



Измерване на обхвата на движение през нощта

Необходими материали, окабеляване и използване на платка Micro:bit

За да програмирате платка micro:bit за събиране на данни за влажност и температура, ще ви е необходим следният хардуер:

- Платка Micro:bit V2 и нейните вградени сензори: Основната програмируема платка, включваща вграден сензор за светлина чрез LED дисплей, вграден сензор за ниво на звука и вграден сензор за температура - приблизително 19 евро на micro:bit
- Micro-USB кабел: За захранване и програмиране на micro:bit
- Външна батерия (по избор): За преносима работа, ако micro:bit трябва да се отдели - Ще намерите официалния калъф за батерия micro:bit, който може да се закупи за около 2,20 евро за опаковка.

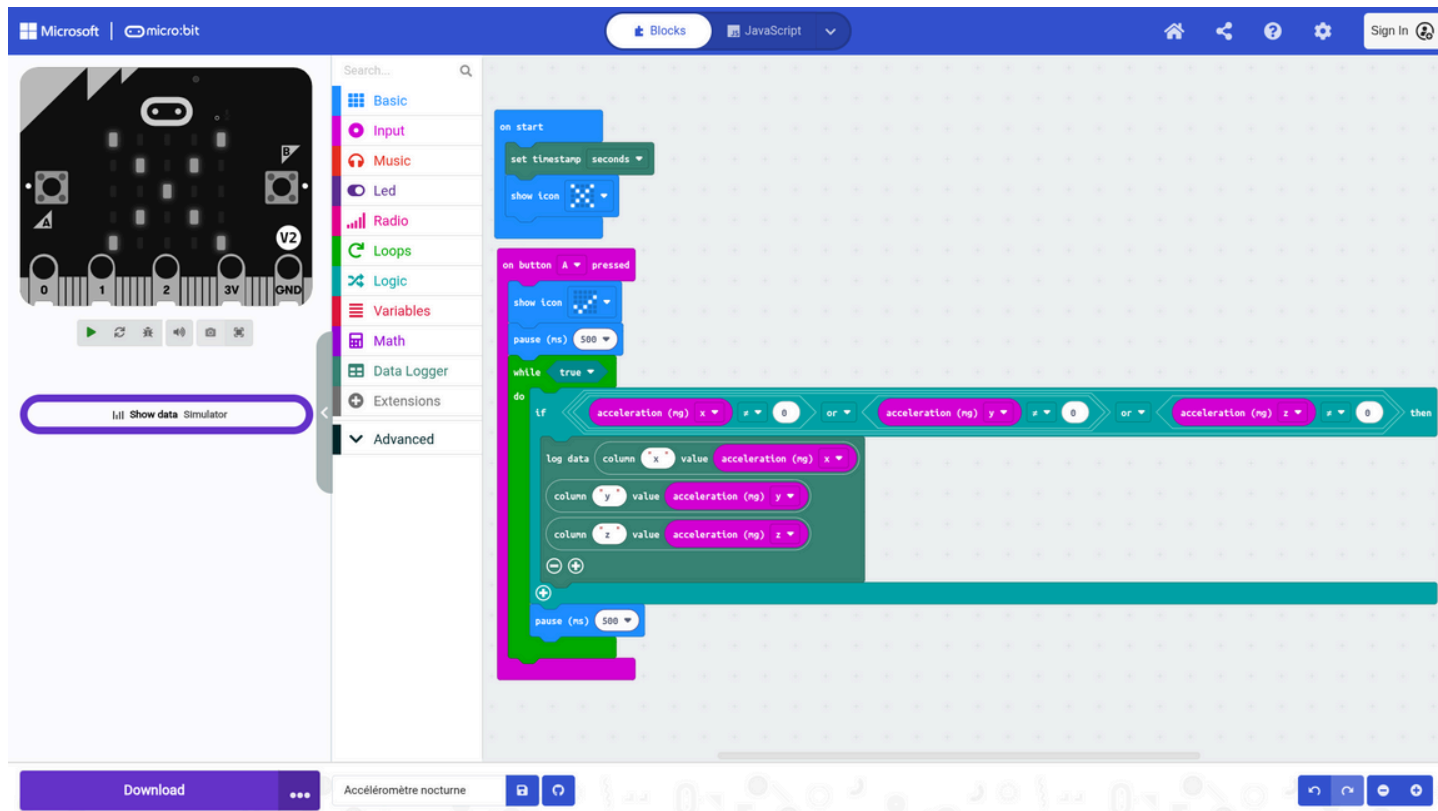
Можете също да закупите комплекта Micro:bit V2, включващ USB кабел и кутия за батерии, за 21 евро за комплект или 177 евро за 10 комплекта.

