SOFE 4610U:
Internet of Things

Lab #1: Introduction to Raspberry Pi & Sensorian and System Setup

Objectives

- Learn the Fundamentals of Raspberry Pi and Sensorian hardware
- Gain experience of setting up a development environment
- Learn the foundations of VMware and Linux systems

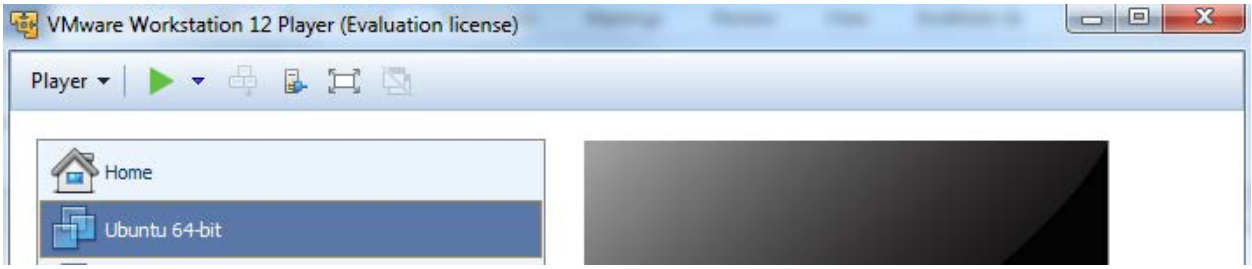
Important Notes

- Work in groups of **two or three** students
- All reports must be submitted as a PDF on blackboard, if source code is included submit your report as PDF and source files as an archive (e.g. zip, tar.gz)
- Save the submission as <lab_number>_<first student's id> (e.g. lab1_100123456.pdf)
- If you cannot submit the document on blackboard then please contact the TA with your submission: **michael.lescisin@uoit.net**

Lab Activity

Software Setup

1. Begin by downloading **VMWare player** to your computer and installing it if your host computer is running Windows. (Virtual Box is another option for this lab).
2. Enter your email to use the evaluation version which is free for all academic use. Next, download a 64-bit Ubuntu.iso image file, and install Ubuntu in VMWare. Download from: *http://releases.ubuntu.com/16.04.3/ubuntu-16.04.3-desktop-amd64.iso*
3. During the installation, you will be prompted to specify memory and virtual hard disk creation, ensure that you select the default settings by assuming that you have more than 2 GB memory spare on your hard drive.
4. Once installed, you will see the Ubuntu from the left side panel



5. If you are using Ubuntu as your host machine, you are welcome to use virtual environment as well so you don't have to worry about editing your path and changing files on your host machine

Hardware Setup

There are two main hardware pieces that are used in this course, Raspberry Pi and Sensorian Shield. As it shown,

