Open Window Detector

Difficulty 🟡

Required materials

- 1 EMDACC4 with the Magnetic sensor
- 1 magnet and some tape to keep it in place
- 1 Micro:bit per group
- 1 Computer per group

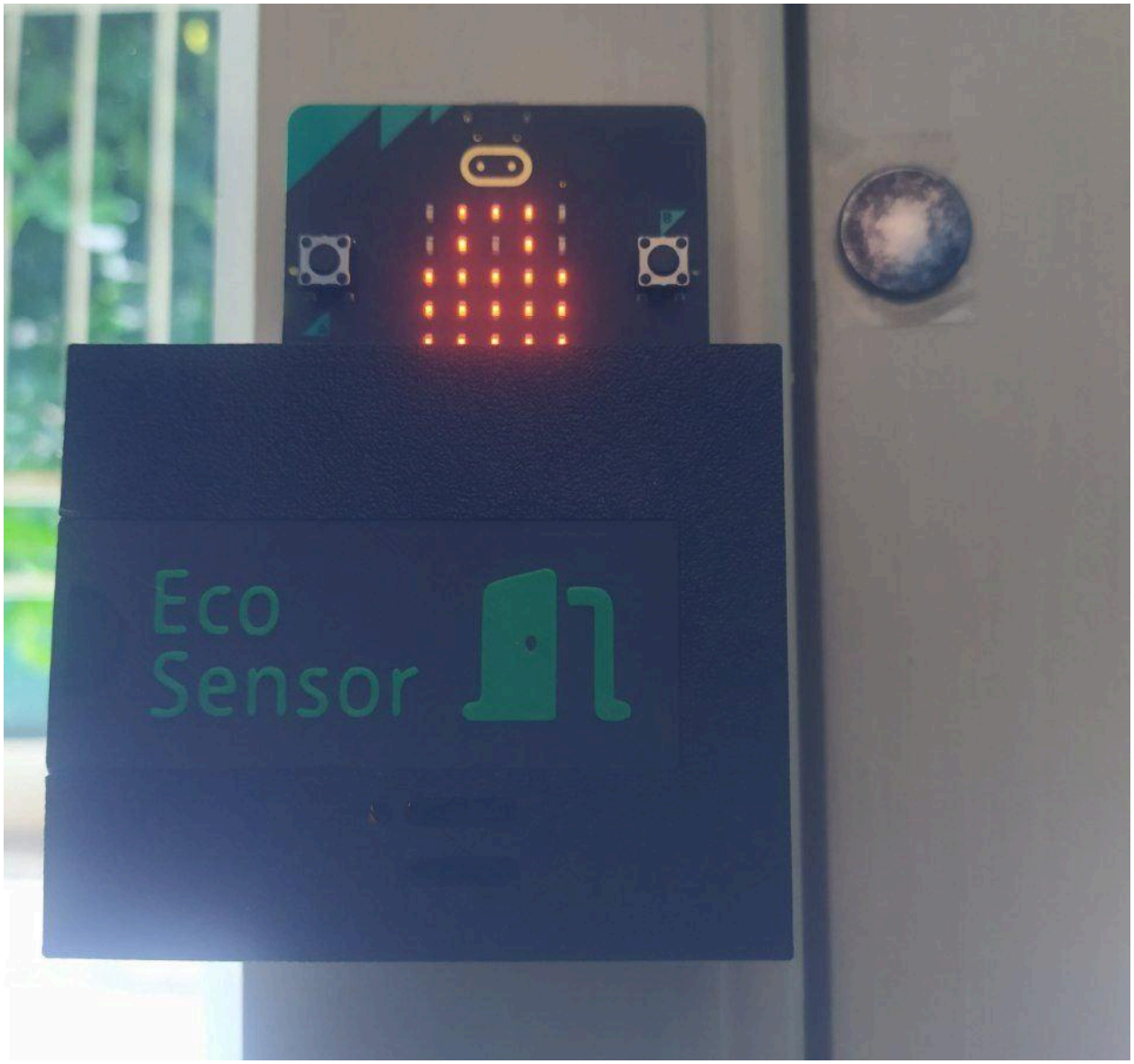
Prerequisites

- Being able to plot data with the MakeCode platform
- Program the central EMDACC4 with the last firmware

1. Deploy the central EMDACC4

After having loaded the firmware on the Micro:bit that will work as the central device, take the EMDACC4 module equipped with the **magnet sensor** and connect the Micro:bit to it. Attach the battery to the module and close the enclosure. Turn it on and check if the symbol of a lock appears on the display: this confirms that the board is properly connected and is already broadcasting magnet sensor data via radio waves.

