Micralyne µTK User Guide

Derek Wong

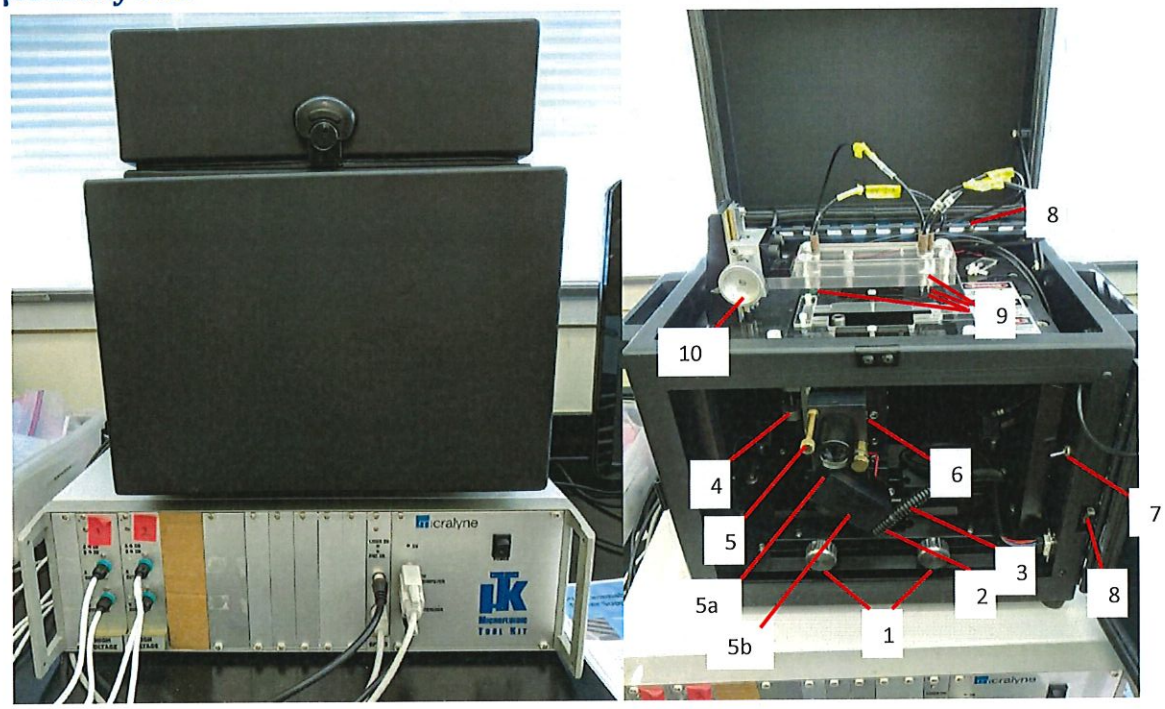
For the Remcho Research Group

Date: 5/30/18

General Information

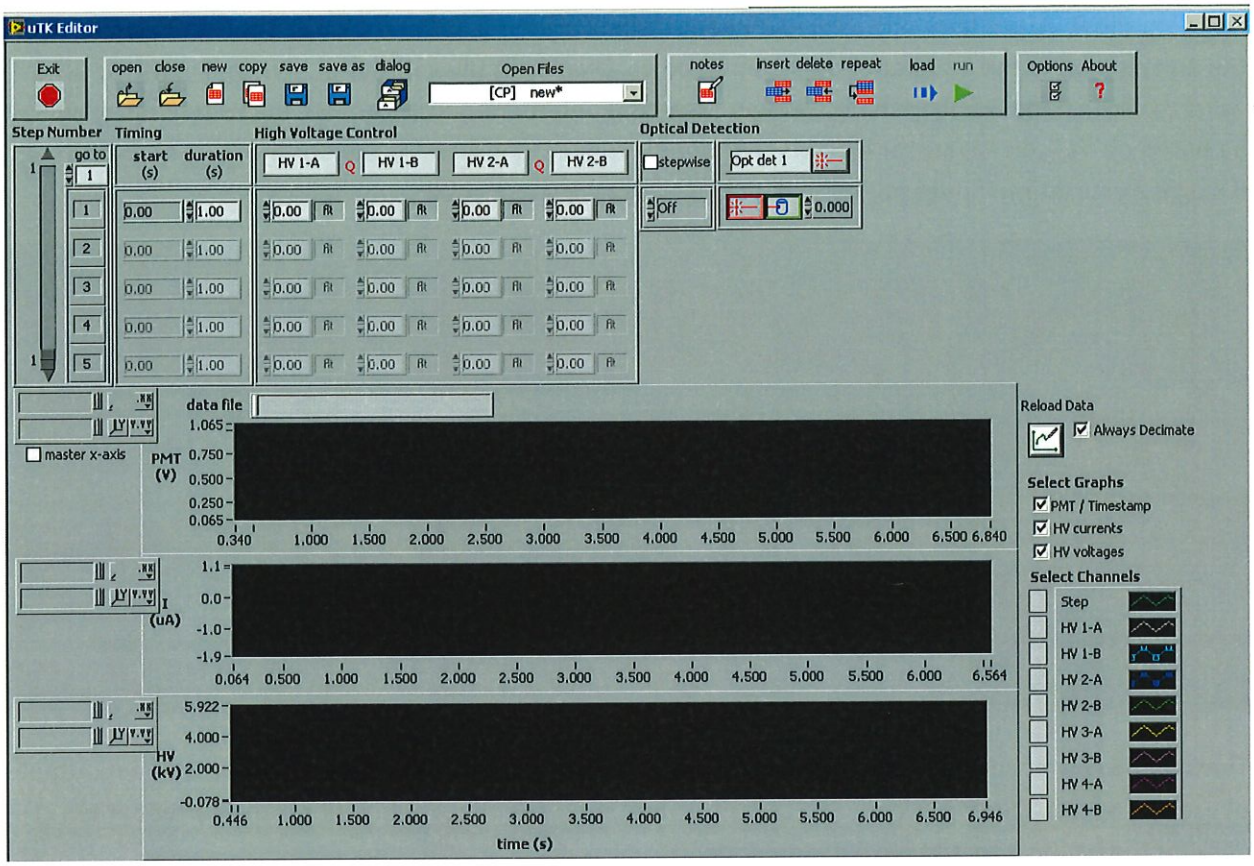
The Micralyne microfluidic toolkit, µTK, is an instrument designed to both control the electrophoresis and detect fluorescence. It has four independent voltage probes, a laser emitter, and a photomultipler tube (PMT). The voltages can range from 0-6kV, the laser is at 532 nm wavelength and the PMT filter is for 568.2 nm wavelength. There is another optics assembly (laser: 635 nm, PMT filter: 670 nm) for the µTK, but the group does not own it. The latches open by flipping it level, then twisting (clasp knobs).