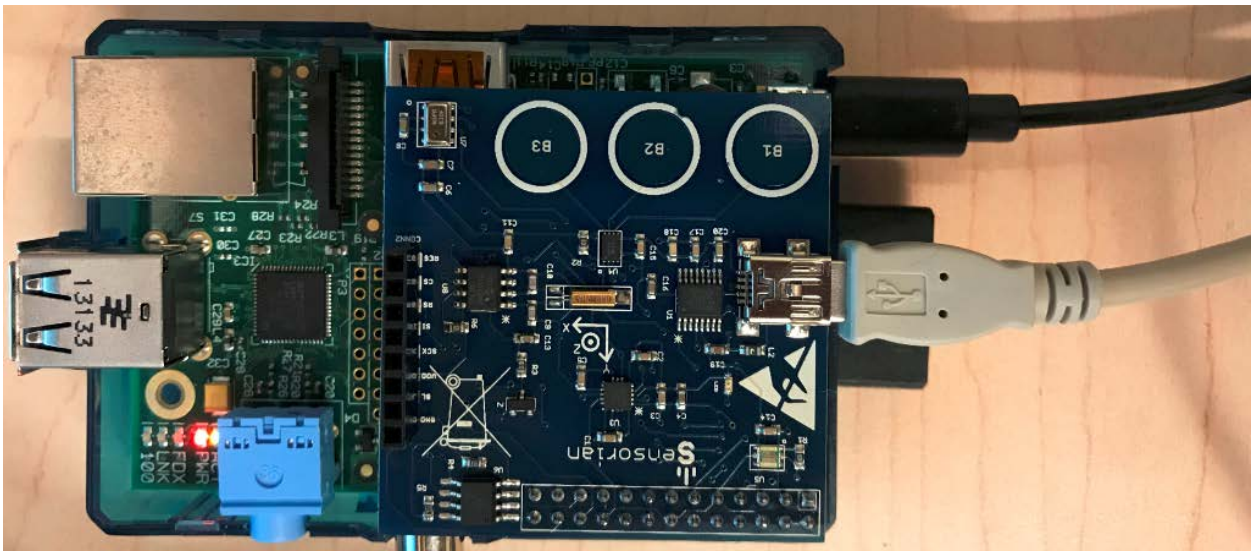


Fig. 1 Raspberry Pi and Sensorian Shield

As seen, the Sensorian board is on the top of the Raspberry Pi board. There are two cables shown in the picture, the light color cable is the mini B USB cable that connects with Sensorian Shield, and the black cable is the micro USB cable that connects with the Raspberry Pi board.

Also, there are RJ45 regular network cables and keyboards where you can plug and play with a monitor that includes HDMI inputs.

Network Setup

As mentioned, the development environment setup is crucial to software engineers or developers. Once the environment is setup, it is much easier to work on the software development.

There are two major ways to set up the network on Raspberry Pi with Sensorian. The software tool to use is PuTTY which has both Windows and Linux versions.

Using Sensorian Shield as the gateway:

When there is no Internet Connection, this method comes handy. The steps can be followed as follows,

1. Install the Sensorian Shield on top of the Raspberry via GPIO pins
2. Power up Raspberry Pi using Micro USB cable

3. Plug the mini B USB cable to Sensorian Shield and connect to your laptop
4. Let the driver install. (For Windows 10 users, there might be issues to download the driver. Please do inform the instructor or lab TA to assist you)
5. Go to Control Panel to find out which USB port has been used for the mini B USB cable
6. Use the COM port from device manager and set speed to 115200 and select Connection type as Serial

