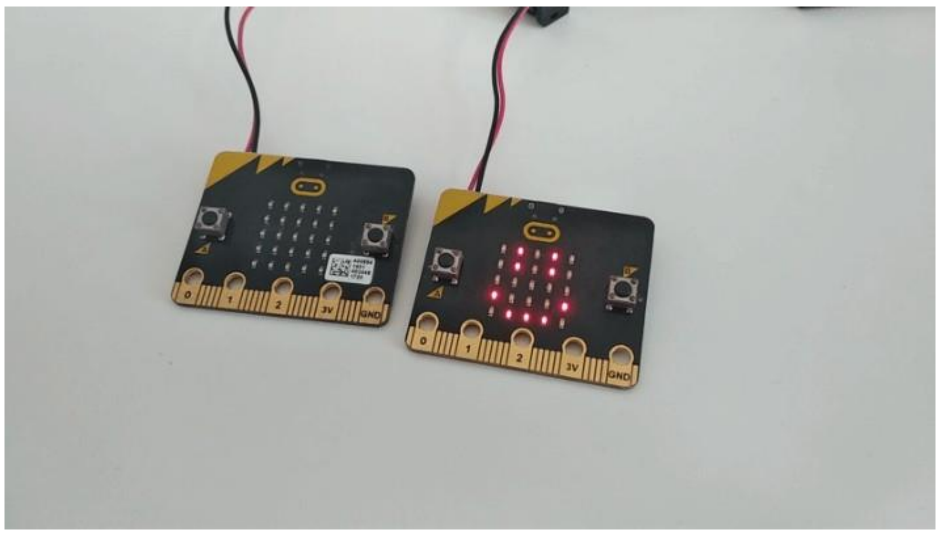
BUTTONS ON MICRO:BIT

PROGRAMMING OF AN ANIMACION



In this lesson, you will learn about the buttons on the BBC micro:bit – how to use them in your own program and how to find out if they were pressed. We will then connect them with the animations on the display.

Material requirements:
BBC micro:bit, USB cable, battery for micro:bit, computer with an access to Internet

We are going to work in the online environment makecode.microbit.org.

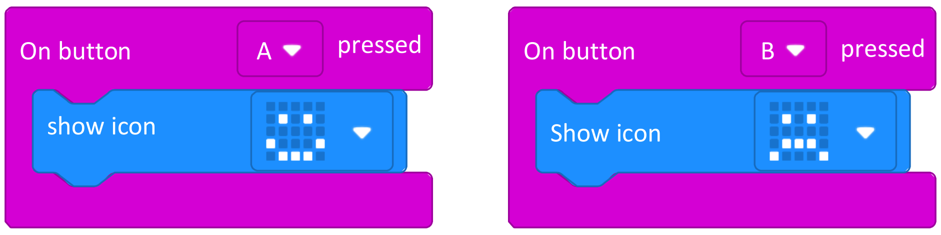
BUTTONS EVERYWHERE AROUND US

Buttons are one of the most common electronic components – they are on controllers, washing machines, game consoles, payment terminals...We use them to control electronics – whenever we want a microprocessor to do something (for example, send an elevator to our floor), we communicate through button.

Micro:bit has 2 buttons that we can use in our program (they are marked as buttons A and B).

Pressing one button

Capturing the pressing of a button is very simple on the micro:bit – just use the command on button A pressed from category Input. Try to program a simple program that will display a happy smiley when you press A and a sad smiley when you press B. First try the program in the simulator and only then upload it to the micro:bit.



