

# Project 3 - Step Counter

For this project, we're going to write some code that interacts with some things on the micro:bit - buttons and the *accelerometer*. You know what buttons are - you press them and they do something. But do you know what an accelerometer is?

An accelerometer is a device that measures movement. A lot of smart phones have accelerometers that can tell if the phone is moving up or down, or side to side. The accelerometer in the micro:bit isn't as powerful as that, but it can detect movement, especially if you shake the micro:bit.



First let's write some code. In your Mu editor, type or copy this code and then flash it to the micro:bit by clicking on the 'Flash' button:

```
from microbit import *

# Define a variable to record steps
steps = 0

while True:
    # Check to see if a step has been taken.
    # If so, display a smile and increase the number of steps by 1
    if accelerometer.was_gesture('shake'):
        steps += 1
        display.show(Image.HAPPY)
        sleep(500)
        display.clear()

    # Check to see if button A has been pressed.
    # If so, display the number of steps taken
    if button_a.is_pressed():
        string_steps = str(steps)
        display.show(string_steps)
        sleep(500)
        display.clear()
```

Let's see if we can tell what this code is doing.

First we create a variable called `steps` with a value of zero.

The part that says “**while True:**” is important. This puts the micro:bit into an *infinite loop*. Remember when we talked about **while** loops and how it's bad to get stuck in an infinite loop? Well this is a case where it's actually useful - the micro:bit is constantly checking to see if it's been moved, forever, until you unplug it or change the code.

Inside that **while** loop, we check to see if the micro:bit has moved, or if it has been shaken. That indicates that we have taken a step, so the code adds 1 to the **steps** variable. We also display a smiley face, then pause for a few seconds and clear the display.

After that, we check to see if a button has been pressed. **button\_a.was\_pressed()** is a function that comes with the microbit library - we get to use it because we wrote **from microbit import \*** at the top of our code.

If the 'A' button is pressed, we take the number of steps that have been added to our **steps** variable so far and convert that number into a string. This is just because **display.show()** can only show strings. Instead of this:

```
>>> 5
```

We now have a **string\_steps** that looks like this:

```
>>> '5'
```

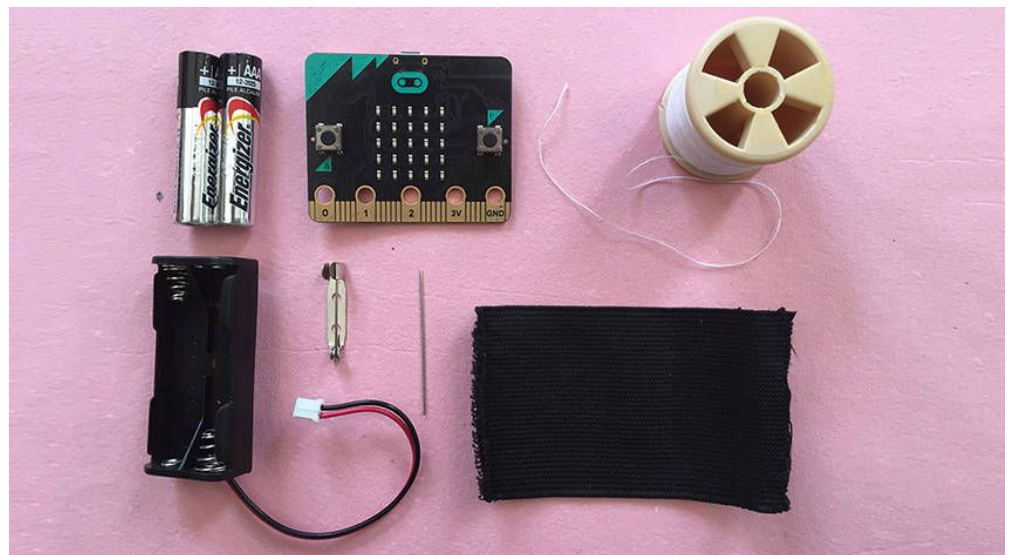
We send that number to the display, leave it there by pausing for a few seconds, then clear the display and start all over.

---

Now comes the fun part! Let's make a wearable badge so that we can walk around and count our steps!

### Supplies you should have:

- 1 micro:bit
- 1 battery holder
- 2 AAA batteries
- thread
- 1 needle
- a piece of elastic
- 1 safety pin





### **Step 1: Stitch the elastic**

Thread your needle, knot the thread, then stitch along the cut edge of the elastic to close it up.



### **Step 2: Roll the elastic to hide edges.**

Roll the top edge over the shorter edge and tuck in so no raw edges are showing.



### **Step 3: Stitch the edge down.**



#### **Step 4: Checkpoint!**

It should now look like a tube.



#### **Step 5: Turn whole tube inside out.**

This will neaten it up and hide stitches.



