RICH Spherical Mirror d0 Test Station Improvements

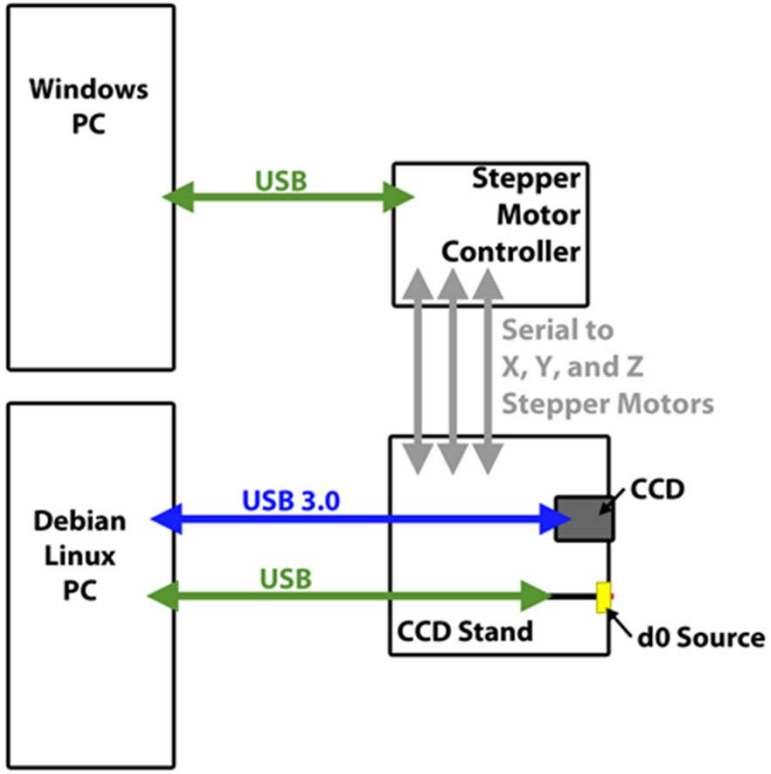
Tyler Lemon

March 20, 2019

d0 Measurement

- Used to measure RICH spherical mirror radius of curvature and surface quality.

- Test station uses:
  - Ximea CCD
  - Thorlabs stepper motors
  - Mirror stand
    - Manual position adjustment
  - Debian PC
    - DAQ, analysis
  - Windows PC
    - Stepper motor controls.



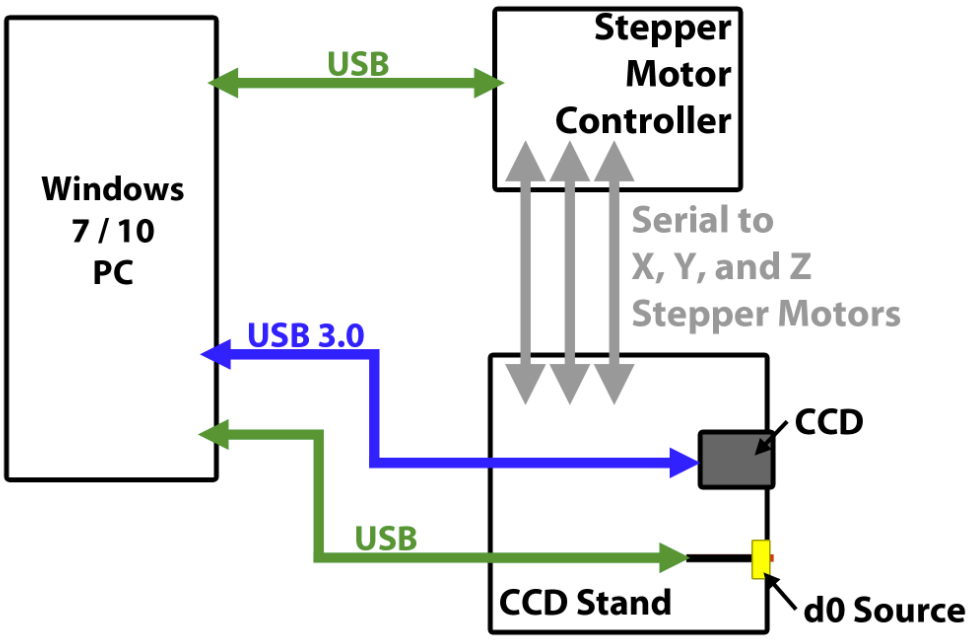
d0 Improvements

- For future RICH sectors, we want to develop a better test station.

