**VNA Measurements**

**Procedure:**

**Check VNA Calibration:**

Turn on the VNA if it is not already on.

Check that the “Cal” button has an orange light, which means the calibration is applied.

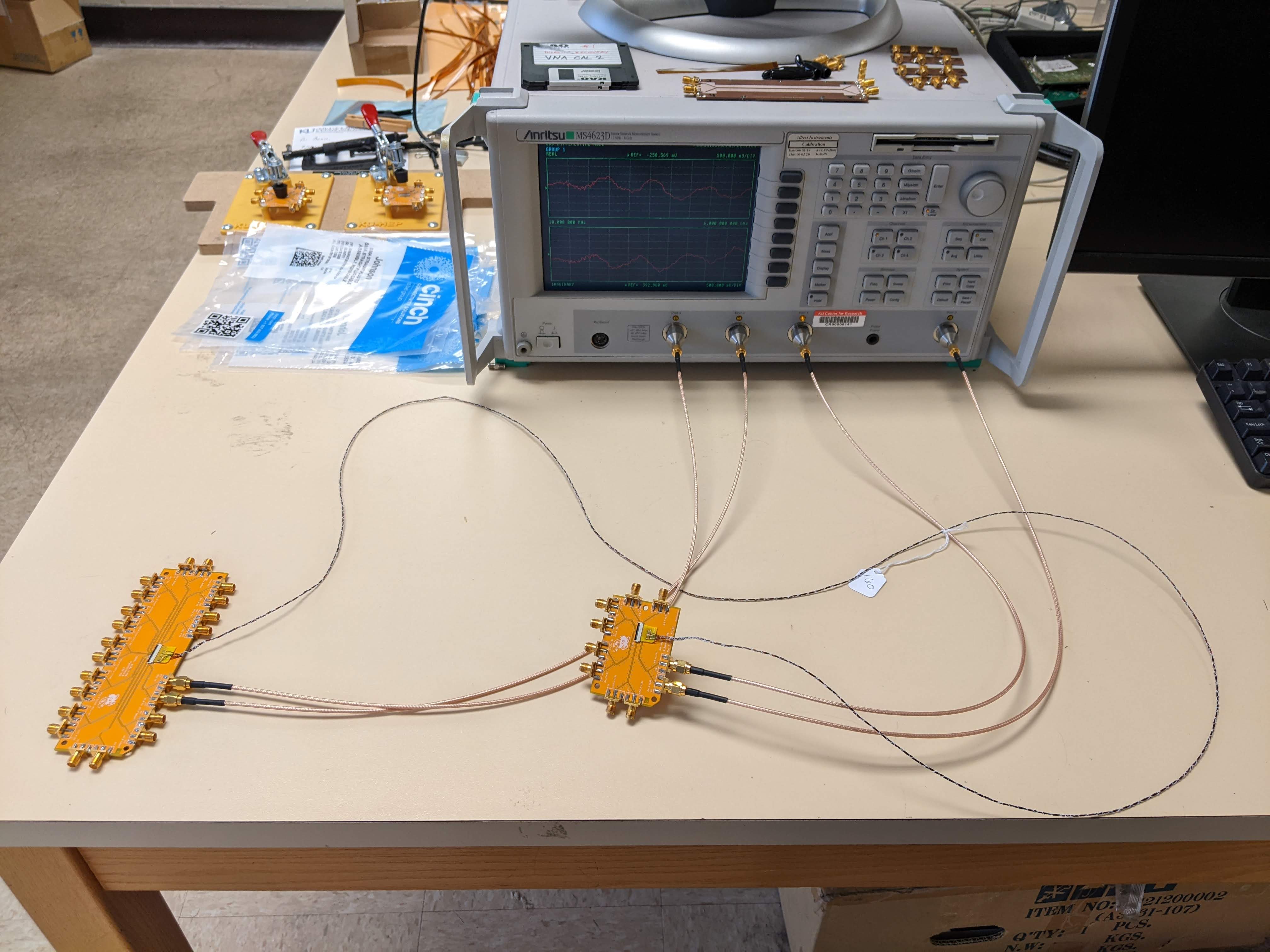
To check which calibration is applied, first make sure the “Clr/Local” button does not have the orange light. If it does, click it to turn it off. Then, click “Cal.” In the menu, make sure that “4 PORT” is selected (highlighted in red) and says “(CAL EXISTS).”

