

Coding and Using Sensors for Measurement

Students will apply the computational concept of variables in order to code and use the temperature sensor on the micro:bit. At this time, they will also learn how to set up and use the class sensor array.

Suggested Timing: Science 10 - 20 min., Coding 20 - 40 min.

Success Criteria:

Students will:

- Accurately and safely use materials and equipment
- Code a sensor (thermometer) in a block-based coding language
- Successfully set up and code the class sensor array

Student Prior Skills and Knowledge:

- Experience with the MakeCode environment and with uploading hex files to the micro:bit.

Materials:

- Sensor Array Set-up Instructions** (for micro:bits) [[.pdf](#)]
- Minds-On 3 Guided Materials** [[.pdf](#)]
- Guided Materials to Code the Sensor Array: Level 1** [[.pdf](#)]
- Guided Materials to Code the Sensor Array: Level 2** [[.pdf](#)] (Optional)
- BLM M3.1: Calibration** [[.doc](#)] [[.pdf](#)] [[fillable .pdf](#)] (Optional) - 1 per student
- BLM M3.2: Calibrating a Temperature Sensor** [[.doc](#)] [[.pdf](#)] [[fillable .pdf](#)] (Optional) - 1 per small group
- Guided Materials Challenge 1: Broadcast Temperature Send and Receive** [[.pdf](#)] (Optional)
- Guided Materials Challenge 2: Display Temperature** [[.pdf](#)] (Optional)
- micro:bits (with USB cords) - 1 per 2/3 students
- Computers – PCs, chromebooks, laptops, iPads, tablets
- Access to internet and MakeCode website (<https://makecode.microbit.org/>) **please note that look of the makecode interface has changed, but the function of the blocks has not changed.*
- Data projector and screen
- Class sensor array

Computational Thinking

Concepts:

- Inputs (temp.)
- Outputs (integers, strings, LED display)
- Sequences
- Variables
- Data

Practices:

- Algorithmic thinking
- Pattern recognition
- Decomposition
- Abstraction
- Data Collection & Analysis

Approaches:

- Tinkering
- Reusing & Remixing



Preparation:

- ❑ Sign out computer lab, laptops, chromebooks, etc., ideally one per 2-3 students.
- ❑ For Part M3.2 and M3.3, students will need the micro:bits and Guided Materials (either on paper or online).
- ❑ You may wish to have a data projector and screen set up for Part M3.2 and for Part M3.3.

Implementation Options:

- There are several options for setting up and coding the main class sensor array (Part **M3.3**). For whichever option you select, the person/people setting up the array will need access to the Device-specific Sensor Array Assembly and Programming Instructions. Instructions for using a micro:bit are included [here](#). The set-up suggestions below are ordered from least to most class time.
 1. You could wire up the devices and program the array yourself outside of class time.
 2. You could have a small group of tech-savvy students set up and program the sensor array.
 3. You could have students watch as you set-up and code the sensor array.
 4. You could have students code and upload the sensor array program to their own micro:bits, and then take turns testing it with the class sensor array to see if it functions as intended.

Minds-On 3.1: Temperature and the ISS

Educators

Students

Step 1 Explain that for this action project, the students will be exploring three aspects of their classroom environment and the space environment - temperature, relative humidity and carbon dioxide level.

Step 2 Have students predict, either in class or by answering the questions online, what they think the temperature is outside the ISS. Record their predictions along a number line drawn on a blackboard or whiteboard. After all students have predicted, explain that the side facing the Sun can warm to **121°C** while the side facing away from the Sun can cool to **-157°C**!

Answer the question:

- *What do you predict the temperature is outside the ISS?*

Have students share ideas as to why the temperatures on the different sides of the ISS would be that extreme. In space, there is no air, so heat cannot be transferred through **conduction** or **convection** (these concepts may need to be explained).

Did you know?

Every thermometer has two parts: a temperature sensor and a means of showing the temperature as a numerical value.

Answer the question:

- *Why do you think the temperatures in space are so extreme?*



Until they are set up to show numerical values, the class sensor array and the micro:bits are not working as thermometers. In the next section, students will program the devices so that they can function as thermometers.

Minds-On 3.2: Code a micro:bit Thermometer

Educators

Step 1

Have each small group of students (2-3) turn on and log into a computer/laptop/ chromebook and navigate to <https://makecode.microbit.org/> or open the MakeCode app.

Step 2

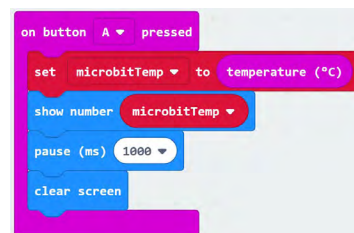
Show the **Minds-On 3 Guided Materials** [.pdf] on screen or have students follow the instructions or video walkthrough in the Student Module.

In this activity, students will be introduced to the Computational Thinking Concept of **Variables**.

Students

Turn on your device and log in. Navigate to <https://makecode.microbit.org/> or open the MakeCode app.

