## Experiment 10: Using the Accelerometer

### Introduction

In this experiment you will look at combining the use of the accelerometer on the micro:bit to measure the orientation of the micro:bit and use it to control the angle of a servo.

Ready to shake, rattle and roll?

#### Parts Needed

You will need the following parts:

* **1x** micro:bit
* **1x** Micro B USB Cable
* **1x** micro:bit Breakout (with Headers)
* **1x** Breadboard
* **5x** Jumper Wires
* **1x** Servo

#### Didn't Get the SIK for micro:bit?

If you are conducting this experiment and didn't get the Inventor's Kit, we suggest using these parts:

### Introducing the Accelerometer

The accelerometer is a component that you won't find in the kit's bag of parts. Why? Because it is on the micro:bit itself! On the back of the micro:bit you can see a number of small chips. One of them is the accelerometer. The micro:bit has an onboard accelerometer that measures gravitational force. Depending on the version that you have, the accelerometer and compass can be on separate ICs or combined into a single IC.

|  |  |
| --- | --- |
| v1.0 w/ Accelerometer on Sepearate IC | v1.5 w/ Combined Accelerometer and Magnetometer |
| *v1.0 w/ Accelerometer on Sepearate IC* | *v1.5 w/ Combined Accelerometer and Magnetometer* |

**Note:** For more information about the motion sensor hardware change, check out this article from micro:bit support.

[**MICRO:BIT SUPPORT: MOTION SENSOR HARDWARE CHANGE**](https://support.microbit.org/support/solutions/articles/19000087020-micro-bit-motion-sensor-change)

An accelerometer is a sensor that measures the gravitational forces pulling on it in all three dimensions of the chip's X, Y and Z axes.



*Visualization of a Common Accelerometer (ADXL345) with Three Axes*

Not only can an accelerometer measure the raw forces pulling on the chip and the object that the chip is sitting on, but it can also detect steps, shakes and other motions that have a specific pattern. On top of that, you can use an accelerometer to simply detect the orientation that the device is in. Did you ever wonder how your phone knows when you turn it from portrait to landscape? It is all because of the accelerometer in your phone!

### Hardware Hookup

Ready to start hooking everything up? Check out the wiring diagram below to see how everything is connected.

|  |  |
| --- | --- |
| Polarized Components  | Pay special attention to the component’s markings indicating how to place it on the breadboard. Polarized components can only be connected to a circuit in one direction. |

#### Wiring Diagram for the Experiment



*Having a hard time seeing the circuit? Click on the wiring diagram for a closer look.*

⚡ **Note:** Controlling a servo with the micro:bit's 3.3V pin is not intended to give you a highly accurate angle of rotation, just basic movement from one position to another to get your project movin'. If you are looking for a higher degree of accuracy from your micro servo, we recommend powering your servo motor with the recommended operating voltage with an external power supply of 4.8V to 6V. Don't forgot to connect ground from the external power supply for reference.

### Run Your Script

*Either copy and paste, or re-create the following code into your own MakeCode editor by clicking the open icon in the upper right-hand corner of the editor window. You can also just download this example by clicking the download button in the lower right-hand corner of the code window.*

**Note:** You may need to disable your ad/pop-up blocker to interact with the MakeCode programming environment and simulated circuit!

### Code to Note

Let's take a look at the code blocks in this experiment.



*If you are having a hard time viewing this code, click on the image above to get a better look!*

#### Acceleration

The acceleration block can be found under the input blocks group. This block returns the force of gravity pulling on a specific axis of the micro:bit (X, Y or Z) and represents that value as a range of numbers between -1023 and 1023. In this case, we measure the X axis, which is the side-to-side tilt of the micro:bit. If you tilt the micro:bit all the way to the left, you will get a -1023 value and all the way to the right is positive 1023.

#### Map

The map block looks intimidating, but it is one of the most useful blocks in MakeCode. The map block takes a given variable that has a known range --- in this case -1023 to 1023 --- and "maps" or scales that value range to another given range. The given range we want is 15 to 165, which is a good *safe* range of rotation for the servo. So, in the end -1023 ends up to equal 0, and 1023 ends up as 165 from the map block.

### What You Should See

At the beginning of the program the servo should move to 90 degrees and then react to the orientation of the micro:bit. If you hold the micro:bit flat, the servo will be at 90 degrees. Then if you tilt the servo to the left, it will move less than 90 degrees toward the value of 15. If you move it to the right, the servo will move toward 165.

