Sustainable Farms - Session 5

How could farming benefit from automation?

Context:
Learners create a program on the micro:bit which logs data about the environment for growing plants, specifically the temperature. The learners focus on how data like this can be used to provide information about the environmental conditions on a farm, to support farmers to make decisions, or automate parts of the farming processes - e.g. switching on irrigation systems during periods of warm weather.

Engineering focus:
Learners will be working as an engineer by improving solutions to farming problems with the use of technology.

Learning time:
1.5 hours

Suggested age group:
11-14 years old

Curriculum for Excellence links:

Third Level:
I can apply my knowledge and understanding of engineering disciplines and can develop/build solutions to given tasks. TCH 3-12a
I can select appropriate development tools to design, build, evaluate and refine computing solutions based on requirements TCH 3-15a
Fourth Level:
I can solve problems through the application of engineering principles and can discuss the impact engineering has on the world around me. TCH 4-12a
I can select appropriate development tools to design, build, evaluate and refine computing solutions to process and present information whilst making reasoned arguments to justify my decisions. TCH 4-15a

Keywords
automation
sensors
Micro:bit
program
code
debug
controlling systems
monitoring systems
environment
conditions
thermometer
data logger
temperature sensor

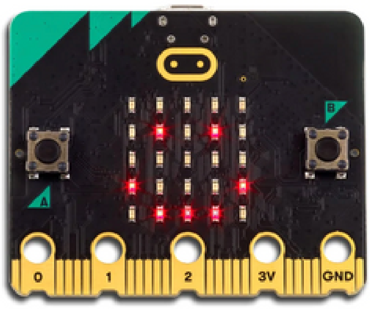
Resources:
- Note - this lesson uses features exclusive to the V2 micro:bits. However if you have V1 micro:bits, or indeed don't have micro:bits at all, you can still complete this lesson using the online simulator at https://makecode.microbit.org/
- computer with access to the internet for MakeCode
- micro:bits V2 (or micro:bit simulator at MakeCode

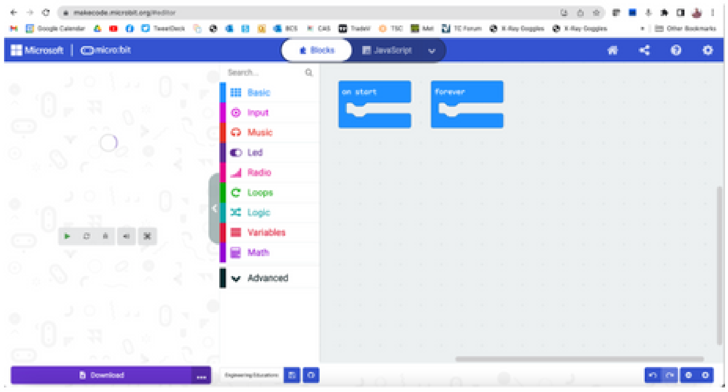
Suggested Activities

1

Display images of a micro:bit and the MakeCode editor, such as those shown below (slide 2). Lead a discussion with Learners:

- Have learners used the micro:bit before? If so, what did they create and how did they create this?
- Have learners used the MakeCode editor before? What does the MakeCode editor remind learners of? Why do they think this? (Make code looks and functions in a similar manner to Scratch - it is a graphical programming language)
- If learners haven't used MakeCode before, but have used Scratch, lead a discussion on what they have previously created in Scratch. How did they create and run these programs?





2

Setting the context:

Show the photo shown (slide 3). What do Learners think this is a photo of? Why do they think this? What might have happened to the plants in the photo? What could have caused this?

Lead a discussion to conclude this is a photo of plants (sunflowers) which have died as a result of not receiving enough water during hot weather. This could be due to extended periods of sunshine without rain.

3

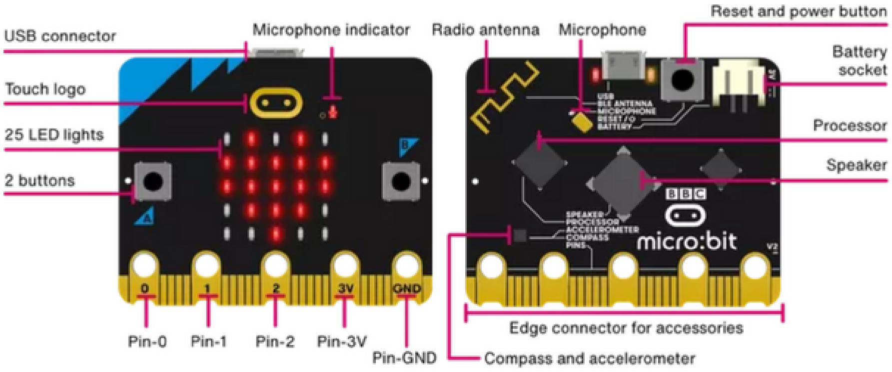
Introducing the challenge:

Explain that in this unit, learners will build a prototype digital device to automatically monitor temperatures on a field to help farmers decide if extra water is required and hopefully prevent crops from dying. Add that as an extension, learners will consider how to add additional code to automate the process of providing additional water.