If the calibration is not applied or does not exist, please refer to the “Loading Calibration” instructions. If you need to perform a new calibration, refer to the “Calibration Procedure” instructions.

**Hardware Setup:**

Connect e-link to the appropriate yellow boards. For example, for a Type 1 e-link, use one 33 pin and one 45 pin yellow board. Insert the Molex connectors with the notch on the left, which is also the side with the white dot on the yellow board.

Connect the four SMA cables from the four VNA ports to the yellow boards for one differential channel (e.g. D0) according to the e-link mapping.



Ports 1 and 2 should be connected to the same yellow board (positive and negative of the same e-link channel), and Ports 3 and 4 should both be connected to the other yellow board (positive and negative of the same e-link channel) according to the e-link mapping. Make sure that Ports 1 and 3 are on opposite ends of one wire (e.g. the positive wire) and that Ports 2 and 4 are on opposite ends of the other wire (e.g. the negative wire). Choosing Port 1 and 3 to be on the positive or negative wire should not affect the measurements.

*Connect Port 1 and Port 2 to the same yellow board, and then connect Port 4 and Port 3 to the other yellow board. Connect such that Port 1 and 3 are on the positive wire and Port 2 and Port 4 are on the negative wire.*

**Data Taking:**

On the Windows 10 computer, open Matlab (we are using version R2020b).

Navigate to the directory R:\BEAN\_GRP\4portvnadata .

In the Matlab shell, type the name of the script “vna\_analysis” and hit enter.

Enter the cable number (e.g. 160).

Enter a file name with the cable number and channel (e.g. TP\_160\_D0). For Type 1 cables, the channel name should be the 33 pin name (e.g. by convention, D0 means D0 on the 33 pin side connected to D1 on the 45 pin side).

When the program finishes, repeat the procedure for the remaining e-link channels by connecting the SMA cables to each channel and taking data.

**Python Setup:**

If python is not already installed, download and install the latest version of python 3.

<https://www.python.org/downloads/>

Check the e-link and SMA cable connections. Then press enter, and the program will begin. Open CMD shell.

Type “python” and hit enter to check that python is installed and in the path.

New commands (where py is python on Windows)

py -m pip install --upgrade pip

py -m pip install numpy

py -m pip install pandas

py -m pip install matplotlib

py -m pip install scikit-rf

py -m pip list

Update to latest version of pip (can only be done per user due to permissions):

pip install --upgrade pip --user

Install required packages:

pip install numpy

pip install pandas

pip install matplotlib

pip install scikit-rf

Check list of installed python packages:

pip list

**Data Analysis:**

Python should be setup (if not, see previous section).

The repository with the code to analyze VNA data is here:

<https://github.com/ku-cms/eLink_Instrumentation>

This is the primary VNA data analysis script:

<https://github.com/ku-cms/eLink_Instrumentation/blob/main/VNA/python/VNA_COMP.py>

Your KU user will need to have access to the R drive. To access the R drive, your KU user needs to be added to the group “phsx\_r\_bean” on groups.ku.edu. You can ask Prof. Alice Bean to be added to this group.

**Make sure data has been taken for all channels of the e-link before running the script. The script analyzes data for all e-link channels.**

***Make sure to use the*** ***Windows Computers. (Cannot be done on Linux)***

Using the file explorer:

To analyze data, first create a directory for the cable that you want to analyze using the cable number on the R drive with this path:

R:\BEAN\_GRP\4portvnadata\VNA\_analysis\Data\<cable\_number>

Then, copy the VNA data files (.vna.txt) for that cable

**from** the folder

R:\BEAN\_GRP\4portvnadata\Cable\_<cable\_number>

**to** the folder

R:\BEAN\_GRP\4portvnadata\VNA\_analysis\Data\<cable\_number>

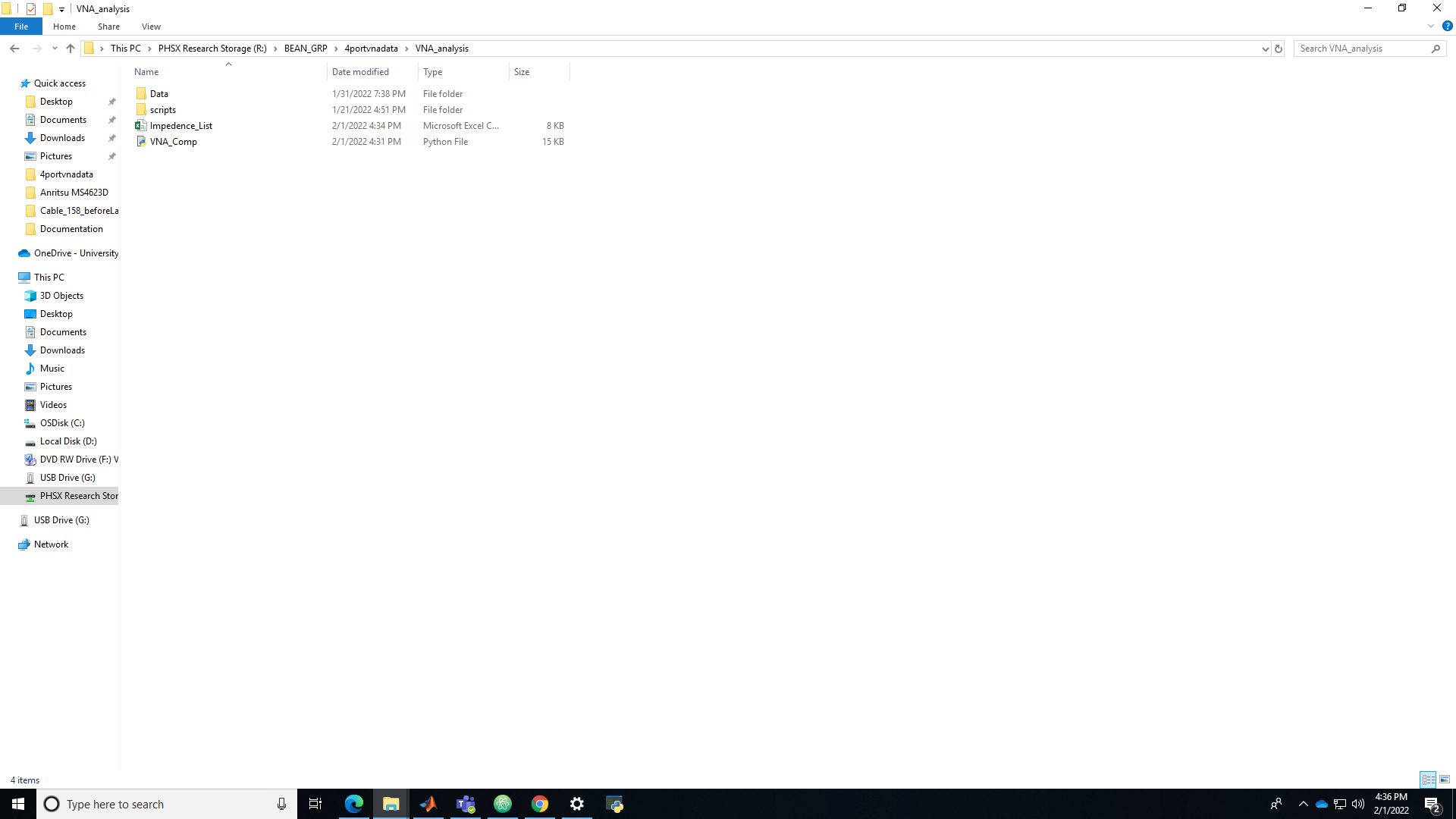
This is the path to the production version of VNA analysis script on the network R drive:

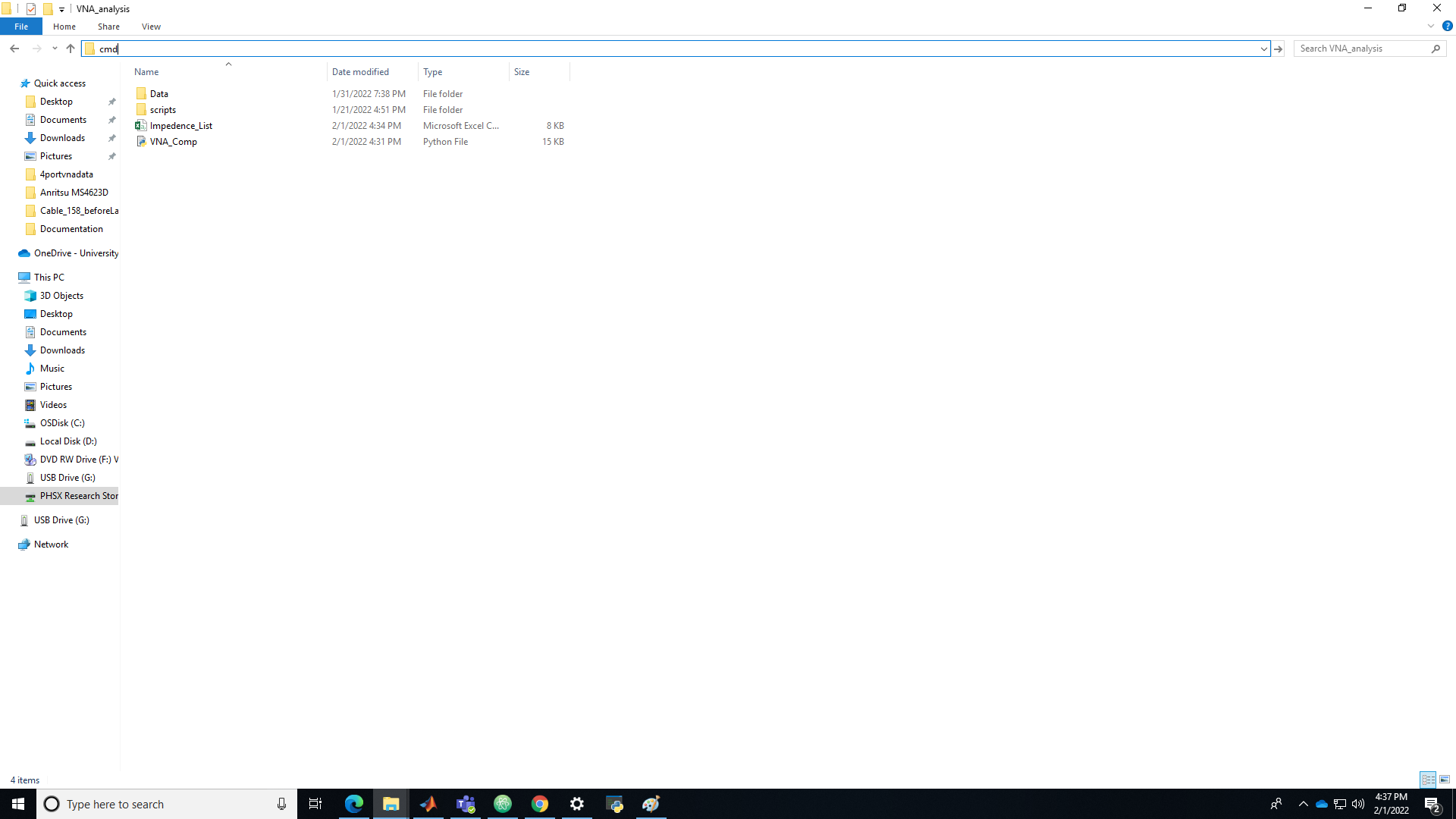
R:\BEAN\_GRP\4portvnadata\VNA\_analysis\VNA\_Comp.py

