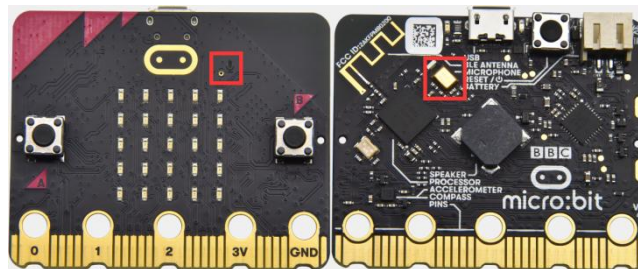


Keyestudio

Project 11: Microphone

1. Description

The micro:bit motherboard boasts a built-in microphone, which can be used to measure the ambient sound level. When you clap, the LED indicator on the micro:bit motherboard will turn on. It can measure the intensity of sound. In this connection, you can make a sound level chart or disco lights that are in tune with the music.



(Sound input hole of microphone) (Microphone)

2. Components Needed

		
Micro:bit * 1	USB Cable * 1	

Keyestudio

3. Test Code

You can upload the code directly from the tutorial (read the "**Development Environment Configuration**" file if in doubt).

Code 1:

```
from microbit import *

while True:
    if microphone.current_event() == SoundEvent.LOUD:
        display.show(Image.HEART)
        sleep(200)
    if microphone.current_event() == SoundEvent.QUiet:
        display.show(Image.HEART_SMALL)
```

Test Result: Download code 1 to micro:bit board and keep the USB cable connected, when applauding, the LED dot matrix displays the ♥ pattern. When the outside world is quiet, it will display the 🧊 pattern.

Code 2:

```
from microbit import *

maxSound = 0

lights = Image("11111:"
               "11111:")
```

Keyestudio

```
"11111:"  
"11111:"  
"11111")  
  
# ignore first sound level reading  
soundLevel = microphone.sound_level()  
sleep(200)  
  
while True:  
    if button_a.is_pressed():  
        display.scroll(maxSound)  
    else:  
        soundLevel = microphone.sound_level()  
        display.show(lights * soundLevel)  
        if soundLevel > maxSound:  
            maxSound = soundLevel
```

Test Result: Download code 2 to micro:bit board and keep the USB cable connected. When you press the A button on the micro:bit motherboard, the LED dot matrix displays the detected maximum sound level value in the environment at this time (**Note: reset the maximum value by pressing the reset button on the back of the micro:bit**). When applauding, the louder the detected sound, the brighter the 25 LEDs.

Keyestudio

4. Code Explanation

from microbit import *	Import the library file of micro:bit
while True:	This is permanent loop, and micro bit executes the code
if microphone.current_event() == SoundEvent.LOUD: display.show(Image.HEART) sleep(200) if microphone.current_event() == SoundEvent.QUET: display.show(Image.HEART_SMALL)	If the microphone detects the sound, LED dot matrix displays ♥ Delay in 200ms If it detects the outside world is quiet, LED dot matrix displays 🧩
print ("Light intensity:", Lightintensity)	BBC microbit REPL window prints the value of the light brightness detected by the light sensor
maxSound = 0	maxSound has an initial value of 0
lights = Image("11111:""11111:""11111:""11111:""11111")	Set Image() to variable lights
soundLevel = microphone.sound_level()	Set microphone.sound_level() to variable soundLevel

Keyestudio

if button_a.is_pressed():

display.scroll(maxSound)

else:

soundLevel =

microphone.sound_level()

display.show(lights * soundLevel)

If button A is pressed, LED dot matrix displays the value of the sound level.

Otherwise,

Set microphone.sound_level() to variable soundLevel

The 25 LEDs are like breathing lights as the sound level changes