NEW CONTROL SOFTWARE FOR CERBERUS 3D NANOINDENTATION SYSTEM

by

Bhakt Vatsal Trivedi
(2010041)

SUPERVISORS:

Mr. Saket Sourav
Research Engineer
IIITDM - Jabalpur

Dr. Graham L. W. Cross
PI, CRANN Nanotechnology Institute
Trinity College Dublin

Computer Science and Engineering Discipline

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY DESIGN AND MANUFACTURING JABALPUR
(24th August to 9th September, 2013)

INTRODUCTION:-

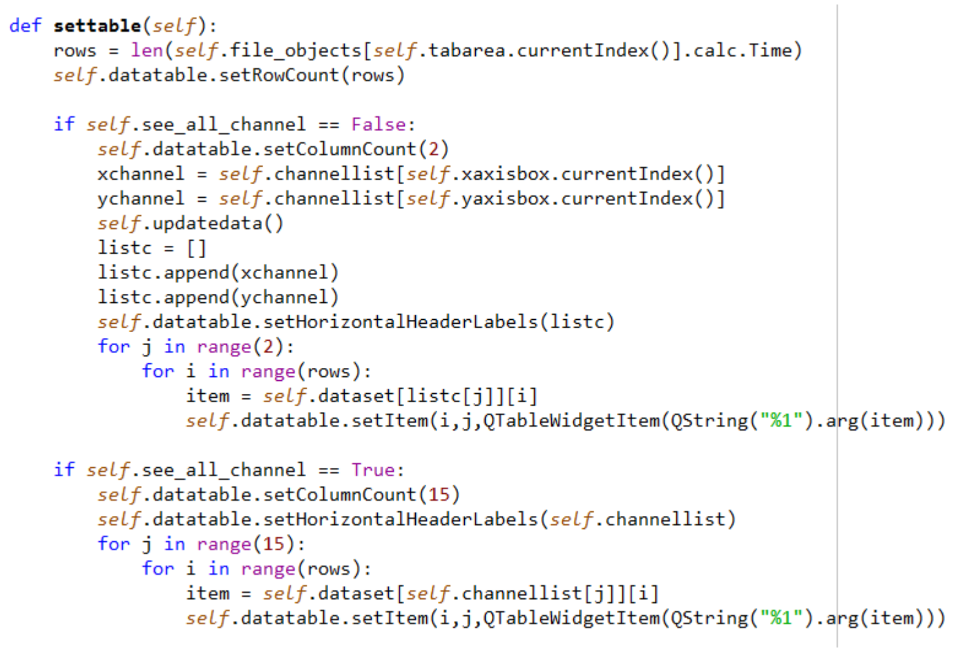
In this phase the main focus was on two major tasks:

- To add features like data explorer, multiple files at same time, saving the file etc., to the GUI.
- To prepare a strategy to connect the CDA to HTBasic control system.

The new features which are added to the GUI are explained later. The strategy to connect to the control system is still not in a rigid form, so it will be mentioned in the next report.

DATA EXPLORER:

After discussion with the people who will be using the GUI, the need for a data explorer came to light. The data explorer should display all the calculated values in accessible way without creating a mess in the GUI.
To achieve this goal, I used QTableWidget of PyQt which is designed for the systematic data view. The following image shows the code for the data explorer.



```
def settable(self):
    rows = len(self.file_objects[self.tabarea.currentIndex()].calc.Time)
    self.datatable.setRowCount(rows)

    if self.see_all_channel == False:
        self.datatable.setColumnCount(2)
        xchannel = self.channellist[self.xaxisbox.currentIndex()]
        ychannel = self.channellist[self.yaxisbox.currentIndex()]
        self.updatedata()
        listc = []
        listc.append(xchannel)
        listc.append(ychannel)
        self.datatable.setHorizontalHeaderLabels(listc)
        for j in range(2):
            for i in range(rows):
                item = self.dataset[listc[j]][i]
                self.datatable.setItem(i,j,QTableWidgetItem(QString("%1").arg(item)))

    if self.see_all_channel == True:
        self.datatable.setColumnCount(15)
        self.datatable.setHorizontalHeaderLabels(self.channellist)
        for j in range(15):
            for i in range(rows):
                item = self.dataset[self.channellist[j]][i]
                self.datatable.setItem(i,j,QTableWidgetItem(QString("%1").arg(item)))
```

MULTIPLE FILES:

The task of opening multiple files is achieved through QTabWidget of PyQt. Each tab has its associated Dataset object which contains all the data and other details.

SAVING A FILE:

To save a file I used the standard python module named **Pickle,** which stores any object to a file. The object is stored serially as byte stream. To retrieve back the object, the module has a concept of unpickling which gets the object back. The following piece of code shows how it is done:

The following code shows how a file in saved (pickling):

