Data Quality Monitoring and Selection:

For analysis of data, the quality of data that we have must be assured that it is good. The criteria upon which it is described as good or bad depends on the Run Group and its physics goals. But some initial quality control can be done by studying the following: the number of good electrons, normalized by the Faraday Cup [FC] charge where the criteria for good electron depends on the individual Run Group based on some cuts and physics goals.

Electron Counts:

The main focus of data quality monitoring is to count electron yield, denoted by Nel, and normalize by the FC charge denoted by FC. This would be done for each hipo file in the DST bank for all RGF runs. For run periods with steady experimental conditions, same DAQ conditions, same trigger conditions, and configurations, theoretically we expect this ratio to be consistent. But change in beam conditions, detector trips, change in target gas type, background conditions, and trigger issues, DAQ issues could cause outliers. So, the main aim of this study is to find such outliers and identify DST files with those values.

Determining the Normalized Electron Yield

First of all, we have to count the total number of electrons that satisfies all good electron selection criteria. The following are the good electron selection criteria:

- PID == 11
- -20 < el_vz < -17 cm
- 50 < lu < 410, lv < 410 and lw < 370
- Nphe > 2
- $EC_{in} + EC_{out} + EPCal > 0.1$
- SF > 0.2
- EPCal > 0.1
- Nel/ event == 1 and First particle

The total number of electrons that satisfies all above selection cuts were counted for all individual DST files and documented in a tabular form.

Determining the Faraday-Cup information for each DST File

For each DST file, we need to calculate the accumulated FC charge. This information is stored in the RUN::Scaler bank which is present in each DST file.
The structure of this bank would look like the following:

```
"name": "RUN::scaler",
"group": 10000,
"item": 14,
"info": "Run-integrated, analyzed scalers.",
"entries": [
    {"name":"fcupgated",  "type":"F", "info":"Beam charge, integrated from beginning of run, DAQ-gated (nano-Coulomb)"},
    {"name":"fcup",       "type":"F", "info":"Beam charge, integrated from beginning of run, ungated (nano-Coulomb)"},
    {"name":"livetime",   "type":"F", "info":"DAQ livetime during one scaler period"}
]
```

We would use the fcupgated beam charge, which contains the faraday cup gated Beamcharge integrated from the beginning of each run.

Similarly, another bank HEL::Scaler is also present which also has this information, along with other scalars information.

```
"name": "HEL::scaler",
"group": 10000,
"item": 16,
"info": "Helicity-gated, analyzed scalers.",
"entries": [
    {"name":"fcupgated",  "type":"F", "info":"Beam charge from Faraday cup, DAQ-gated (nano-Coulomb)"},
    {"name":"fcup",       "type":"F", "info":"Beam charge from Faraday cup, (nano-Coulomb)"},
    {"name":"slmgated",   "type":"F", "info":"Beam charge from SLM, DAQ-gated (nano-Coulomb)"},
    {"name":"slm",        "type":"F", "info":"Beam charge from SLM (nano-Coulomb)"},
    {"name":"clockgated", "type":"F", "info":"Clock, DAQ-gated"},
    {"name":"clock",      "type":"F", "info":"Clock"},
    {"name":"helicity",   "type":"B", "info":"Helicity state, HWP-corrected"},
    {"name":"helicityRaw","type":"B", "info":"Helicity state"}
]
```

The one difference to note in "fcupgated" Beamcharge between these two banks is:

- RUN::Scaler - beam charge from FC integrated from beginning of run
- HEL ::Scaler - beam charge from FC

Missing FC cup information for a group of runs:

Following the above-mentioned procedures, while we were trying to extract beam charge information for each DST file of RGF we noticed that for Runs: 12857 – 12877, there was not any scalar information for these runs. We even tried to look for alternative scalars that could lead us to extract beam charge from these scalars' information. But none of the scalars below were available for the above runs.

- No FC -gated [RUN::Scaler and HEL:: Scaler Bank]
- No FC-ungated [RUN::Scaler and HEL:: Scaler Bank]
- No SLM-gated
- No SLM-ungated

So, we had to find a way to calculate any measurable quantity from available scalars or any combination of scalars that could have a one-to-one correspondence with beam charge.

