

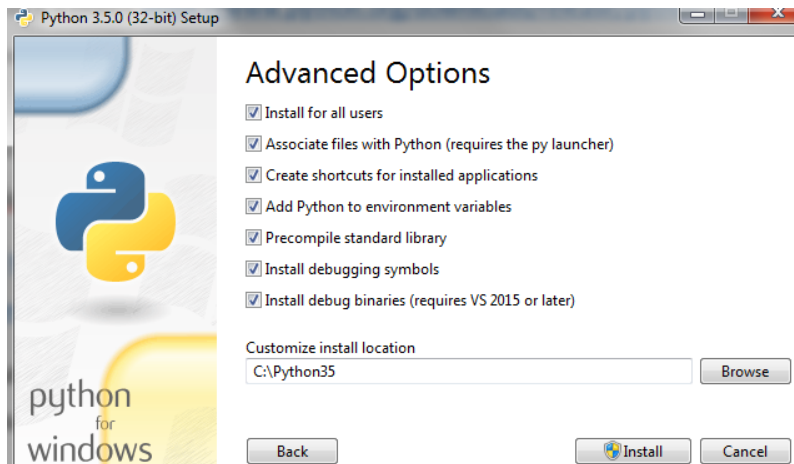
This manual shows how to set-up PyVISA and other python packages to remotely get waveform from oscilloscope and network analyzer

Note: If you've already installed Python packages, please uninstall this and delete its corresponding folder. On this manual, you will download and install suggested packages to avoid software conflict.

1. First, install following packages. ALL of them are free

- a. NI VISA for Windows (free): <http://www.ni.com/download/ni-visa-15.0.1/5693/en/>
- b. NI- 488.2 for Windows (free): <http://www.ni.com/download/ni-488.2-16.0.0/6132/en/>
- c. Python 3.5.0 (free), 32 bit: <https://www.python.org/downloads/release/python-350/>

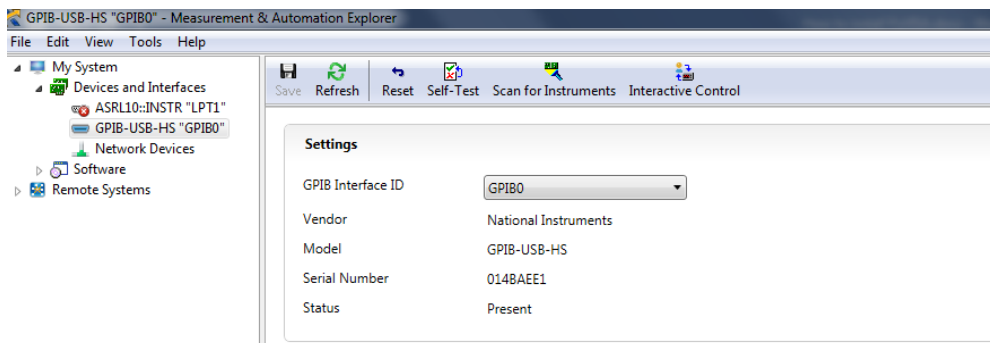
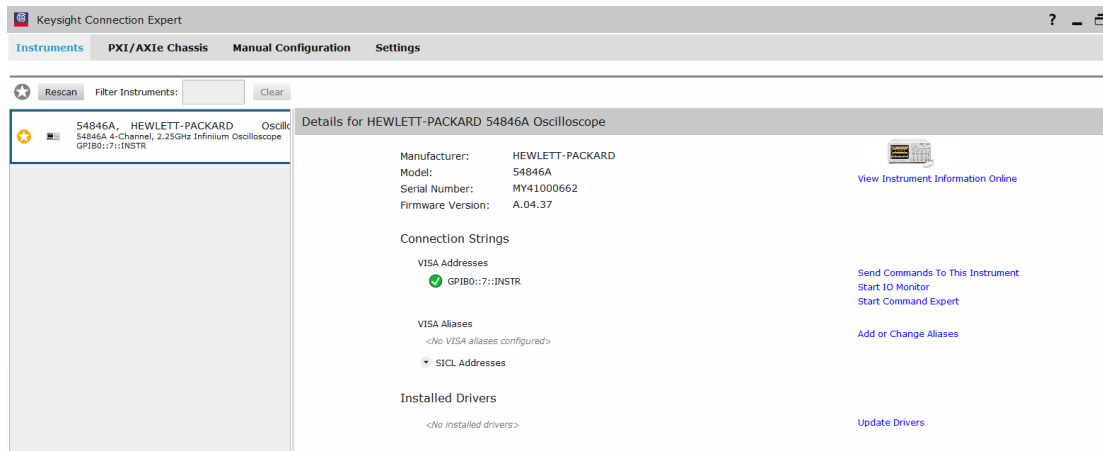
Choose to customize installation, create a folder **C:\Python35**. Check on options