Model the Micro:bit:

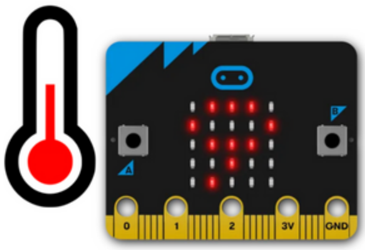
4

Explain to learners that we will be using the micro:bit to prototype our solutions. Add that we will first explore how these work by exploring some simple code (slide 4).



Hand the Micro:bits out to learners. Can learners find the thermometer on the micro:bit which we can use to measure temperature? Note - This is a trick question as learners won't be able to see it - it is located inside the processor!

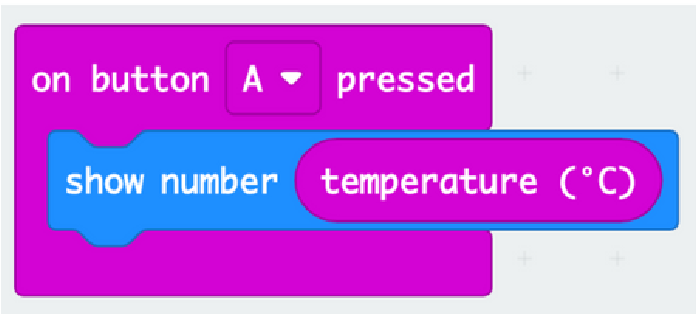
The micro:bit uses the digital thermometer to measure the temperature of the micro:bit's processor, but we can also use it to get a fairly accurate measurement of the temperature around the micro:bit (slide 5).



Code Exploration Part 1 - A simple temperature sensor

5

Explain to learners they will be adding the following code to their micro:bits. Before doing so, what do learners predict the code will do when run? Why do they think this? Can they talk through the code with a partner to help their predictions? (Slide 6)



Code Exploration Part 1 Cont.

6

Ask learners to add the code to their micro:bit and run it. Were their predictions correct? Did anything surprise them? When this code is run, pressing the button A will cause the micro:bit to display the temperature value it measures on the screen of LEDs.

Explore the following questions as a class:

- Could this simple device be used to keep track of the temperature on a field?
- What are the problems with this device so far?
- How could we improve it?

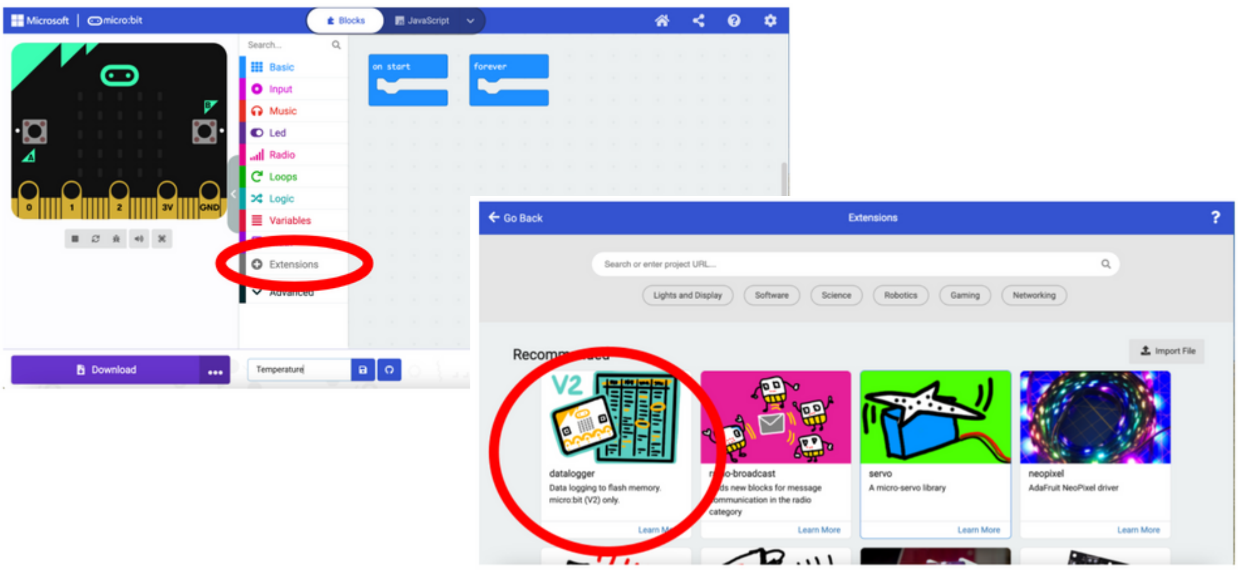
Lead a discussion to conclude that it would be impractical for someone to have to walk to temperature sensors across different fields to push the button and take a measurement - plus this would then have to be recorded somewhere!