- Why?
  - Gain network access by moving away from using Debian PC.
    - Only Red Hat Enterprise Linux, Mac OS, and Windows supported and allowed network access at JLab.
  - Automation of the testing process.
    - Currently, user has to execute terminal commands for every step.
  - Enable multiple measurements for a single mirror.
    - Due to time it took to manually enter commands, one image was taken at each position.

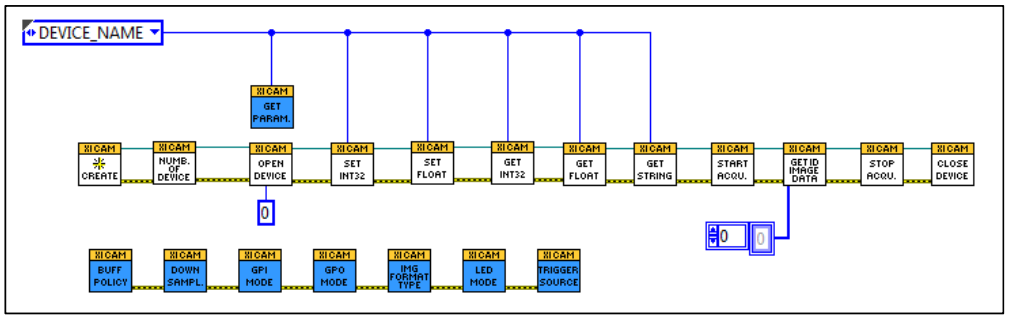
Suggestions for Improvement

- Develop a version of the d0 calculation able to run on Windows.
  - Windows 7 vs Windows 10 not an issue.
    - Possibility of timing differences is irrelevant.
    - New program will still be faster than manually typing in commands.
- Develop a single LabVIEW program for all steps of the test.
  - CCD data acquisition
  - Data analysis
  - Stepper motor movement

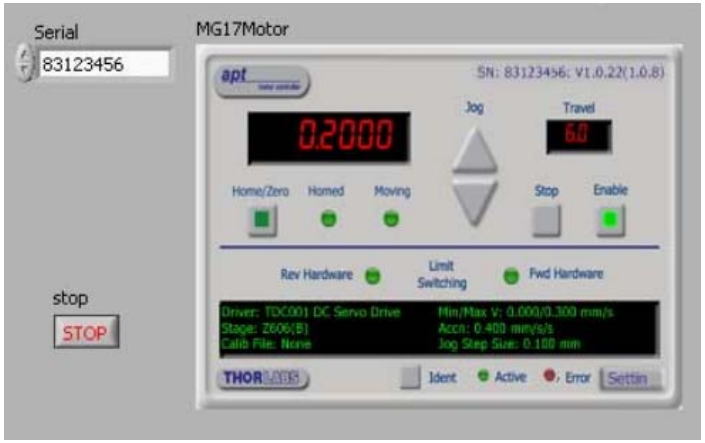


Benefits of LabVIEW

- Built-in user interface
- Can be run remotely.
  - User can remotely access test PC from another location allowing test to be performed in as low-light conditions as possible.
- Manufacturer-developed subVIs already exist for Ximea CCD and Thorlabs stepper motors.



Existing Ximea CCD SubVIs



Existing Thorlabs Stepper Motor SubVIs

Progress of Improvements

- d0 calculation's C/ROOT code successfully converted to Python.
  - C used for calculations, ROOT for plotting
  - Converted to Python to gain better understanding of how d0 is calculated.

- Development of LabVIEW d0 calculation in progress.
  - Attempting to replicate steps entirely in LabVIEW for easier integration into final program.
  - If significantly slower or less accurate, will look into running either the C or Python program from LabVIEW.

