

Prelab 1.1 IoT Sensor Data Collection 1 - Temperature Sensor

Learning Goals

- Students will be able to prepare the basic software requirements in Raspberry Pi to measure sensor data.
- Students will be able to code using basic python functions related to date and time

1.1 Introduction

In Lab 1, we will use a digital temperature sensor, DS18B20 module (<https://www.analog.com/en/products/ds18b20.html>, available on Dec. 19, 2024), to measure atmospheric temperature. It uses a capacitive thermistor to measure the surrounding air and splits out a digital signal on the digital pin.

The DS18B20 module is simple to use but requires careful timing to grab data.

Warning! The only real downside of this sensor is you can only get new data from it once every 750 milliseconds.

There are many applications using the temperature sensor, such as HVAC (Heating, Ventilation, and Air Conditioning), testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, humidity regulator, and so on.

Prelab 1 is to prepare the DS18B20 module use for Lab 1 and to get familiar with Python. In all lab manuals, Python means Python3 if no further descriptions. We will use Raspberry Pi and Python. Lab manuals will mainly describe the use of 'Terminal'. However, if you are a beginner of Python, you can also use Thonny IDE, the default software of Raspberry Pi OS. If you are interested, visit and follow this tutorial (<https://roboticsbackend.com/thonny-ide-raspberry-pi-os/>, available on Dec. 19, 2024).

1.2 Python Package Installation

First, you need to make Raspberry Pi have the latest packages.



Raspberry Pi - Terminal

```
sudo apt update
```

```
sudo apt upgrade
```

To install Python additional packages, we will use 'pip' package. The 'pip' is the standard package manager for Python. It enables you to install and manage additional packages that are not part of Python library. Let's install 'pip' package first.



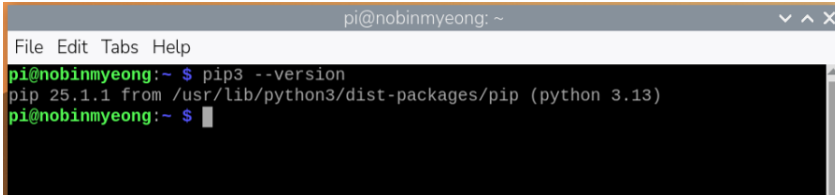
Raspberry Pi - Terminal

```
sudo apt install python3-pip
```

To check 'pip' version, you can use the command 'pip --version' in 'Terminal'. To see all installed Python packages and their versions, use 'pip list' in 'Terminal' as below.

```
pip3 --version
```

```
pip3 list
```



```
pi@noblinmyeong: ~  
File Edit Tabs Help  
pi@noblinmyeong:~ $ pip3 --version  
pip 25.1.1 from /usr/lib/python3/dist-packages/pip (python 3.13)  
pi@noblinmyeong:~ $
```

Figure 1 Terminal window to check pip version

Task 1.1

Capture your terminal window as Figure 1 after checking pip version and attach it to the report below:



```
pi@robertclaud: ~  
pi@robertclaud:~ $ pip3 --version  
pip 20.3.4 from /usr/lib/python3/dist-packages/pip (python 3.9)  
pi@robertclaud:~ $
```

To access data from DS18B20, we will use 1-Wire interface of Raspberry Pi. The steps for enabling the 1-Wire interface are described in GUI (VNC access). If you want to do it in CLI, please use `sudo raspi-config`. Continue to follow the procedure below for the remainder.

1. Click 'applications menu' (Pi logo on the upper left side)
2. Click 'Preferences'
3. Click 'Control center'
4. Click 'Interfaces' tab
5. Click 'Enable' of '1-Wire' as Figure 1.
6. Reboot Raspberry Pi