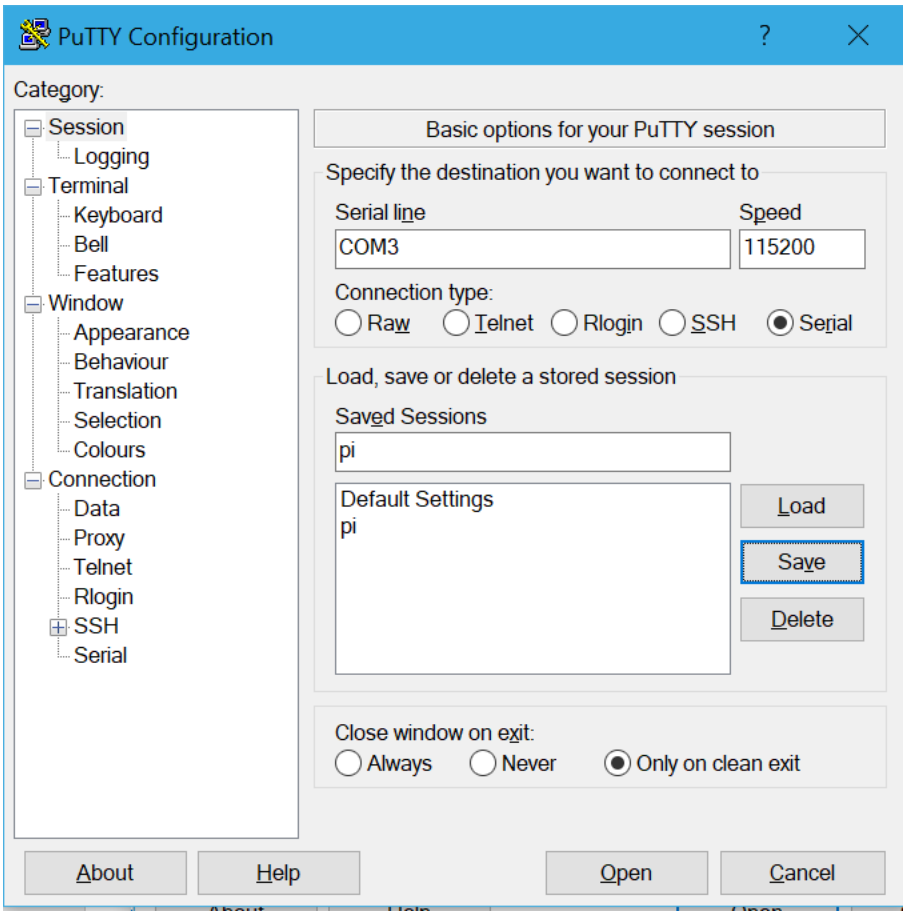


Fig. 2 Putty Configurations

7. To confirm the settings: 115200 8 N 