Beamcharge extraction using clock [DAQgated] and beamcurrent:

The RAW::epics bank present in the clas12 data stream contains the EPICS information. Using information from this bank,
We tried to interpret beam charge as:

Beamcharge = clock. current
Clock = DAQ gated clock from HEL::Scaler bank
Current = Current from RAW::epics bank [2C21, 2H01 and 2C24]

So, for each reading of the clock in the HEL::Scaler bank we multiplied that clock with Current [I2C21] and summed each event in a file for each DST file.

To verify that the beamcharge extracted from this method gives the correct result we have to show these results work in the files that already have beam charge available in RUN::Scaler or HEL::Scaler and show the one-to-one correspondence

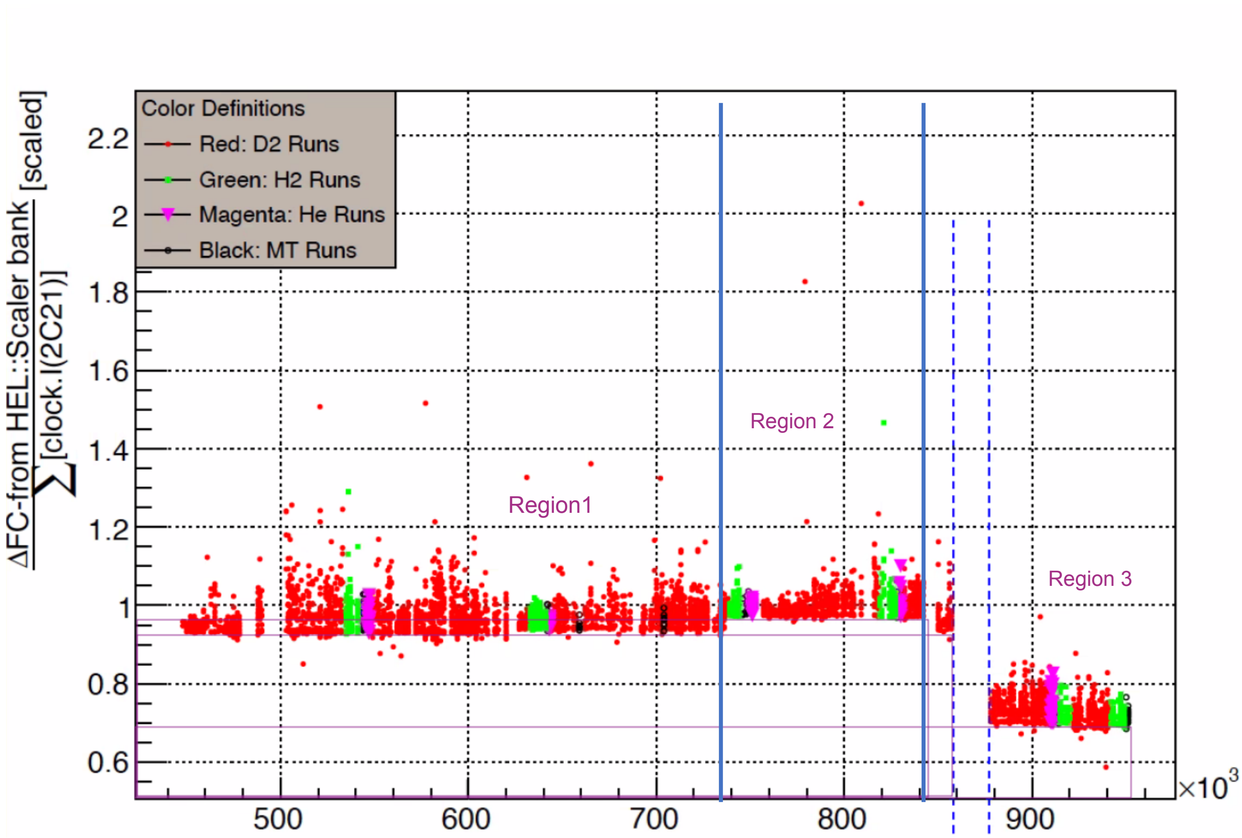

Fig : The ratio of beamcharge to clock.I[2C21] is plotted against Run-File numbers.

