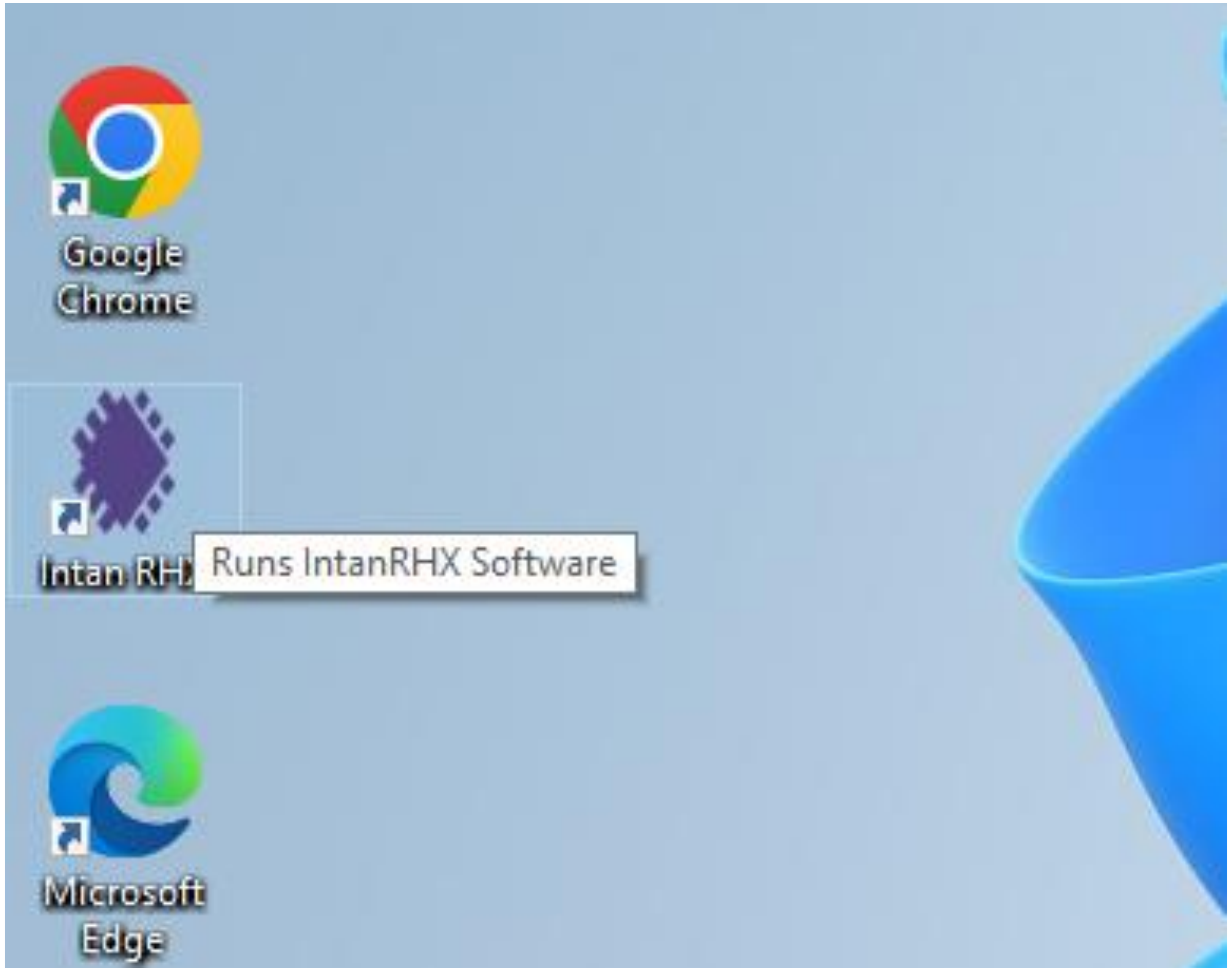
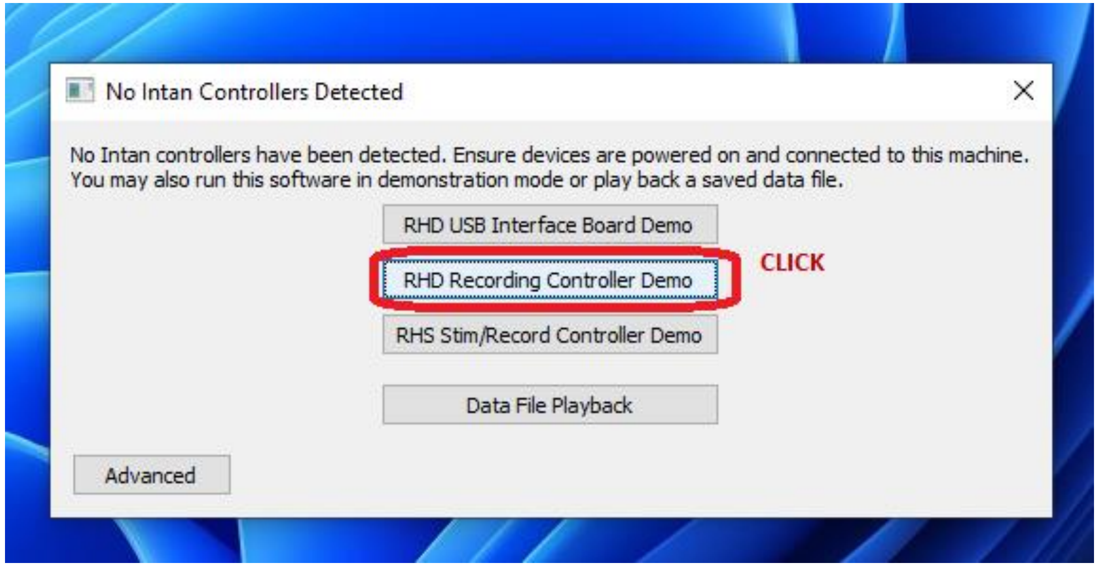
User Manual to run Plotter.py