Now fix the EMDACC4 device and the magnet to the door or window you want to monitor. Check the picture below as a reference.



2. Connect the Micro:bit to the computer

Turn on the computer and open a web browser (Chrome is recommended).

Go to the MakeCode website (https://makecode.microbit.org/), create a new project, and give it a meaningful name, for instance **“Open Window Detector”**.

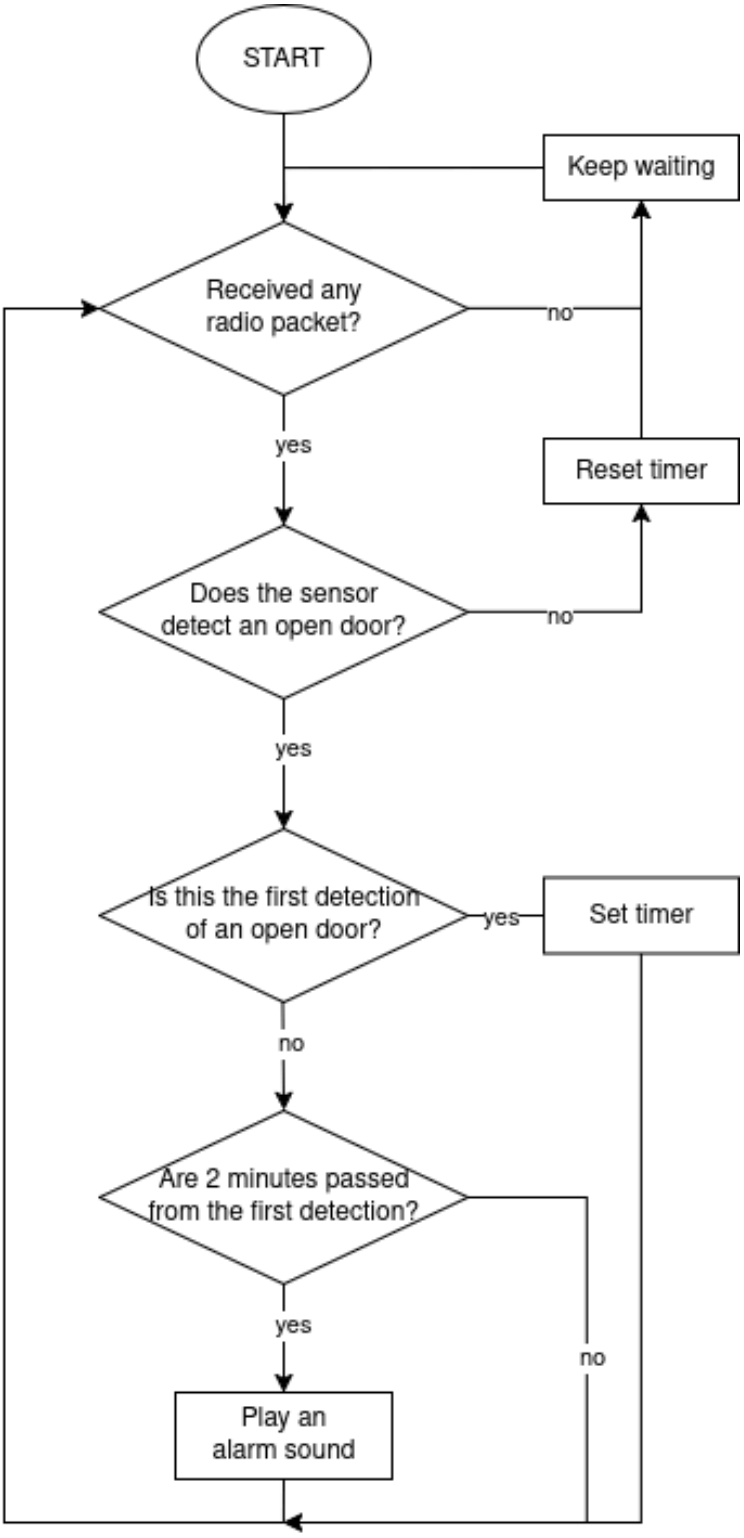
Plug the Micro:bit to the USB port of the computer and pair it with the MakeCode editor.

Write a simple plotting program to see which are the values that the sensor is broadcasting and monitor how the values change according to the presence of the magnet close to the sensor.

3. Design the algorithm

The goal of this activity is to make the peripheral Micro:bit **detect when a window or a door is left open for too long** and send an alarm if it happens so. This could be useful as a reminder to close doors/windows to reduce the waste of energy used to cool or heat a room.

Let us start imagining how it should work. At the end of the process you should have something similar to the following flowchart.



