KENNESAW STATE UNIVERSITY

CPE 4040: Data Collection and Analysis, Spring 2023

# Laboratory Report #1
## Lab 1: Getting Started with Raspberry Pi

Team Members: Neal Jarzen & Clarence Barron

Electrical and Computer Engineering

Kennesaw State University

Faculty: Dr. Jeffrey L Yiin

Date of Lab Experiment: January 23, 2023

# I. Objective

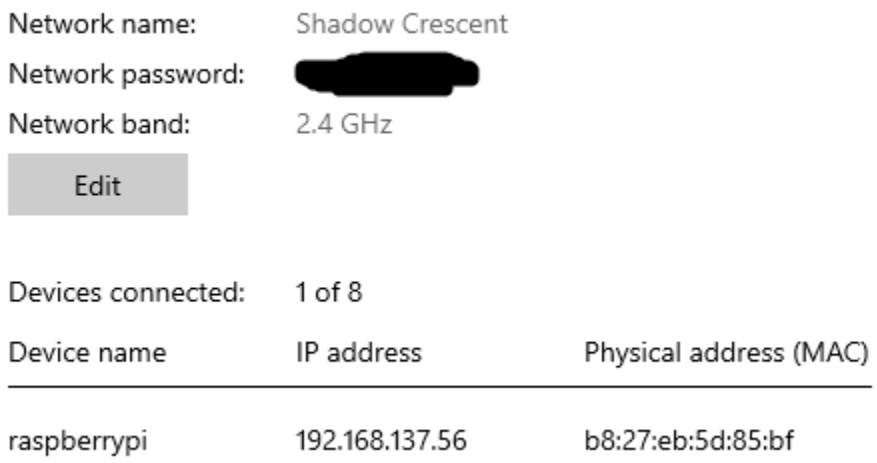
The objective of lab 1 was to learn how to install an Operating System and update packages within a Raspberry Pi. We also need to understand how remote connections work when connecting to a Raspberry Pi as well as the required tools it takes to access and control a Pi. Finally, we are to execute a sample code to get familiar with code execution within a Pi.

# II. Material List

1. Raspberry Pi 3 or 4
2. Power supply adapter
3. Micro SD card (16+GB)
4. Ethernet cable
5. (optional) USB Keyboard, mouse and HDMI monitor or TV
6. We also needed to download and install Putty, Advanced IP Scanner and WinSCP

# III. Lab Procedures and Results

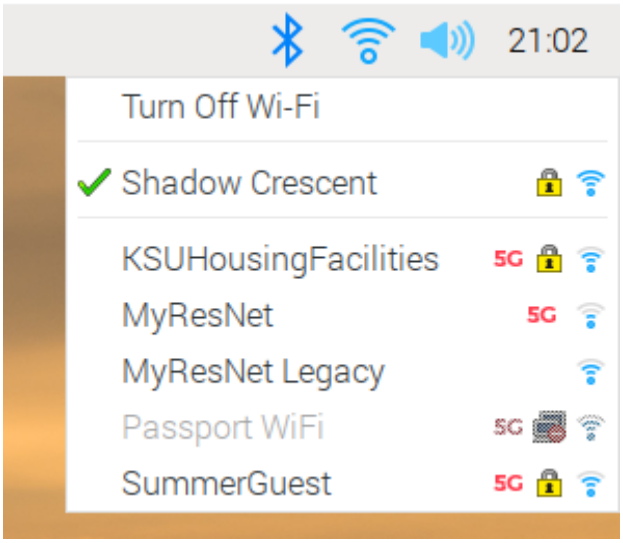
IP configuration found via Hotstop on a Windows computer:



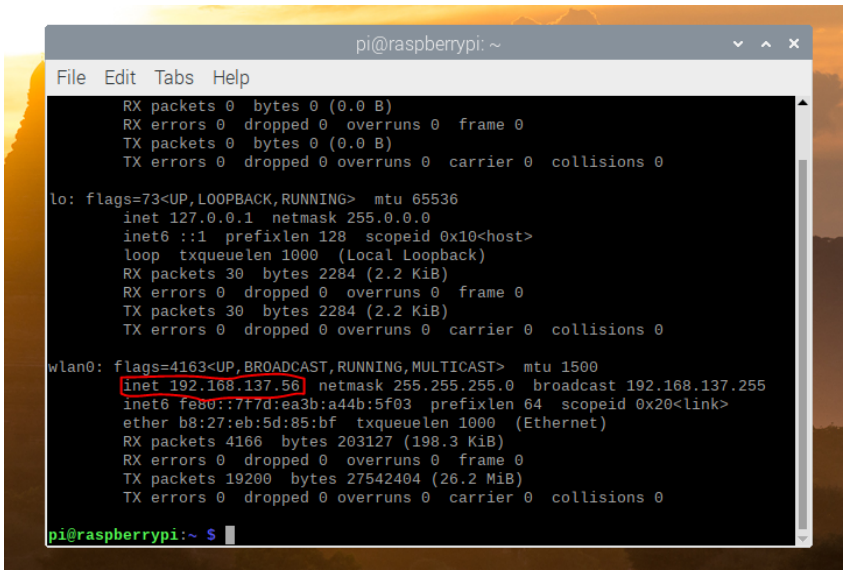
## --- Start of Procedure ---

1) Setting up Raspberry Pi Wifi through a Mobile hotspot

a. To connect the Raspberry Pi to the internet, we need to create a hotspot using either a computer or mobile device. For Windows devices, Go to Settings -> Network & Internet -> Mobile Hotspot
b. Turn on “Share my Internet Connection with Other Devices”. That will enable the mobile hotspot. You can edit the name and password for the hotspot for ease of use for later, but it is not needed.
c. To access the Pi’s interface, you must use a monitor to access the interface with an HDMI connector. Once accessed, go to the top right corner, and click the Wi-Fi symbol at the top. Locate and enter the credentials that are given from the created hotspot.

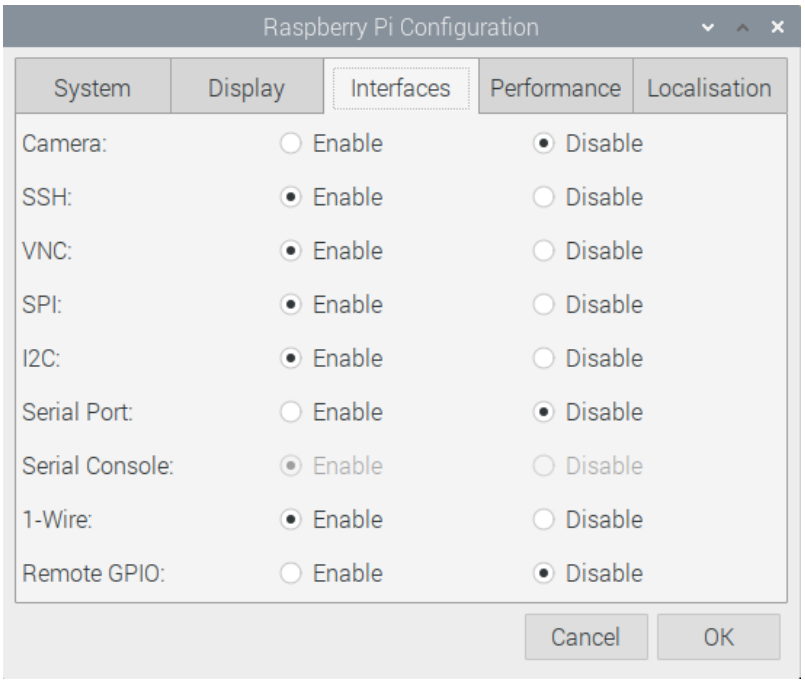


2) To find the IP Address on the Raspberry Pi, open the terminal in the top left corner and type “ifconfig” with in it to pull up the IP address which is under “inet”. Write down and save this for later. This will be important to use the SSH functionality for your Pi.




3) Enabling SSH and Connecting to the Pi via Headless Connection:

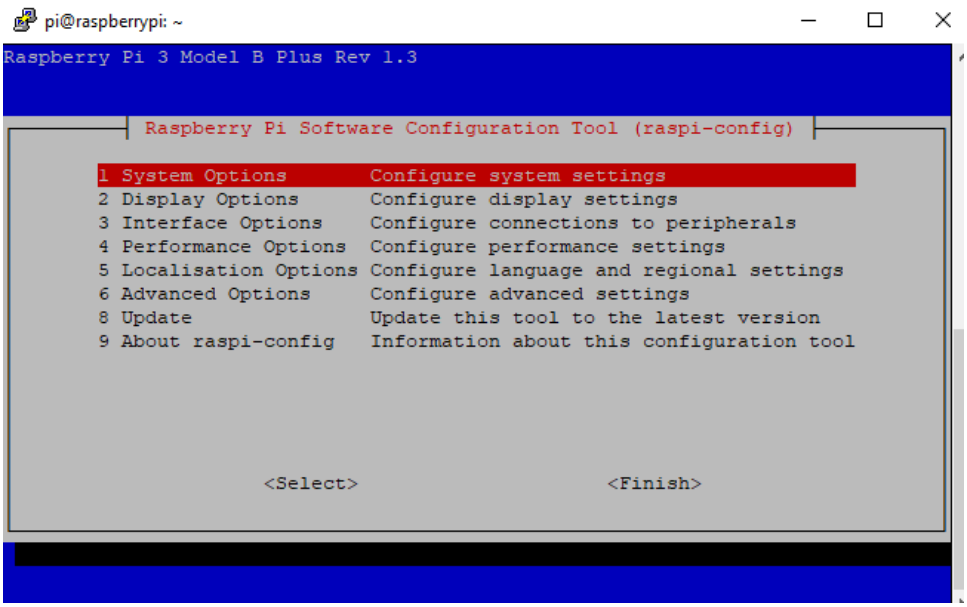
a. To enable the SSH on the Raspberry Pi, go to the Raspberry icon and go to Preferences -> Raspberry Pi Configuration -> Interfaces, then set SSH to “Enable” as shown below.