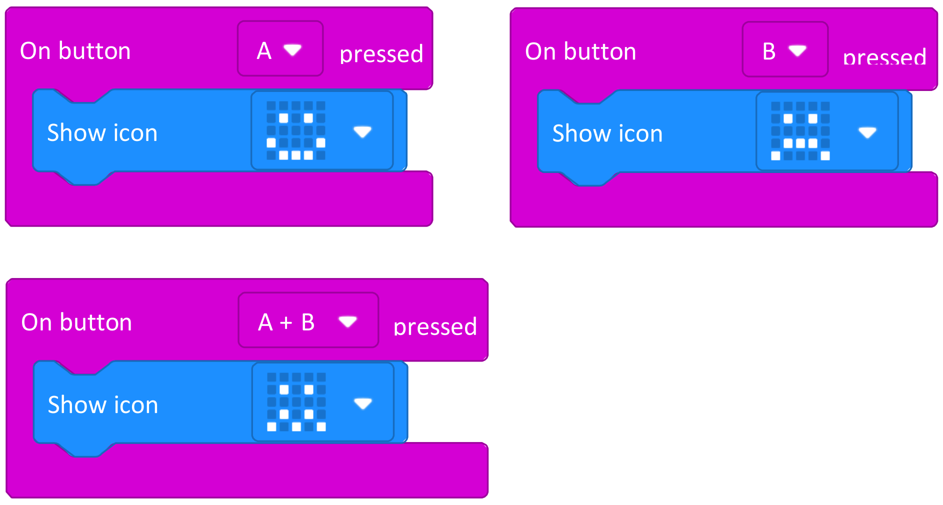
On button A pressed
show icon

On button B pressed
Show icon

Edit and download: http://makecode.microbit.org/_WotWzfe3edtk

Pressing both buttons

Let's try to add a third smiley to our simple program – this time confused. We will display this by pressing both buttons at the same time. But how do you test the program in the simulator? If you notice carefully, after adding the command on button A + B pressed a new white A + B button will appear in the simulator on the bottom of the right side of the micro:bit. Of course, you won't find it on a real micro:bit, it is only in the simulator, so you can try pressing both buttons at the same time.