1, go to Serial under SSH

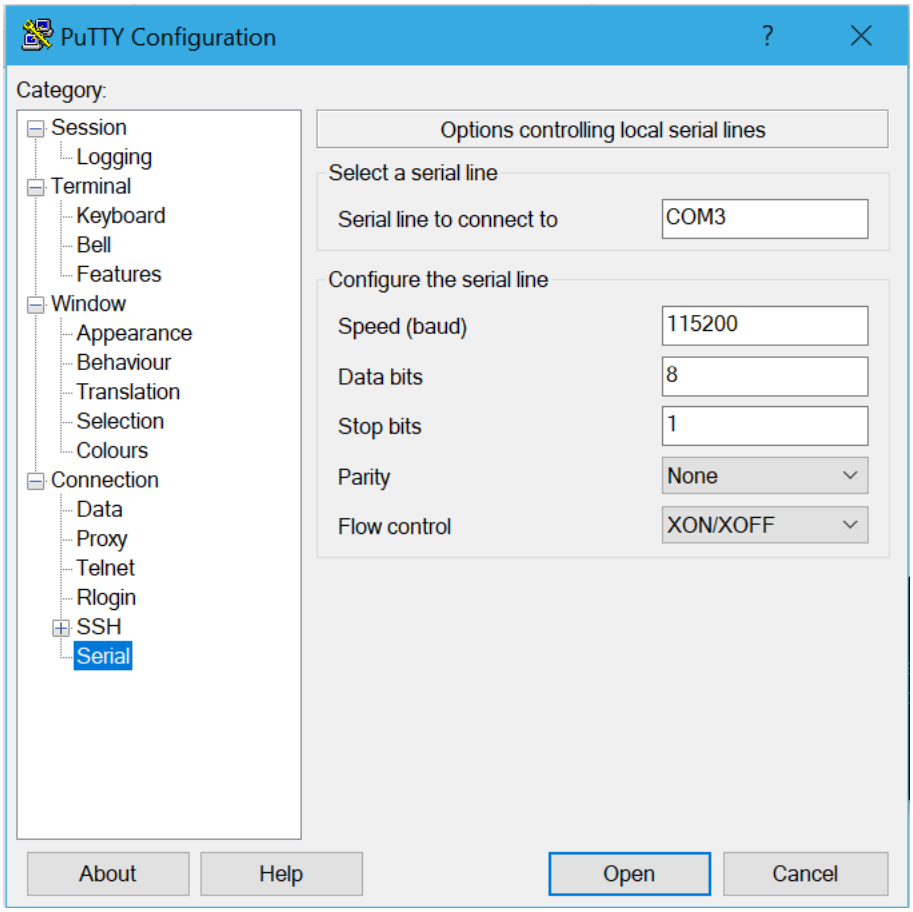
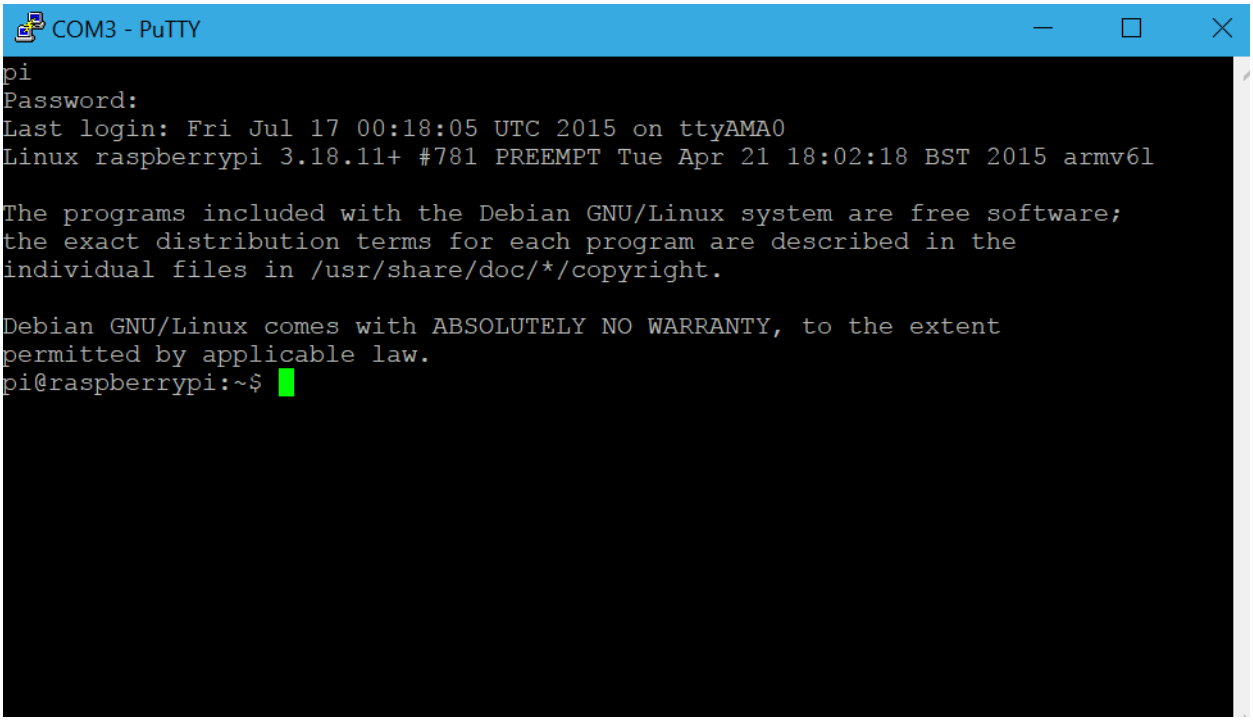


