



Misurazione dell'ampiezza del movimento durante la notte

Materiali richiesti, cablaggio e utilizzo di una scheda Micro:bit

Per programmare una scheda micro:bit in modo che raccolga umidità e temperatura, avrai bisogno del seguente hardware:

- Scheda Micro:bit V2 e relativi sensori integrati: la scheda principale programmabile comprende un sensore di luce integrato tramite il suo display LED, un sensore di livello sonoro integrato e un sensore di temperatura integrato - Circa 19 EUR per micro:bit
- Cavo micro-USB: per alimentare e programmare il micro:bit
- Batteria esterna (opzionale): per il funzionamento portatile se è necessario staccare il micro:bit. Troverai la custodia ufficiale per la batteria del micro:bit disponibile per l'acquisto a circa 2,20 EUR a confezione.

È anche possibile acquistare il kit Micro:bit V2 comprensivo di cavo USB e batteria al prezzo di 21 EUR per kit o 177 EUR per 10 kit.

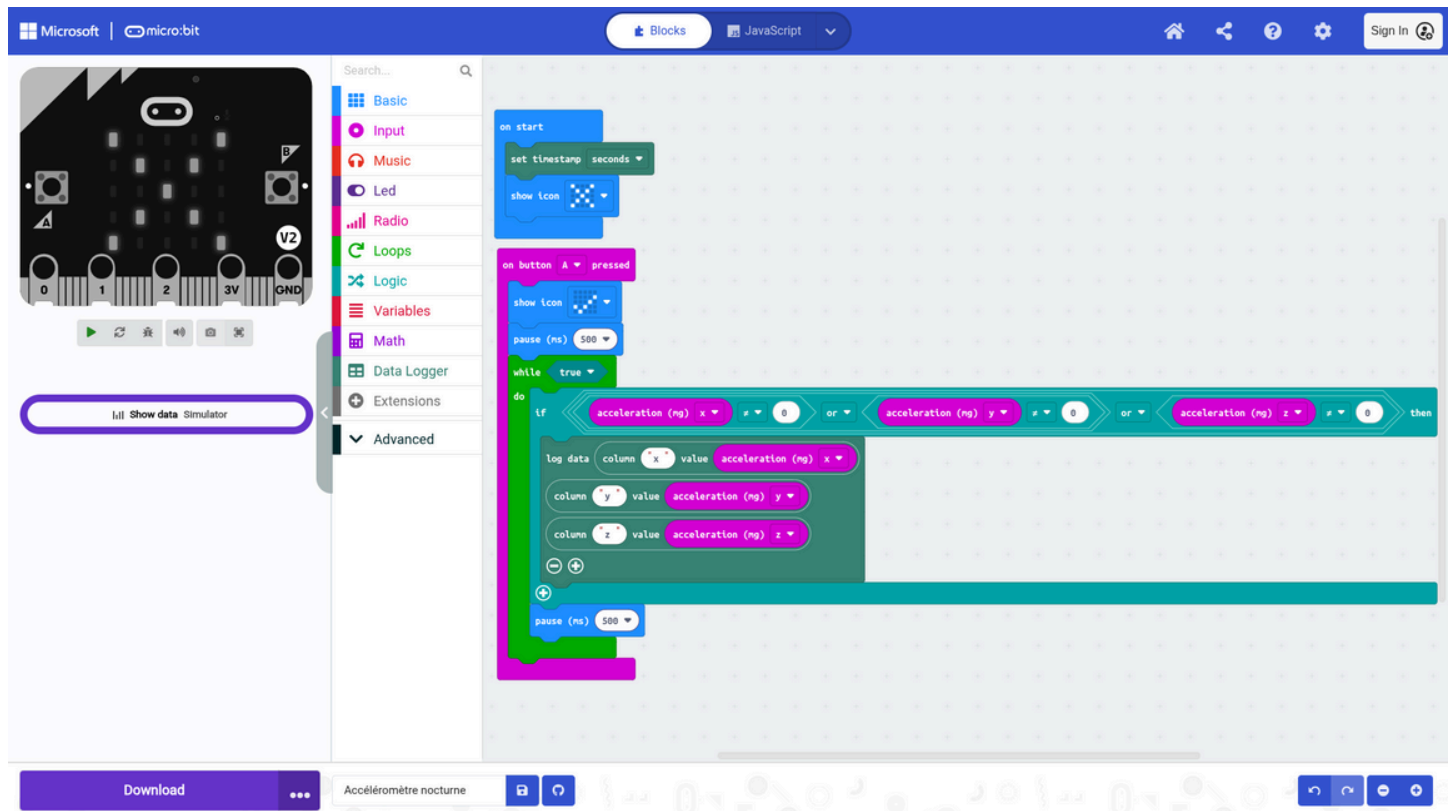
- Computer o tablet: per scrivere e caricare il codice.
- Ambiente di programmazione: editor online MakeCode