We could improve the device if it automatically measured the temperature AND recorded this for us. Explain that the micro:bit is capable of this using its data logger extension which we will explore now.

Code Exploration Part 2 - Collecting data over time

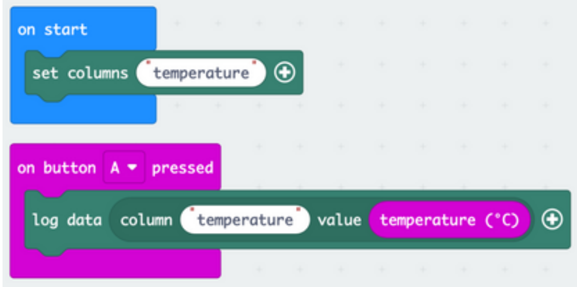
7

Show learners how to activate the data logger extension commands by selection 'extension' and 'data logger' (slide 8 and 9) as shown below:



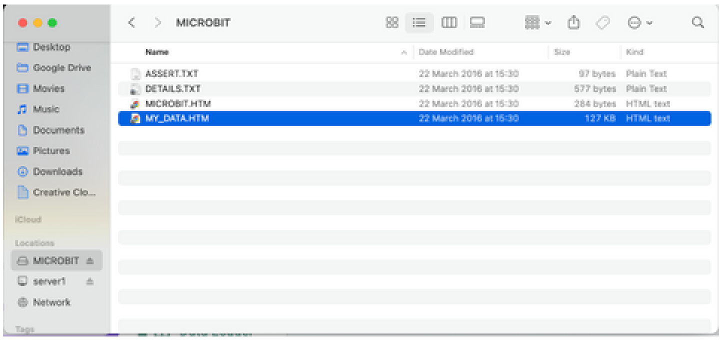
Code Exploration Part 2 cont.

Display the code below which uses a selection of the 'data logging' commands. Can Learners predict what the code will do when run? Why do they think this? Note - a full explanation of this code is shared in the following 'Explain' section (slide 9).



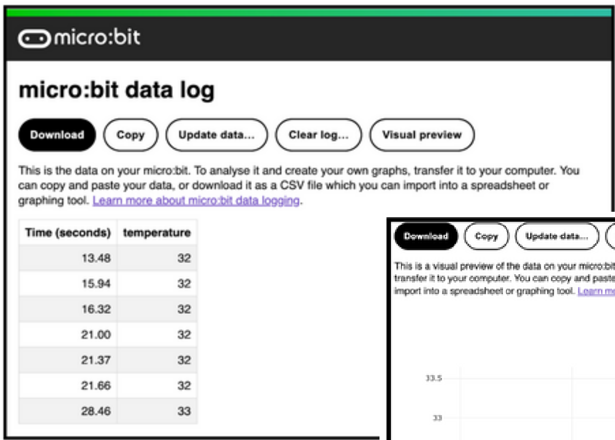
Ask learners to add the code to their micro:bits and then press button A several times.

7

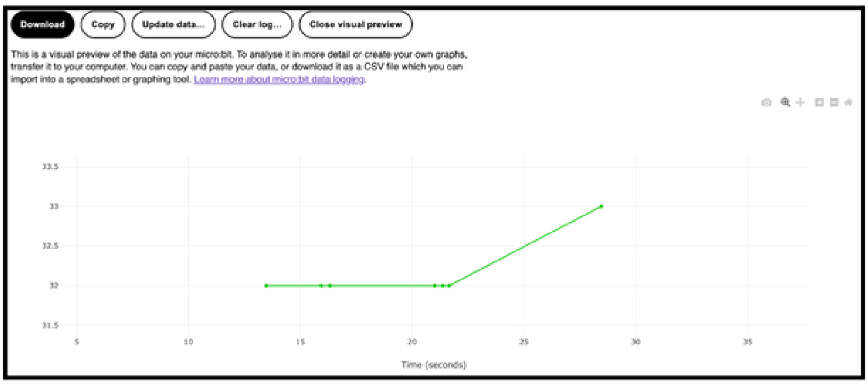


Ask learners to now reconnect their micro:bit to their computer, and click on 'My Data' (slide 10).