- Компютър или таблет: За писане и качване на код.
- Програмна среда: онлайн редактор MakeCode

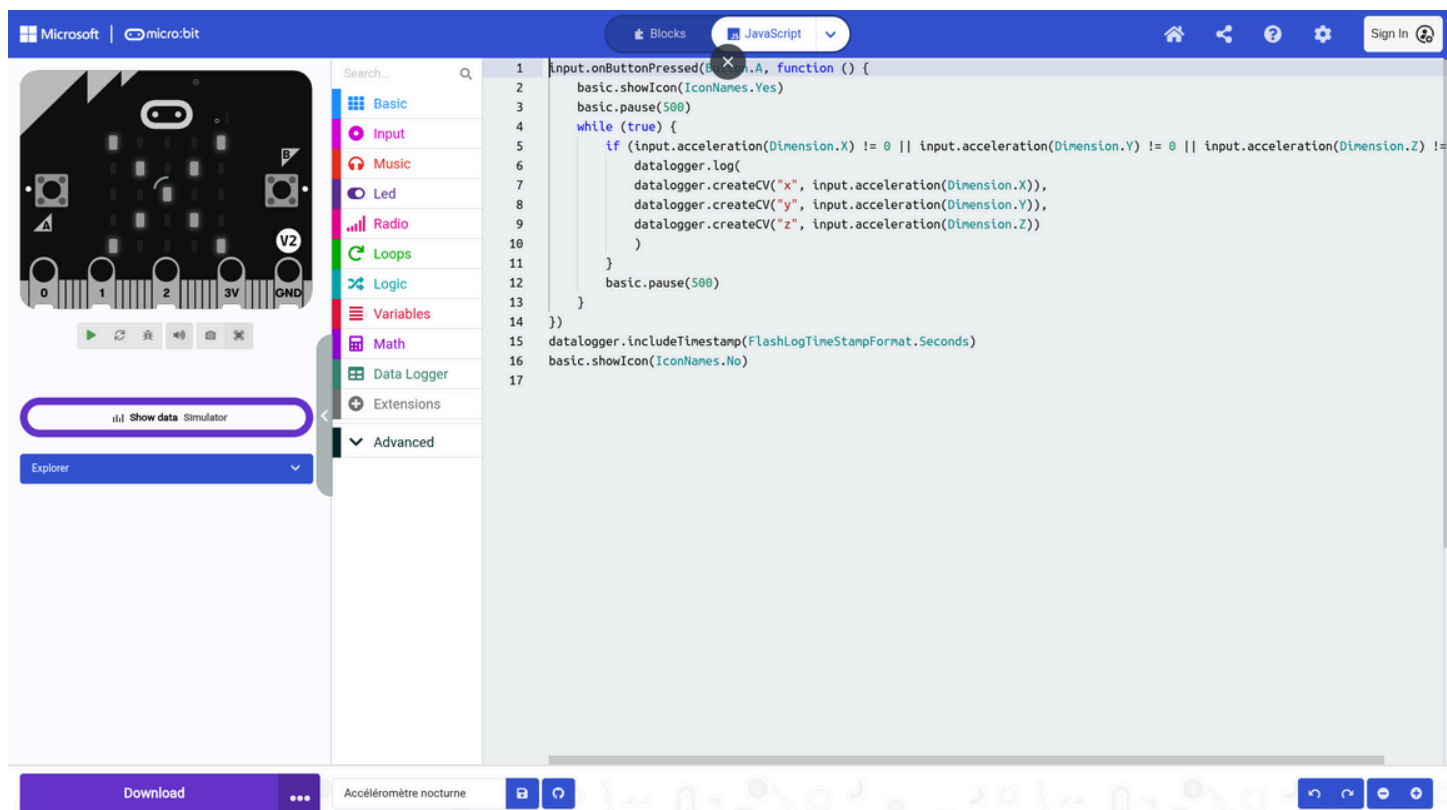
Следвайте тези стъпки, за да програмирате, поставите, запишете и извлечете данни за околната среда, използвайки micro:bit.