In X-axis the axis labelling should have added 12 in beginning.

For example: For axis label of 500, which is actually 500000

- It is Run- 12500 and extra three 000 refers file numbers that is in name of DST File.
- A DST File inside Run 12555 and filenumber 015 would have X- label:555015

Note: The Gap between 2 dotted Vertical lines refers to runs that has scalars information missing in RUN::Scalar bank and HEL::Scalar bank

Based on the graph above we divided our Run period into 3 different Regions:

- Region1: Runnumber < 12736 **or** (Runnumber> 12845 **and** Runnumber < 12877)
- Region2: Runnumber> 12736 **and** Runnumber < 12845
- Region3 : Runnumber > 12877

We see that a linear correspondence between beamcharge from HEL::Scaler bank and [clock. current[I2C21]]

Though the ratio is consistent, we observed two different natures of reading:

1. Normal reading in FC from HEL::Scaler bank and High [clock.current[I2C21]]
   For example:
   Run: 12448 File:000
   FCcup from HEL::Scaler 44240.2
   Clock. I_{2C21} 46685.6
   Ratio: 0.95
2. Low reading in FC from HEL::Scaler bank and Low [Current.Clock[I2C21]]
   [BY low reading we mean low in relative comparison to corresponding reading in RUN::Scaler bank]
   For example:
   Run: 12449 File:005
   FCcup from HEL::Scaler 346.543
   Clock. I_{2C21} 361.02
   Ratio:0.96

The following 2D histogram shows relationship between the clock.current[I2C21] and FC from HEL::Scaler bank for all 3 regions.

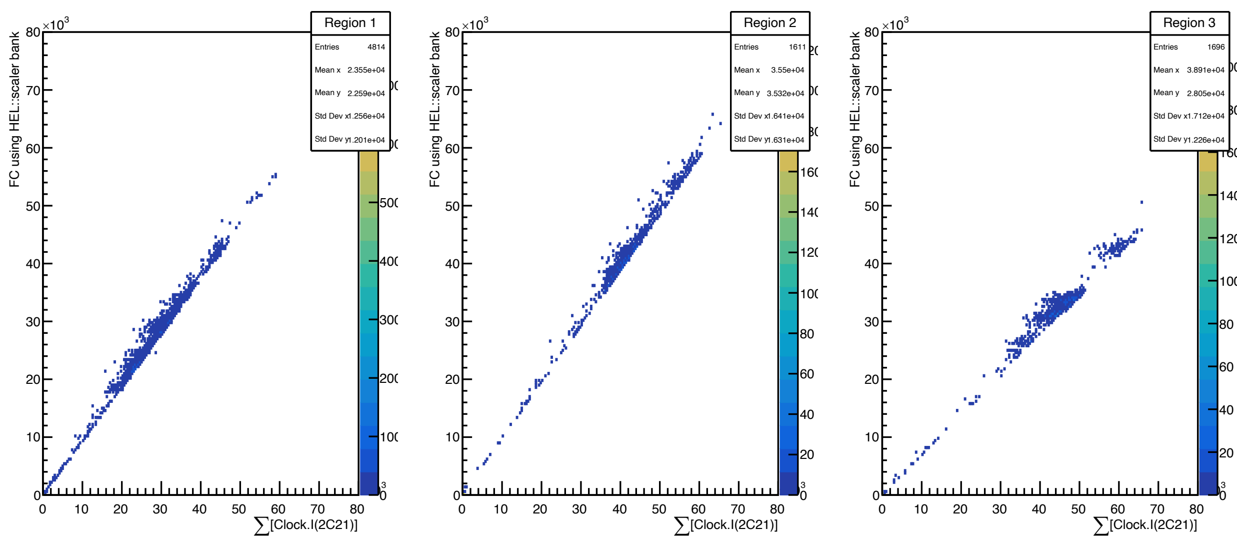

Fig: Histogram [From left to right goes region 1 to region 3] of FC beamcharge measured from HEL::Scaler bank in Y-axis vs clock.current[I2C21] in X-axis.