Segui questi passaggi per programmare, posizionare, registrare e recuperare dati ambientali utilizzando micro:bit.

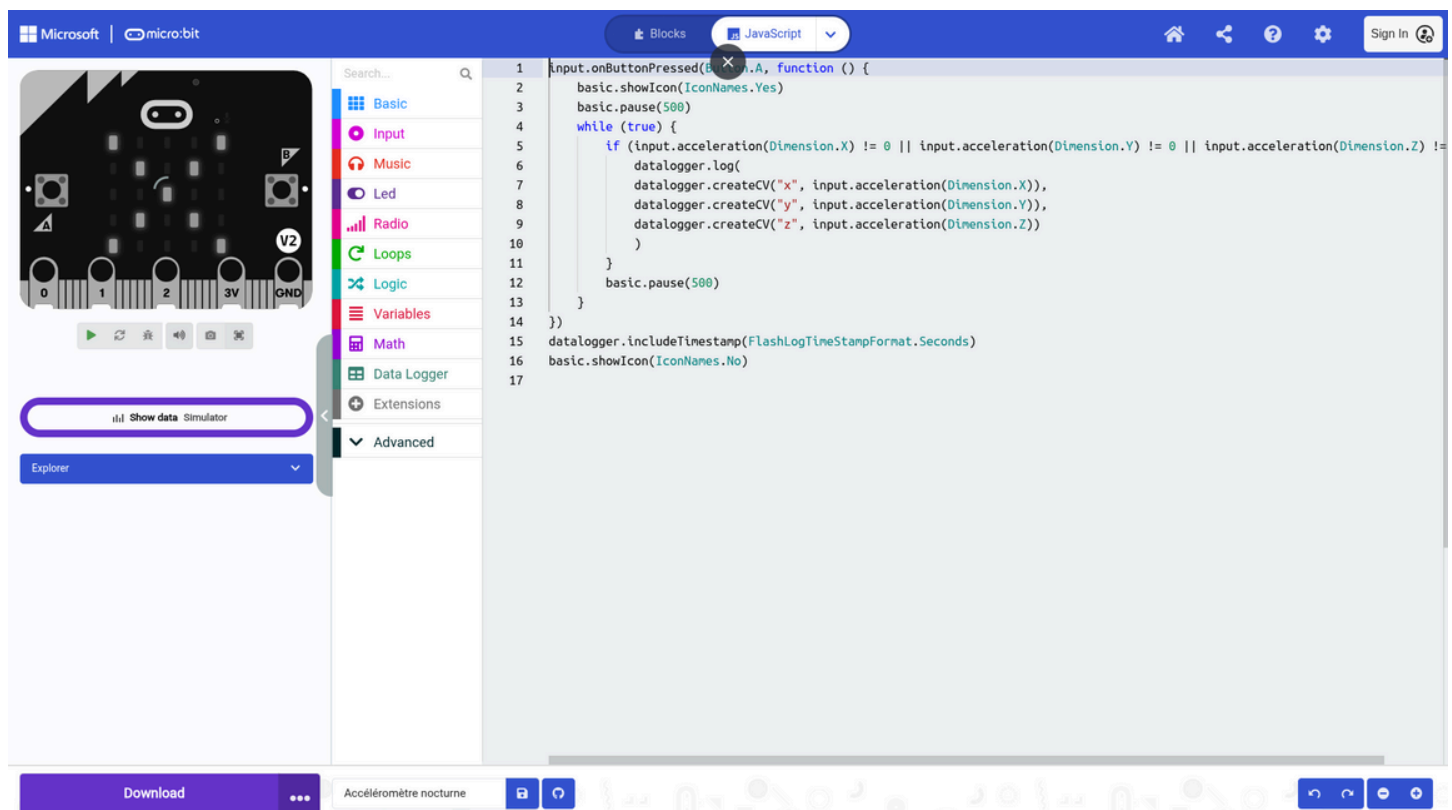
Fase 1 - Programmare la scheda Micro:bit