- d. Text editor (free): <https://atom.io/>
- e. Keysight IO library to check instrument connections and address (free)
<http://www.keysight.com/main/software.jsp?id=2175637&pageMode=CV&cc=US&lc=eng>

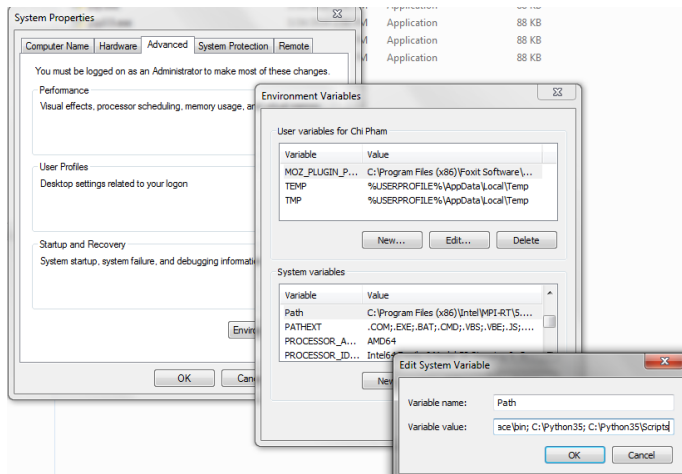
2. Enable all running services by Agilent and NI, then check Keysight Connection Expert and NI Max

See results of installing below



3. After installing all above packages, process to install pyVISA package

- Add **Python** to your system path by navigating to *Start menu->Control panel->System->Advanced tab->Environment Variables* and adding: **C:\Python35** and **C:\Python35\Scripts** to the **PATH** variable.
- Restart your computer



- c. Download PyVISA 1.8 : <https://pypi.python.org/pypi/PyVISA>
- d. Extract this package PyVISA to **C:\Python35**
- e. Open command line on Window, then navigate to PyVISA-1.8 folder
- f. Type: *python setup.py install*

This should automatically install PyVISA to the C:\python\Lib\site-packages\ directory

```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Chi Pham>cd ..
C:\Users>cd ..
C:\>cd Python35
C:\Python35>cd PyVISA-1.8
C:\Python35\PyVISA-1.8>python setup.py install
running install
running bdist_egg
running egg_info
writing dependency_links to PyVISA.egg-info\dependency_links.txt
writing PyVISA.egg-info\PKG-INFO
writing top-level names to PyVISA.egg-info\top_level.txt
reading manifest file 'PyVISA.egg-info\SOURCES.txt'
reading manifest template 'MANIFEST.in'
no previously-included directories found matching 'docs\_build'
no previously-included directories found matching 'docs\_themes\.git'
warning: no previously-included files matching '*.pyc' found anywhere in distrib

```

- g. Test your installation
 - Using command line, navigating C:\Python35
 - Type: python. Then, type

```

>>> import visa
>>> rm = visa.ResourceManager()
>>> print(rm.list_resources())