Comparison of C and Python Results

Mirror	Source	d0 [mm]	d0Err [mm]	DX [mm]	DY [mm]	X [mm]	Y [mm]
1	Python	1.500	0.020	1.474	1.452	5.313	5.467
	C	1.509	0.020	1.474	1.452	5.288	5.475
	Difference	0.009	0.000	0.000	0.000	0.025	0.008
2	Python	1.420	0.020	1.408	1.468	5.637	5.786
	C	1.429	0.020	1.408	1.469	5.632	5.788
	Difference	0.009	0.000	0.000	0.001	0.005	0.002
2C	Python	1.200	0.020	1.226	1.248	5.198	6.110
	C	1.190	0.020	1.227	1.248	5.173	6.125
	Difference	0.010	0.000	0.001	0.000	0.024	0.014
3	Python	1.640	0.020	1.622	1.485	5.682	5.087
	C	1.629	0.020	1.622	1.529	5.649	5.087
	Difference	0.011	0.000	0.000	0.044	0.032	0.000
3C	Python	1.300	0.020	1.331	1.325	4.829	5.868
	C	1.290	0.020	1.331	1.325	4.861	5.912
	Difference	0.010	0.000	0.000	0.000	0.032	0.043

Mirror	Source	d0 [mm]	d0Err [mm]	DX [mm]	DY [mm]	X [mm]	Y [mm]
4	Python	1.820	0.020	1.738	1.820	6.413	5.588
	C	1.829	0.020	1.738	1.821	6.418	5.567
	Difference	0.009	0.000	0.000	0.001	0.005	0.021
4C	Python	1.400	0.020	1.375	1.375	5.566	6.314
	C	1.390	0.020	1.375	1.375	5.562	6.334
	Difference	0.010	0.000	0.000	0.000	0.004	0.020
5	Python	1.420	0.020	1.375	1.397	4.917	5.659
	C	1.409	0.020	1.375	1.397	4.936	5.682
	Difference	0.011	0.000	0.000	0.000	0.019	0.023
5C	Python	1.360	0.020	1.353	1.441	5.269	5.863
	C	1.371	0.020	1.353	1.441	5.274	5.841
	Difference	0.011	0.000	0.000	0.000	0.005	0.022
6	Python	1.780	0.020	1.650	1.776	5.159	4.966
	C	1.771	0.020	1.650	1.777	5.173	4.982
	Difference	0.009	0.000	0.000	0.000	0.014	0.016

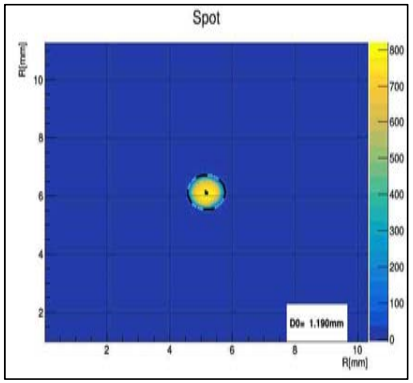
Largest d0 Difference

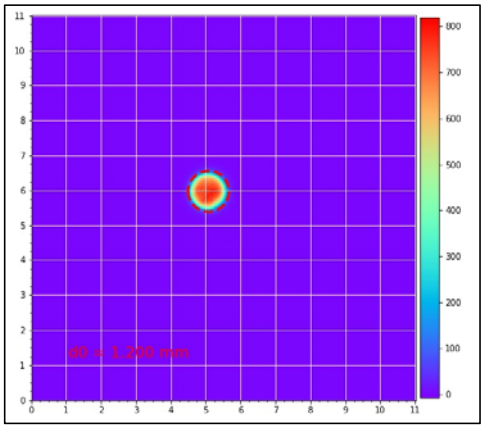
	d0 [mm]	d0Err [mm]	DX [mm]	DY [mm]	X [mm]	Y [mm]
AVERAGE DIFFERENCE	0.010	0.000	0.000	0.005	0.017	0.017
STANDARD DEVIATION	0.001	0.000	0.000	0.013	0.011	0.012