Collega la scheda Micro:bit: collega la scheda micro:bit al computer su cui hai creato il programma utilizzando l'editor MakeCode. Una volta collegata, la scheda micro:bit apparirà sul computer come un disco rimovibile (ad esempio, "MICROBIT"). Scrivi il programma: apri l'editor MakeCode per creare un programma che raccolga dati sul rumore utilizzando i sensori integrati nella scheda di programmazione Micro:bit V2. Assegna un nome chiaro al tuo progetto prima di iniziare. Una volta nell'editor, dopo aver creato il tuo nuovo progetto, verrai indirizzato alla schermata predefinita "out of the box" e dovrai installare un'estensione. Le estensioni in MakeCode sono gruppi di blocchi di codice che non sono direttamente inclusi nei blocchi di codice base di MakeCode. Le estensioni, come suggerisce il nome, aggiungono blocchi per funzionalità specifiche. Sono disponibili estensioni per un'ampia gamma di funzionalità molto utili, che aggiungono funzionalità per gamepad, tastiera, mouse, servocomandi e robotica e molto altro. Nelle colonne di visualizzazione dei blocchi, fai clic sul pulsante ESTENSIONI. Nell'elenco delle estensioni disponibili, trova l'estensione Datalogger che verrà utilizzata per questa attività. Fai clic sull'estensione che desideri utilizzare e nella schermata principale apparirà un nuovo gruppo di blocchi.