They will now see the temperature which was measured and stored every time they clicked button A, as shown below. The time stored relates to how long has elapsed after the program was loaded onto the micro:bit before the button is pressed (slide 11 and 12).



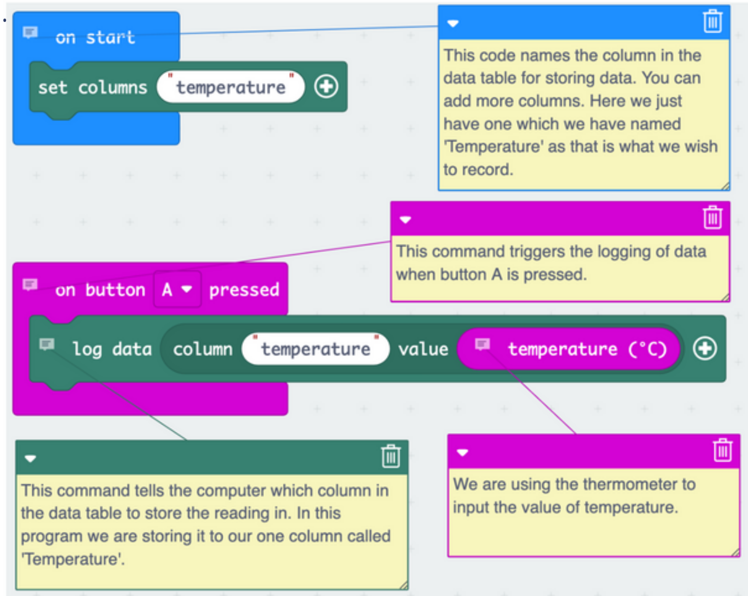
Time (seconds)	temperature
13.48	32
15.94	32
16.32	32
21.00	32
21.37	32
21.66	32
28.46	33



Summarise learning

8

We have now shown how the micro:bit can store temperature data for us, but so far our solution still requires us to manually press the button A to take a reading. Remind learners that our ideal solution would do this automatically for us, which is the challenge they are about to tackle (slide 13).
But first, if required, recap the function of each element of code in our last exploration to consolidate learners' understanding. Explanations appear below in the form of code comments.

