Exercise 1: Intro to Arduino and Raspberry Pi*

EENG450AB: Systems Exploration, Engineering, and Design Laboratory

Vibhuti Dave and Tyrone Vincent

Department of Electrical Engineering
Colorado School of Mines

Spring 2018

1 Introduction

The purpose of this exercise is to get familiar with a Raspberry Pi computer, their hardware features and how to interact with them using a PC/laptop. These basic exercises will set up the foundation to be able build a much more complex system later.

2 The Hardware

The Raspberry Pi is a fully functionally computer. It has a dedicated processor, memory, and a graphics driver for output through HDMI. It runs a specially designed version of the Linux operating system. You have been provided with the Raspberry Pi 3 with Raspbian (OS) and OpenCV preinstalled on it and ready to go. If you are ever interested in starting with a brand new kit and doing this on your own, follow the tutorials on installing Raspbian Jessie and OpenCV. The Raspberry Pi will be used to detect and process images (computer vision).

3 Python

For this course, Python will be used as the default scripting language. (C/C++ can be used if preferred. Personal experience suggests, Python is a good choice when it comes to computer vision. Simple and intuitive, although it is slow).

Go through the following tutorial on Python: `https://www.datacamp.com/courses/intro-to-python-for-data-science/`. This tutorial contains videos, so if you want to go through it in lab, please bring some headphones. The tutorial will take about 4 hours to complete.

4 Raspberry Pi

4.1 Raspberry Pi Setup and Software

Hook up the USB keyboard, and mouse to the Raspberry Pi board. Hook up the monitor using the HDMI cable. Power up the board and check the monitor to see the OS load. One of the first things to do is to change the keyboard layout to US (English). This can be done from Menu. Also ensure that the Pi is connected to the internet. Connect to "CSM

Wireless" and register the computer. The wireless connection will be slow but is still helpful. Although the OS comes with a pretty decent GUI, it will be necessary to work at command line quite a bit for the course of the project. Some basic commands and how to use them can be found here. Here is the Raspberry Pi technical documentation.

Open up the command terminal and type the following command

```
sudo nano /etc/default/keyboard
```

Change `gb` to `us` in this file.

Note: `sudo` lets you run commands as an administrator (super user). `nano` is a text editor.

Create a directory for your projects using mkdir. Code for future exercises and projects should be saved in this directory. The image on the SD card of the Pi should have the Python IDE, called IDLE 2 pre-installed on it. You can access this using the menu at the top left of your screen on the Pi.

1. The data camp tutorial introduced basic python and numpy objects, functions and methods. You will also want to also learn about conditions, loops, user input, exception handling and how to create functions. The python software foundation has a tutorial at `https://docs.python.org/2/tutorial/` (this tutorial is for Python 2, which is what you will be using). Go through the tutorial chapters 4, 7, and 8, using the Python IDE on your PI to try out code mentioned in these chapters. You should bookmark this page in case there are other topics that you will need for future projects (e.g. Section 5 on Data Structures may be useful). You should also bookmark the Python Standard Library documentation.

2. There is a data file on Canvas called datafile.txt. Copy this file to your Pi (use a usb drive or go to canvas using the internet browser on the Pi). This file generates a list of 100 integers. You can load this data using the following code:

   ```
   with open('datafile.txt','r') as f:
       b = eval(f.read())
   f.close
   ```

   Create a program on your PI in python that will display the following information about this list:

   (a) The maximum

   (b) The minimum

   (c) The index where the number 38 is located

   (d) The numbers that are repeated the most, and the number of times they are repeated (use a list comprehension rather than for loops to calculate)

   (e) A sorted list (convert to numpy array and use appropriate method)

   (f) All even numbers, in order, (Use a list comprehension rather than for loop to calculate)

5 Documentation

1. Save your a well commented and easily readable code, and upload a copy to the assignment link on Canvas.
2. Upload your weekly reflection logs to Canvas.
3. Write a 1-2 page response for the following design scenario exercise : The case below is an example of a senior design project. How would you approach this project? What steps would you take or processes would you use? What challenges would you anticipate? Focus on the describing your general approach, rather than explaining or designing a particular solution. Assume you have two semesters and reasonable resources available to complete the project.

Design Case Study

The purpose of this project is to further advance efforts in manufacturing working antennas using 3D printing techniques. The anticipated result is a working GPS antenna manufactured using 3D printing techniques (leveraging the ADAPT center at Mines).

Project Description

In today's RF industry, rapid prototyping gives a company advantages over the competition in terms of quicker design maturity by providing the ability to perform multiple iterations of a design until optimized faster. 3D printing of antennas can improve our current process of rapid prototyping and increase our advantage over the competition. Colorado School of Mines is working with Ball Aerospace to advance our 3D printing capabilities through the ADAPT Center. This project will leverage from the 2016-2017 senior design team's (anTEAMa) final results and will focus on optimizing the current RF and mechanical design, surveying 3D printing vendors early on for cost and schedule, procurement of materials, building and testing antenna. The final result will be a working GPS antenna manufactured using 3D printing techniques.

This documentation must be uploaded by the date given in the syllabus.

6 Demonstration

Be able to demonstrate your code. Print the rubric associated with this handout, and get graded by a instructor or teaching assistant when you are ready to demonstrate the results.