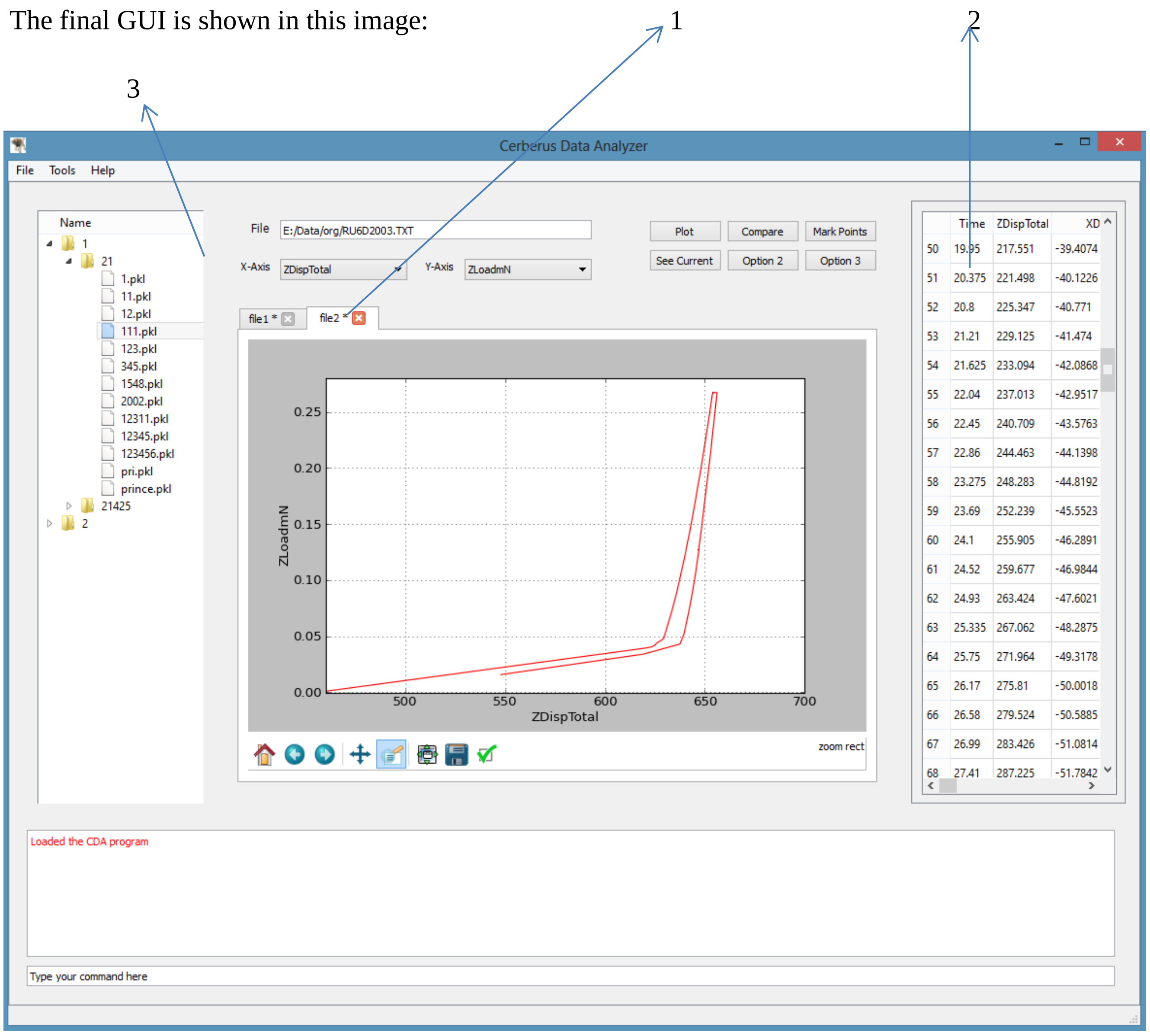
```
output = open("projects\\" + self.current_project + "\\" + self.current_exp + "\\"+ name , "wb")
pickle.dump(self.file_objects[self.tabarea.currentIndex()], output, -1)
output.close()
self.file_objects[self.tabarea.currentIndex()].filename = name

self.tabarea.setTabText(self.tabarea.currentIndex(), _translate("MainWindow", self.file_objects[self.tabarea.currentIndex()].filename, None))
self.file_objects[self.tabarea.currentIndex()].saved = True
```

The following code shows how a saved file is opened (unpickling):

```
indexItem = model.index(index.row(), 0, index.parent())
filePath = self.model.filePath(indexItem)
self.file_to_load = filePath
inputfile = open(filePath, "rb")
self.file_objects.append(pickle.load(inputfile))
```

RESULTS AND DISCUSSIONS:

The final GUI is shown in this image:



File: E:/Data/org/RU6D2003.TXT

X-Axis: ZDispTotal   Y-Axis: ZLoadmN

	Time	ZDispTotal	XD
50	19.95	217.551	-39.4074
51	20.375	221.498	-40.1226
52	20.8	225.347	-40.771
53	21.21	229.125	-41.474
54	21.625	233.094	-42.0868
55	22.04	237.013	-42.9517
56	22.45	240.709	-43.5763
57	22.86	244.463	-44.1398
58	23.275	248.283	-44.8192
59	23.69	252.239	-45.5523
60	24.1	255.905	-46.2891
61	24.52	259.677	-46.9844
62	24.93	263.424	-47.6021
63	25.335	267.062	-48.2875
64	25.75	271.964	-49.3178
65	26.17	275.81	-50.0018
66	26.58	279.524	-50.5885
67	26.99	283.426	-51.0814
68	27.41	287.225	-51.7842

```
Loaded the CDA program
```

1. Multiple tabs to open any number of files at the same time.
2. Data explorer to see the data of the file in the current tab.
3. Saved files in “.pkl” format.

CONCLUSION:

The target for the next phase is to develop a concrete strategy for connecting this GUI to the Control System for real time data analysis.