Стъпка 1 - Програмирайте платката Micro:bit **Свържете платката Micro:bit:** Свържете платката micro:bit към компютъра, на който сте създали програмата, използвайки редактора MakeCode. След като бъде свързана, платката micro:bit ще се появи на компютъра като сменяем диск (напр. "MICROBIT"). Напишете програмата: Отворете редактора MakeCode, за да създадете програма, която събира данни за шум, използвайки вградените сензори на програмната платка Micro:bit V2. Дайте ясно име на проекта си, преди да започнете. След като влезете в редактора, след като създадете новия си проект, ще бъдете отведени до екрана по подразбиране "изцяло от кутията" и ще трябва да инсталирате разширение. Разширенията в MakeCode са групи от кодови блокове, които не са директно включени в основните кодови блокове на MakeCode. Разширенията, както подсказва името, добавят блокове за специфична функционалност. Има разширения за широк спектър от полезни функции, добавяйки геймпад, клавиатура, мишка, серво, роботика и много други. В колоните за показване на блокове щракнете върху бутона РАЗШИРЕНИЯ. В списъка с налични разширения намерете разширението Datalogger, което ще се използва за тази дейност. Щракнете върху разширението, което искате да използвате, и на главния екран ще се появи нова група блокове.

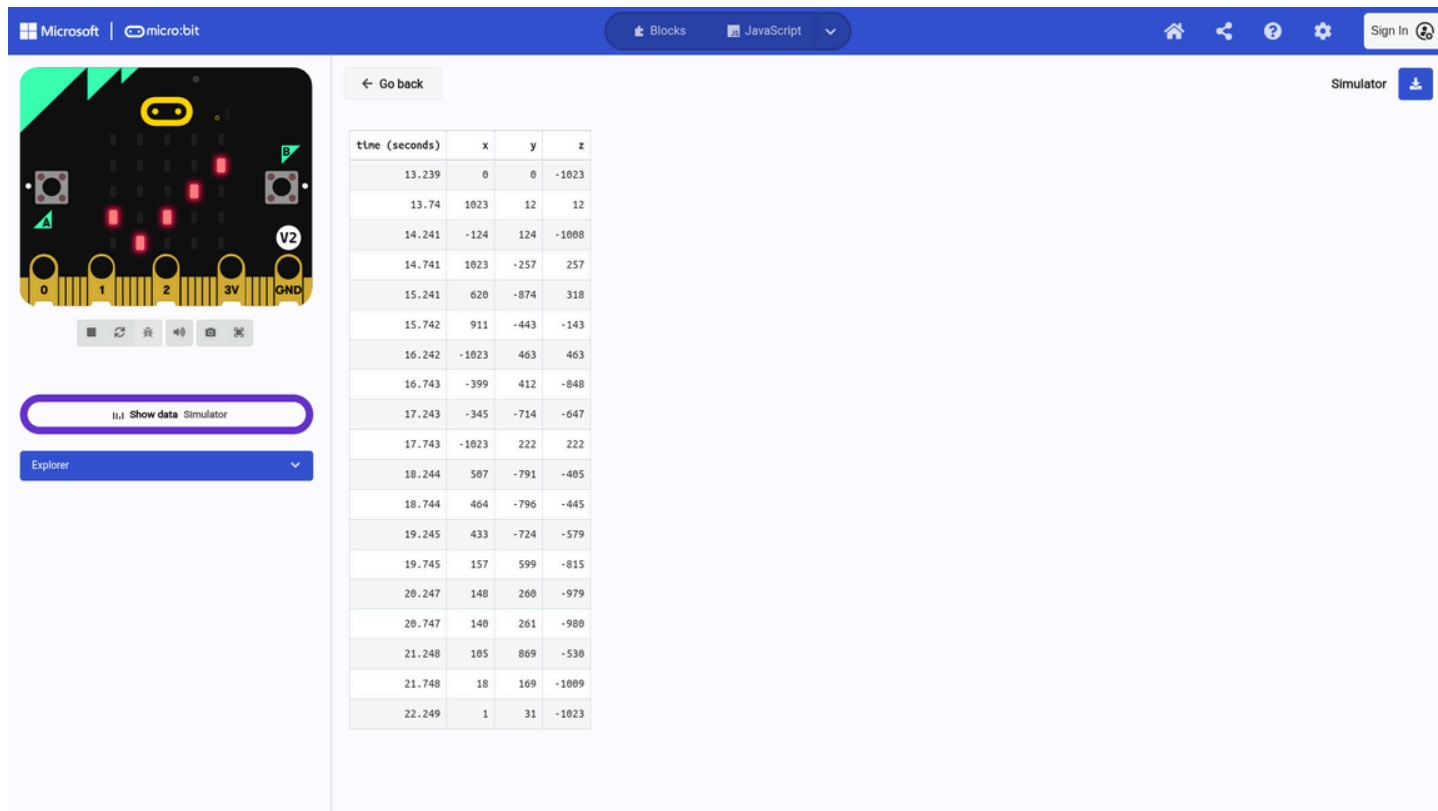
След това можете да започнете да организирате блоковете си, като следвате кода, предоставен по-долу (добавете безкраен цикъл, запазете данните в регистратора на данни...).



Можете също да копирате и поставите кода в редактора на Javascript.



Тествайте програмата, използвайки симулатора MakeCode.