Comparison of Plots from ROOT and Python

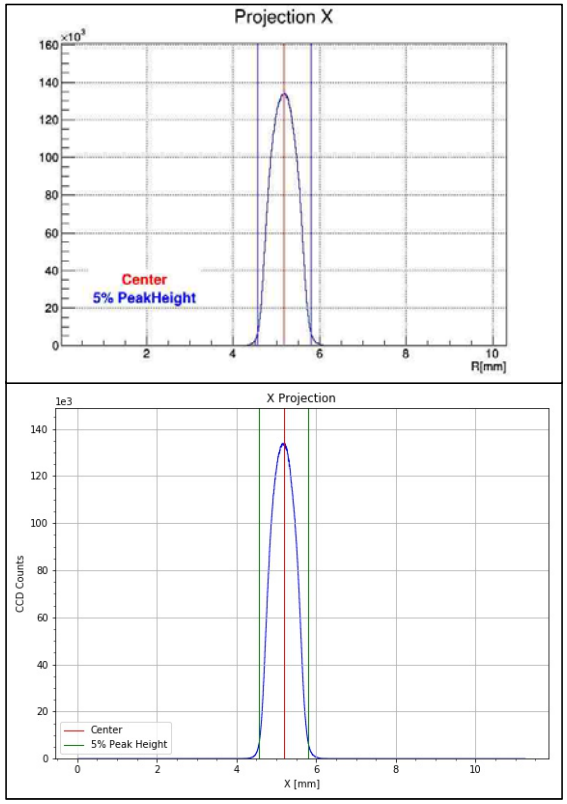
ROOT

Python

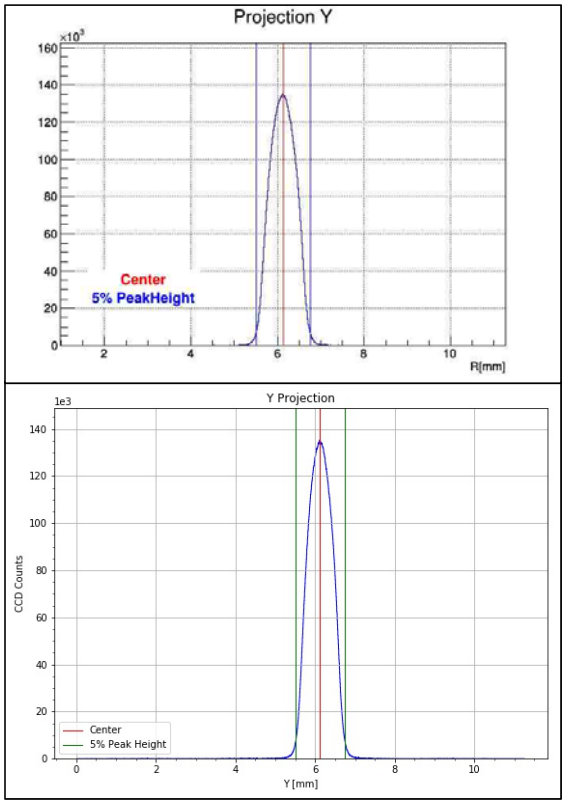




Imaged of d0 spot from CCD
(colors are different due to color mapping in C/ROOT vs Python)



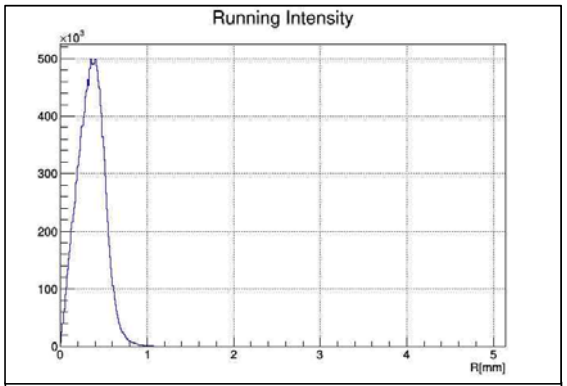
X-projection of d0 spot used to find center of spot on x-axis (apparent offset between graphs due to axis limits).



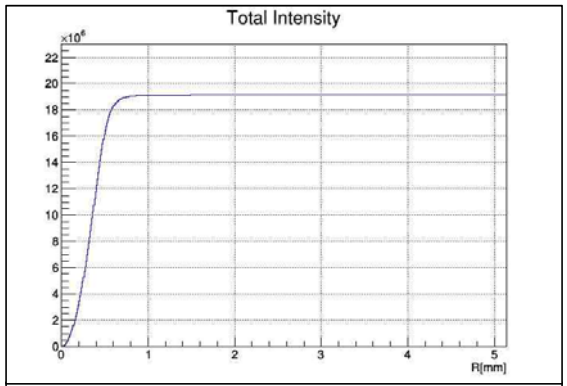
Y-projection of d0 spot used to find center of spot on y-axis (apparent offset between graphs due to axis limits)

Comparison of Plots from ROOT and Python

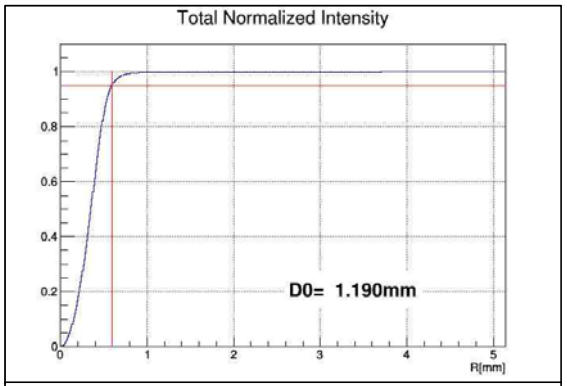
ROOT



Summed pixel counts for in rings of increasing diameter, centered at spot center.

