```
# Written by Todd Micallef
# linux_gpib_readme.txt

#    Rev 1.0 Initial release
#    Rev 1.1 Several capitalization mistakes corrected
#            Some steps added that were missing
#            Added i2cdetect step to verify hex address of BME280
sensor
#    Rev 1.2 added note that Wifi country change will prompt for
reboot. Otherwise no reboot needed
#            added note that thp_server_config.ini will need to be
modified for BME280 address change
#    Rev 1.3 All python files have been converted to python3
compatible code
#            Added sqlite logging option. The script is inside the
thp_client directory and is called from a cron job
#            Using Adafruit pitft installation guide. New issue is
that the 320x240 displays are scaled x2 to 640x480 requiring the /
boot/config.txt to be corrected
#            https://learn.adafruit.com/adafruit-pitft-28-inch-
resistive-touchscreen-display-raspberry-pi/easy-install-2
#            The Adafruit code is now python3 compatible
#            The latest Raspberry Pi OS https://www.raspberrypi.com/
software/operating-systems/
#            Since this script was written to support the RPI 3B and
above, the 64-bit version of the OS has been tested only on the latest
script
#            Raspberry Pi OS with desktop  Release date: April 4th
2022 is the tested download
#            Raspberry Pi Imager should be used since the default pi
user is no longer the default login. Enable ssh under advanced
options.
#            The blank ssh file is not needed if ssh is enabled under
advanced options
#            Modify the localization setting in the advanced options
#            https://www.raspberrypi.com/software/

# https://www.seeedstudio.com/blog/2020/07/07/raspberry-pi-rtc-
tutorial-using-ds1307-and-ds3231-rtcs-with-raspberry-pi-m/
```

The pi or other created user will only have ssh access. Root access is not enabled for security reasons
It is recommended for security reasons not to use the old default password of 'raspberry'
It is assumed the installer has some experience using ssh and knows the IP address of the RPI
Do not update/upgrade the software before running this script. It will be performed before the gpib software is installed

Other packages will also be installed. Some are in support of other features other than linux-gpib

For BME280 support the thermo_installer.run file should also be copied

BME280 support will need the sensor attached to the I2C bus. Use the 3.3v supply for the sensor.
ds1307 or ds3231 support will need the RTC hardware attached to the I2C bus. Connect it in parallel to the BME280

VNC support does not use the default realvnc server and is deleted during the update. tightvnc and x11vnc are used due to their wide support by vnc clients

Linux-GPIB support will need the USB interface plugged in before the drivers are installed. The NI USB controller works best and a hot plug script is added.

gpib_config should not be needed on reboot. This is handled automatically now.

Installation Steps:

1. Download and install the Raspberry Pi OS to a sd card using the Raspberry Pi Imager software (v1.7.2 was tested with no issues)
2. The sd card will need to be reattached to the pc if it was software ejected so that 2 files can be copied to its /boot directory
3. Copy the files "install_linux_gpib.sh" and "thermo_installer.run" for BME280 and TFT hat support
4. Eject the sd card using your OS software utility and insert the sd card into the RPI. Turn on the RPI
5. It may take a few minutes as the OS boots and does some automatic configuration
6. ssh pi@'IP address' (or whatever user was used during the imaging process) and enter the password
7. sudo raspi-config
8. "Interface Options" -> SPI, and I2C should be enabled
9. Do not change boot options yet. They will be configured later
10. Finish and exit raspi-config. 'Yes' to reboot if prompted. (reboot prompted if WiFi country is changed)
11. ssh into the RPI (step 6) if the RPI was rebooted
12. cd /boot
13. bash install_linux_gpib.sh
14. 'y' to update and install system software.
15. 'y' to reboot once the update finishes.
16. Plug in the usb based gpib adapter before installing the linux-gpib software
17. ssh into the RPI (step 6) if the RPI was rebooted
18. cd /boot
19. bash install_linux_gpib.sh

20. 'n' to update software
21. 'y' to install linux-gpib
22. 'y' or 'n' to install teckit
23. 'y' or 'n' to install samba
24. 'y' or 'n' to install hardware support for LCD hat
24a. PiTFT hat support:
'1' to install PiTFT hat drivers
Select the correct PiTFT hat (2)
Select the rotation. (270 puts the HDMI port on top)
'n' to put console output on the display.
'y' to mirror HDMI output on the display
'y' to reboot. Next option will fail unless the pi is rebooted
24b. goodtft hat support:
'2' to install goodtft hat drivers
The installation will run and the RPI will reboot to the desktop