Region 1

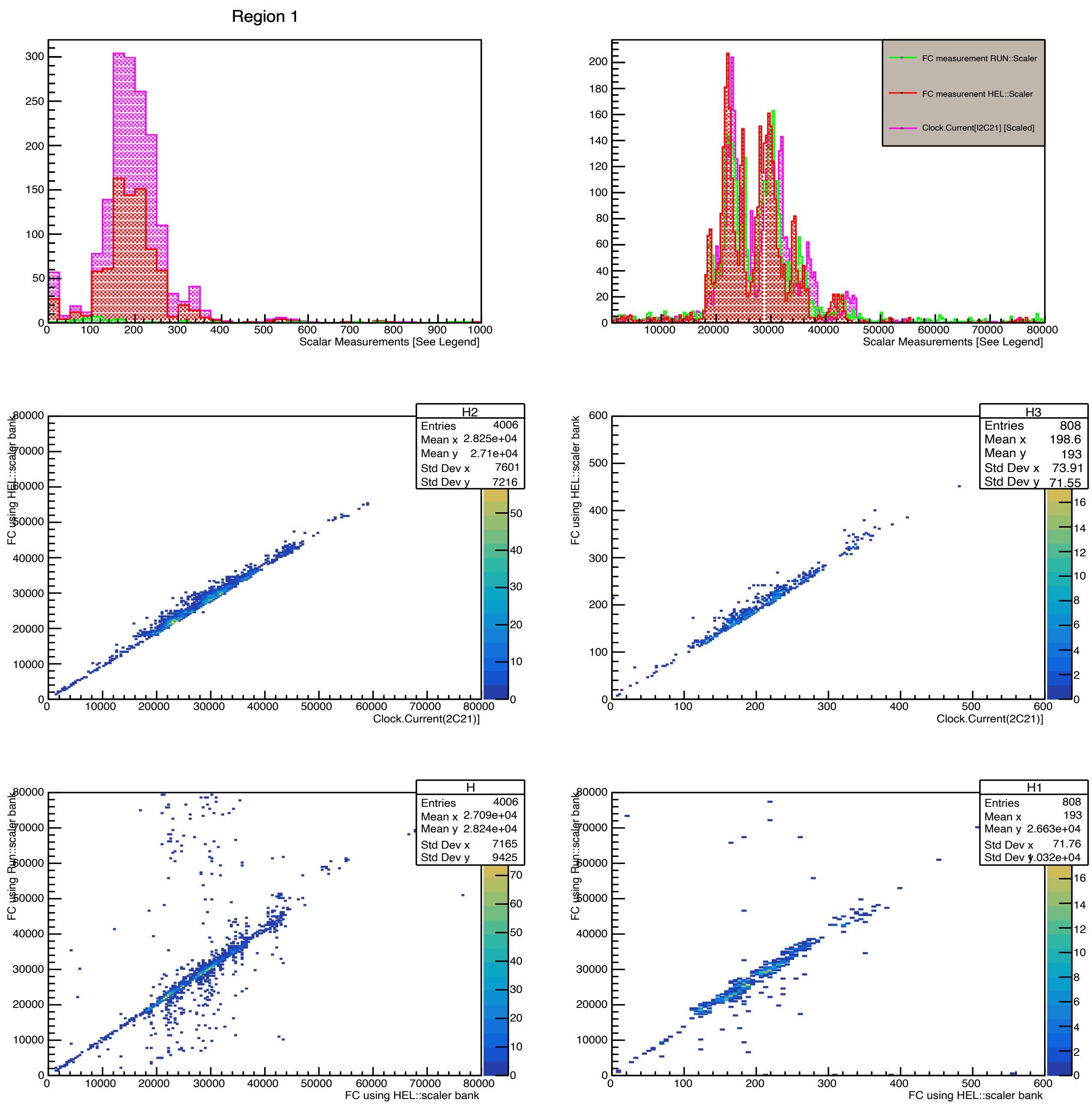

Fig:

Top Left and Right: 1D histogram of scalar measurements [beamcharge from RUN::scaler bank, HEL::scaler bank and clock.current[I2C21]]

Middle left and right: 2D histogram of the beamcharge from HEL::scaler bank in Y-axis vs clock.current[I2C21] in X-axis

Bottom Left and right: Histogram of FC measurements using RUN::Scaler bank in Y-axis vs of FC measurements using HEL::Scaler bank in X-axis,