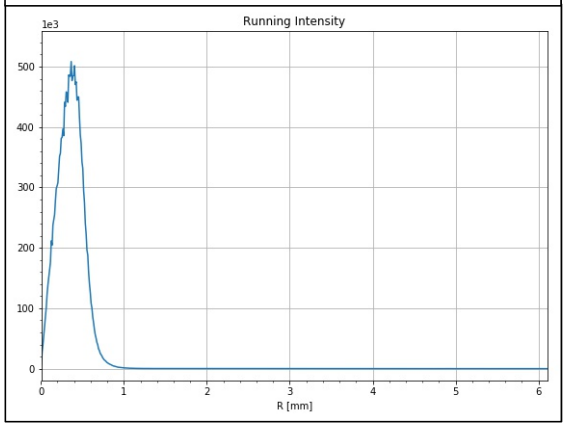


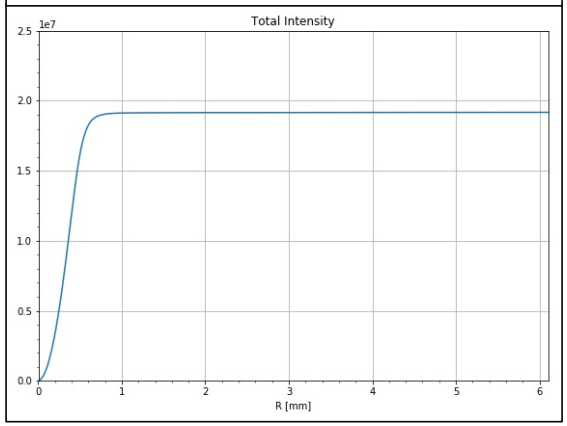
Summed pixel counts for in circles of increasing diameter, centered at spot center.

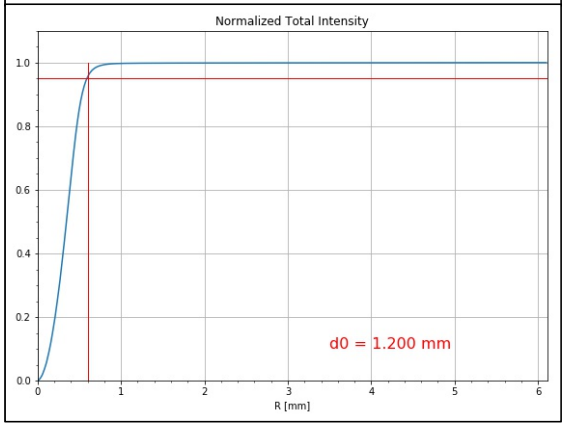


Normalized "Total Intensity" plot.
X value for plot's value closest to 0.95 is d0.

Python







Obstacles Faced

- Finding cause of differences between C and Python calculation results.
  - In theory, results should be the same because same calculation method and same data are used.
  - Possible cause is an indexing mistake in Python program
    - CCD pixels are 0.0055 mm x 0.0055 mm; 0.01 mm (average difference) is ~2 pixel lengths off.
    - 0.01 mm is also the step size used for running intensity and total intensity calculation.

- Implementing calculation in LabVIEW.
  - Initial impressions are that iterating through all elements of the CCD array (size 2048 x 2048) for calculations takes a very long time in LabVIEW.
  - Attempting to use LabVIEW's built-in Python interface, but having problems pointing LabVIEW to the modules used in the Python program.

Conclusion

- DSG is investigating ways to improve d0 measurement tests for RICH spherical mirrors.

- The original C executable has successfully been converted to Python.
  - Average difference in d0 result between C and Python is 0.010 ± 0.001 mm.
  - Further investigation underway to find source of difference.

- Development of a LabVIEW 2018 program is underway.
  - Program will handle all test station tasks (CCD data acquisition, analysis, and CCD stand stepper motor movement).