# Goal 14 - Life under water – Ocean Health Monitor

Getting started

A group of marine scientists have asked you to develop a prototype floating sensor node that they can leave in the ocean and that transmit data to them so they can study climate change in the sea.

Success criteria

* Design and create a floating sensor node using a micro:bit
* The beacon must transmit sensor data to a gateway micro:bit every 10 seconds
* The beacon must also transmit its unique ID number (there will be lots of nodes!)
* The gateway micro:bit must be able to show the data on its LED screen

Input process output (IPO)

|  |
| --- |
| **Node** |
| **Input** | **Process** | **Output** |
| Water temperatureAir temperatureAir pressureHumidityAir qualityLight intensityAccelerometer (waves)Compass | Format dataUnique ID number  | Transit data via radioTransmit unique ID number |

|  |
| --- |
| **Gateway** |
| **Input** | **Process** | **Output** |
| Data via radioUnique ID number via radio | Format the data | Display the data on the LEDs |

Pro-tip

The scientists want data from as many sensors as possible, if you can use other sensor peripherals make sure that you make full use of all the additional sensors.

Building the prototype

The prototype should be designed and made to float but this does not need to be tested as the scientists are only interested in the sensor data and not the floating part as this will be made by a specialist boat designer.

In the following example the air temperature will be sensed and transmitted from the node to the gateway.

Pro-tip

The ‘on radio received’ event can only be created once due to hardware limitations, to work round this you can simply concatenate (join) your sensor data and send it all once, or use more than one micro:bit.

Example code 1 (node)

|  |  |
| --- | --- |
|  | This example sets the radio group to 1, this is important as both the node and the gateway need to be set to the same group or they will not be able to communicate. A variable called ‘ID’ is created to store the node’s unique ID.Another variable is created called ‘AirTemp’ and this is set to the micro:bit temperature sensor reading.Next, we create another variable called ‘Data’ to hold the joined (or concatenated) ID and ‘AirTemp’ variables.The ‘Data’ variable is then sent over radio.  |

Example code 1 (gateway)

|  |  |
| --- | --- |
|  | As with the node we also need to set the radio group to the same as the node or they will not be able to communicate. Here, the micro:bit waits to receive the variable ‘Data’. It then shows the variable on the LEDs.  |

Graphing the temperature data

Example code 2 (node)

|  |  |
| --- | --- |
|  | In this example we again set the node radio group to 1 and will also do the same on the gateway.Again, we create a variable called ‘Air Temp’ and set it to the micro:bit’s temperature sensor reading.Then the ‘AirTemp’ variable is sent over radio to the gateway. |

Example code 2 (gateway)

|  |  |
| --- | --- |
|  | As with the node we also need to set the radio group to the same as the node or they will not be able to communicate. This is where things are a little different!In this example we take the received variable and plot a bar graph using the LED blocks.You may need to think about adding some ‘pauses’ on the node to slow the data transmission rate down to make the batteries last longer! |

Pro-tip

Sending data in real time can use up batteries on these devices quickly and these monitors need to be at sea for long periods of time. Think about how you can send the data from the nodes less frequently but still give a good representation of temperature changes over time.

Graphing the acceleration data

Example code 3 (node)

|  |  |
| --- | --- |
|  | These blocks are set up as in example 2 and set the radio group to 1 (this is also done on the gateway below).Here the acceleration data is sent over radio. In the previous example we set a variable to the temperature reading and send that. Here the acceleration data is just sent. Do you think this makes a difference? |
| A graph of data logged from a micro:bit and Mu | In this example the Y axis data is sent, what is measuring? |

Example code 3 (gateway)

|  |  |
| --- | --- |
|  | As before the radio group is set to the same as the node so that they can communicate. The blocks here ‘listen’ for transmitted numbers and then add them to a variable called ‘recievedNumber’.A graph is then plotted on the LED screen showing a real time graph of the accelerometer data,Why might this be useful data for the scientists? |

|  |
| --- |
| If you have access to any additional peripherals, you will need some custom extension blocks in MakeCode. To find these blocks you need to click the ‘Extensions’ tab: |
|  | search for the control board that you are using.  |

Test time!

**BEWARE: micro:bits and peripherals are not waterproof!** Electricity and water do not mix well, and you can permanently damage the electronics by getting them wet. You can test this prototype on dry land!

Stretch tasks

* Add a separate waterproof temperature sensor that can be submerged to measure the temperature at different depths
* Using the compass on the micro:bit, modify your prototype to provide a wind direction to the gateway (you will need to do this on a separate micro:bit)
* Create another node and have them both transmit different data to the gateway
* Create a user interface (UI) for the gateway to allow the user to see the data from separate nodes
* Add a LoRa or WiFi transmitter to your prototype and transmit the data over longer distances
* Add a pH meter to also measure ocean acidity (you may need to use an analogue to digital signal converter)

Final thoughts

This project explores several different computational techniques to gather useful data to help the marine scientists. Can you think of other types of sensors that may help them? What other data could be useful to them across an ocean?