µTK layout

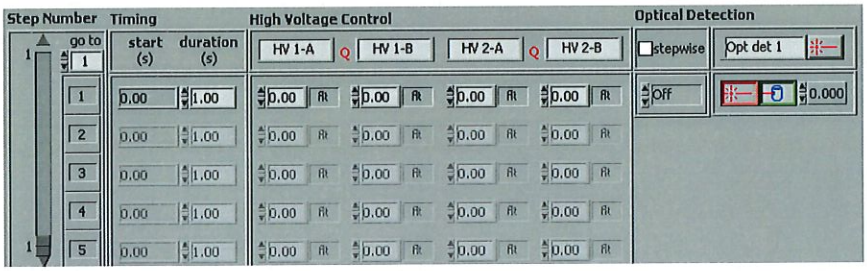


inside the µTK

1) Course X (horizontal) adjustment: releases the base blocks
2) Fine X (horizontal) adjustment
3) Fine Y (horizontal) adjustment
4) Fine Z (vertical) adjustment
5) Screw pin for holding optics option in place
    a. Visual lens (currently in position)
    b. PMT
6) Focus versus scatter brass knob for laser
    a. In – focus
    b. Out – scatter
7) Switch to turn on LED's if laser is on
8) Safety interlock for when door is open or closed
9) Voltage probes – platinum wire