b. Go into Putty or any other SSH software on your computer and copy the IP address that was copied from the ifconfig section into the section in Putty. Once logged in, you will put in the log in credentials. As a default, the username will be “pi” and the password will be “raspberry”. Congratulations! You are now connected with your Pi via headless connection!

4) Software Configuration

a. After logging in with the SSH, in the terminal type “**sudo raspi-config"** to access the configuration menu. Afterward, you will be taken to this menu. To control the menu, the Up and Down arrow keys will navigate the menu options and the Left and Right arrow keys will control the select and back options. Hit enter to confirm the choice highlighted.

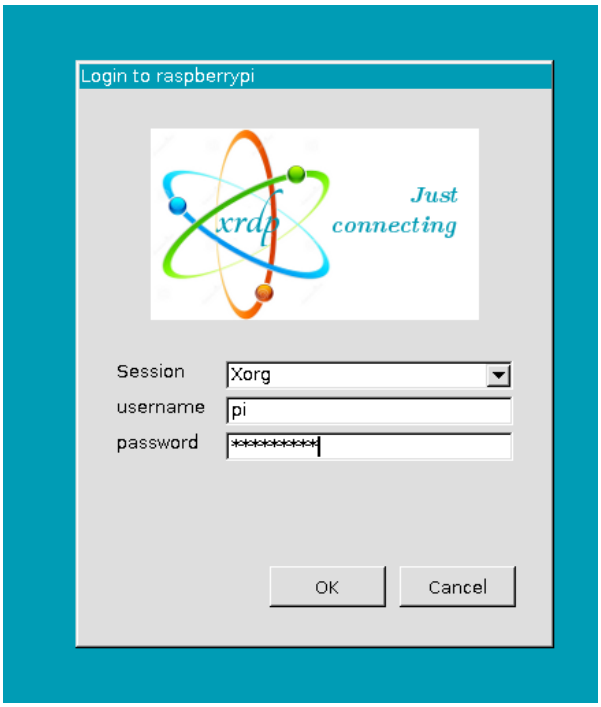



b. Go to the interface options and enable the VNC, I2C, SPI and 1-Wire by going over the option and hitting select to enable them. You must do this individually. After changing the settings and hitting Finish, you will be asked to reboot the system. Hit yes to apply the changes.

5) After accessing the Pi via SSH again and logging in, enter "**sudo apt-get update**" or "**sudo apt-get upgrade**" to update and install all packages.

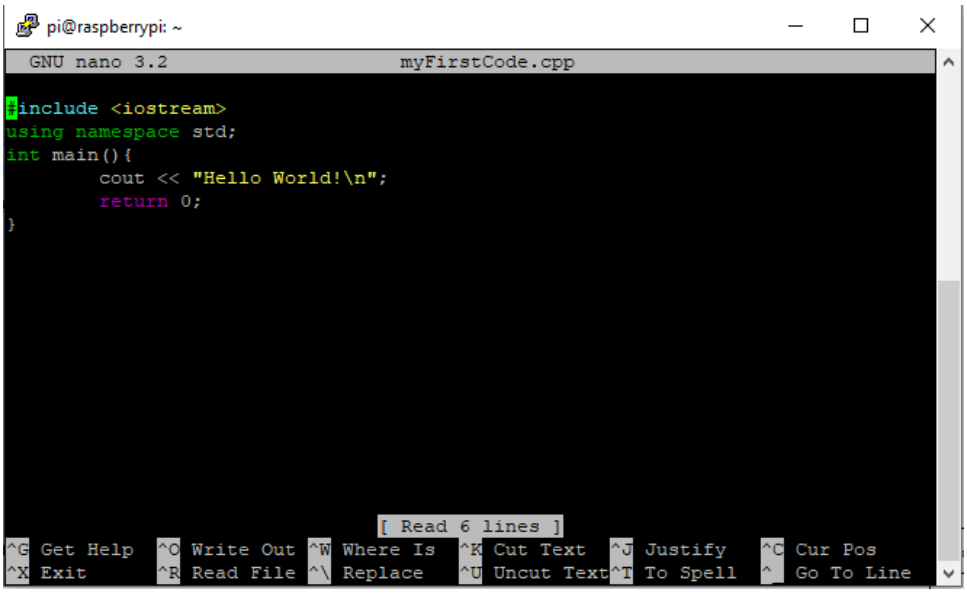
6) The next step is to setup a remote desktop connection. To start we must install the latest remote desktop protocol packages to allow for this. Enter "**sudo apt-get install xrdp"** to achieve this.

