

The Hardware/Software Connection

OVERVIEW

Activity Overview: Students will get an introduction to programming a physical micro:bit. The micro:bit can only hold one program at a time so students will program the LEDs to flash something. They will learn about loops, pauses, and plotting.

Target Grade Level: K-2,3-5

Target Subject: Computer Science, Hardware/Software

Notes for Educators: Start with the built in show icon block so that students can see premade icons before creating their own.

Materials:

Website: <https://makecode.microbit.org/>

Micro:bit

Device with web browser

[OVERVIEW](#)

[PRE-TEST](#)

[OBJECTIVES](#)

[CATCH/HOOK](#)

[ACTIVITY INSTRUCTIONS](#)

[REVIEW](#)

[POST-TEST](#)

[STANDARDS](#)

[CROSS-DISCIPLINARY](#)

[COMPUTER SCIENCE](#)

[Domains:](#)

[Practices:](#)

[Specific CS Standards:](#)

PRE-TEST

1. What is a Micro:bit?
2. How does a pause block work?
3. What is the difference between a finite and infinite loop?

OBJECTIVES

Identify the difference between the finite loop and the infinite loop.

Identify where to place a pause block in the code.

Create a program that controls the LEDs on the micro:bit.

CATCH/HOOK

What if you could have a mini billboard in your pocket? Let's learn about the micro:bit?

<https://www.youtube.com/watch?v=Wuza5WXiMkc>

ACTIVITY INSTRUCTIONS

Students will learn that the micro:bit is a piece of hardware. Show a picture of the micro:bit that shows all the integrated components. In order for the LEDs to light, the students will need to write a program (software) that controls the lights of the LEDs. Students will watch: <https://www.youtube.com/watch?v=qqBmvHD5bCw>

to learn about LEDs on the micro:bit.

Students will then be introduced to the micro:bit programming interface. They will create a program that allows an icon to blink a certain number of times and another icon that blinks a certain number of times. This will reinforce the concept of a finite loop. They will then place inside the forever loop so that the pattern will continue indefinitely.

Next students will create their own icon(s) using the blank show LEDs block and place into the different loops to create their own design. Then students will be introduced to plotting the LEDs, if they've finished creating their own icons.

Finish with <https://www.youtube.com/watch?v=0GpF9z1rKBM>

To relate the portable billboard in their pocket with digital billboards, made of LEDs.

REVIEW

Students will show off their programs on the micro:bit to each other. They will talk about their programs and the reason they chose to design them the way they did.

Together as a group, have a discussion about what they enjoyed, what was a challenge and what they would do differently next time. Also ask what they would like to learn more about in the future.



POST-TEST

1. What is a Micro:bit?
2. How does a pause block work?
3. What is the difference between a finite and infinite loop?

STANDARDS

COMPUTER SCIENCE

Domains and Specific Standards:

Computing Systems, Algorithms & Programming

PRIORITY

5.CS.HS.01 Demonstrate and describe the function of common components of computing systems (hardware and software) (e.g., use a browser, search engine).

PRIORITY

5.AP.A.01 Using grade appropriate content and complexity, compare and refine multiple algorithms for the same task and determine which is the most appropriate.

PRIORITY

5.AP.C.01 Using grade appropriate content and complexity, create programs that include sequences, events, loops, and conditionals, both individually and collaboratively.

SUPPORTING

5.AP.M.01 Using grade appropriate content and complexity, decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process.

SUPPORTING

5.AP.M.02 Using grade appropriate content and complexity, modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.

PRIORITY

5.AP.PD.01 Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences.

SUPPORTING

5.AP.PD.04 Using grade appropriate content and complexity, describe choices made during program development using code comments, presentations, and demonstrations.

Practices:

Practice 3.3 Recognizing and Defining Computational Problems

Practice 6.3 Testing and Refining Computational Artifacts

Practice 5.2 Creating Computational Artifacts

Practice 3.2 Recognizing and Defining Computational Problems



Practice 5.3 Creating Computational Artifacts

Practice 6.2 Testing and Refining Computational Artifacts

Practice 7.2 Communicating About Computing

Practice 7.2 Communicating About Computing