Region 2

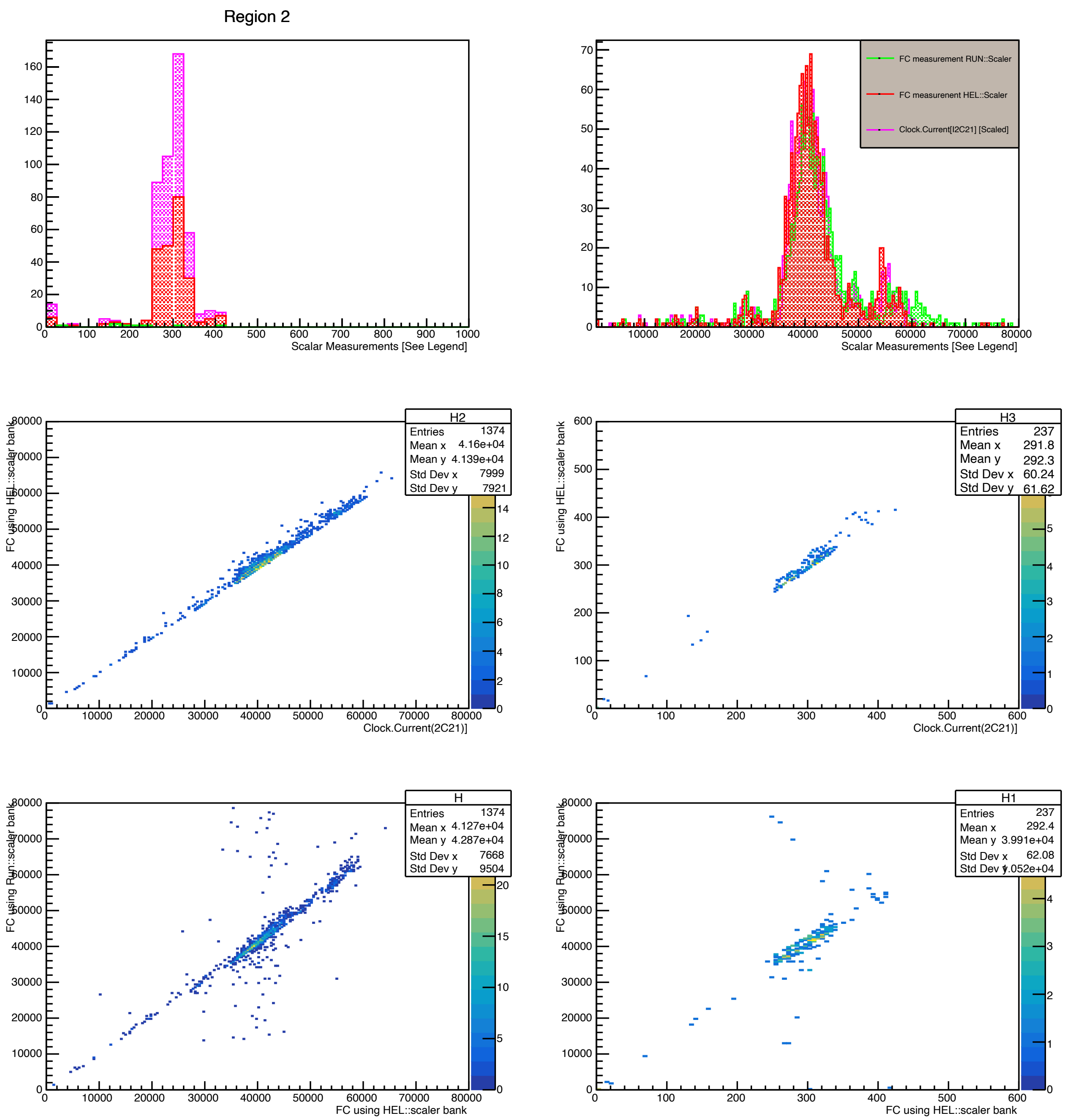

Fig:

Top Left and Right: 1D histogram of scalar measurements [beamcharge from RUN::scaler bank, HEL::scaler bank and clock.current[I2C21]]