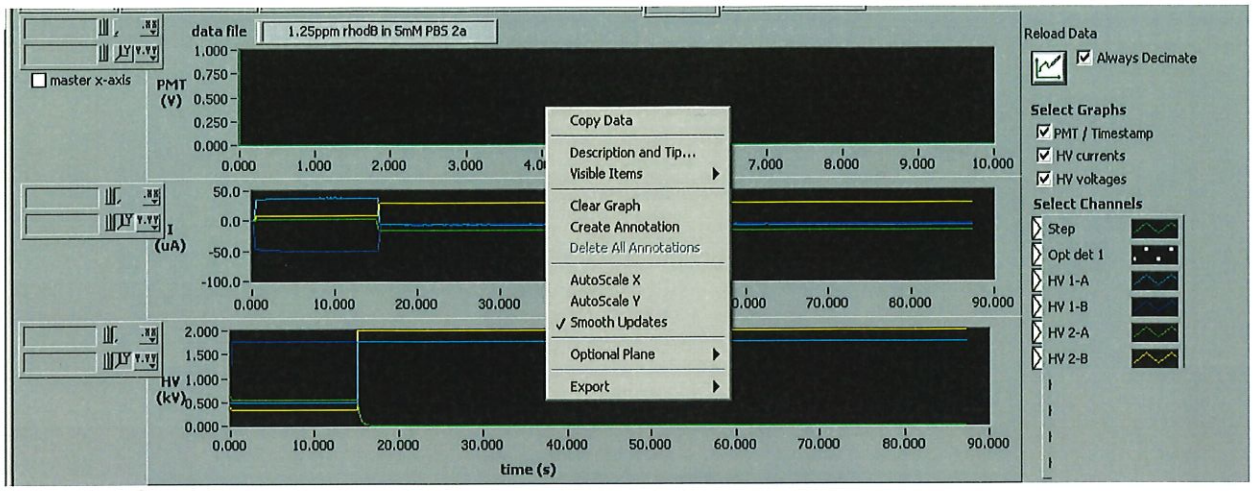


This is the GUI the user will interact with.



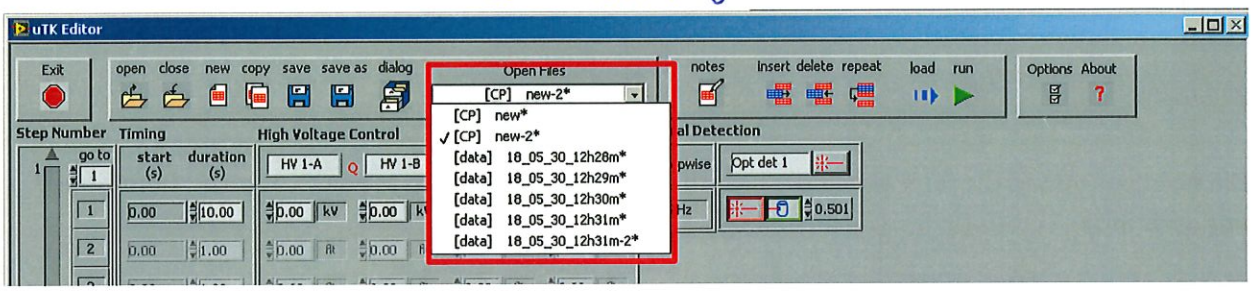
The voltages, step times, and acquisition setup are all set in the screenshot above. The stepwise check box allows the user to set a different PMT setting for each step. The default is the same laser and PMT settings for all steps. The button to the right of "Opt det 1" turns on the laser when the µTK isn't running. The laser will only turn on when the top lid of the µTK is closed. The bottom lid can be open. For the control program steps, having the button depressed means the laser is on. The green outline button next to it is the PMT. The number to the right of that represents the gain which maxes out at 0.800. The number below stepwise represents the acquisition rate (frequency) of the PMT. It has values from 0 – 200 Hz.

To jump from step to step, the user can either use the "go to" box at the top left or the slider. The step put in will be moved to the top.