```

```

Processing PyVISA-1.8-py3.5.egg
creating c:\python35\lib\site-packages\PyVISA-1.8-py3.5.egg
Extracting PyVISA-1.8-py3.5.egg to c:\python35\lib\site-packages
Adding PyVISA 1.8 to easy-install.pth file

Installed c:\python35\lib\site-packages\pyvisa-1.8-py3.5.egg
Processing dependencies for PyVISA==1.8
Finished processing dependencies for PyVISA==1.8

C:\Python35\PyVISA-1.8>import visa
'import' is not recognized as an internal or external command,
operable program or batch file.

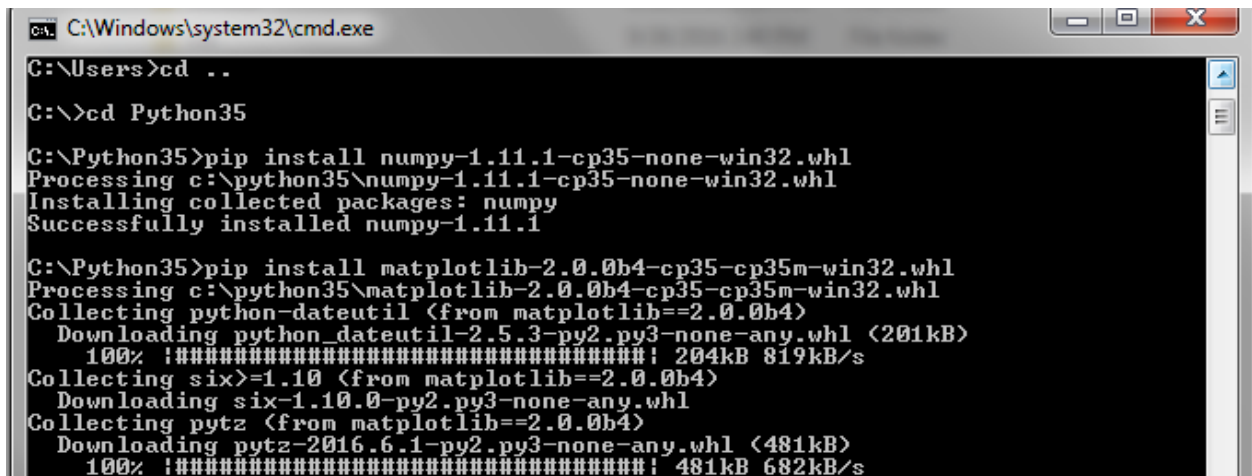
C:\Python35\PyVISA-1.8>cd ..
C:\Python35>import visa
'import' is not recognized as an internal or external command,
operable program or batch file.

C:\Python35>python
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:16:59) [MSC v.1900 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import visa
>>> rm=visa.ResourceManager()
>>> print(rm.list_resources())
<<ASRL10::INSTR', 'GPIB0::7::INSTR'>
>>>

```

4. Install Python packages: numpy and Matplotlib

- Upgrade your pip
`python -m pip install --upgrade pip`
- First, download two packages
`matplotlib-2.0.0b4-cp35-cp35m-win32.whl`
<http://www.lfd.uci.edu/~gohlke/pythonlibs/#matplotlib>
`numpy-1.11.2rc1+mkl-cp35-cp35m-win32.whl`
<http://www.lfd.uci.edu/~gohlke/pythonlibs/#numpy>
- Copy the downloaded packages to C:\Python35
- Open command line on Windows, navigate to C:\Python35
- Run pip to install these packages:
Pip install matplotlib-2.0.0b4-cp35-cp35m-win32.whl
Pip install numpy-1.11.2rc1+mkl-cp35-cp35m-win32.whl



