lasts for 10 seconds. In the beginning, we put the sprite in the middle-bottom of LEDs.
Enemies always show up at the top of LEDs.
When the game starts, enemy shows up randomly.

```
on start
    start countdown (ms) 10000
    set score 0
    set Sprite to create sprite at x: 2 y: 4
    set Enemy to create sprite at x: pick random 0 to 4 y: 0
    show icon
    while is running
    do pause (ms) pick random 0 to 2000
       Enemy set x to pick random 0 to 4
    show number score
```

Set the score as 0 in the beginning.

Use a Boolean condition as long as "game is running".

Show the final score once the game finishes.

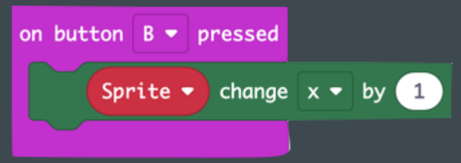
@CoraEdTech

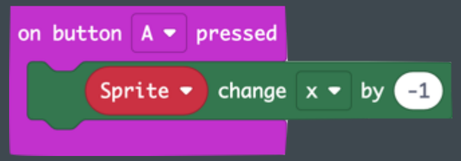
Shooting Game

Use bullets to shoot your goals - let's see how many scores you can get!

Step 2:

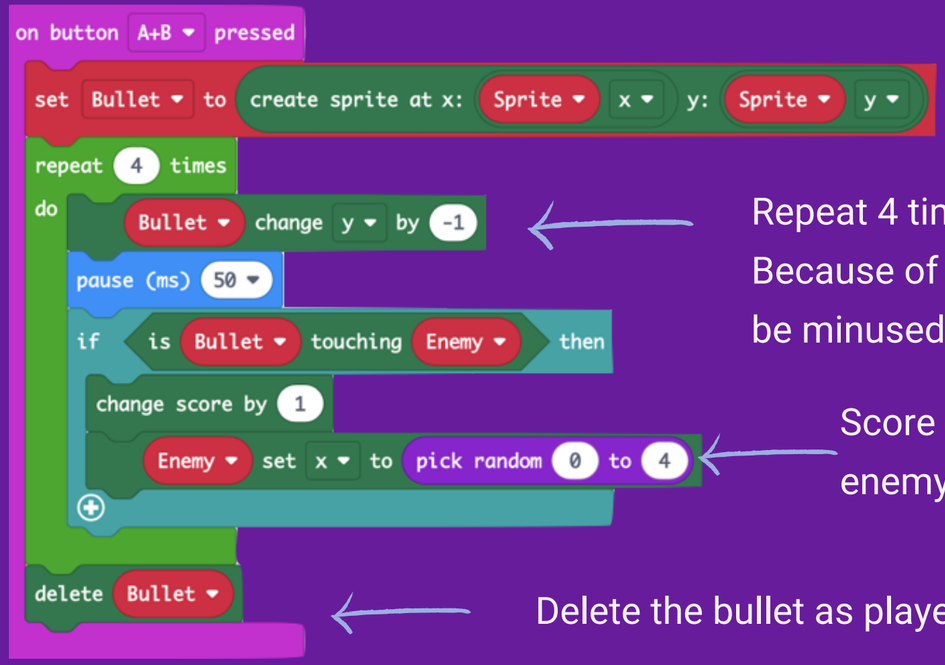
Use buttons A and B to control the movement of sprite (it can be moved side to side).





Step 3:

When button A+B is pressed, the sprite will shoot bullets. As the y axis has 5 grids, it needs to move up 4 steps to shoot enemies. If the bullet touches enemy, players will score one.



The bullet always appears as the same x axis like the sprite.

Repeat 4 times until the bullet reaches to the top of LEDs. Because of the y axis setting on Micro:Bit, the y needs to be minused when bullet moves up.

Score one if the bullet touches the enemy, also the enemy will move to another place randomly.

Delete the bullet as players can shoot a new one.