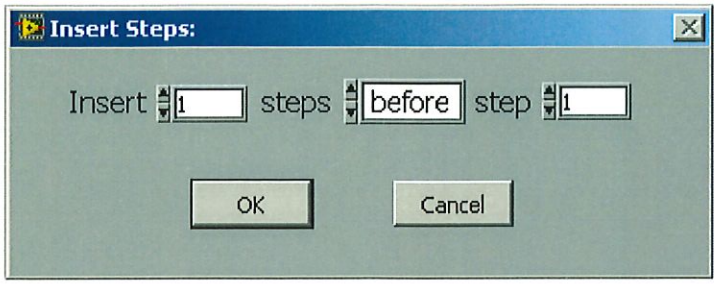
The one two above is the dropdown menu for the x y axes control. The one above is a right click menus for graph options and exporting.

CP - control program

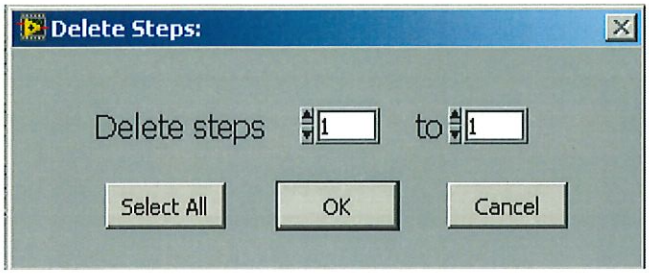


To view different files clicking on the drop down below "Open Files" will pull up this menu. An asterix after means that the file has not been saved or edited since its last save.

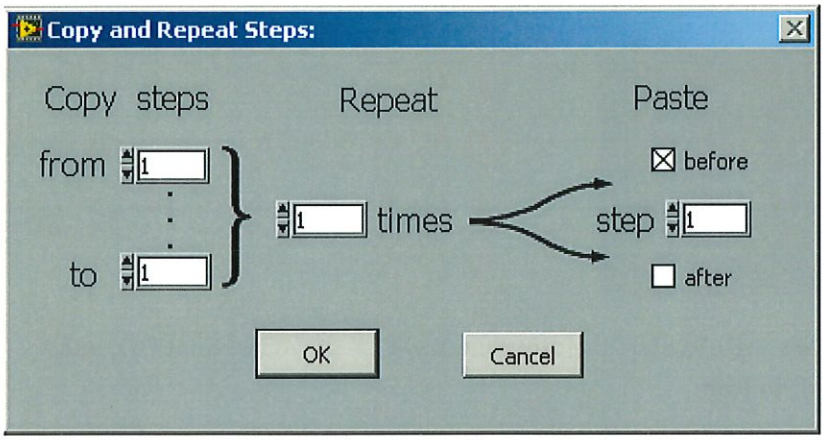
The top row of icons shown above will be how the user saves and opens files to the computer.



Insert will prompt how many lines to add and where to add it. There is an option to add before or after a step number. The inputted step(s) will be a copy of the first step in the control program.



Delete will ask for the range of steps to remove. If the range were to be invalid (7 to 6 for example), the program will automatically adjust the other step value to be valid (eg. 6-6, change to 7-6, will automatically adjust to 7-7).



Repeat will ask for which range of steps to copy, how many repetitions, and where to insert (before or after which step).