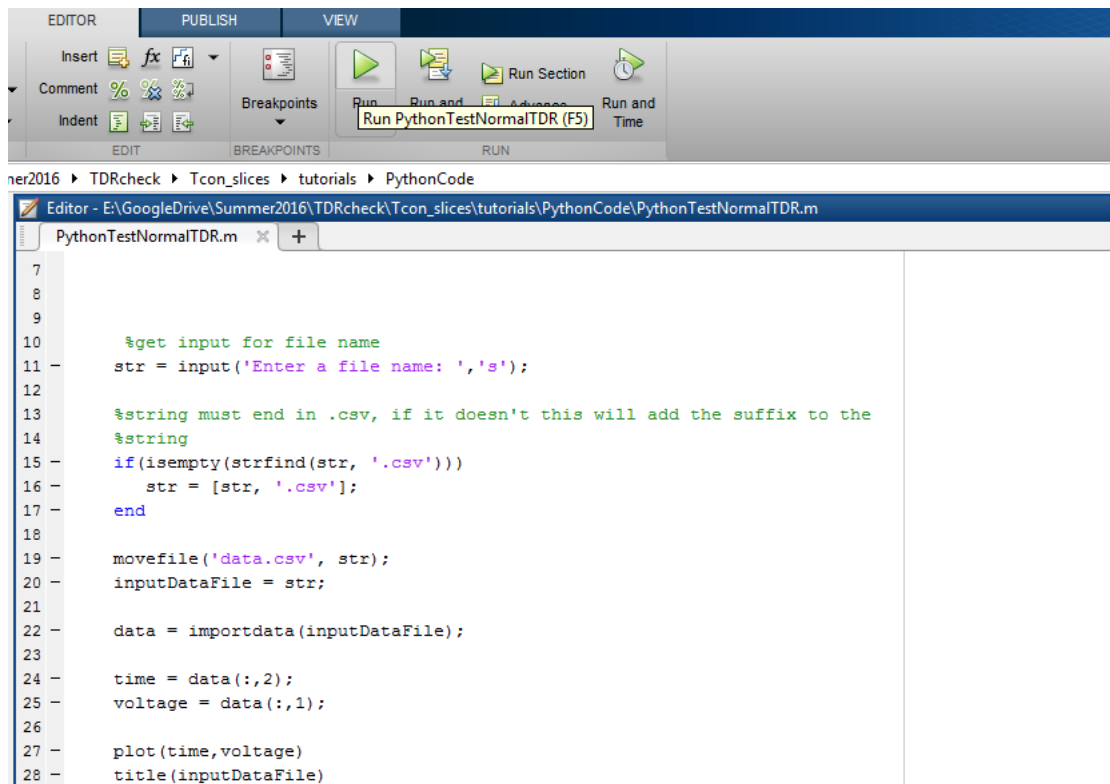
```
C:\Windows\system32\cmd.exe
C:\Users>cd ..
C:\>cd Python35
C:\Python35>pip install numpy-1.11.1-cp35-none-win32.whl
Processing c:\python35\numpy-1.11.1-cp35-none-win32.whl
Installing collected packages: numpy
Successfully installed numpy-1.11.1
C:\Python35>pip install matplotlib-2.0.0b4-cp35-cp35m-win32.whl
Processing c:\python35\matplotlib-2.0.0b4-cp35-cp35m-win32.whl
Collecting python-dateutil <from matplotlib==2.0.0b4>
  Downloading python_dateutil-2.5.3-py2.py3-none-any.whl <201kB>
    100% |#####| 204kB 819kB/s
Collecting six>=1.10 <from matplotlib==2.0.0b4>
  Downloading six-1.10.0-py2.py3-none-any.whl
Collecting pytz <from matplotlib==2.0.0b4>
  Downloading pytz-2016.6.1-py2.py3-none-any.whl <481kB>
    100% |#####| 481kB 682kB/s
```

5. Next steps show how to take out measured data using MATLAB (example with normal TDR and TDT).

- Open **NormalTDR.py** using any text editor, change GPIB address (highlight part on figure below) if this address is different with your current setting, which shows on Keysight Connection Expert.

```
NormalTDR.py x
# visa is used for interfacing with instruments↓
# numpy is used for number operations, arrays↓
# pylab is used for plotting↓
# os is used for directory operations↓
import visa↓
import numpy as np↓
import pylab↓
import os↓
↓
# Choose your instrument to work with↓
rm = visa.ResourceManager()↓
my_instrument = rm.open_resource('GPIB0::7::INSTR')↓
↓
# Set the channel source↓
my_instrument.write('WAV:SOUR CHAN3')↓
↓
# WAV:XINC? returns the duration between data points on the x-axis↓
xinc = float(my_instrument.query('WAV:XINC?'))↓
↓
```