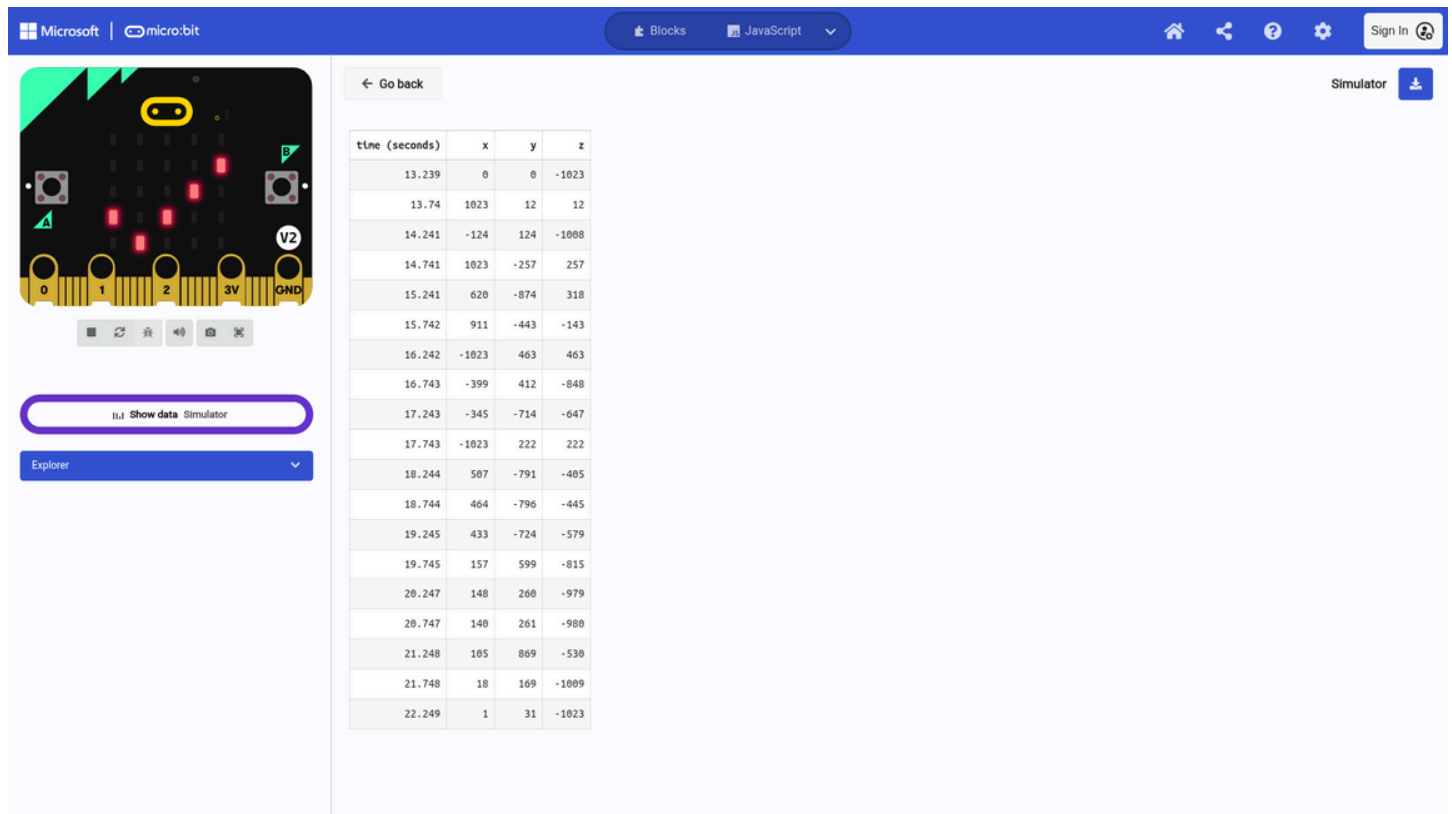
Quindi puoi iniziare a organizzare i tuoi blocchi seguendo il codice fornito di seguito (aggiungi un ciclo infinito, salva i dati nel datalogger...).



Puoi anche copiare e incollare il codice nell'editor Javascript.



Testare il programma utilizzando il simulatore MakeCode.



time (seconds)	x	y	z
13.239	0	0	-1023
13.74	1023	12	12
14.241	-124	124	-1008
14.741	1023	-257	257
15.241	620	-874	318
15.742	911	-443	-143
16.242	-1023	463	463
16.743	-399	412	-848
17.243	-345	-714	-647
17.743	-1023	222	222
18.244	507	-791	-405
18.744	464	-796	-445
19.245	433	-724	-579
19.745	157	599	-815
20.247	148	260	-979
20.747	140	261	-980
21.248	105	869	-530
21.748	18	169	-1009
22.249	1	31	-1023

Una volta che il programma funziona correttamente sul simulatore, trasferiscilo sul tuo Micro:bit: clicca su "Carica" in MakeCode per generare un file .hex. Questo file contiene il programma compilato che permetterà alla scheda di funzionare.

Copia il file .hex dalla cartella dei download nell'unità rimovibile "MICROBIT".

Una volta copiato il file, la scheda si riavvierà automaticamente ed eseguirà il codice.

Fase 2: Posizionare il Micro:bit e iniziare a registrare i dati

Una volta programmato, posiziona il micro:bit in modo che raccolga i dati necessari, utilizzando un bracciale. Alla fine di questa guida troverai alcuni tutorial per creare un bracciale fai da te per il micro:bit. Utilizza un power bank per alimentare costantemente il micro:bit durante la registrazione (il kit per il micro:bit ti dà accesso a contenitori di batterie facilmente posizionabili sulla scheda).

