

ELEN30013 Workshop 6: Rangefinder and Servo Part 2

Total marks: 30 points

Attendance: 10 points

This workshop will expand upon the previous workshop that covered the basics of the HCSR04 ultrasonic range finder and servo motor to interface with Matlab for data collection and plotting using version **2019a or newer**.

Software Interfacing

Task 1 (5 points)

The first stage of this workshop is focused on establishing a serial connection to the Arduino. To do this you will need to have a program on the Arduino that is making use of serial communication. For this first task we will be looking at the case where the Arduino is writing and Matlab is reading.

Arduino Uno code:

```
// Arduino
int counter = 0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  Serial.print("Count is: ");
  Serial.println(counter);
  counter++;
  delay(1000);
}
```

We will use some of the following functions in Matlab, the others are listed as references.

Matlab serial functions:

Function Name	Reference Link
serialport	https://au.mathworks.com/help/matlab/ref/serialport.html
read	https://au.mathworks.com/help/matlab/ref/serialport.read.html
readline	https://au.mathworks.com/help/matlab/ref/serialport.readline.html
writeline	https://au.mathworks.com/help/matlab/ref/serialport.writeline.html
write	https://au.mathworks.com/help/matlab/ref/serialport.write.html

flush

<https://au.mathworks.com/help/matlab/ref/serialport.flush.html>

Task 2 (5 points)

In your Matlab script clear the workspace as there can only be one connection to the Arduino and each variable counts as a connection. Also, ensure that there are no programs such as the Arduino serial monitor are running.

```
%% Matlab
clear
close all
clc

device = serialport("COM1",9600);
% Be careful copying the quote marks they may need retyping
% replace COM1 with the Arduino comport
```

Task 3 (5 points)

Use the Matlab workspace to investigate `serialport` object, `read` and `readline` before moving on. Make use of the `readline` function to read in data from the Arduino in your program.

Task 4 (5 points)

Extract the number from the string from `readline`. Use any method you wish.

The following code is just a hint for one possible approach:

```
[num_ix_start,num_ix_end] = regexp("Counter :123987","[0-9]+")
```

Note: do not use the name `'line'` as it is a function as well.

Task 5 (No mark. Include in the Week 7 presentation)

Put all of the previous steps together to generate an updating plot of the time vs serial data.

You may wish to make use of one of the following Matlab time based function:

```
datetime("now")
now()
tic() toc()
```

Task 6 (No mark. Include in the Week 7 presentation)

Modify your Arduino code to read the distance of the ultrasonic rangefinder in us every second and plot the time vs data in real time. The Arduino print should be of the form:

Reading:123\n

Extend your string pattern recognition to include:

- Optional plus or minus sign
- Optional decimal place
- Optional decimal values

Hint: <https://regexr.com/>

Create additional string processing to extract the text at the start up to the colon character and directly modify the figure object axes xlabel property.

Task 7 (No mark. Include in the Week 7 presentation)

Note: In preparation for the presentation the following must be included as part of the presentation in at least some form.

Making use of the previous task and the previous workshop put it all together to create a scanning ultrasonic range finder that plots in real time. Using the Arduino the servo motor should move the ultrasonic range finder to a given angle then the ultrasonic range finder should measure the distance and finally send the data via serial as:

angle,distance\n

Sending the servo pulse duration and the ultrasonic echo duration is also a valid option but ensure that the calibration mapping is applied.

Matlab should then receive the data via a serial port in real time and process the string so that a plot of the detected distances is plotted. You may wish to use `polarplot(., polarscatter(.` or `rotz(.`. Within the plot the servo motor should be positioned at the origin.

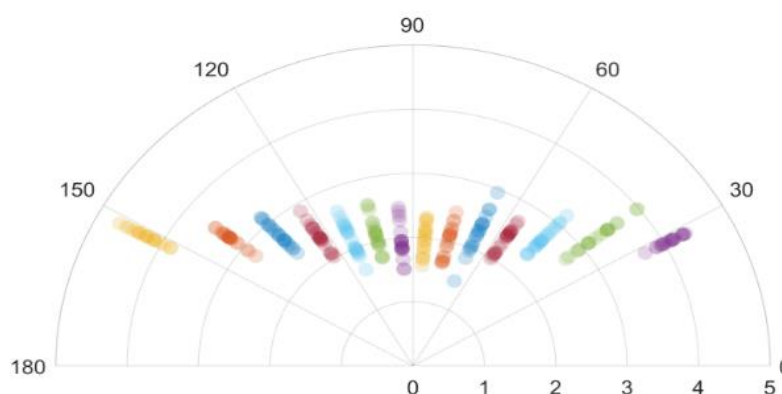


Figure 1: Plotting example detecting a wall