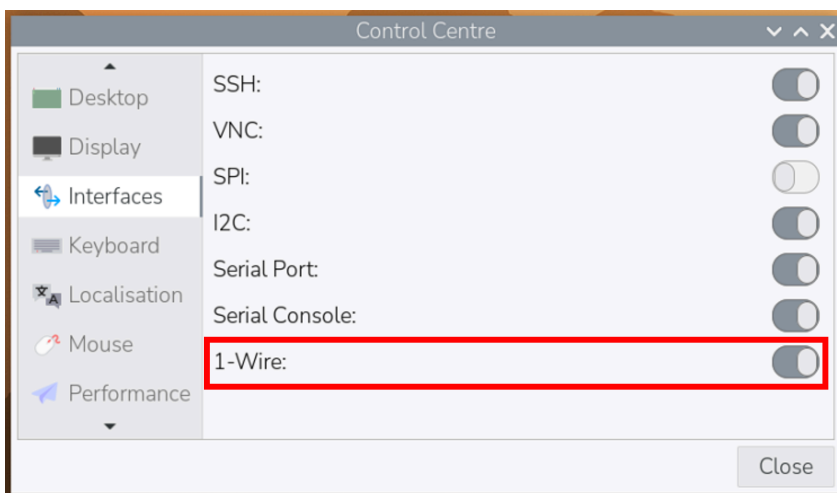
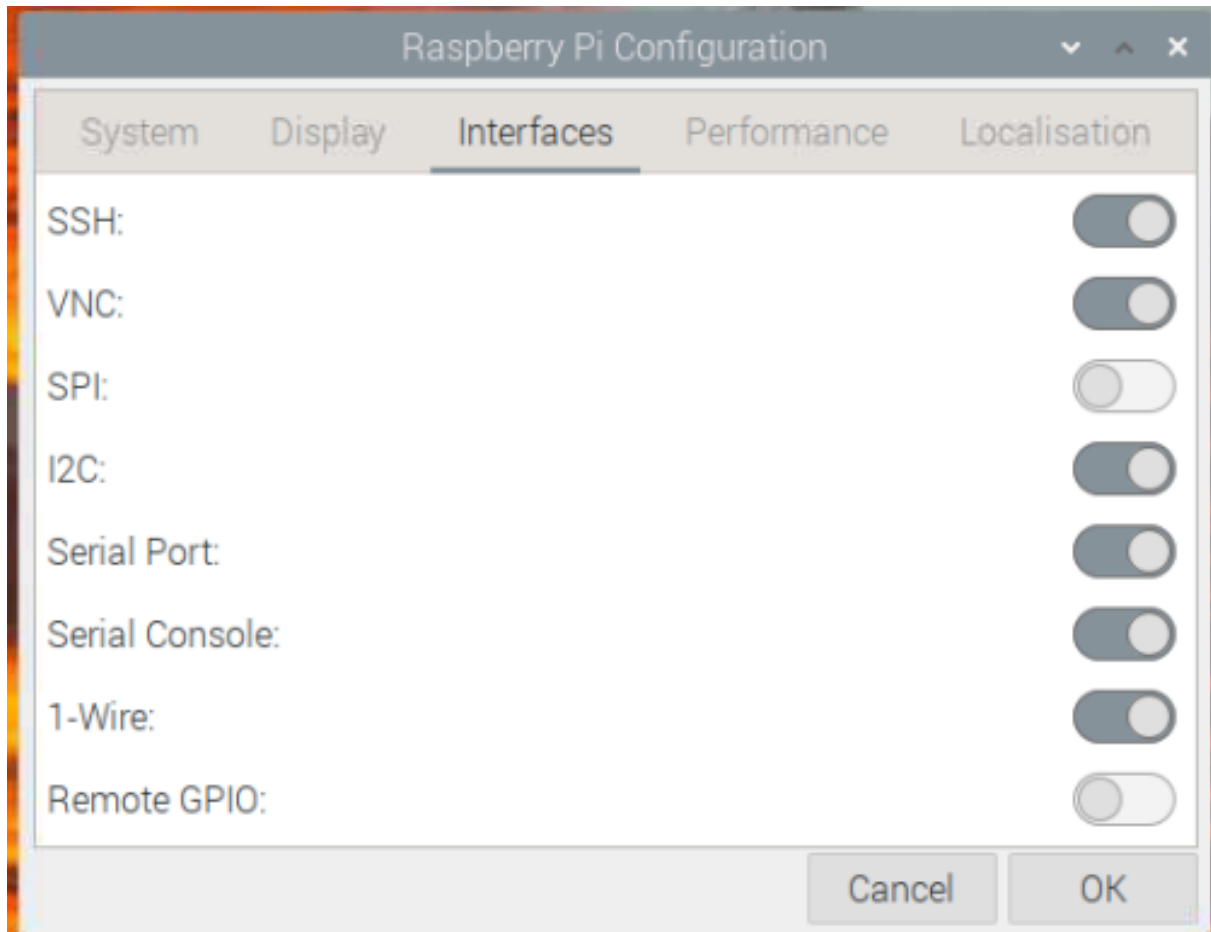


Figure 2 Enabling 1-Wire

Task 1.2

Capture your terminal window as Figure 2 to see the configuration and attach it to the report below:



Finally, install 'matplotlib' Python package by using 'pip3' command. We will use 'matplotlib' package to visualize data in the following lab sessions.

Raspberry Pi - Terminal

```
sudo pip3 install matplotlib
```

Please continue to [Prelab 1.2 here](#).