След като програмата ви работи правилно на симулатора, прехвърлете я на вашия Micro:bit: щракнете върху „Качване“ в MakeCode, за да генерирате .hex файл. Този файл съдържа компилираната програма, която ще позволи на платката да работи.

Копирайте .hex файла от папката за изтегляния на сменяемото устройство „MICROBIT“.

След като файлът бъде копиран, платката автоматично ще се рестартира и ще изпълни кода.

Стъпка 2: Поставете Micro:bit и започнете да записвате данни

След като програмирате, позиционирайте micro:bit, за да събирате необходимите данни, като използвате гривна за ръка. В края на това ръководство ще намерите няколко уроци за създаване на DIY гривна за ръка за micro:bit. Използвайте външна батерия (power bank), за да захранвате micro:bit непрекъснато по време на запис (комплектът micro:bit ви дава достъп до кутии за батерии, които лесно се позиционират на платката).

Преди лягане, натиснете бутона "A" на MicroBit, за да започнете да записвате данни чрез програмата.

Стъпка 3: Извличане на данни и подготовка на картата за следващата сесия на запис

Всяка сутрин, за да избегнете загуба на данни, препоръчваме да изключите micro:bit от захранването, за да спрете записа на данни, и да свържете micro:bit към компютъра си, за да получите достъп до файла, компилиран през нощта от регистратора на данни (който ще се нарича „MY_DATA.HTM, наличен на четеца micro:bit).

Копирайте този файл на компютъра си и го преименувайте с днешната дата (напр. BOARD1_NAME_YYYY-MM-DD.HTM).

След като копирате и преименувате файла, изтрийте файла MY_DATA.HTM от платката MicroBit, за да освободите място и да позволите запис на нови данни.