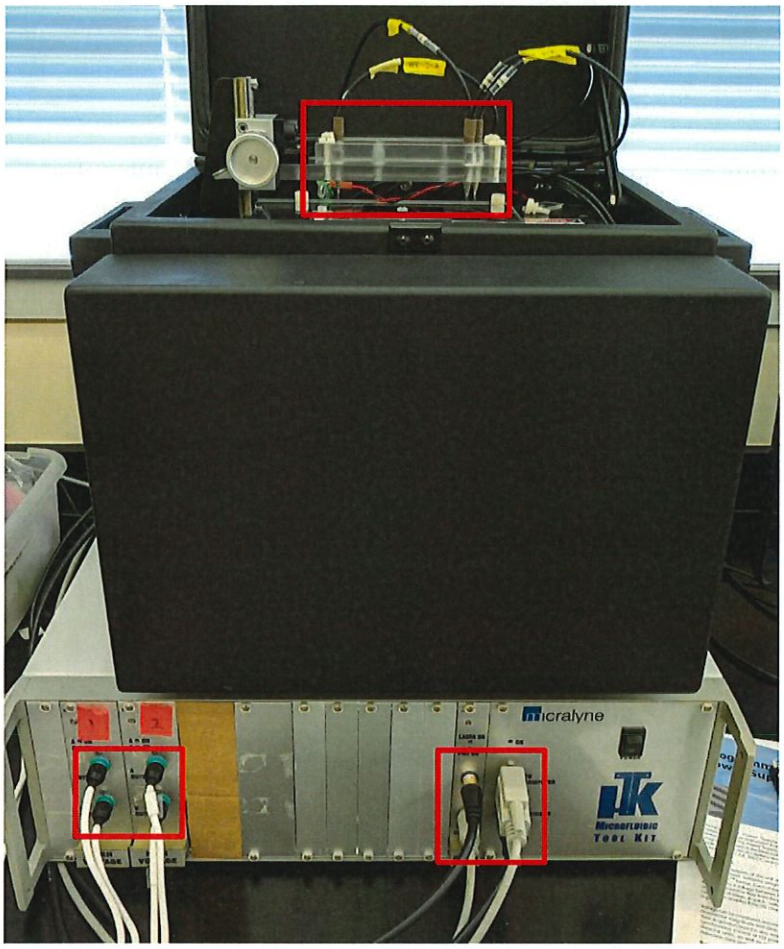
Load will load the current Control Program, CP, into the power supply unit onboard chip.

One should see this pop up briefly:

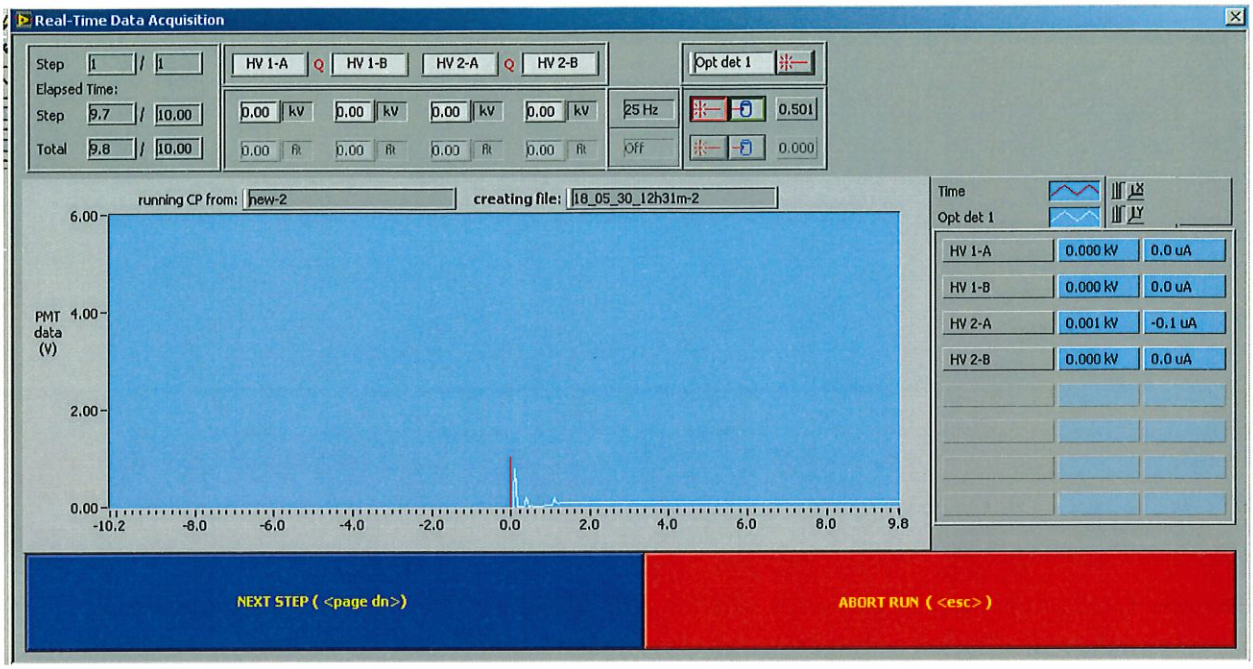
Operating the tool

Pre-operation

Before turning on the µTK, confirm that all the electrical and data connections are good. These include the connections of the outside insulated wires to the high voltage power supply boards and the back of the µTK highlighted in the picture below.