7) After installing the remote desktop package onto the Pi, just like Putty, input the IP Address of the Pi to connect to it. Afterward, you will be taken to the following screen. Just like in the terminal for Putty input the username and password of the Pi.




8) After logging into the Pi, you will reach the homepage. Just like how you would use the Pi on a separate monitor, go to the terminal and type "**ifconfig**" to make sure the information is the same as what you would use to SSH connection.

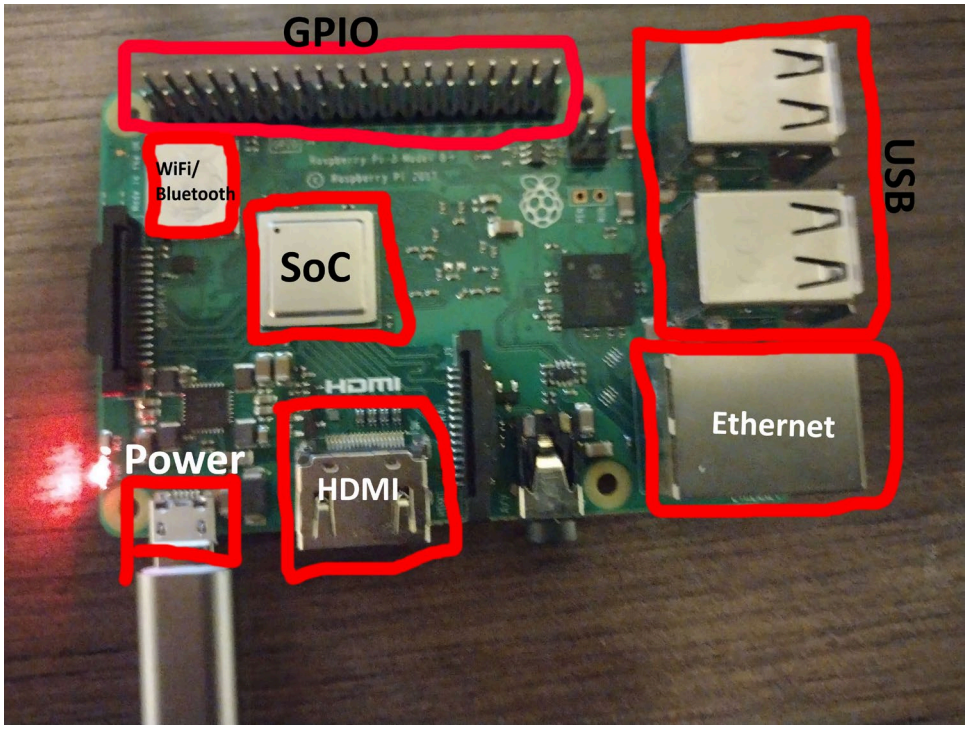
9) Our final step is to run some test code in C++. Begin by opening the RPi remote desktop terminal and then open a nano text editor. Now create a file with a "**.cpp**" extension by entering "**sudo nano myFirstCode.cpp**". You must then enter the code shown below into the nano editor.




After entering the code, enter "**Ctrl+X**", then follow it by entering Y successfully saving the file and then exit the nano text editor. Next, we need to compile the program using g++ and make our program executable. Enter "**g++ myFirstCode.cpp -o myFirstCode**" to achieve this. Last, we can execute the program by entering "**./myFirstCode**"

**--- End of Procedure ---**

The model of the Raspberry Pi is a Model 3B+.



## IV. Conclusion

In this lab, we learned how to install an operating system, set up remote connections, and compile code within a Raspberry Pi. Learning about remote desktop connection was one of the more interesting sections of the lab. Being able to connect to the Pi over the network and have a full GUI proved to be a huge benefit for us. SSH connection was also a useful tool to have for the future. We liked Remote Desktop Connection more than SSH. This is due to SSH only having command line functionality. We know that SSH is faster and more on the technical side, but the remote desktop feature allows us to have a better handle on the Pi's functionality and more hands-on control rather than remembering commands. The only issue that we encountered during the lab session was connecting the Pi to the internet; however, this was resolved easily after finding a monitor that supports HDMI.