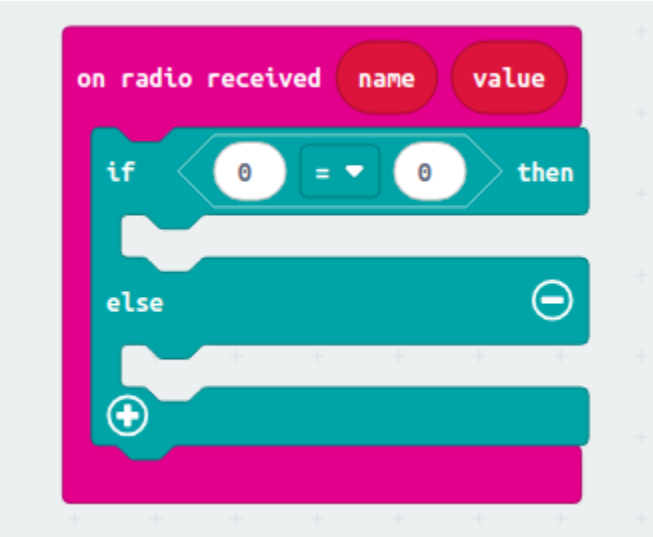
Do not worry if it does not come like this at the first attempt: writing algorithms is done in steps, you start from a vague idea and then you refine it adding the details and checking the corner cases. Do not rush to the code or you might forget something important!

4. Program the Micro:bit

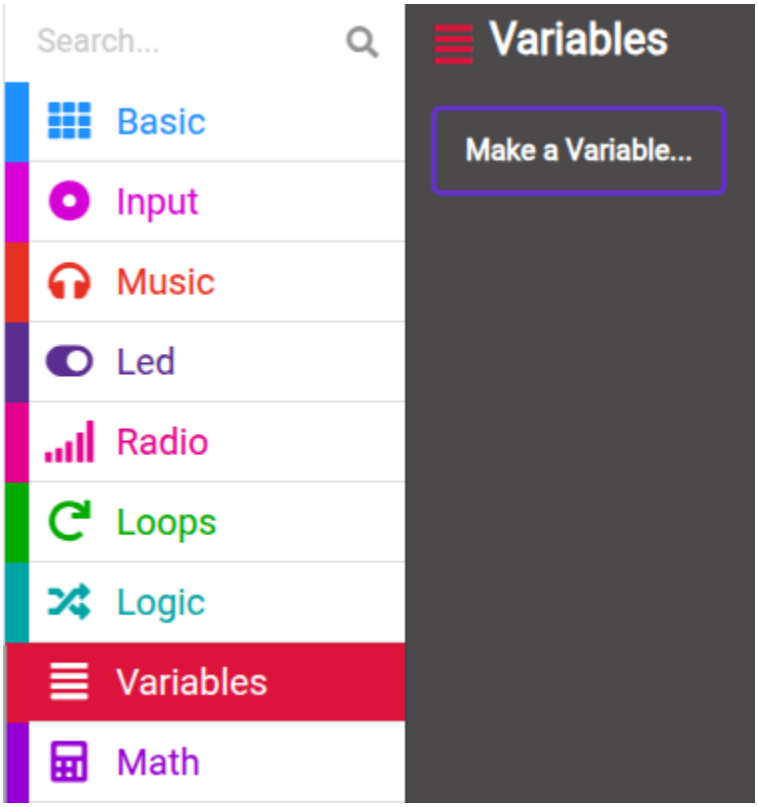
After having the algorithm clear in mind, it is possible to go on with the real programming of the device.

Everything starts with the receiving of a radio packet from the central EMDACC4 device so a block “on radio received <name> <value>” will be needed.

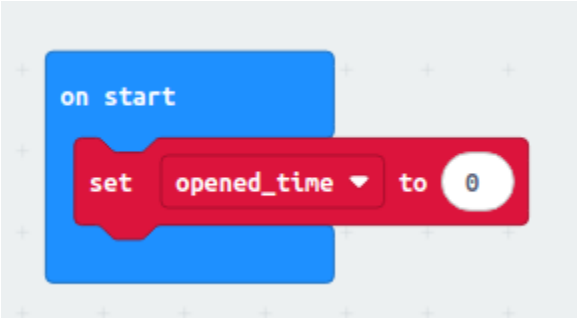
Whenever we receive a new packet from the central device we must evaluate a condition, we find all the decision and evaluation related blocks under the “Logic” menu, in particular we need a “if … then … else” block with a comparison condition that we are going to connect together.



The condition we need to evaluate is whether the door is open (value will be 0) or closed (value will be 1). In order to keep track of the status of the door and its changes, we need to declare a variable that will be used to record the time at which we are notified that the door has been opened. We can declare the variable “opened_time” from the menu “Variables” and clicking on “Make a Variable”.

