1 Milestone I – initial setup and tests

The **main objectives** of this first milestone are:

- To learn how to set up and interact with a Raspberry Pi
- To get familiar with SCP command to copy files from PC to Raspberry Pi.
- To test simple python script with command line arguments

In terms of output, **students will be asked to demonstrate** several commands and programs discussed throughout this section.

1.1 Material

The main materials used throughout this project are:

- Raspberry Pi 4
- Grove Base Hat for Raspberry Pi
- Grove Base Sensor Kit for Raspberry Pi
- Dongles and cables (Ethernet cables, USB-C to USB adaptors, ethernet switches, etc.)

> Due to limited availability, not every group will get the *full* Sensor Kit. however, separate sensors (the same as in the kit) are also available in the lab.

Carefully go through the sensor kit introduction to get to know the hardware basics of this project.

1.2 Raspberry Pi configuration

There are various ways to configure a Raspberry Pi. The first thing one needs to do is installing an image (of the operating system) on the Pi.

1.2.1 Image installation

- Download latest Pi imager at https://www.raspberrypi.org/software/
- Open the imager
- Choose Raspberry Pi Device: Raspberry **Pi 4**
- Choose Raspberry Pi OS (Legacy, 32 bit): **Bullseye** with desktop
- Choose storage: select SD card or USB device
- Open Edit Settings
  - set default username as ***pi*** and password as ***raspberry*** (**Don’t change default username and password**)
  - enable SSH in SERVICES and Use password authentication
- Save the settings and Write
- Eject SD card / USB

> - https://projects.raspberrypi.org/en/projects/raspberry-pi-setting-up
> - https://www.raspberrypi.org/software/

1.2.2 Connect to the Raspberry Pi

After installing the image on the Pi, there are several ways to connect to it. One option is to **directly access** the Pi by connecting a monitor (HDMI) and a keyboard to it. Another option is **remote access**, for example, a SSH connection over WiFi, or a physical connection with an Ethernet cable between the Pi andyour computer. **This document will use both of these options**. The connection between the Pi and your laptop will go **via an Ethernet cable** (you can use the switches provided in the lab). This gives us **local access to the Pi**. Enabling SSH is necessary if you want to connect to the Pi locally or remotely. If you have forgotten to enable SSH in settings, you still can do it with the following steps:

- injecting the SD card (or USB stick) once again into your computer (with image already installed)
- opening the "`boot`" partition (this is a folder)

This boot-folder include several files. **Create** “ssh” file in this directory. You don’t have to add anything in this file (i.e. an empty text file,no extension). The SSH file will enable the SSH service.

In most cases, you will be able to access the Pi over SSH (using your computer's Command Line Interface (CLI)) with:

```
$ ssh pi@raspberrypi.local
```

followed by typing the **password** "raspberry".

If, for some reason, this does not work, for example, the local hostname does not get resolved, you need to **determine the IP address of the Pi** (for instance, by using some kind of IP scanner) or you can assign a static IP in different ways.As already mentioned, one can always connect an external monitor (+ keyboard) to the Pi, and find out itsIP address like that, e.g., with:

```
$ ifconfig eth0
```

From here on out, it is ***assumed*** the IP address of the Pi is `192.168.1.100` (note that this is just an example). In this case, the previous login becomes:

```
$ ssh pi@192.168.1.100
```

Again, with the same password.

> From now on, the notation `pi@raspberrypi.local` will be used. If necessary (e.g. domain not resolving automatically), just replace this with the actual IP address, after "@".

Since we also want to **connect our Pi to the outside world** (clouds), we have to make sure it is connected to a public network (WAN) as well, in this case **over WiFi**. As such, one has to configure a wpa_supplicant on the Pi. While we can use our local connection to do the configuration, we can also directly configure files on the disk (SD card). Let's do that, starting with:

- injecting the SD card (or USB stick) once again into your computer (with image already installed)

- opening the "`boot`" partition (this is a folder)

Create wpa_supplicant.conf in this directory:

1. Put the required configuration in this file in order to connect to the WiFi network. Just create a new file "`wpa_supplicant.conf`", and add the following lines to it:

```
country=BE
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
  ssid="your-wifi-name"
  scan_ssid=1
  psk="your-wifi-pw"
  key_mgmt=WPA-PSK
  priority=1
}
```

The WiFi name and password should be the ones **configured on your router**. Now, eject the SD card from your computer and insert it on your Raspberry Pi. Then, power on your Raspberry Pi. Your Raspberry Pi should be connected to the WiFi network. One can **verify this** by logging in to the Pi, and executing the following command (can be any public domain):

```
$ ping www.google.com
```