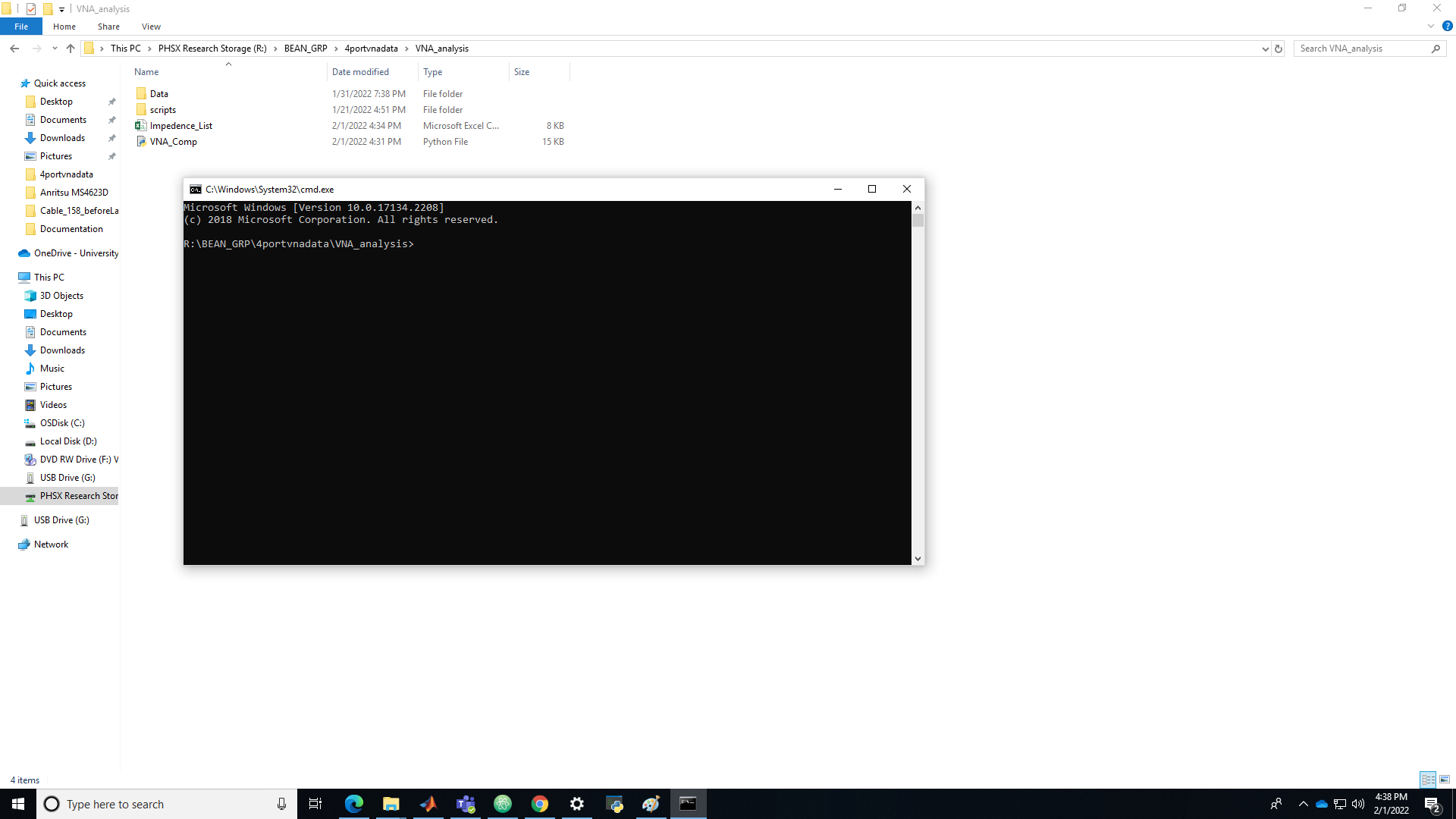
Open file explorer on the Windows 10 computer and navigate to this directory: R:\BEAN\_GRP\4portvnadata\VNA\_analysis

Click on the full path tab at the top, remove the path, and enter “cmd.”

A Command Prompt shell should open with the working directory “R:\BEAN\_GRP\4portvnadata \VNA\_analysis” .







You can enter the command “cd” to print the current working directory.

Note that Windows CMD commands are not the same as macOS terminal commands (Unix based) or GNU/Linux commands.

See a list of Windows CMD commands here (or search online):

[Windows CMD commands: A list of command prompt codes - IONOS](https://www.ionos.com/digitalguide/server/know-how/windows-cmd-commands/)

**Make sure data has been taken for all channels of the e-link before running the script. The script analyzes data for all e-link channels.**

Run python analysis script and enter the requested parameters.

python VNA\_Comp.py

Or (Windows alternate)