Middle left and right: 2D histogram of beamcharge from the HEL::scaler bank in Y-axis vs clock.current[I2C21] in X-axis

Bottom Left and right: Histogram of FC measurements using RUN::Scaler bank in Y-axis vs of FC measurements using HEL::Scaler bank in X-axis,

Region 3

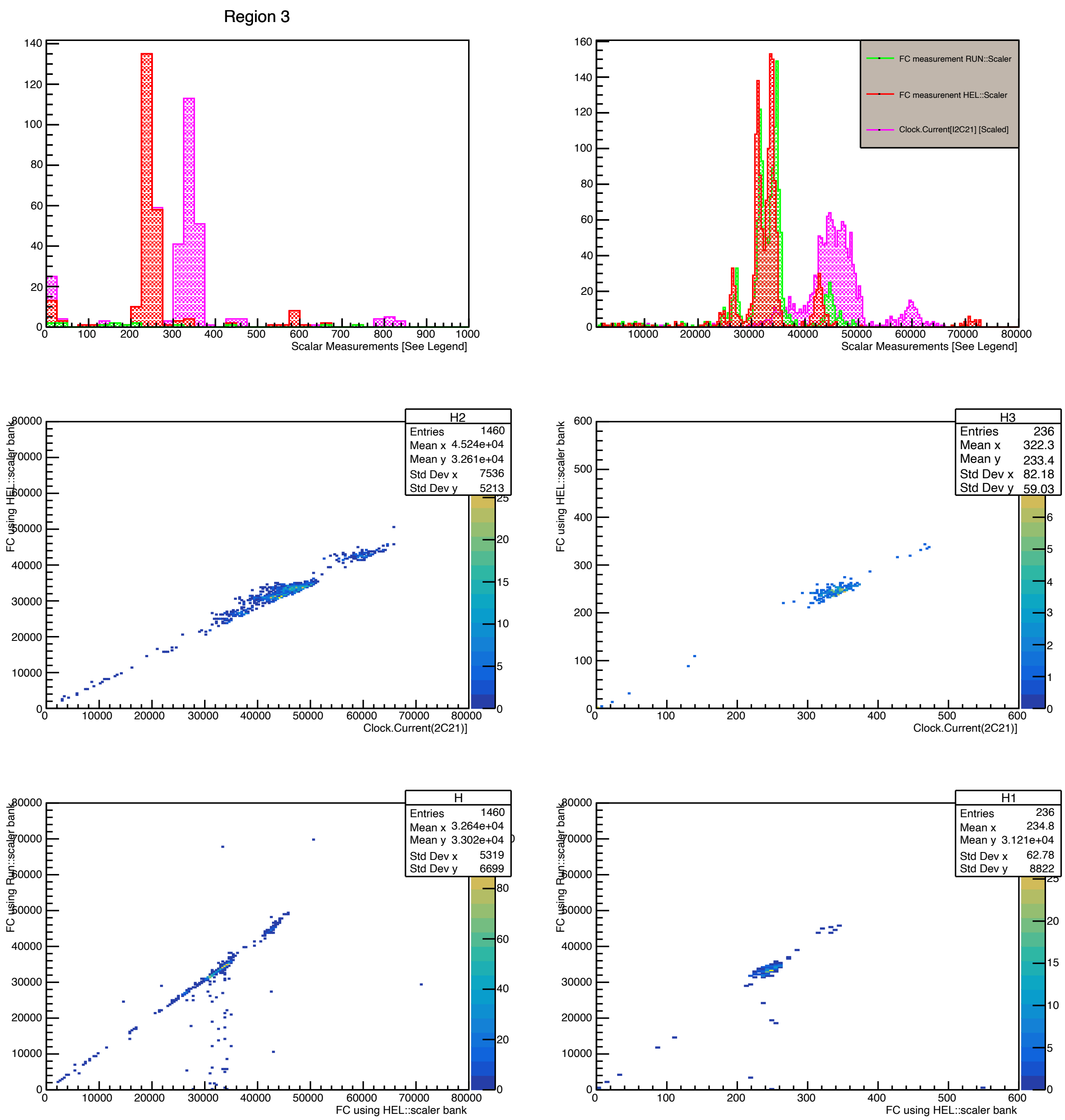

Fig:
Top Left and Right: 1D histogram of scalar measurements [beamcharge from RUN::scaler bank, HEL::scaler bank and clock.current[I2C21]]
Middle left and right: 2D histogram of beamcharge from HEL::scaler bank in Y-axis vs clock.current[I2C21] in X-axis
Bottom Left and right: Histogram of FC measurements using RUN::Scaler bank in Y-axis vs of FC measurements using HEL::Scaler bank in X-axis,