This should **output** messages (pongs) similar to:

```
[...]
64 bytes from lhr35s11-in-x04.1e100.net (...): icmp_seq=1 ttl=59 time=24.5 ms
64 bytes from lhr35s11-in-x04.1e100.net (...): icmp_seq=2 ttl=59 time=27.3 ms
64 bytes from lhr35s11-in-x04.1e100.net (...): icmp_seq=3 ttl=59 time=28.7 ms
64 bytes from lhr35s11-in-x04.1e100.net (...): icmp_seq=4 ttl=59 time=22.5 ms
64 bytes from lhr35s11-in-x04.1e100.net (...): icmp_seq=5 ttl=59 time=24.4 ms
[...]
```

- https://www.raspberrypi.org/documentation/remote-access/
- https://www.learnrobotics.org/blog/raspberry-pi-without-a-monitor/
- https://www.eightforums.com/threads/trying-to-make-a-conf-file.24021/
- https://www.raspberrypi.org/documentation/configuration/raspi-config.md
- https://tutorials-raspberrypi.com/raspberry-pi-remote-access-by-using-ssh-and-putty/

1.2.3 Install VNC viewer as remote desktop

While one could easily setup and use the Pi through the command prompt (e.g. over SSH), it is also possible to install VNC viewer, i.e. a "remote desktop", as also explained here. Professional Software

Developers use virtual desktops through VNC server in Pi, such that developers can simultaneously access the same Pi to manage different tasks such as coding, debugging, and monitoring system performances at the same time. It is mandatory that each team member should work with Pi in parallel and for that you could use virtual instances of VNC viewer.

- https://www.raspberrypi.org/documentation/remote-access/vnc/README.md
- https://www.realvnc.com/en/connect/download/viewer/
- https://www.youtube.com/watch?v=GnzRS3AgW5U

In some cases, you also need to set the resolution on the Pi (e.g. to 16:9), as discussed in this article.

1.2.4 Command line arguments in Python

Create the following python script in your laptop (multiply.py).

```
import  sys

def multiply(x,y):
    return(x*y)

if __name__ == '__main__':
    num1 = float(sys.argv[1])
    num2 = float(sys.argv[2])
    result=multiply(num1,num2)
    print("Result=",result)
```

1.2.5 Copy files from PC to Raspberry Pi (secure copy)

When working with remote software and data, it is often desired to copy files from one system to another in a secure manner. Especially when you cannot physically reach the remote system, you are not able to use USB sticks/flash drives to copy data. As such, a short introduction on Secure Copy (SCP) can be found here.

Log in to the Pi, and create a new directory (choose a directory name):

Watch out when copying code that you maintain the right python indentation (spaces at the beginning of a code line).

```
$ ssh pi@raspberrypi.local
```

Subsequently, in the console of the Pi:

```
$ mkdir ~/Documents/my-ib2a
```

In your own system (terminal/prompt), execute the following command (change the directory names if necessary):

```
$ scp multiply.py pi@raspberrypi.local:~/Documents/my-ib2a
```

On the Raspberry Pi, in the right directory, you can now run the following (the second command allows for user input):

```
$ python3 multiply.py <num1> <num2>
```

Secure copies with the `SCP`-command can be executed in both directions, that is, copying files from Pi to your PC, and vice versa.

- https://www.raspberrypi.org/documentation/remote-access/ssh/scp.md
- https://howchoo.com/pi/how-to-transfer-files-to-the-raspberry-pi
- https://www.rickmakes.com/copy-files-from-windows-10-to-a-raspberry-pi-using-scp/
- https://stackoverflow.com/questions/39784136/how-to-add-external-arguments-in-python-script

1.2.6 Familiarize with basic CLI commands

If you are new to this kind of computer interactions, for instance, working in command prompts, development can be difficult or inefficient at first. You may use any online resources to practice CLI commands. Demonstrate in milestone I video that you have executed these basic commands.

1.2.7 General tutorials (optional)

User interfaces:

- https://projects.raspberrypi.org/en/projects/raspberry-pi-using/5
- https://projects.raspberrypi.org/en/projects/custom-pi-desktop/

Terminal / command prompt / console:

- https://projects.raspberrypi.org/en/projects/pacman-terminal/1
- https://projects.raspberrypi.org/en/projects/raspberry-pi-command/

General:

- https://learn.adafruit.com/an-illustrated-shell-command-primer/moving-and-copying-files-mv-and-cp