Analyzing Hall B Solenoid FastDAQ Data with Python

Brian Eng
2022-11

Analyzing Hall B Solenoid FastDAQ Data with Python

The Hall B magnet fastDAQ system records data at 10 kHz on voltage taps (21 in total) and current readout on both the solenoid and torus. This data is saved to ROOT files that are saved in 2 GB chunks in order to make retrieval faster/easier. There are two utility programs that also utilize the ROOT API, one to view the data and make plots and another that will convert the ROOT file to an ASCII file. In normal cases the first program is sufficient to view the data around an event, zooming in to times of interest and adding specific voltages or comparators.

However this only provides viewing the data, for additional analysis or performing any calculations on the data the second ROOT program is needed to export the data. Once exported other programs/utilities can be used to perform this analysis. For extremely short (on the order of a few single-digit ms) time spans Excel is normally used as it is fairly straight forward. For longer time spans the time to even open the file becomes prohibitive not even taking into accounts performing calculations or plotting.

- **Existing ROOT programs for fastDAQ data work well for viewing the data, but don't easily provide ways to perform analysis**
- **Python program was implemented to parse exported ASCII files into a Pandas dataframe**
- **Additional calculations and plotting are significantly easier**

09/29/22Detector Support Group

1

Analyzing Hall B Solenoid FastDAQ Data with Python

In order to perform this analysis in a timely manner a Python program was written in order to parse and perform calculations on the data. The program uses the Pandas library to import the ASCII file as a dataframe. An optimization to save this dataframe as a pickle file, which is a serialized Python object. This saves having to reading/parsing the ASCII file when changes to the calculation code are done. The time for reading/parsing an ASCII file with ~1.2M entries is around 5 seconds versus around 1 second for the pickle file. Once in a dataframe performing calculations becomes trivial, e.g. some math done were multiplying voltages by current to get power or calculating the area under the curve both of which are single lines of code.

With the Python program additional calculations and plots can be produced from the fastDAQ data..