Fig.3 Port Settings

8. Click on open to get on the Raspberry pi



From this point on, you should be able to get on board to start exploring the Linux environment on the Raspberry pi.

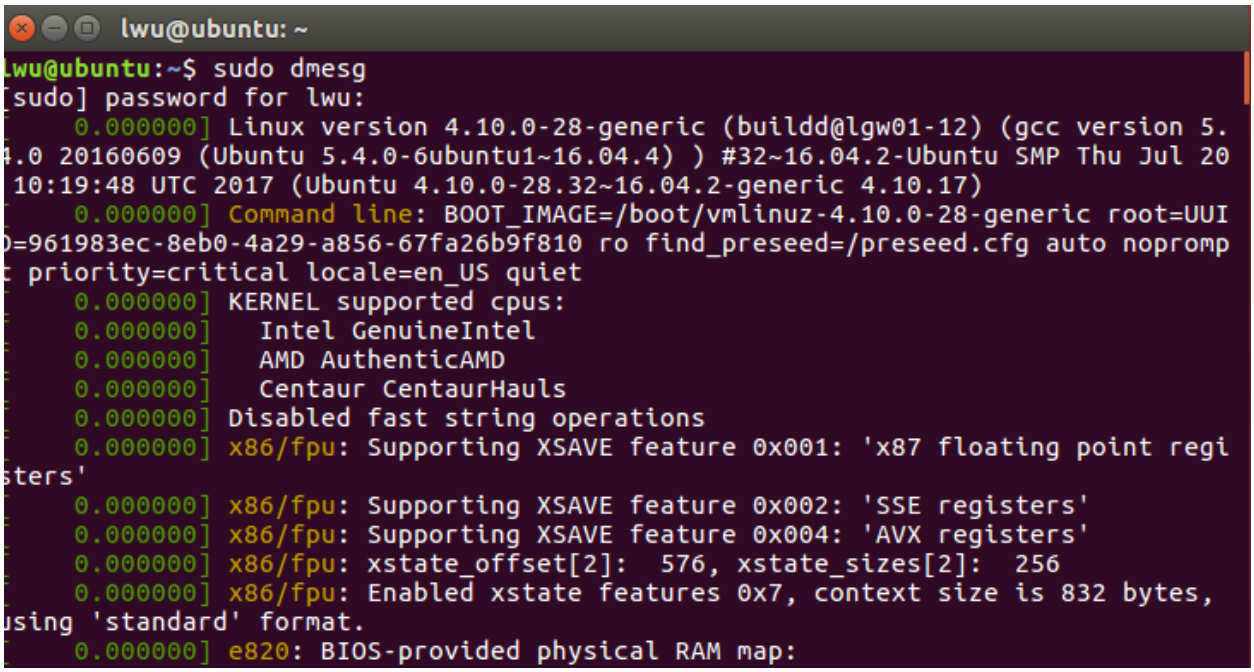
Lab Tasks

Network Configurations:

For security reasons, the network setup should be connecting the network cable provided directly between the Raspberry Pi and your laptop. This way will prevent illegal intrusion from other laptops logging into the Pi.

1. Start the Virtual Machine and power up the Ubuntu image.
2. Run command to display device drivers presented in the kernel to relevant hardware.

```
sudo dmesg
```



3. Update the package cache by running the command

```
sudo apt-get update
```

4. Install PuTTy

```
sudo apt-get install putty
```

5. Add your username to the *dialout* group, then logout and log back in.

```
sudo usermod –aG dialout USERNAME
```

6. Open PuTTY to on the Ubuntu VM to get ready to access the Raspberry Pi.

In order to keep privacy and security for each group, please use the network cable provided to connect the Raspberry Pi and host as shown below.

After powering up the Raspberry Pi, the host and Pi should obtain an auto-configuration IP address for their Ethernet interfaces. To obtain these IP address from the Pi, with the serial Mini B USB connection to the Pi, run the following command:

```
ifconfig eth0
```

- A route to the Raspberry Pi must be made in the Ubuntu OS. Within the Ubuntu VM run the following command:

```
route
```

- Create the route to the Raspberry Pi by running the command:

```
sudo route add IP_OF_PI gw GATEWAY_IP
```

  Replacing IP_OF_PI with the IP address of the Pi's Ethernet interface and GATEWAY_IP with the IP address obtained in the previous step.

- You should now be able to ping the Raspberry Pi from the virtual machine by running the command:

```
ping IP_OF_PI
```

replacing IP_OF_PI with IP address of the Pi's Ethernet interface. It should return:

```
64 bytes from ...
```

if all is working correctly.

Network Login

- Close the current PuTTY serial session open on the virtual machine.
- Open PuTTY once again on the Raspberry Pi this time selecting SSH as the Connection type, enter the IP address of the Pi's Ethernet interface as Host Name (or IP address) and leave Port as the default 22.
- Click Open and accept the hostkey.
- You should now have SSH console access with port default setting 22 to the Raspberry Pi.

Turn on Sensorian LED

This is the lab task that you will write C code to turn on and off Sensorian LED every 3 seconds. The bcm2835.c, bcm2835.h, Utilities.c and Utilities.h will be given to you. You can download Winscp to transfer files to the Raspberry Pi. You also are welcome to use direct transfer protocol in Linux to transfer files. You can use the following code as the reference.

```
while (1)
   {
      digitalWrite(LED_PIN, HIGH);
      delay_ms(1000);
      digitalWrite(LED_PIN, LOW);
      delay_ms(1000);
   }
```

Show demo to lab instructor or lab TA when you complete both tasks.

Deliverables

Complete all lab tasks, for task 1 how to logon to the Raspberry Pi from your host, write the step by step instructions including screenshots; for task 2, submit the complete code

and Makefile along. Please note, all lab report will have title pages, introduction, content of the lab tasks, and conclusion.