import wx

import UniversalLibrary as UL

"""app = wx.App(False) # Create a new app, don't redirect stdout/stderr to a window.

frame = wx.Frame(None, wx.ID\_ANY, "Kepco 50-8M Control") # A Frame is a top-level window.

frame.Show(True) # Show the frame.

app.MainLoop()"""

voltage=0

current=0

Conv = 1;

BoardNum = 0

Chan = 3

Gain = 20

var = 1.05

EngUnits=0;

VtoA = 0.34

MAXCURRENT = 4;

MINCURRENT = -4

isClosed = False

curUp = False

class Main(wx.Frame):

def \_\_init\_\_(self, parent,title):

wx.Frame.\_\_init\_\_(self,parent,title=title,size=(800,800))

#self.control = wx.TextCtrl(self)

self.CreateStatusBar()

self.InitUI()

self.checkGain();

'''Button Stuff'''

#y = self.ask(message = 'Enter the voltage here')

#print y

'''Exit menu stuff'''

filemenu = wx.Menu()

ex = filemenu.Append(wx.ID\_EXIT, "&Exit", "Exit the program")

self.Bind(wx.EVT\_MENU, self.OnExit, ex)

menubar=wx.MenuBar()

menubar.Append(filemenu,"&File")

self.SetMenuBar(menubar);

self.Show(True)

def InitUI(self):

pnl = wx.Panel(self)

self.btnV = wx.Button(pnl, 20, "Set Voltage Input", (50,50))

self.btnA = wx.Button(pnl, 20, "Set Current Input",(200,50))

self.quote = wx.StaticText(pnl, label ="Voltage And Current Control", pos=(0,0))

#self.textField = wx.TextCtrl(self, pos=(50,150));

self.btnV.Bind(wx.EVT\_BUTTON, self.OnVoltagePress)

self.btnA.Bind(wx.EVT\_BUTTON, self.OnCurrentPress)

self.txtV = wx.TextCtrl(pnl,pos = (50, 150) , size =(140,30))

self.txtV.SetValue("Volts")

self.txtA = wx.TextCtrl(pnl,pos = (200, 150) , size =(140,30))

self.txtA.SetValue("Amperes")

self.labelV = wx.TextCtrl(pnl, pos = (50,220), size = (140,35),style=wx.TE\_READONLY)

self.labelV.SetValue("Voltage is 0 V");

self.labelA = wx.TextCtrl(pnl, pos=(200,220), size = (140,35),style=wx.TE\_READONLY)

self.labelA.SetValue("Current is 0 A");

self.SetSize((500, 400))

self.SetTitle('Kepco 50-8M Control')

self.Centre()

self.Show(True)

def OnExit(self, event):

global isClosed

isClosed = True

self.setVoltage(0);

self.setCurrent(0)

self.Close(True)

def vLabelUpdate(self,volt):

self.labelV.SetValue("Voltage is "+str(volt)+" V")

def aLabelUpdate(self,amp):

self.labelA.SetValue("Current is "+str(amp)+" A")

def setVoltage(self, volt):

global voltage

voltage=volt;

def getVoltage(self):

global voltage

return voltage;

def setCurrent(self, amp):

global current

current=amp;

def getCurrent(self):

global current

return current;

def OnVoltagePress(self,event):

vString =self.txtV.GetValue();

print(vString);

self.updateVoltage(float(vString));

def OnCurrentPress(self,event):

aString =self.txtA.GetValue();

print(aString)

self.updateCurrent(float(aString));

def checkGain(self):

global Gain,var,Conv

if(Gain==200):

Conv = 10.0/9\*var

elif(Gain==23):

Conv = (10.0-0.19)/17

elif(Gain==15):

Conv = 5.0/13

def updateVoltage(self,volt):

global voltage, EngUnits,BoardNum,Chan,Gain,Conv,VtoA;

self.setVoltage(volt);

EngUnits = self.getVoltage();

DataValue = UL.cbFromEngUnits(BoardNum, Gain, EngUnits\*Conv, 0)

UL.cbAOut(BoardNum, Chan, Gain, DataValue)

self.vLabelUpdate(voltage);

self.aLabelUpdate(voltage\*VtoA)

def updateCurrent(self,amp):

global current, EngUnits,BoardNum,Chan,Gain,Conv,VtoA, curUp,MAXCURRENT,MINCURRENT;

if amp<= MAXCURRENT and amp>= MINCURRENT:

self.setCurrent(amp);

EngUnits = self.getCurrent()/VtoA;

DataValue = UL.cbFromEngUnits(BoardNum, Gain, EngUnits\*Conv, 0)

UL.cbAOut(BoardNum, Chan, Gain, DataValue)

self.aLabelUpdate(current);

self.vLabelUpdate(current/VtoA)

curUp = True;

curUp = False;

else:

self.aLabel.setValue("Your current is too high: It must be between 4 and -4 Amps")

def wasCurrentUpdated(self):

global curUp;

return curUp;

def isItClosed(self):

global isClosed

return isClosed

app = wx.App(False);

frame = Main(None, 'Small Editor')

app.MainLoop()