Prima di andare a letto, premere il pulsante "A" sul MicroBit per avviare la registrazione dei dati tramite il programma.

Fase 3: Recuperare i dati e preparare la scheda per la successiva sessione di registrazione

Ogni mattina, per evitare qualsiasi perdita di dati, consigliamo di scollegare il micro:bit dalla fonte di alimentazione per interrompere la registrazione dei dati e di collegare il micro:bit al computer per accedere al file compilato durante la notte dal datalogger (che si chiamerà "MY_DATA.HTM, disponibile sul lettore micro:bit).

Copia questo file sul tuo computer e rinominalo con la data odierna (ad esempio, BOARD1_NAME_YYYY-MM-DD.HTM).

Dopo aver copiato e rinominato il file, eliminare il file MY_DATA.HTM dalla scheda MicroBit per liberare spazio e consentire la registrazione di nuovi dati.

Ripetere il procedimento per la sessione successiva, ovvero la sera successiva prima di andare a letto.

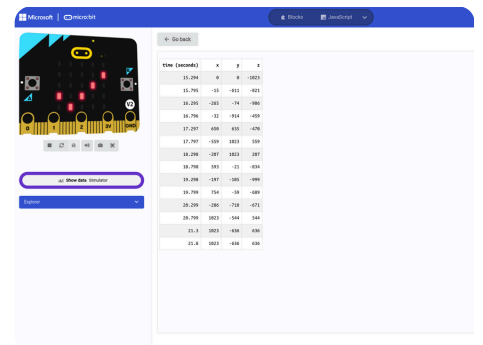
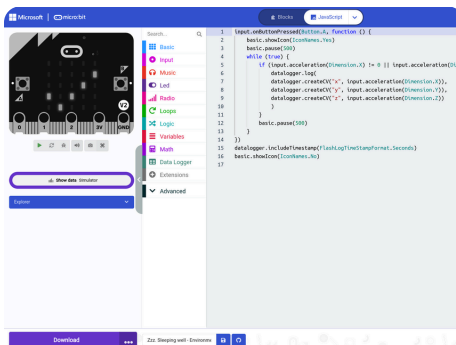
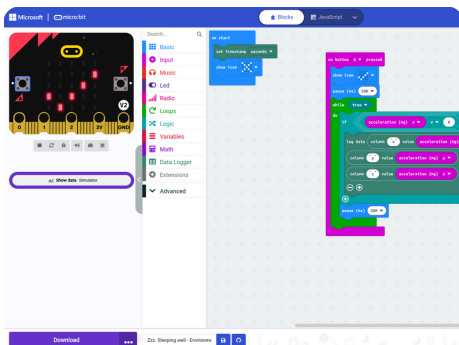
Al termine del periodo di raccolta, sarà possibile recuperare tutti i file raccolti sulle diverse schede micro:bit. Una volta aperti, i file dati saranno accessibili in formato HTML. Forniranno tutti i dati raccolti e potranno essere scaricati in formato .csv.

Utilizzare e comprendere il codice

Ecco il codice JavaScript utilizzato per programmare una scheda micro:bit in modo che raccolga regolarmente dati di movimento:



```
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)
```



Come funziona il codice?

Questo programma misura i valori di accelerazione acquisiti dall'accelerometro. Ogni 500 millisecondi (l'intervallo può essere modificato a 10 secondi, 5 minuti, 2 volte all'ora, ecc.) il programma verifica se la scheda si sta muovendo e, in tal caso, compila i dati in un "datalogger" da cui è possibile scaricare un file .csv.