Runs IntanRHX Software

STEP 1



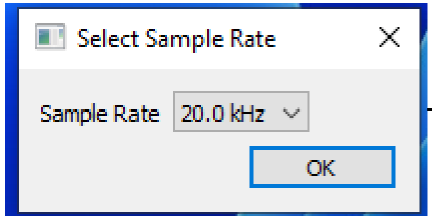
No Intan Controllers Detected

No Intan controllers have been detected. Ensure devices are powered on and connected to this machine. You may also run this software in demonstration mode or play back a saved data file.

RHD USB Interface Board Demo
RHD Recording Controller Demo
RHS Stim/Record Controller Demo
Data File Playback
Advanced

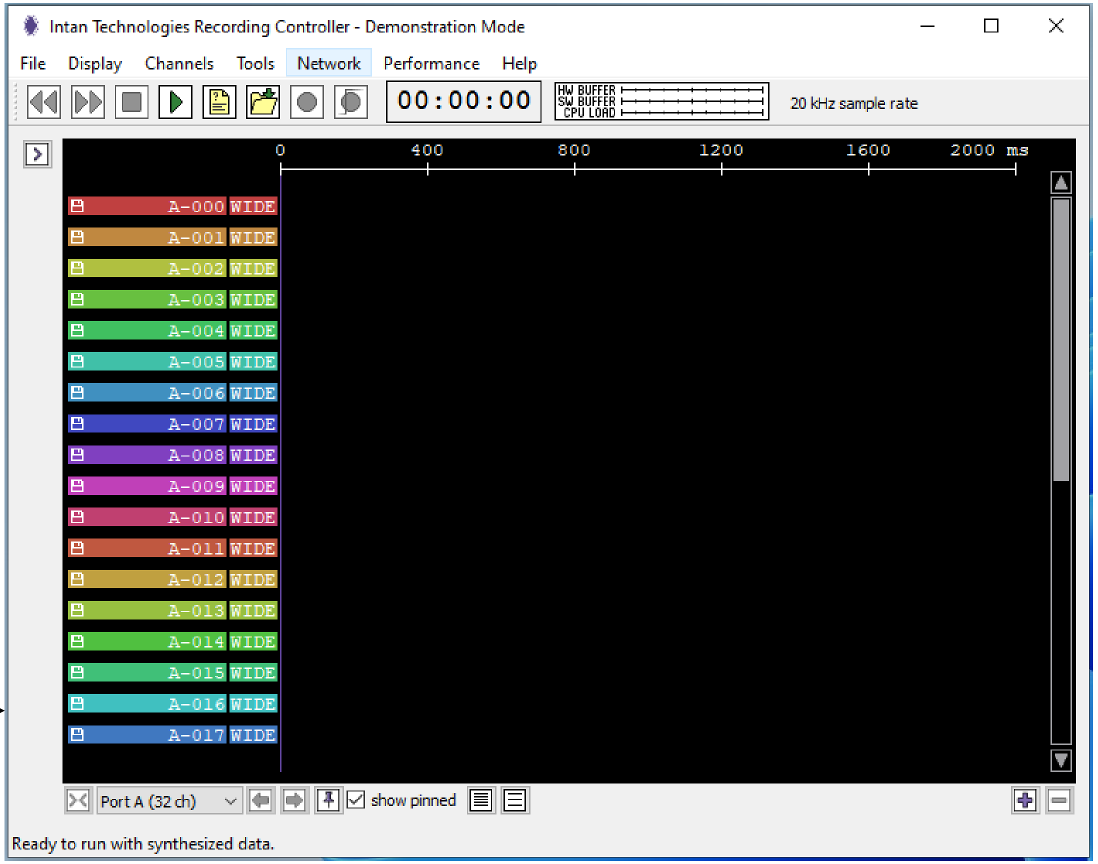
CLICK

STEP 2



Select Sample Rate

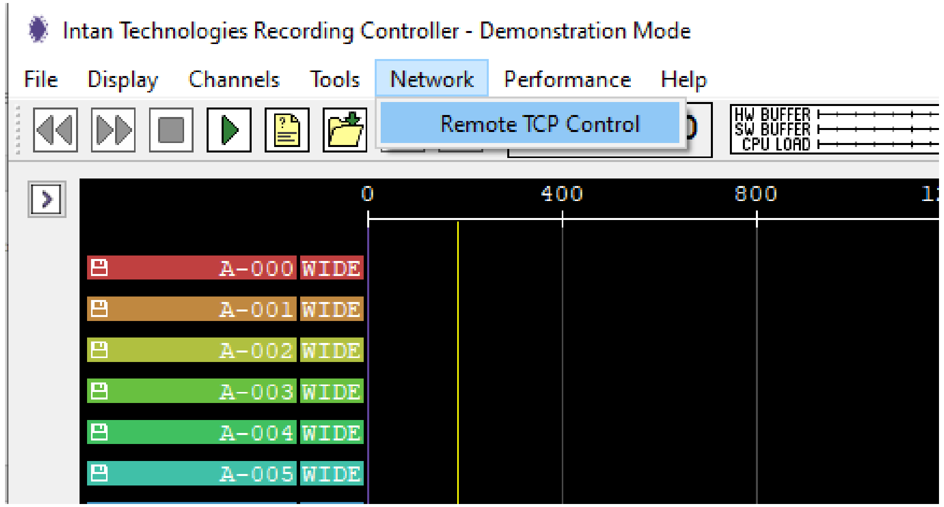
Sample Rate: 20.0 kHz



Intan Technologies Recording Controller - Demonstration Mode

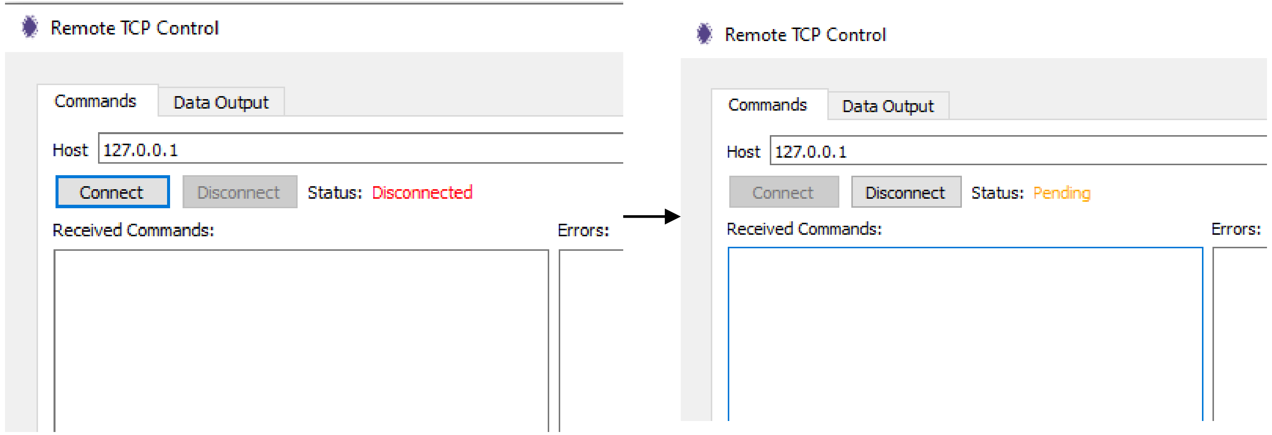
Ready to run with synthesized data.