py VNA\_Comp.py

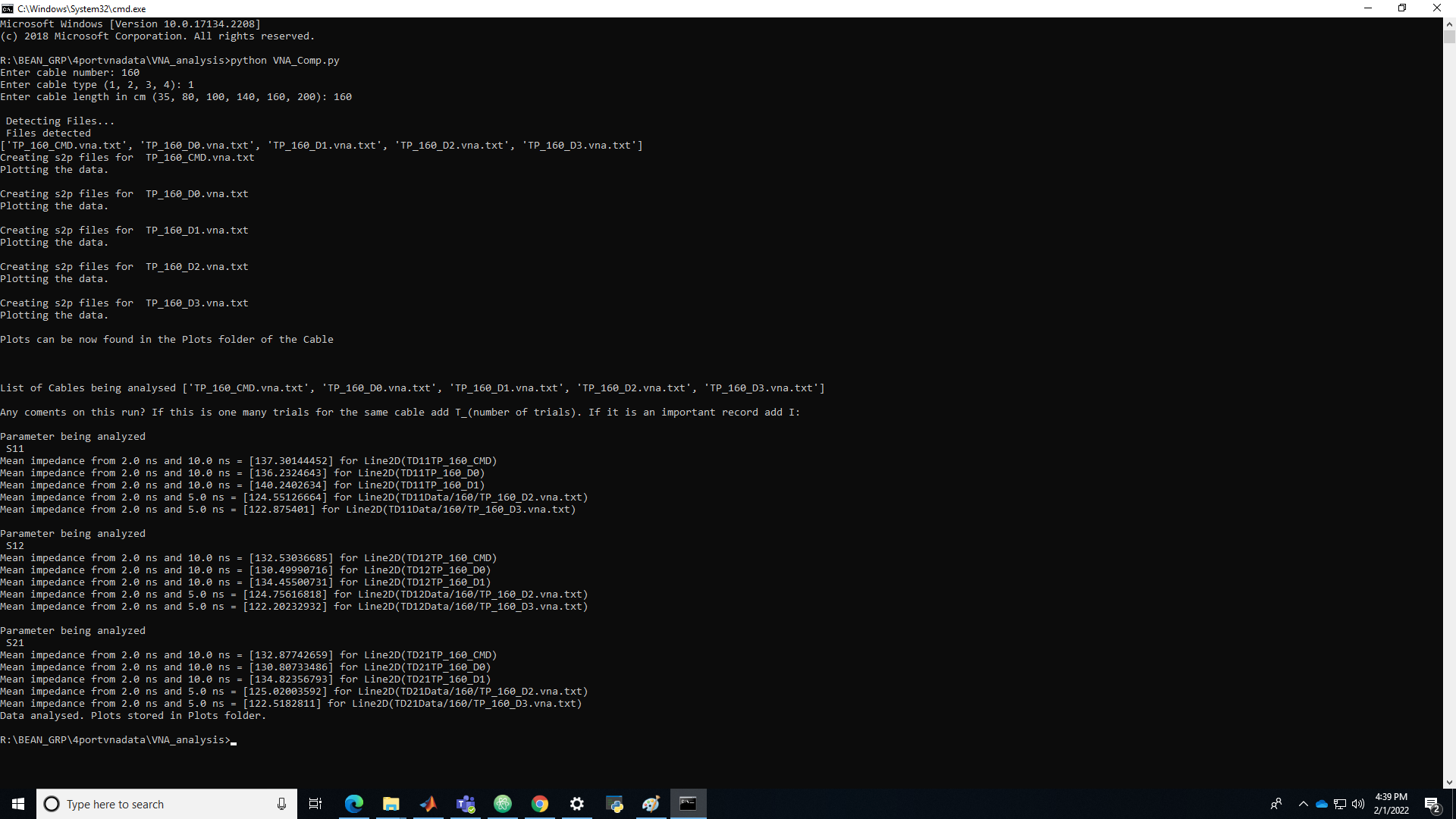
* Enter cable number.
* Enter cable type (1, 2, 3, 4).
* Enter cable length in cm (35, 80, 100, 140, 160, 200).