b. Open **PythonTestNormalTDR.m** on MATLAB. Go to Editor, hit RUN or F5

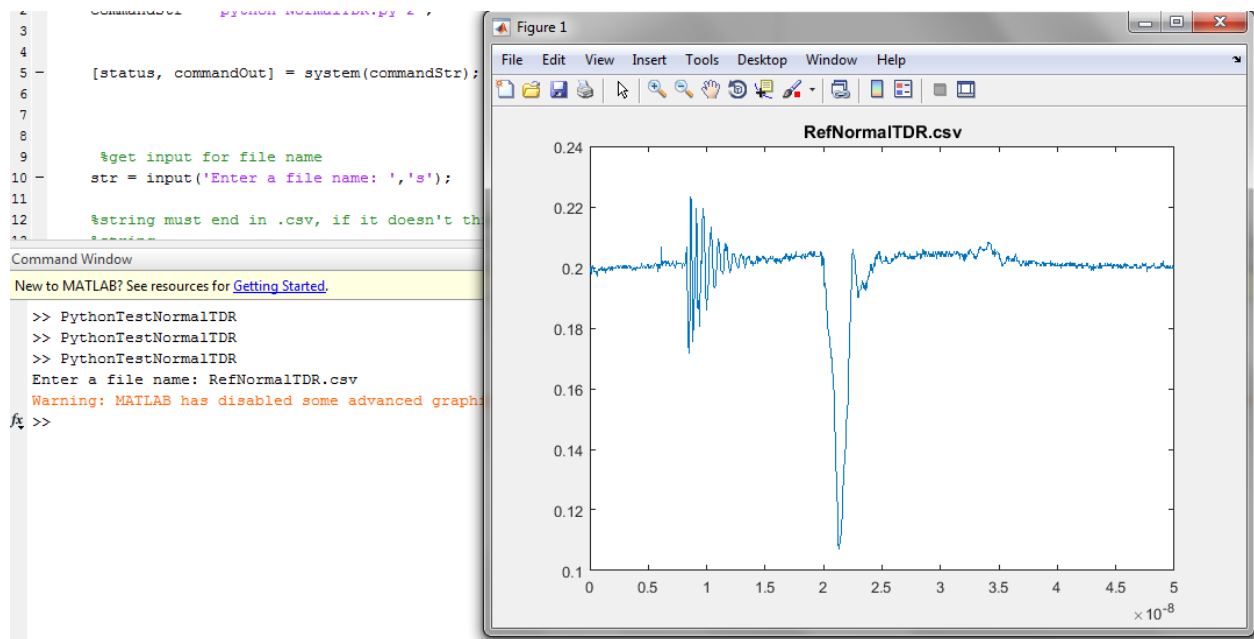


The image shows the MATLAB Editor interface. The top toolbar includes tabs for EDITOR, PUBLISH, and VIEW. The EDITOR tab is active, showing a toolbar with icons for Insert, Comment, Indent, Breakpoints, Run, Run Section, Run and Time, and Run PythonTestNormalTDR (F5). The main window displays the script PythonTestNormalTDR.m with the following code:

```
7
8
9
10 %get input for file name
11 str = input('Enter a file name: ','s');
12
13 %string must end in .csv, if it doesn't this will add the suffix to the
14 %string
15 if isempty(strfind(str, '.csv'))
16     str = [str, '.csv'];
17 end
18
19 movefile('data.csv', str);
20 inputDataFile = str;
21
22 data = importdata(inputDataFile);
23
24 time = data(:,2);
25 voltage = data(:,1);
26
27 plot(time,voltage)
28 title(inputDataFile)
```

c. This will ask you to Enter the file name for collected data on Command Window.

For example, type: **RefNormalTDR.csv**. The measured plot will pop up and you can see the RefNormalTDR.csv file on the folder directory where you are running the program



d. Repeat the steps for **PythonTestVSWR.m**