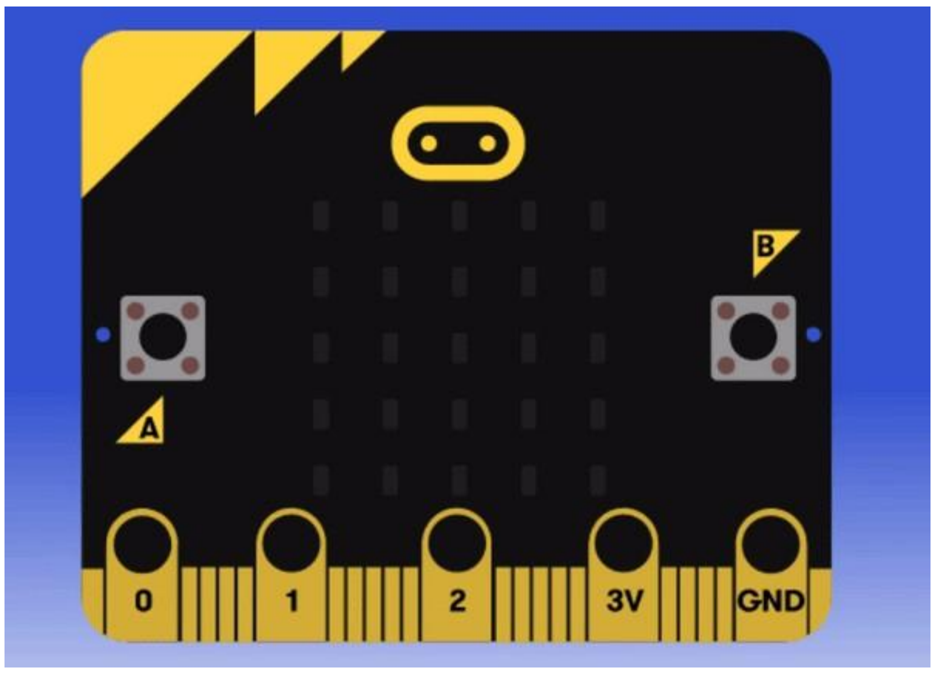
On button A pressed
Show icon

On button B pressed
Show icon

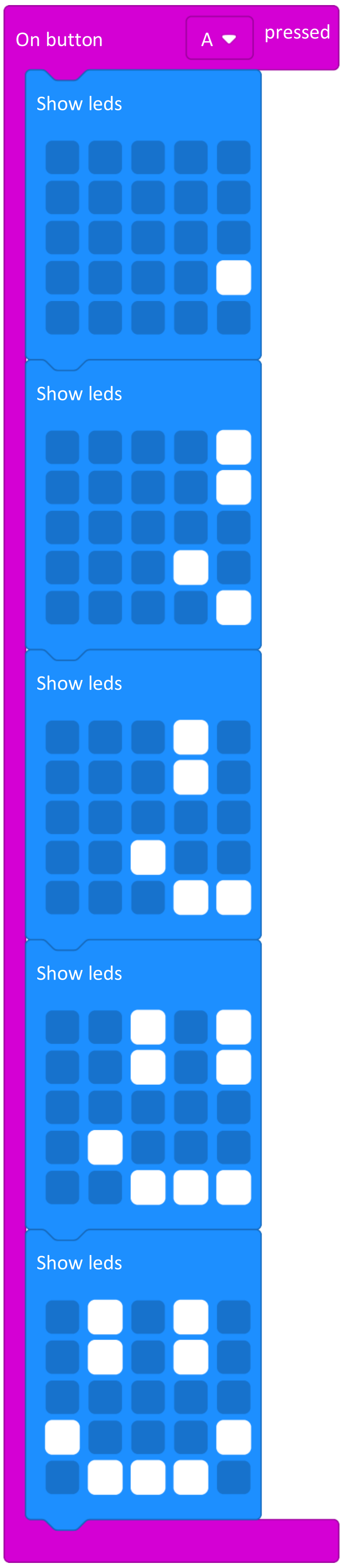
On button A + B pressed
Show icon

ANIMACION – SMILEY IS COMING AND LEAVING

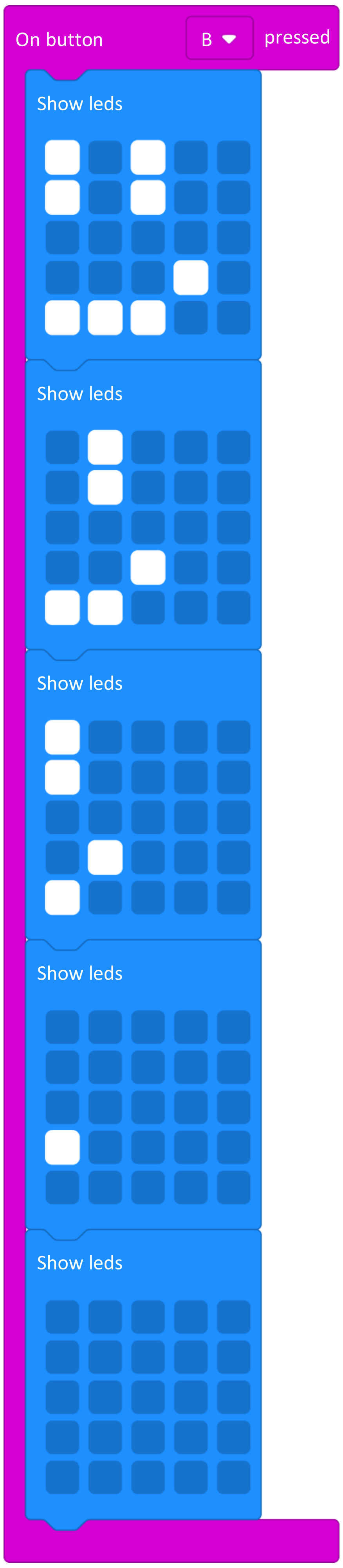
Until now, we simply displayed images without any animation. But now we will change that – we will create an animation of a smiley, which comes to us and then leaves.



We want to achieve that when we press the A button, the smiley gradually moves from the right side until we see entire smiley. When we then press the B button, the smiley will move further to the left and gradually leave. For each frame of the animation, we will use a separate command show leds.



```
On button A pressed
  Show leds
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ ■
    □ □ □ □ □
  Show leds
    □ □ □ □ ■
    □ □ □ □ ■
    □ □ □ □ □
    □ □ □ ■ □
    □ □ □ □ ■
  Show leds
    □ □ □ ■ □
    □ □ □ ■ □
    □ □ □ □ □
    □ □ ■ □ □
    □ □ □ ■ ■
  Show leds
    □ □ ■ □ ■
    □ □ ■ □ ■
    □ □ □ □ □
    □ ■ □ □ □
    □ □ ■ ■ ■
  Show leds
    □ ■ □ ■ □
    □ ■ □ ■ □
    □ □ □ □ □
    ■ □ □ □ ■
    □ ■ ■ ■ □
```