Un file .csv (Comma-Separated Values) è un formato di file di testo utilizzato per archiviare dati tabulari (ad esempio in una tabella o un foglio di calcolo). Ogni riga del file rappresenta una riga di dati e ogni valore all'interno di una riga è separato da un delimitatore (spesso una virgola, ma a volte un punto e virgola o una tabulazione). È possibile recuperare i dati da un file .csv in un programma di fogli di calcolo come Excel o LibreOffice Calc. In Excel, apri il programma, fai clic su File > Apri, seleziona il file .csv e configura i delimitatori, se necessario, utilizzando lo strumento di importazione. In LibreOffice Calc, segui una procedura simile: fai clic su File > Apri, seleziona il file e utilizza la procedura guidata di importazione per scegliere il delimitatore (ad esempio, virgola o punto e virgola). In entrambi i casi, i dati vengono visualizzati in formato tabella, pronti per l'analisi.

Questo programma è progettato per registrare i dati dell'accelerometro su un MicroBit quando viene premuto il pulsante "A" e memorizzarli in un datalogger con i valori per gli assi X, Y e Z.

Inizializzazione dell'evento di pressione del pulsante "A": quando l'utente preme il pulsante "A" sul MicroBit, viene attivata la funzione `input.onButtonPressed(Button.A, function() {...})`. Ciò impedisce che i dati vengano salvati non appena la scheda viene collegata.

Visualizzazione dell'icona "Sì" durante l'esecuzione: prima di avviare la registrazione dei dati, il programma visualizza l'icona "Sì" (`basic.showIcon(IconNames.Yes)`) per 500 millisecondi (0,5 secondi) per indicare che il processo di registrazione è iniziato.

Pausa di 500 millisecondi: dopo aver visualizzato l'icona "Sì", il programma attende 500 millisecondi utilizzando `basic.pause(500)`.

Ciclo infinito di raccolta dati: il programma entra in un ciclo `while` (vero) infinito. Ciò significa che i dati verranno raccolti e registrati all'infinito finché il MicroBit non verrà spento o riavviato.

Controllo dei dati dell'accelerometro: a ogni iterazione, il programma verifica se uno qualsiasi dei valori di accelerazione (sugli assi X, Y o Z) è diverso da zero. Questo viene fatto con la condizione:

```
se (input.accelerazione(Dimensione.X) != 0 || input.accelerazione(Dimensione.Y) != 0  
|| input.accelerazione(Dimensione.Z) != 0)
```

Se uno qualsiasi di questi valori è diverso da zero (il che significa che è stato rilevato un movimento), il programma registra questo dato nel datalogger.

Registrazione dati: i valori di accelerazione per gli assi X, Y e Z vengono registrati utilizzando la funzione `datalogger.log()`: questa funzione crea un record dei valori di accelerazione ogni volta che i valori sono diversi da zero, con un timestamp per ogni record. Il timestamp viene aggiunto automaticamente utilizzando la seguente riga (spiegata più avanti).

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

Pausa di 500 millisecondi prima della lettura successiva: dopo aver salvato i dati, il programma si ferma per 500 millisecondi (`basic.pause(500)`) prima di riprendere le letture dei valori dell'accelerometro.

Timestamp dei dati (incluso tramite `datalogger.includeTimestamp`): oltre alla funzione del pulsante, il comando `datalogger.includeTimestamp(FlashLogTimeStampFormat.Seconds)` viene utilizzato per includere un timestamp in secondi per ogni record nel datalogger. Ciò significa che ogni record sarà accompagnato dal tempo trascorso, in secondi, dall'avvio del programma.

Visualizzazione dell'icona "No" prima dell'esecuzione: prima che l'utente prema il pulsante "A", il programma visualizza un'icona "No" (`basic.showIcon(IconNames.No)`) per indicare che il MicroBit è in attesa dell'azione dell'utente.

Dati: l'accelerometro restituisce valori in milli-g per ciascun asse (X, Y, Z). Ciò significa che se l'accelerometro rileva un'accelerazione di 1000 milli-g, ciò corrisponde a un'accelerazione di 1 G (o $9,81 \text{ m/s}^2$) su quell'asse.

Aggiungi una fascia da braccio

Sono disponibili diverse risorse per aiutarti a fissare la scheda Micro:bit a un braccialetto, consentendo agli studenti di indossarla durante la notte. Ecco alcuni tutorial e accessori utili per aiutarti in questo passaggio:

[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](#)