25. At this point linux-gpib has finished installing if the RPI rebooted on the last step. You can quit here or continue to add BME280 support
26. The Desktop should now be on the lcd if Pitft support was installed
27. ssh into the RPI (step 6) if the RPI was rebooted
28. Before adding support for the BME280, it will be important to know its address on the i2c bus.
Run the following command and note its hex address: "i2cdetect -y 1"
The number will be in hex. ie 76 = 0x76 or 77 = 0x77
The datasheet only lists two possible hex addresses. If a RTC module is attached, it should also be displayed.
ie. the DS3231 can be at address 0x68.
29. cd /boot
30. bash install_linux_gpib.sh
31. Select 'n' until prompted to add Fluke 1620 emulation using the BME280
32. Select 'y'
33. Select 'n' to display THP data on the lcd. Otherwise, select 'y' for headless display. Selecting 'y' will not install the gui server
34. Select 'y' or 'n' to enable logging of data to a sqlite database 1 rdg/min default (can be changed in crontab)
35. Select 'y' or 'n' to enable x11vnc server to remotely view LCD display
36. Create a password for the user login for x11vnc
37. Select 'y' or 'n' to save the password in /etc/x11vnc.pass
36. Select 'y' or 'n' to install support for a ds1307 or ds3231 hardware RTC
Select '1' or '2'
37. Select 'y' or 'n' to enable tightvnc server to remotely access the desktop

38. Create a password for the remote login
39. Select 'y' or 'n' to have a view-only password.
36. Select 'y' to reboot
37. If the THP data isn't displayed, the launcher.sh file in /home/pi may not have the right permissions. sudo chmod +x launcher.sh and then 'sudo reboot'. Also, the /root/thp_server/thp_server_config.ini file might need the BME280 address changed.
38. If the screen is oriented in the wrong direction, it can be changed in the /boot/config.txt file at the bottom. Default is 270 degrees
39. If the diplayed readings are too small, the Adafruit installer used a scale factor. Correct it by doing the following:

```
       ssh into the RPI (step 6)
       sudo nano /boot/config.txt
       arrow down until you find the line: hdmi_cvt=640 480 60 1 0 0 0 (for 320x240 pitft) and change it to: hdmi_cvt=320 240 60 1 0 0 0
       press Ctrl-x and select 'y' to save the changes then Enter/Return to exit
       type 'sudo reboot now' to restart the RPI
```

Enable 82357B manually:
1. sudo modprobe agilent_82357a
2. lsusb and get bus + device numbers
3. sudo fxload -D /dev/bus/usb/001/005 -t fx2 -I /usr/share/usb/agilent_82357a/measat_releaseX1.8.hex   replace 001 and 005 with the correct bus + device numbers
4. Step two again.. device should have incremented
5. Step 3 again with new device number
6. sudo gpib_config

Modify gpib.conf file:
1. sudo nano /usr/local/etc/gpib.conf
2. under the 'interface' section look for the line with 'board_type'
3. If the NI GPIB-USB-HS was attached during installation, the board type should be "ni_usb_b"
4. If the Agilent 82357B was attached during installation, the board type should be "agilent_82357a"
5. Change the entry if needed
6. Ctrl-x and 'y' to confirm if changes were made

BME280 Usage:
The installed software talks to a python based server through a telnet connection. The telnet port is 10001, the same as the Fluke 1620.
The code is written to be multi-threaded so that multiple connections can be made without interfering with other users.

There are two sample scripts included in the /home/$USER/thp_client sub-directory.(ie. /home/pi/thp_client) The first one (test_client.py)

is very basic and returns two different queries.
'python3 test_client.py' returns both temp+rh and temp+rh+bp. A Fluke 1620 only returns temp+rh with the command (READ?1). Calling READ?3 returns all three values.

'test_client_adv.py' returns some extra info about the RPI that was used to create the RPI lcd gui display.

The thp_client directory may be copied to another device that also has the BME280 software installed on it. Just edit the test_client*.py files and change the line:
thp = thp_client.THP_socket('127.0.0.1',port) to reflect the new IP address.

The thp_db.py file is also optionally installed in the thp_client directory and is called from a cron job. The file thp.db is a sqlite database that contains the data from the BME280. An application may be written to read out and display the values
The thp_db.py file has the information needed to figure out how the database table is created and the names of the fields.

The /root/thp_server directory contains the thp_server.py script which is the main BME280 server program. It doesn't need an lcd to function.
The other main file is the thp_server_gui.py file. It is responsible for displaying the data on the lcd if installed.
A copy of the thp_client.py file is also here. It is needed for the gui to query the thp_server.

There is also a thp_server_config.ini file. It has the settings for both the thp_server and the thp_server_gui
The BME280 address is set here and so is the ability to enable logging for debugging purposes. The values should normally be 0 for logging.
The BME280 address should either be 0x76 or 0x77
There are also some calibration constants for gain and offset of the BME readings. They are unique to each sensor.