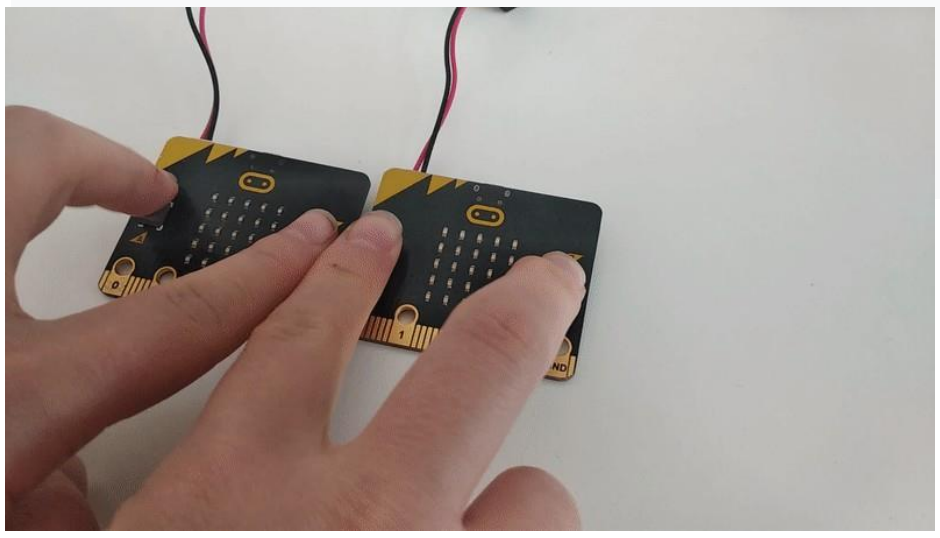


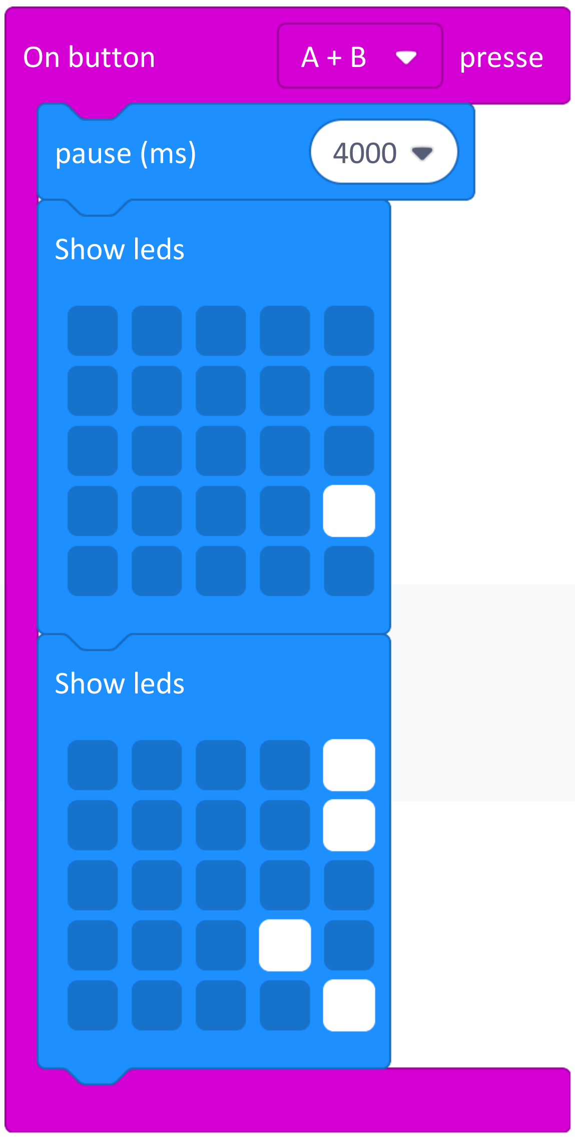
```
On button B pressed
  Show leds
    ■ □ ■ □ □
    ■ □ ■ □ □
    □ □ □ □ □
    □ □ □ ■ □
    ■ ■ ■ □ □
  Show leds
    □ ■ □ □ □
    □ ■ □ □ □
    □ □ □ □ □
    □ □ ■ □ □
    ■ ■ □ □ □
  Show leds
    ■ □ □ □ □
    ■ □ □ □ □
    □ □ □ □ □
    □ ■ □ □ □
    ■ □ □ □ □
  Show leds
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
    ■ □ □ □ □
    □ □ □ □ □
  Show leds
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
```