STEP 3



Intan Technologies Recording Controller - Demonstration Mode

STEP 4



Remote TCP Control

Host 127.0.0.1

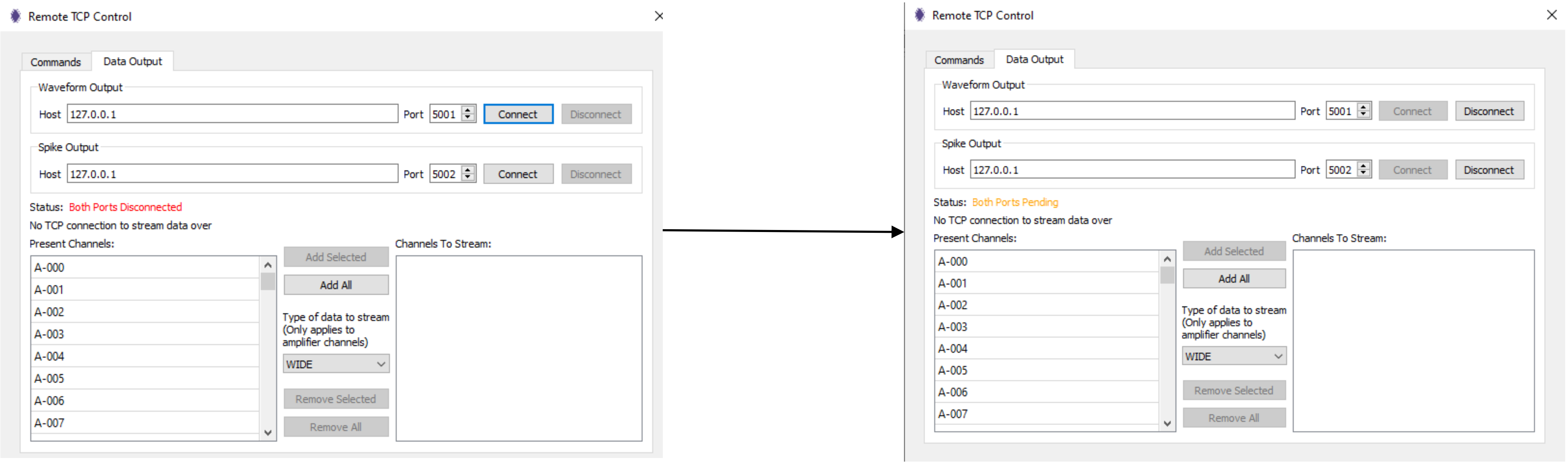
Status: Disconnected

Remote TCP Control

Host 127.0.0.1

Status: Pending

STEP 5



Remote TCP Control

Waveform Output
Host 127.0.0.1 Port 5001

Spike Output
Host 127.0.0.1 Port 5002

Status: Both Ports Disconnected
No TCP connection to stream data over

Type of data to stream (Only applies to amplifier channels): WIDE

Remote TCP Control

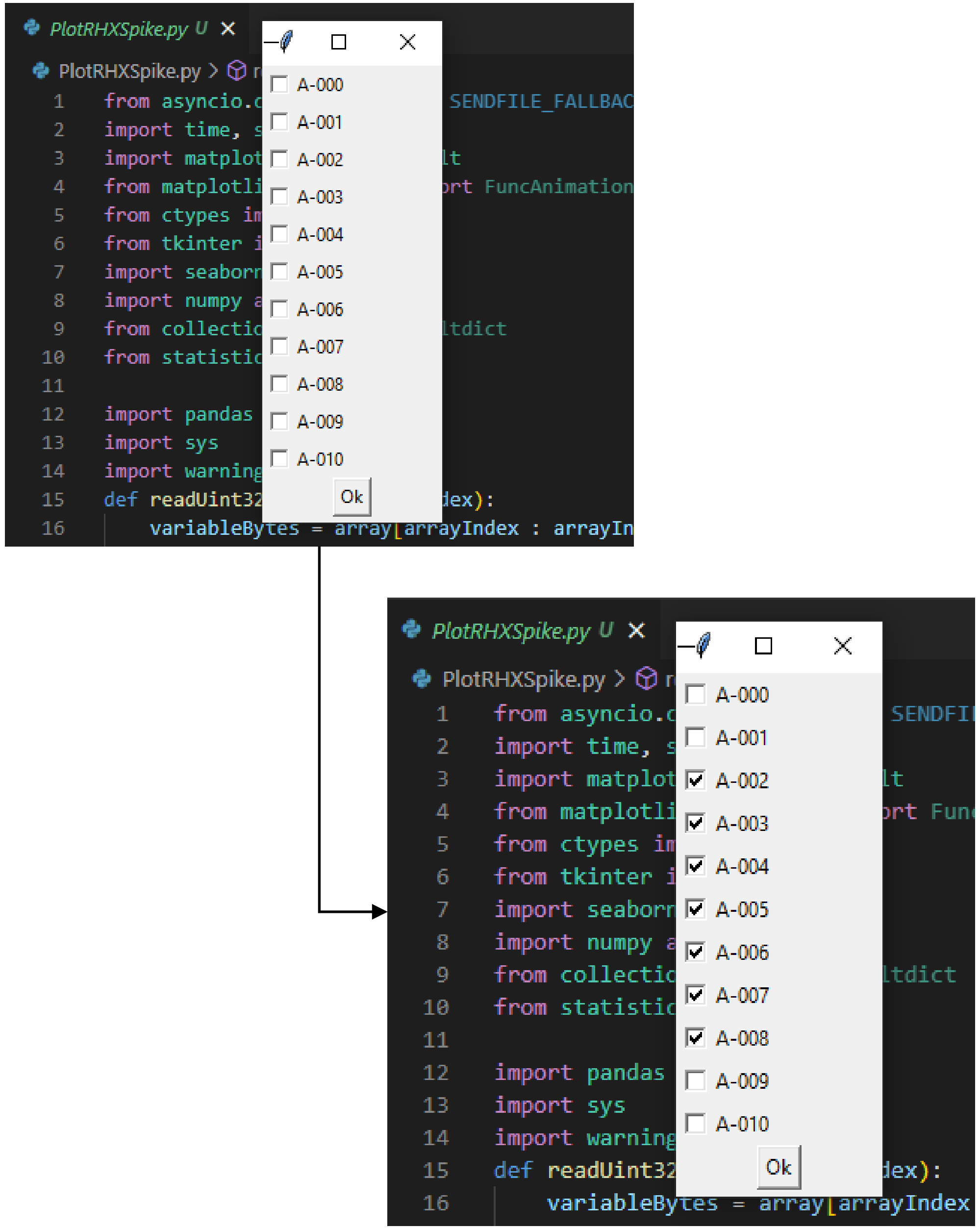
Waveform Output
Host 127.0.0.1 Port 5001

Spike Output
Host 127.0.0.1 Port 5002

Status: Both Ports Pending
No TCP connection to stream data over

Type of data to stream (Only applies to amplifier channels): WIDE

STEP 6



```
from asyncio.c          SENDFILE_FALLBAC
import time, s
import matplot          lt
from matplotli          ort FuncAnimation
from ctypes im
from tkinter i
import seaborn
import numpy a
from collectio          ltdict
from statistic

import pandas
import sys
import warning
def readUint32          dex):
    variableBytes = array[arrayIndex : arrayIn
```