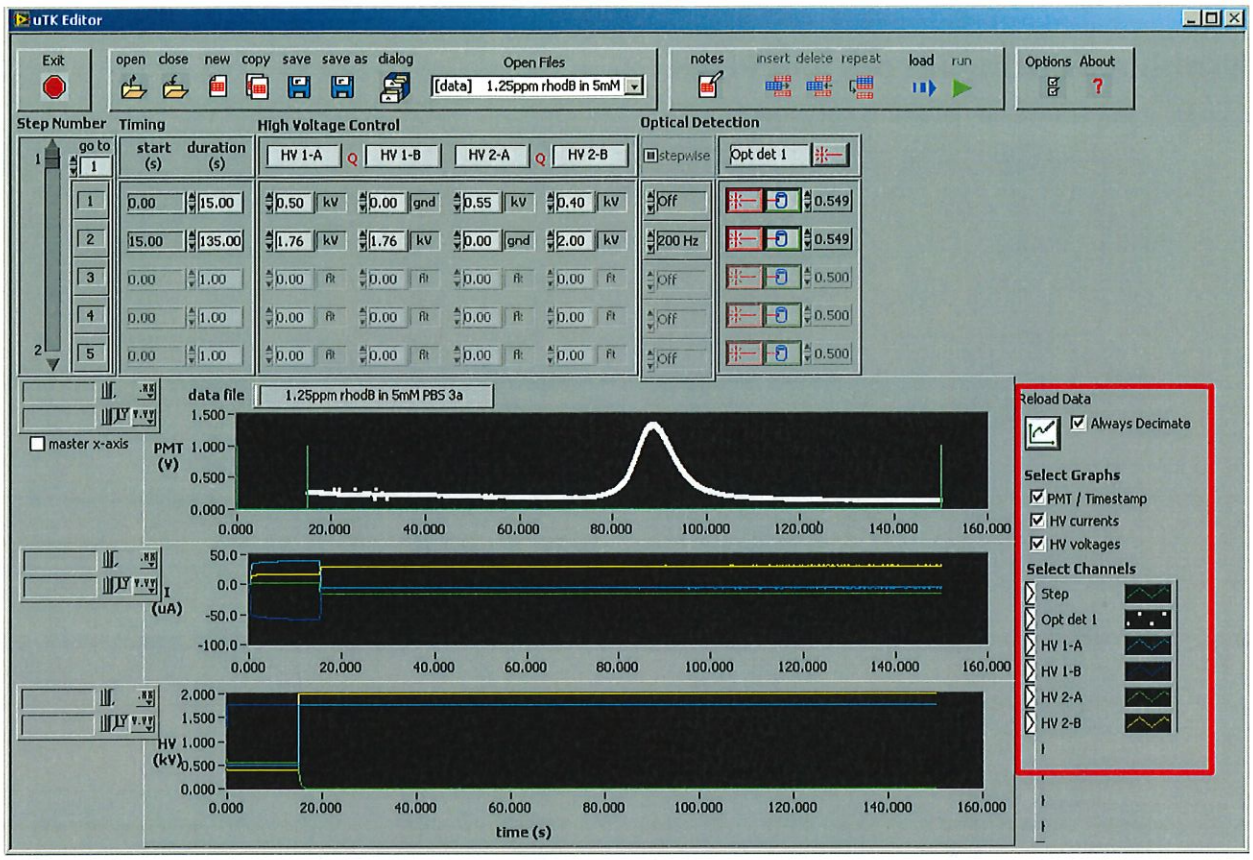


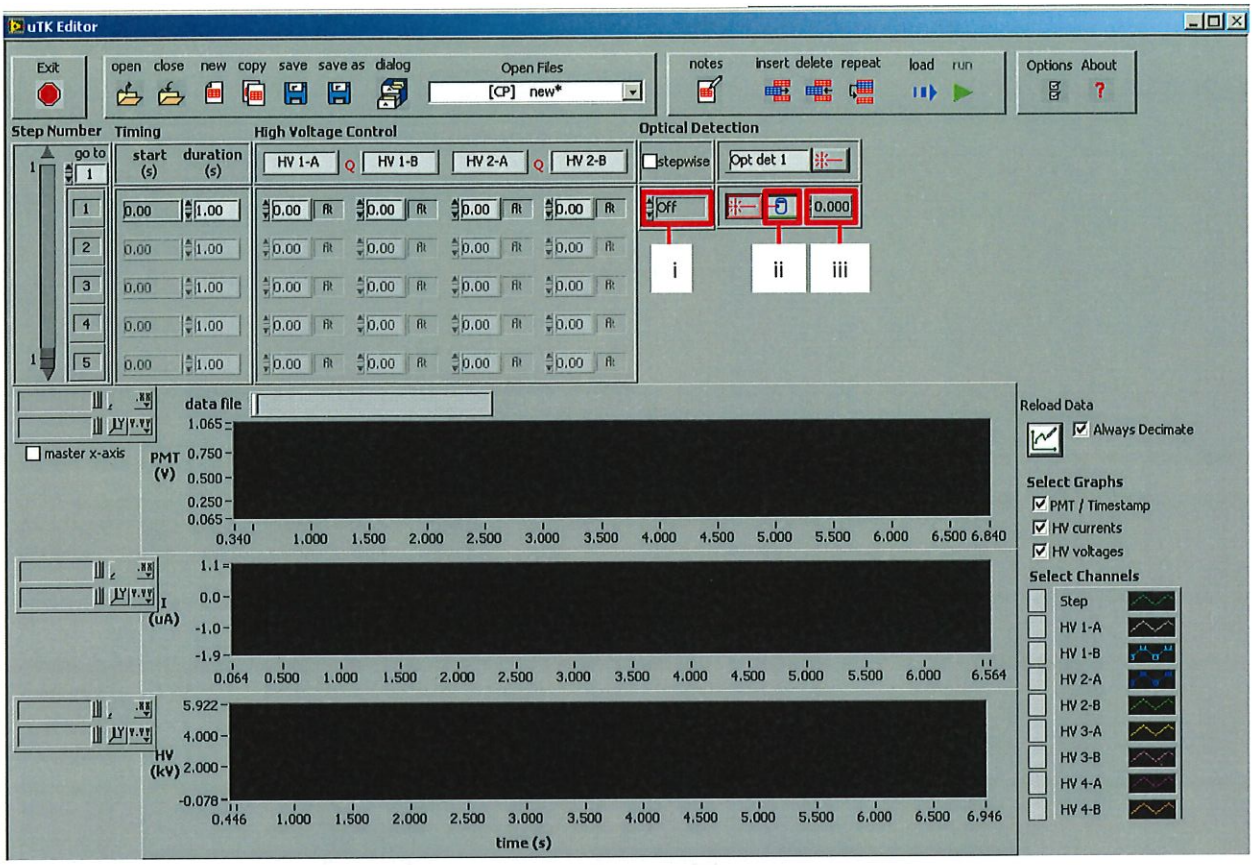
(in the pictures, visual lens is the current viewing option). If you see a detection that looks like below, you probably forgot to turn the optics back to the PMT or the PMT gain is too low:



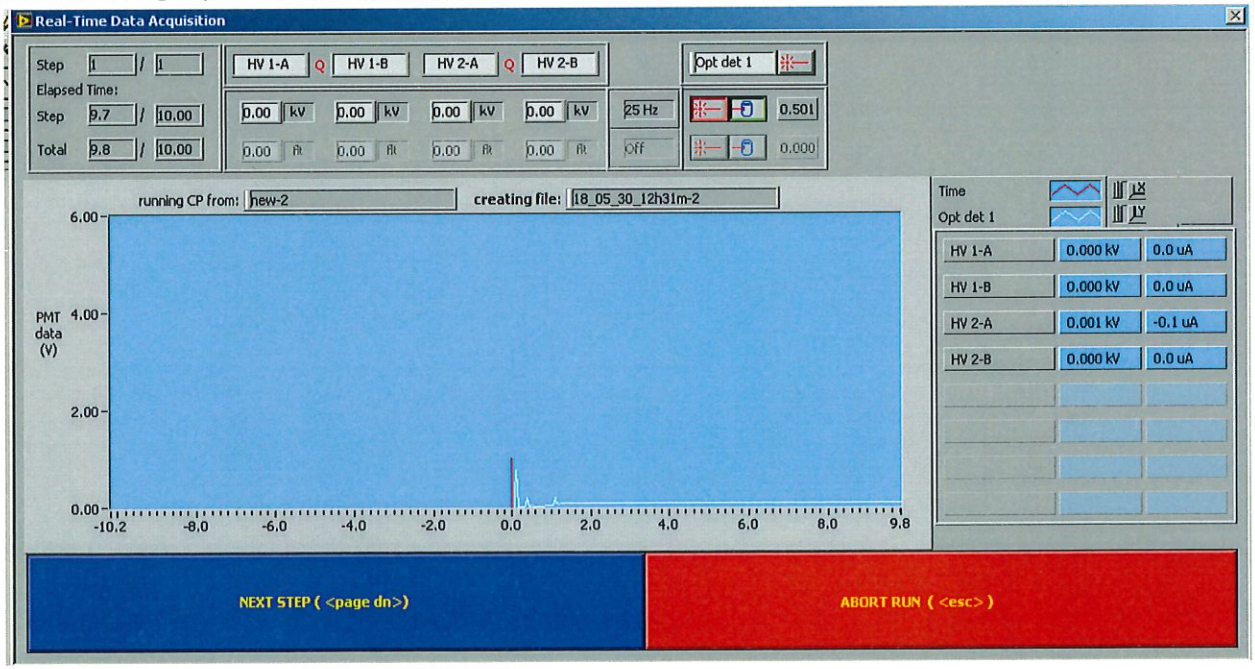
Operation

Load a previous control program in via the "Open" button previously mentioned. Afterwards, hit the "Load" button to download the program to the high voltage board. Lastly, click the "Run" button to actually run the program. This will pull up the screenshot below where it will show the currents, voltages, and last 20 seconds of PMT data. The user can skip to the next step or cancel the experiment with the two large button at the bottom. Otherwise it shows the current step at the top and the step following along with the time at the left. **Warning: If the PMT goes over 5V, there is a chance to burn out the PMT. There is normally some noise in the data, so if the PMT data value is a flat line around 5V, abort the run and change the PMT gain.** Also if the current output goes above 100 or below -100 μA, the μTK will abort.


uTK Editor
Exit
open close new copy save save as dialog
Open Files
[data] 1.25ppm rhodB in 5mM
notes insert delete repeat load run
Options About
Step Number
Timing
High Voltage Control
Optical Detection
go to
start (s)
duration (s)
HV 1-A
HV 1-B
HV 2-A
HV 2-B
stepwise
Opt det 1
data file
1.25ppm rhodB in 5mM PBS 3a
master x-axis
PMT (V)
I (uA)
HV (kV)
time (s)
Reload Data
Always Decimate
Select Graphs
PMT / Timestamp
HV currents
HV voltages
Select Channels
Step
Opt det 1
HV 1-A
HV 1-B
HV 2-A
HV 2-B



2) There is a large spike at the beginning when I turn on the PMT and laser or it isn't steady.



a. Fix:

   i. Turn on the laser one step before

3) Re-updating LabVIEW Metholody

For GUI to work properly the uTK folder must in the base C: drive (a.k.a the folder must be C://uTK). The update path was to load the uTK LabVIEW 5 file and open it with LabVIEW 7. A runtime engine will work fine for this part. Afterwards it was opened in LabVIEW 2012. Each time make sure to save the changes that occur as it recompiles. At this point, LabVIEW said which blocks were not working properly. I (Derek) did two things at this point: In the included library, I deleted out the blocks and had LabVIEW update the code with its own library instead of an outside one. (Library is the storage of code snippets, saved programs effectively). Then for the ones that weren't working properly still, I checked to see if they were obsolete and replaced them with the current versions of they were (e.g. xscaleinfo became xscaleranger).