# CoderDojo Bray : Ultrasonic Sushi

The basic Arduino program to start you off with the LED Dojo Sushi challenges is provided: **range**.

“range” is a simple program which controls a HC-SR04 Ultrasonic sensor, and prints out some raw numbers which are a function of the distance to the nearest object in front of the distance sensor.

Using this “range” program as a base, we have a series of challenges to go deeper into controlling the sensor and learning some programming techniques.

If you can complete these Challenges, then you have mastered using the HC-SR04 sensors to measure distance with your Arduino. Congratulations!

## Challenge #0 – Getting Started

### Level

Beginner : You need to complete this challenge to get started with working with the HC-SR04 and your Arduino.

### Challenge

* Get the Arduino IDE installed on your computer
* Build the “range” sketch
* Build a circuit on your breadboard with the HC-SR04 sensor connected to the Arduino
* Download the sketch to your Arduino, open up the Serial Monitor and observe the numbers printing out from the monitor
* Point your distance sensor at different targets: does the number printed out change?

### Extra Credit

* Draw a graph showing the sensor reading for different ranges to the target
* For extra super credit, use Excel to make a chart of the sensor reading vs distance
* Look up the special Arduino function calls made in the Arduino Reference Guide

### Aims

Get the Arduino IDE installed on your computer and correctly communicating with an Arduino board connected to it over USB, with the HC-SR04 sensor working correctly.

When you have this working, your environment is up and running and you are ready for the rest of the Ultrasonic Sushi challenges.

### Resources

An Arduino, USB cable, breadboard, sensor and some wires, and a laptop.

### Notes for Mentors

See the coderdojobray site for troubleshooting information, if the Arduino IDE will not communicate with the Arduino board.

## Challenge #1 – Refactoring

### Level

Intermediate

### Challenge

Refactor the code to put the ranging part of the code into a function.

“Refactoring” is what we call modifying the software to make it clearer, or easier to read, or easier to use. In this challenge you will move the code which does ranging into a function.

* Make a copy of the range sketch
* Look up “functions” in the Arduino reference
* Modify the code:
  + Make a function which gets the raw range and returns it to the calling function
  + Modify the loop() to use this function instead of the existing code

### Extra Credit

* Modify the ranging function so that it can work with multiple ranging sensors: pass in the sensor to use for ranging as a parameter to the function

### Aims

Learn about functions and how to write your own functions.

Learn that you can write a program which looks the same from the outside in very different ways.

### Resources

An Arduino, USB cable, breadboard, sensor and some wires, and a laptop.

### Notes for Mentors

See the coderdojobray site for troubleshooting information, if the Arduino IDE will not communicate with the Arduino board.

## Challenge #2 – Calibration

### Level

Intermediate

### Challenge

Calibration means “comparison between measurements”.

We know that the ranging sensor gives us a number, which changes depending on the distance to the nearest object. But we don’t know what that number means exactly: it would be better if our program printed out the distance in centimetres – because we understand centimetres.

We want to modify the program so that the output is in cm.

* Make a copy of the range sketch
  + If you’ve already completed Challenge #1, use that modified sketch instead
* Get the sketch working on your Arduino
* Make a series of measurements to targets at different distances
* Work out how to translate the sensor reading into the distance in cm
* Modify the code so that instead of printing out the raw sensor reading, it prints out the distance in cm

### Extra Credit

* Write some functions which will convert the raw range numbers from the sensor into different distances: cm, m, km, inches, feet, yards …
* Test your code using the tape measure.
* Which distances are useful do you think?

### Aims

Learn about how to relate numbers coming back from a sensor to the physical world.

### Resources

An Arduino, USB cable, breadboard, sensor and some wires, and a laptop. And a tape measure ☺

### Notes for Mentors

See the coderdojobray site for troubleshooting information, if the Arduino IDE will not communicate with the Arduino board.

## Challenge #3 – On Command

### Level

Advanced

### Challenge

We want to modify our circuit so that we add in a pushbutton, and the Arduino takes a measurement only when the pushbutton is pressed.

Read Tutorial 5 (<http://arduino.coderdojobray.com/AT_L5>) on our tutorial site to see an explanation of pushbuttons and how to connect them up.

* Modify your circuit to add in a push button on your breadboard, connected to a digital input pin on your Arduino.
* Make a copy of the range sketch
  + If you’ve already completed Challenge #1, use that modified sketch instead
* Modify the code so that the Arduino only takes a measurement and prints it out when the button is pressed.

### Extra Credit

* Make a circuit with two pushbuttons on it
* Modify the code so that:
  + When the boards boots up, it is not taking measurements at all
  + When you press one pushbutton, and Arduino starts taking measurements and printing then out every 20 seconds
  + Each time you press that pushbutton, the Arduino takes the measurements twice as fast: i.e 20 seconds -> 10 seconds -> 5 seconds -> 2.5 seconds -> 1.25 seconds …
  + Each time your press the other button, the Arduino takes measurements twice as slowly: i.e. 1.25 -> 2.5 -> 5 -> 10 -> 20
    - … until it gets to 20 seconds, then it stops taking measurements.

### Aims

Learn about how controlling your board using an external input. Learn about control loops and timing.

### Resources

An Arduino, USB cable, breadboard, sensor and some wires, some pushbuttons, and a laptop.

### Notes for Mentors

See the coderdojobray site for troubleshooting information, if the Arduino IDE will not communicate with the Arduino board.