#### **Step 6: Stitch the pin.**

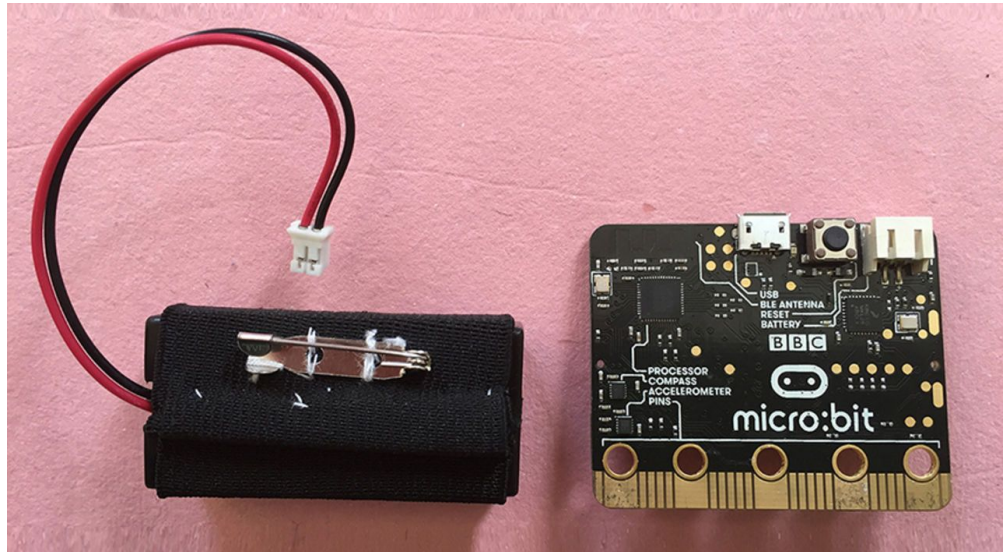
Stitch the safety pin securely to one side of the tube, preferable with the seam just below the badge pin, so that when it's worn, it hangs well.





**Step 7: Insert the battery holder into the elastic tube.**

After inserting 2 AAA batteries, squeeze the battery holder into the tube. It should be a snug fit.



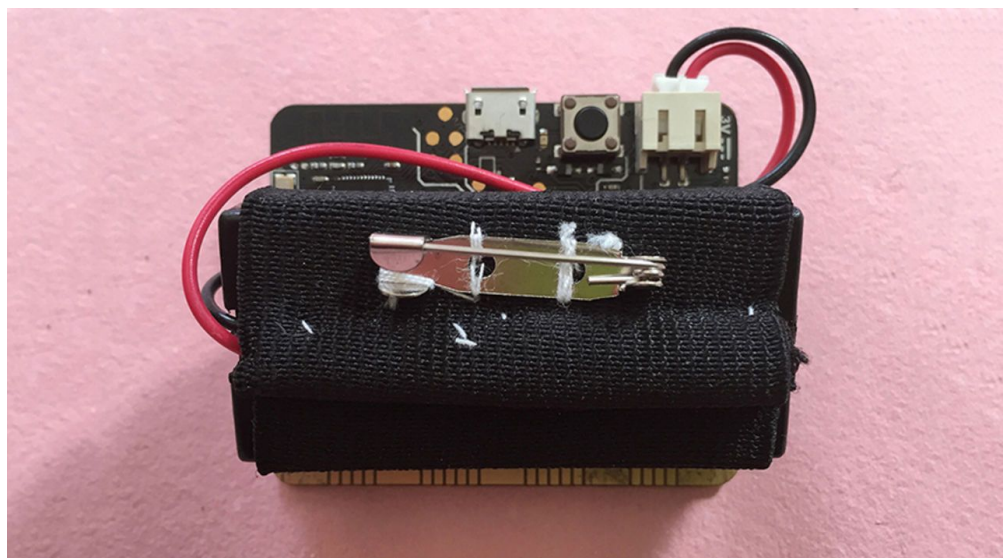
**Step 8: Sew a holding stitch.**

Knot your thread again and place a stitch on the opposite side to the badge pin in the center edge of the elastic.



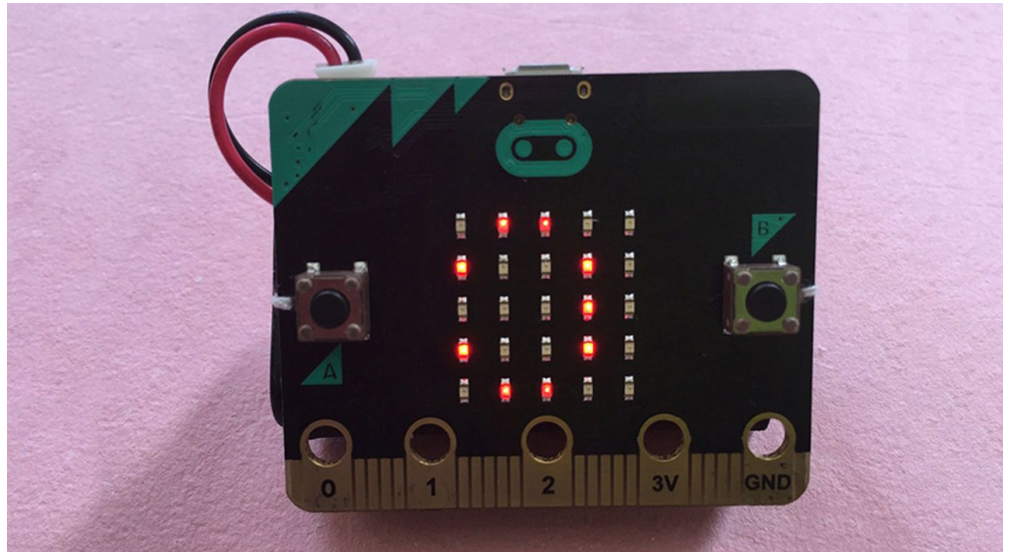
**Step 9: Sew elastic to the micro:bit.**

Place the tube onto the back of the micro:bit and sew through the holes located on both sides of the micro:bit, right next to the buttons.



### Step 10: Plug in the power.

Your micro:bit is now ready to wear. Pin it to your shoe, or perhaps the leg of your pants, and walk around to see how many steps you can measure.



---

For more projects you can do with your micro:bit, check out these resources:

- **The MicroPython guide to BBC micro:bit**  
<https://www.microbit.co.uk/python-guide>
- **MicroPython/micro:bit documentation:**  
<http://microbit-micropython.readthedocs.io/en/latest/>
- **Micro:bit projects on Instructables:**  
<http://www.instructables.com/howto/microbit/>