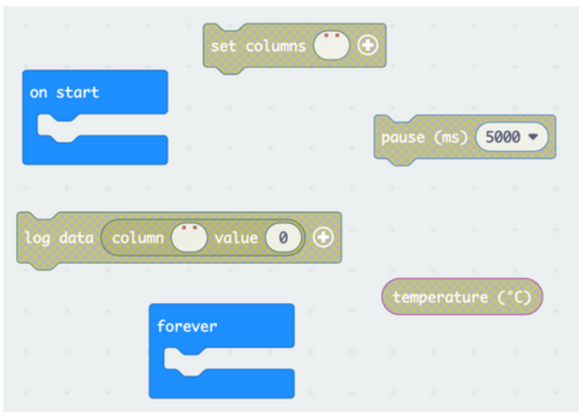


Parson's Problem

9

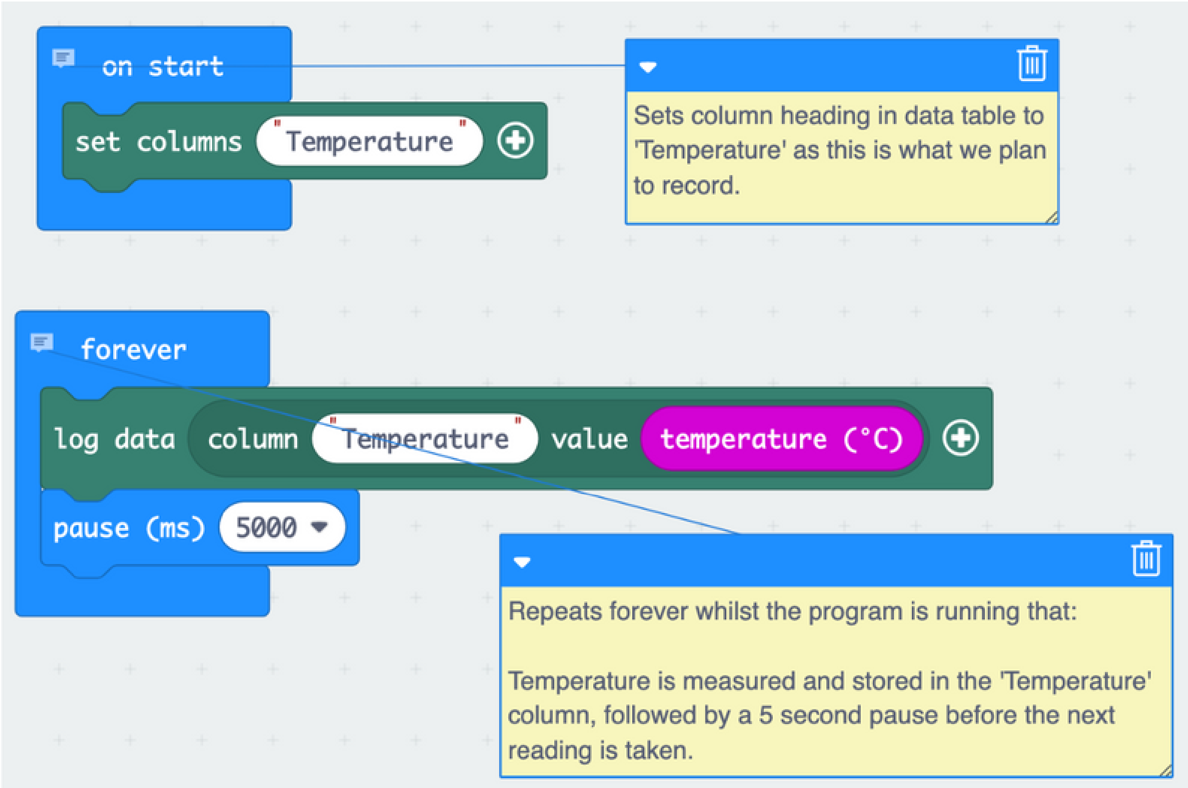
A Parson's problem scaffolds programming tasks by providing learners all the code they need to complete a problem but doesn't show how the code should be combined (slide 14).

Ask learners to add the following commands to their micro:bit project and task them to combine the code to complete our challenge. I.e. When the program is run, it automatically measures and stores temperature data for us at regular intervals.



Solution with comments (slide 15)

10



11

Evaluate

Lead a discussion with learners to evaluate the success of their program. Did their program:

- Automatically log the temperature at set time intervals?
- Store the temperatures measured so these could be viewed at a later date and used to decide if additional water was required for crops?

Discuss with learners whether they encountered any bugs in their program - what were these and how did they remove them?

reflect on how learners have been working as agricultural engineers in this challenge (slide 16).

Further links:

- Micro:bit Education Foundation https://microbit.org/
- Information about the new (2022) micro:bit V2 https://microbit.org/new-microbit/

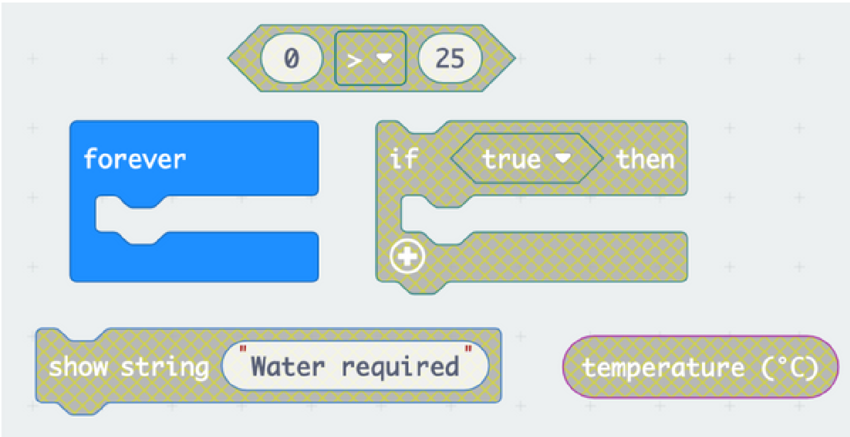
Information about data logging using the micro:bit https://microbit.org/get-started/user-guide/data-logging/

Take it further

Improve

In the introduction to this resource, we mentioned the idea of a device not only measuring and storing temperature, but also triggering the flow of water if temperatures were excessive (slide 17).

As an extension challenge, can learners combine the following code (and add it to their existing program), to prototype the idea that when a certain excessively high temperature is reached, i.e. 30 degrees, the micro:bit automatically displays a message warning 'Watering required' - simulating the triggering of a watering system being activated?



Solution with comments (slide 18)