Повторете процеса за следващата сесия, тоест следващата вечер преди лягане.

В края на периода на събиране ще можете да извлечете всички файлове, събрани на различните micro:bit дъски. След отваряне, файловете с данни ще бъдат достъпни в HTML формат. Те ще предоставят всички

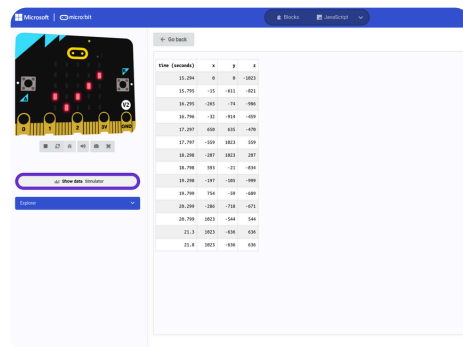
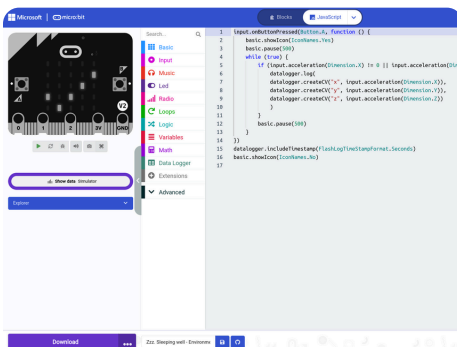
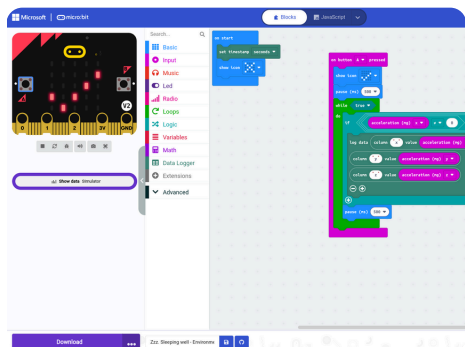
събрани данни и ще ви позволят да ги изтеглите във формат .csv.

Използвайте и разберайте кода

Ето JavaScript кодът, използван за програмиране на платка micro:bit за редовно събиране на данни за движение:



```
input.onButtonPressed(Button.A, function () {
    basic.showIcon(IconNames.Yes)
    basic.pause(500)
    while (true) {
        if (input.acceleration(Dimension.X) != 0 || input.acceleration(Dimension.Y) !=
0 || input.acceleration(Dimension.Z) != 0) {
            datalogger.log(
                datalogger.createCV("x", input.acceleration(Dimension.X)),
                datalogger.createCV("y", input.acceleration(Dimension.Y)),
                datalogger.createCV("z", input.acceleration(Dimension.Z))
            )
        }
        basic.pause(500)
    }
})
datalogger.includeTimestamp(FlashLogTimeStampFormat.Seconds)
basic.showIcon(IconNames.No)
```



Как работи кодът? Тази програма измерва стойностите на ускорението, заснети от акселерометъра. На всеки 500 милисекунди (интервалът може да се промени на 10 секунди, 5 минути, два пъти на час и т.н.) програмата проверява дали картата се движи и ако е така, компилира данните в „datalogger“, от който може да се изтегли .csv файл.



.csv (стойности, разделени със запетая) файлът е текстов файлов формат, използван за съхраняване на таблични данни (например в таблица или електронна таблица). Всеки ред във файла представлява ред с данни и всяка стойност в рамките на реда е разделена с разделител (често запетая, но понякога точка и запетая или табулация). Възможно е да извлечете данни от .csv файл в програма за електронни таблици, като Excel или LibreOffice Calc. В Excel отворете програмата, щракнете върху Файл > Отвори, изберете .csv файла и конфигурирайте разделители, ако е необходимо, като използвате инструмента за импортиране. В LibreOffice Calc следвайте подобен процес: щракнете върху Файл > Отвори, изберете файла и използвайте съветника за импортиране, за да изберете разделителя (например запетая или точка и запетая). И в двата случая данните се показват в табличен формат, готови за анализ.