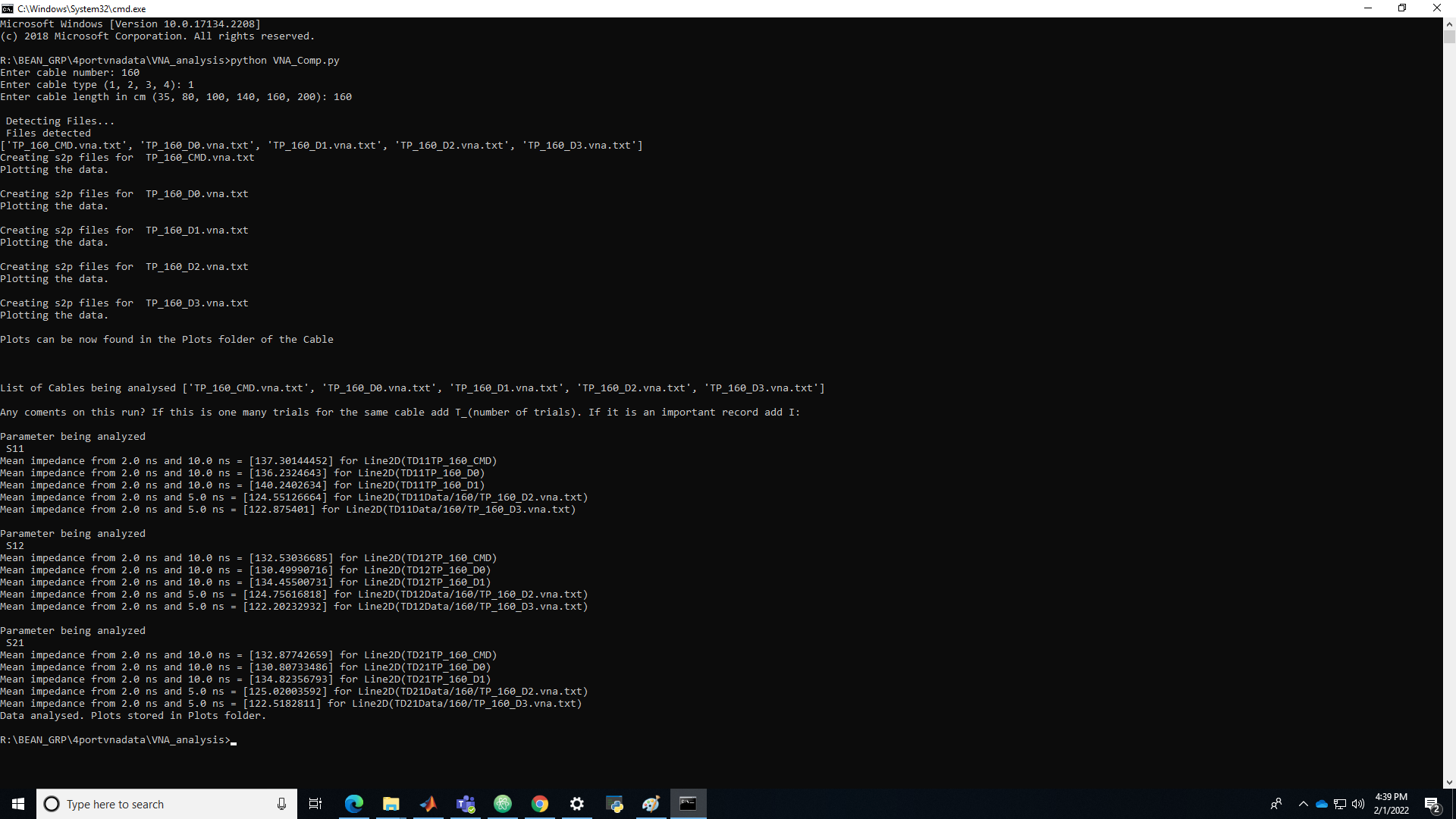
Note: Type 1 cables have 45 pin to 33 pin paddle boards. To analyze data from Type 0 cables (33 pin to 33 pin paddle boards), enter Type 1 for this script, since both Type 0 and Type 1 cables have the same five channels and same channel names.

At the end, enter comments (if any) and press enter. If not comments, just press enter.

The program prints impedance values and saves them in a csv file.

The program also creates plots.





**Debug:**

You may see the following error:

'python' is not recognized as an internal or external command, operable program or batch file.

Python needs to be downloaded and installed. When installing, choose to have python added to the path. Python needs to be in the path to work in the CMD shell. See the “Python Setup” section for details on the python packages that need to be installed.

<https://www.python.org/downloads/>

[python is not recognized windows 10 - Stack Overflow](https://stackoverflow.com/questions/47539201/python-is-not-recognized-windows-10)

**Recording Results:**

Save plots to the R drive and to the corresponding directory for the cable on Microsoft Teams.

Record impedances for this cable in this table in Microsoft Teams:

Documents/Cable production/TP\_Cables\_Production2020.xlsx

Example plots (cable 160, channel D0):

TODO:

Left plot y-axis range: [-75, 75]

Right plot y-axis range: [0, 200]

