**Calibration:**

Please refer to this document: [*https://docs.google.com/document/d/1MJ3P9vXbwytBE8vtJQ59elSed23MnNGf8xYxmcMEyBU/edit*](https://docs.google.com/document/d/1MJ3P9vXbwytBE8vtJQ59elSed23MnNGf8xYxmcMEyBU/edit)

***Usage:***

1. Install the MAX6675 library in arduino IDE
2. Connect the arduino hardware with PC
3. Select proper Board & Port in the IDE (tools>port; tools>board)
4. Verify, compile & upload the code to the board.
5. Connect the Max6675 module with board & sensor terminals.
6. Place the sensor properly on the testing heat bed.
7. Open serial monitor through Arduino IDE & turn on the heater to check if data is coming properly in the serial buffer.
8. Close the arduino serial monitor & turn off the heater.
9. Check the serial port number in the IDE (go to tools>port in the arduino IDE; e.g. “/dev/ttyACM0” or “/dev/ttyACM2” etc.)
10. Open the “serial\_datalogger” python script in any text editor & correct the serial port address, if necessary & save it.
11. Open linux terminal & run the script using the following command & put user password to give USB access to the python script.

* sudo python serial\_datalogger.py (it is assumed that python is already set up in the PC)

1. In the script folder a new “log.csv” file will be created & updated by the python script. Do not open it until the data logging is done. For reference, the logged data can be seen on the linux terminal as well.
2. Turn on the heater & collect the data from 25\*C (or ambient temp.) to 105\*C & hit “ctrl+c” to stop the script.
3. Go to the script folder & rename the log.csv file to a specific file name.
4. Run the python script again, turn off the heater & collect the data from 105\*C to 25\*C (or ambient temp.) & hit “ctrl+c” to stop the script.
5. Go to the script folder & rename the log.csv file to a specific file name.

NOTE: step 7 & 8 are entirely optional. One can directly goto step 9 & carry out the experiment.

***Result:***

Time vs. Temperature data will be obtained for different heaters.

***Troubleshooting:***

1. **No data stored in log.csv file**: make sure that the arduino serial terminal is closed, else the python script will not be able to access the same serial port if it is already taken by arduino.
2. **Permission denied error:** make sure to use sudo while running the script.
3. **No such file or directory:** make sure if python is properly setup in the PC.
4. **Serial port not found:** check the serial port in the Arduino IDE & put the same on the python script.
5. **Garbage data in the log file:** Make sure that the baud rates are same in both the Arduino & python code. For example, 115200 is already set in both codes.
6. **Import error:** Make sure to install all the necessary python libraries; serial, io, datetime etc.

***Resources:***

1. **Installing arduino library:** <https://www.arduino.cc/en/Guide/Libraries#toc4>
2. **Installing python from source code:** <https://tecadmin.net/install-python-2-7-on-ubuntu-and-linuxmint/> (tested-OK in Ubuntu Mate 16.04.5 LTS)
3. **Installing python via Deadsnakes PPA:**

sudo add-apt-repository ppa:deadsnakes/ppa

sudo apt-get update

sudo apt-get install python2.7

Other versions, such as python2.4 or python3.6, etc. are also available.

**Source:** <https://askubuntu.com/questions/101591/how-do-i-install-the-latest-python-2-7-x-or-3-x-on-ubuntu>

1. **Installing python serial library:**

sudo apt-get install python-serial python3-serial

sudo pip install pyserial

**Source:**

<https://pythonhosted.org/pyserial/pyserial.html>