The completed code for this task should look something like this:





Minds-On 3.3: Setting up the Sensor Array

There are two parts to setting up the Class Sensor Array - the **hardware** (devices and wiring) set-up the **software** (coding) set-up.

Time permitting, have all students write the code for the class sensor array and then upload it to their own micro:bits. If time is not available, you could do it yourself or delegate the coding to one or more students.

Students may choose to code the class sensor board on MakeCode using either Level 1 code (easier) or Level 2 code (more challenging).

You may choose to code the class sensor board on MakeCode using either Level 1 code (easier) or Level 2 code (more challenging) coding.

Hardware

The Let's Talk Science custom sensor board attaches to a micro:bit using the alligator clip jumper leads provided in the kit.

Either yourself, or students, can attach the custom sensor array board to a micro:bit. See the [Sensor Array Set-up Instructions](#) (for micro:bits) for more details.

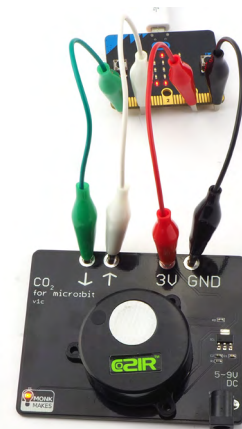
If students are setting up the hardware, review the micro:bit safety instructions with them. Check to make sure that the leads are correctly attached and that a suitable power supply is being used.

Once the code has been uploaded to the micro:bit, it should be ready to take readings and display values for temperature, relative humidity and carbon dioxide.

If you will be the person connecting the custom sensor array board to a micro:bit, review the [micro:bit safety instructions](#) with your teacher.

Connect the custom sensor board to a micro:bit using the alligator clip jumper leads provided. Connect the sensor board to an appropriate power source.

micro:bit		Sensor Board
0	→	In
1	←	Out
2	(no connection)	
3V	+	3V
GND	-	GND



Upload the sensor array code to the micro:bit. The LED display should now begin to display values for temperature, relative humidity and carbon dioxide.

Level 1

Level 1 - Data is displayed when buttons are pushed (Button A - CO₂, Button B - Temperature, Buttons A+B - Relative humidity). See the [.pdf](#) or the student instructions for more details.

Level 1 - Data is displayed when buttons are pushed (Button A - CO₂, Button B - Temperature, Buttons A+B - Relative humidity). Instructions for this are available as:

- Step-by-Step guided material online
- A [.pdf](#) that can be printed
- A [video walkthrough](#)

Level 2

Level 2 - Data is displayed continuously as a scrolling set of values with units. See the [.pdf](#) or the student instructions for more details.

Level 2 - Data is displayed continuously as a scrolling set of values with units. Instructions for this are available as:

- Step-by-Step guided material online
- A [.pdf](#) that can be printed
- A [video walkthrough](#)

Minds-On 3.4: Extensions

Science

- Have students look around the room for other measuring devices. What kinds are there?
- Discuss with students the advantages and disadvantages of using a digital thermometer, like the micro:bit, to a traditional alcohol thermometer.
- Hand out **BLM M3.1** [[.doc](#)] [[.pdf](#)] and have students read the top two paragraphs silently to themselves or have them read the paragraphs out loud. Have students explain in their own words what “accurate,” “standard” and “calibrate” mean. With an elbow partner, have students answer the two questions at the bottom of the page. Have students share their responses. For temperature, they may use their prior knowledge to suggest using water at either its boiling point or freezing point. For humidity, they may or may not have ideas of their own.



Calibrating a Temperature Sensor

- Alcohol thermometer -1 per group
- Temperature sensor on sensor array - each group will need access
- Plastic cup - 1 per group (one empty and one full of water)
- Ice cubes - enough to fill one plastic cup per group
- Plastic spoons - 1 per group
- BLM M3.2:** Calibrating a Temperature Sensor - 1 per group [[.doc](#)] [[.pdf](#)] (Optional)

Preparation:

- Gather the required materials and prepare a set for each group.
 - Prepare several trays of ice cubes in advance or bring in a bag of ice.
 - Arrange students into small groups.
- Review the steps of the **Calibrating a Temperature Sensor** activity on **BLM M3.2** with the class. Distribute the materials. Ask students what they would call a series of steps like this. They may suggest words like “instructions” or “recipe.” Explain that a series of steps to be done can also be called an [algorithm](#).

Calibrating a Humidity Sensor

This activity is best done as a demonstration unless you have multiple humidity sensors.

- 125 mL (½ cup) measuring cup - 1
- Table salt - 60 mL (¼ cup)
- Water
- Large resealable bag
- Humidity sensor on sensor array

Explain what you are doing, out loud, to the students as you follow the steps.

1. Fill the measuring cup approximately halfway to the top with table salt (approximately ¼ cup).
2. Add water until the salt is damp and looks like wet sand. Do not put in too much water.
3. Carefully put the measuring cup with the salt/water into the resealable plastic bag.
4. Place the humidity sensor into the bag, away from the measuring cup. **Make sure that the salt/water does NOT get on the sensor.** Seal the bag around any cords or wires.
5. Place the bag somewhere that is out of direct sunlight and not near a heating or cooling vent as the temperature should be fairly constant. Leave the sensor array in the bag for 12 hours.
6. While still in the bag, take a humidity reading. It should read 75%. If it does not, it will need calibrating.