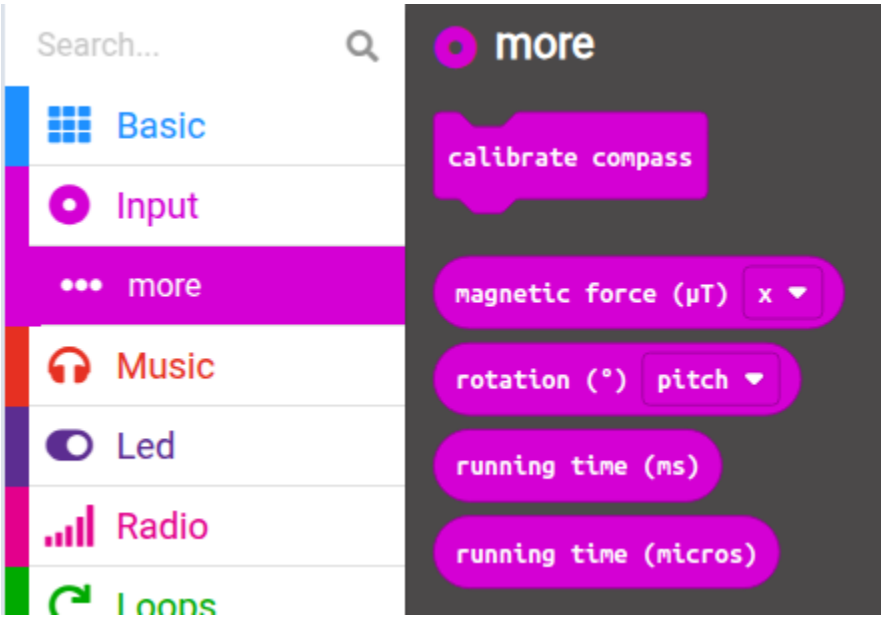


Every time we create a variable we must set a value for it and it is generally a good idea to do it inside the “on start” block to keep everything tidy and under control.

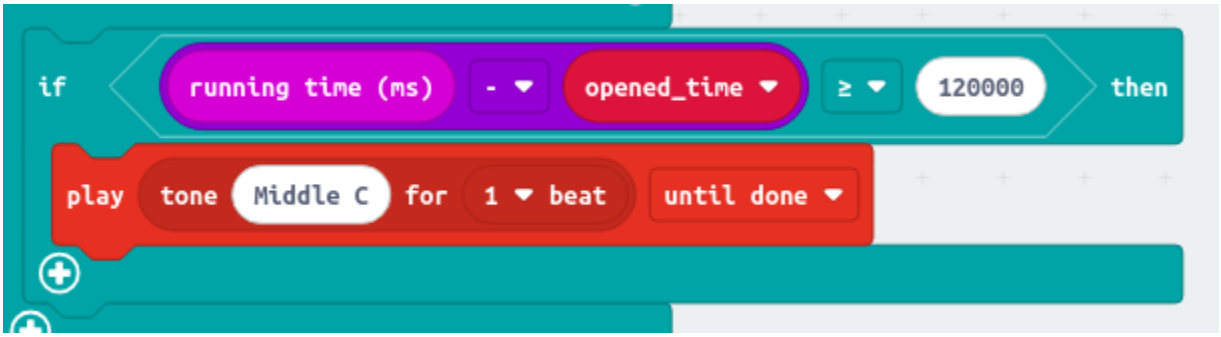


Setting “opened_time” to 0 will be as saying “the door is closed and there is no need to keep track of elapsed time”. This instruction will be repeated every time we receive a packet that tells us that the door is closed.

In case we detect an open door, if it was previously measured as closed (another condition involved), we set “opened_time” to the current time in milliseconds (ms). We can take this value from the menu “Input - more”. In this way we can keep track of the time in which we first measured the fact that the door has been opened.



Assembling the code as it is described in the flowchart, we will reach a point in which we need to measure how much time has passed since we first detected that the door has been left open. Since we stored the time at which the door has been measured as open, we just need to subtract it to the current time to have the number of milliseconds passed. If we want to play an alarm if the door has been left open for more than 2 minutes (120 seconds), we should write a similar code:



If the condition is met we can play a tone for one beat that will work as an alarm.

After having double checked the code, you can load onto the Micro:bit to check if it works as expected!

Ideas

1. Try changing the alert response: instead of playing a sound, make the Micro:bit flash or display something.
2. Combine this with other data: what if the device only warns about the window being open when the temperature outside is below 18 °C?

Reference code