Тази програма е предназначена да записва данни от акселерометъра на MicroBit при натискане на бутон "A" и да ги съхранява в регистратор на данни със стойности за осите X, Y и Z.

Инициализиране на събитието за натискане на бутон "A": Когато потребителят натисне бутон "A" на MicroBit, функцията `input.onButtonPressed(Button.A, function () {...})` се задейства. Това предотвратява запазването на данни веднага след свързването на платката.

Показване на иконата „Да“ по време на изпълнение: Преди да започне запис на данни, програмата показва иконата „Да“ (`basic.showIcon(IconNames.Yes)`) за 500 милисекунди (0,5 секунди), за да покаже, че процесът на запис е започнал.

Пауза за 500 милисекунди: След показване на иконата „Да“, програмата изчаква 500 милисекунди, използвайки `basic.pause(500)`.

Безкраен цикъл за събиране на данни: Програмата влиза в безкраен цикъл `while (true)`. Това означава, че данните ще се събират и записват безкрайно, докато MicroBit не бъде изключен или рестартиран.

Проверка на данните от акселерометъра: На всяка итерация програмата проверява дали някоя от стойностите на ускорението (по осите X, Y или Z) е различна от нула. Това се прави с условието:

```
ако (вход.ускорение(Размер.X) != 0 || вход.ускорение(Размер.Y) != 0 ||  
вход.ускорение(Размер.Z) != 0)
```

Ако някоя от тези стойности е различна от нула (което означава, че е засечено движение), програмата записва тези данни в регистратора на данни.

Записване на данни: Стойностите на ускорението за осите X, Y и Z се записват с помощта на функцията `datalogger.log()`: Тази функция създава запис на стойностите на ускорението, когато стойностите са различни от нула, с времеви печат за всеки запис. Времевият печат се добавя автоматично, като се използва следният ред (обяснен по-късно).

```
datalogger.log(  
    datalogger.createCV("x", input.acceleration(Dimension.X)),  
    datalogger.createCV("y", input.acceleration(Размер.Y)),  
    datalogger.createCV("z", input.acceleration(Размер.Z))  
)
```

Пауза за 500 милисекунди преди следващо отчитане: След запазване на данните, програмата прави пауза за 500 милисекунди (`basic.pause(500)`), преди да възобнови отчитането на стойностите на акселерометъра.

Времеви печат на данните (включен чрез `datalogger.includeTimestamp`): Освен функцията на бутона, командата `datalogger.includeTimestamp(FlashLogTimeStampFormat.Seconds)` се използва за включване на времеви печат в секунди за всеки запис в регистратора на данни. Това означава, че всеки запис ще бъде придружен от изминалото време в секунди от стартирането на програмата.

Показване на икона „Не“ преди изпълнение: Преди потребителят да натисне бутона „А“, програмата показва икона „Не“ (`basic.showIcon(IconNames.No)`), за да покаже, че MicroBit чака действие от потребителя.

Данни: Акселерометърът връща стойности в милиграми за всяка ос (X, Y, Z). Това означава, че ако акселерометърът засече ускорение от 1000 милиграми, това съответства на ускорение от 1G (или $9,81 \text{ m/s}^2$) по тази ос.

Добавете лента за ръка

Има няколко налични ресурса, които ще ви помогнат да прикрепите платката Micro:bit към гривна за ръка, което ще позволи на учениците да я носят през нощта. Ето някои полезни уроци и аксесоари, които ще ви помогнат с тази стъпка:

[Smart Coding Watch Kit - micro:bit](#)

[Duct Tape Watch](#)

[BBC micro:bit wrist holder | mattoppenheim](#)

[Yahboom Wrist:bit wearable watch kit based on BBC Micro:bit V2/V1.5 board](#)

[CHARGE for micro:bit](#)