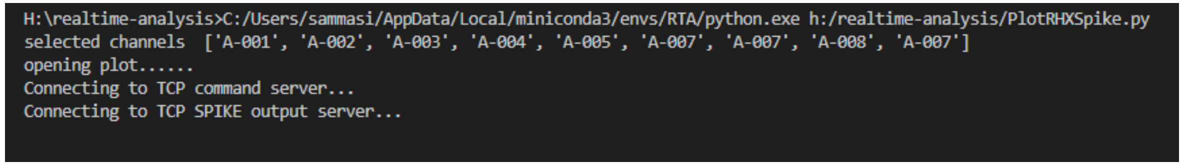
- □ A-000
- □ A-001
- □ A-002
- □ A-003
- □ A-004
- □ A-005
- □ A-006
- □ A-007
- □ A-008
- □ A-009
- □ A-010

- □ A-000
- □ A-001
- ☑ A-002
- ☑ A-003
- ☑ A-004
- ☑ A-005
- ☑ A-006
- ☑ A-007
- ☑ A-008
- □ A-009
- □ A-010

STEP 7

Once ‘Ok’ clicked , you will see

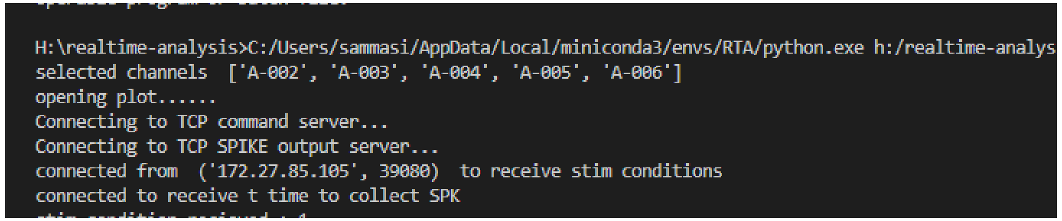
- two empty matplotlib plots(figure 1 & figure 2) open
- Plotter.py waiting for response from Texture_task_wTCPIP.m



```
H:\realtime-analysis>C:/Users/sammasi/AppData/Local/miniconda3/envs/RTA/python.exe h:/realtime-analysis/PlotRHXSpike.py
selected channels  ['A-001', 'A-002', 'A-003', 'A-004', 'A-005', 'A-007', 'A-007', 'A-008', 'A-007']
opening plot......
Connecting to TCP command server...
Connecting to TCP SPIKE output server...
```

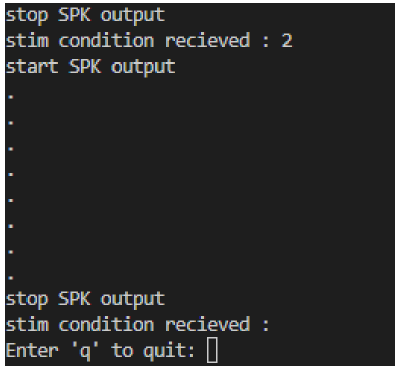
STEP 8

Once the connection is established , plotter.py will be ready to receive stim_cond & SPK_outut ON OFF signal



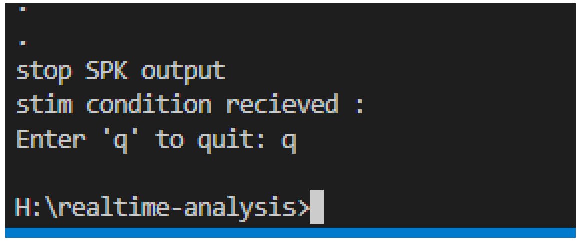
```
H:\realtime-analysis>C:/Users/sammasi/AppData/Local/miniconda3/envs/RTA/python.exe h:/realtime-analys
selected channels  ['A-002', 'A-003', 'A-004', 'A-005', 'A-006']
opening plot......
Connecting to TCP command server...
Connecting to TCP SPIKE output server...
connected from  ('172.27.85.105', 39080)  to receive stim conditions
connected to receive t time to collect SPK
stim condition recieved : 1
```

When the texture_taskwTCPIP.m is terminated, the plotter.py terminal would look something like this



```
stop SPK output
stim condition recieved : 2
start SPK output
.
.
.
.
.
.
.
.
stop SPK output
stim condition recieved :
Enter 'q' to quit: 
```

And to quit plotter.py, enter ‘q’



```
.
.
stop SPK output
stim condition recieved :
Enter 'q' to quit: q

H:\realtime-analysis>
```

The spike data from the trials will be stored in “CH_stim_SPK_data.csv”

To run the code again ,follow steps 3-7