Note: to properly calibrate the humidity sensor, it needs to be undisturbed for a minimum of 12 hours.



Calibrating a Carbon Dioxide Sensor

This activity can be done as a demonstration with the class, or by the educator in the morning before class.

1. Prepare the sensor array to take readings.
2. The carbon dioxide level of a room first thing in the morning (before students come in) should be roughly similar to that of outside air (400 ppm).
3. Whatever value the carbon dioxide sensor shows (above or below 400 ppm), should either be added to or subtracted from future readings (e.g., if it was 410, then all future readings should subtract 10 ppm from the values).
4. The carbon dioxide sensor is now ready to take readings.

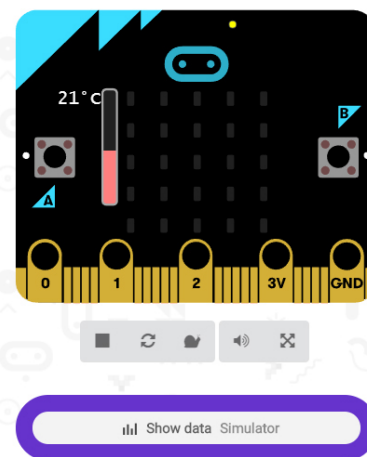
Coding

- If students only coded the Level 1 display for the class sensor array, have them try coding the Level 2 display. Discuss with students the advantages and disadvantages of each display method.
- Have students learn how to use the radio send and receive function of the micro:bit by sending temperature readings from one micro:bit to another (See **Challenge 1** in the Student Module).
- Have students learn how to display temperature values using a tens column and a ones column (See **Challenge 2** in the Student Module).
- The MakeCode simulator has a feature that lets you see streamed data. If students create the following code using blocks from the Basic, Input and Series categories:

```

forever
  serial write value "temp" = temperature (°C)
  
```

They will see the **Show data Simulator** button appear under the simulator. When they click the button, they will see the temperature values scrolled along the top and actual values listed below. When they change the temperature in the simulator, they will see the values change. These simulator values can even be exported using the download button in the top right-hand corner.



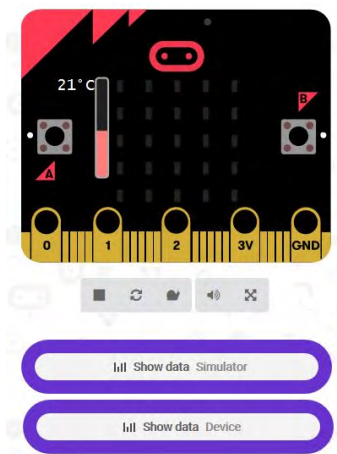


- The [MakeCode app for Windows 10](#) has an even better data streaming and logging feature. The app allows you to stream real time data from the micro:bit itself, rather than only from the simulator. For example, to stream micro:bit temperature data, students would create the following code using blocks from the Basic, Input and Series categories:

```

forever
  serial write value "temp" = temperature (°C)
  
```

They will see a **Show data Device** button appear below the Show data Simulator button. When they click the button, they will see the temperature values **of the micro:bit** scrolled along the top and actual values listed below. When they change the temperature of the micro:bit, such as by blowing on it, they will see real-time values appear on screen. The values can also be exported as a CSV file using the download button in the top right-hand corner.



- On the MakeCode website is a DIY project for setting up the micro:bit to take [soil moisture](#) readings. The page includes links for assembling the equipment, coding the micro:bit and related projects.

Minds-On 3.5: Additional Resources

Background Information

- <http://tomatosphere.letstalkscience.ca/Resources/library/ArticleId/4671/precision-and-accuracy.aspx> (Accessed March 27, 2019)
This backgrounder, on the Tomatosphere™ website, explains the difference between accuracy and precision.

Online Resources

- https://science.nasa.gov/science-news/science-at-nasa/2001/ast21mar_1 (Accessed March 27, 2019)
This web page, from the NASA website, explains how the ISS's thermal control system keeps astronauts safe and comfortable.
- <https://www.aicompanies.com/education/calibration/why-is-calibration-important/> (Accessed March 27, 2019)
This web page, from Advanced Instruments, explains why calibration is important.

- <https://www.co2meter.com/blogs/news/7512282-co2-sensor-calibration-what-you-need-to-know> (Accessed March 27, 2019)
This web page, from CO2Meter.com, discusses why and how to do carbon dioxide sensor calibration.
- https://www.rotronic.com/media/productattachments/files/b/e/beginners_guide_to_humidity_measurement_v0_1.pdf (Accessed March 27, 2019)
This PDF, *A beginner's guide to humidity measurement*, from the National Physical Laboratory in the United Kingdom, has a wealth of information about humidity and how it is measured.