ANIMACION ACROSS MICRO:BIT

You will need 2 micro:bits for this activity. If you only have one, connect with a friend who also has micro:bit and try to program the animation together.

We have a nicely animated smiley, it comes and goes on the display, the next step is to animate it across several micro:bits. It will gradually appear on one micro:bit and then move on to the other. So we will create two animations, on one micro:bit it will start immediately after pressing both buttons at the same time and on the other micro:bit it will take place few seconds later. But how to determine after how many seconds later the smiley should be displayed on the second micro:bit? The easiest way is to measure the time with a stopwatch and then use the pause (ms) command at the beginning of the animation for the second micro:bit.





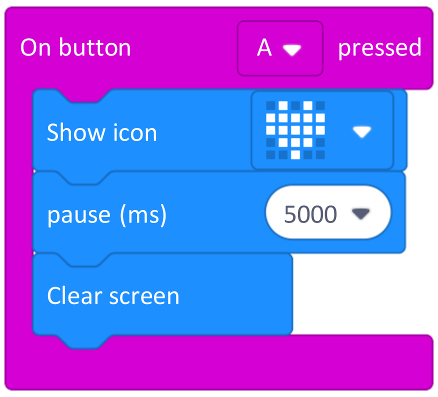
```
On button  A + B ▾  presse
  pause (ms)  4000 ▾
  Show leds
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ ■
    □ □ □ □ □
  Show leds
    □ □ □ □ ■
    □ □ □ □ ■
    □ □ □ □ □
    □ □ □ ■ □
    □ □ □ □ ■
```

Upload an animation without pause to one micro:bit and to the second one with pause before an animation.

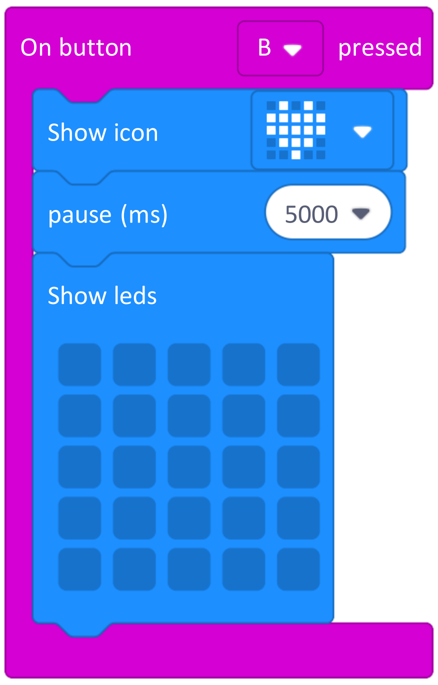
HEART FLASH

The last task is to create a “flash“of heart. – after pressing the button, the heart is displayed for 5 seconds and then disappears.

We already know how to display an image for 5 seconds, but how to clear the screen? There are two ways to do this – either with the command show leds, which we leave blank, or with the command Clear screen from the Basic category.



```
On button  A ▾  pressed
  Show icon  ♥ ▾
  pause (ms)  5000 ▾
  Clear screen
```



```
On button  B ▾  pressed
  Show icon  ♥ ▾
  pause (ms)  5000 ▾
  Show leds
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
    □ □ □ □ □
```

LESSON SUMMARY

In this lesson, we learned how to use the micro:bit buttons in our programs. We will use them often as the buttons are one of the most used electronic components. However, this is not the only way we can tell the micro:bit to do something – in the next lesson we will look at how to control the micro:bit using motion.