After studying all of the above plots, hence we decided that we will be using RUN::Scaler bank and clock. current to extract beam charge for the runs that have missing information. We followed the Beamcharge extraction procedure for all three regions separately, which is explained below.

Region 1:

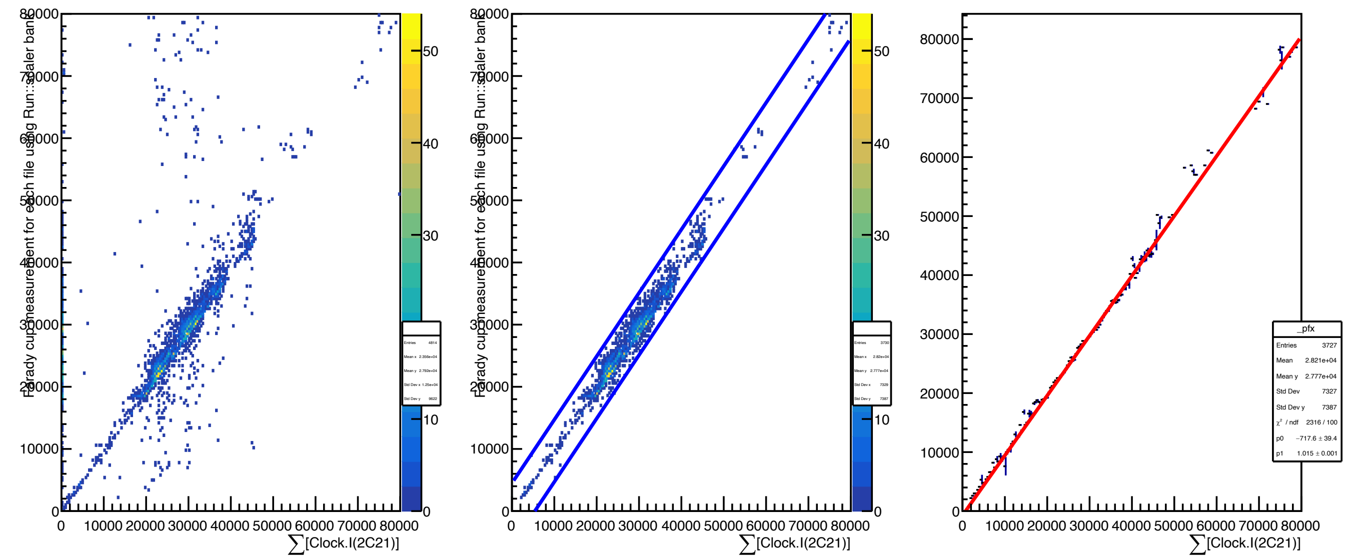

Fig: **Left**: 2D histogram of beamcharge from RUN::scaler bank in Y-axis vs clock.current[I2C21] in X-axis

Middle: Same 2D histogram as in left plot but only focusing in the region between 2 diagonal lines

Right: Profile histogram of middle plot is drawn and a 1D fit of profile is done.

We followed following procedures:

1. If ∑[clock.current] was less than 1000 we use reading in RUN::scaler bank as beamcharge
2. If the reading/data lies in the region as shown in middle plot above, we use reading in in RUN::scaler bank as beamcharge
3. If reading lies outside described by above points 1. and 2., we used fit parameters from the fit above right plot to extract beamcharge from If ∑[clock.current].

The equation of diagonals in middle picture above are:

Top line1: $(Y - 4500)/X = 1.02$

Bottom line2: $(Y + 5500)/X = 1.02$

Parameters of profile fit:[also shown in stat box]

p0 = -717.6. p1= 1.015

For region 2 and 3 also similar procedures were applied.

Region 2:

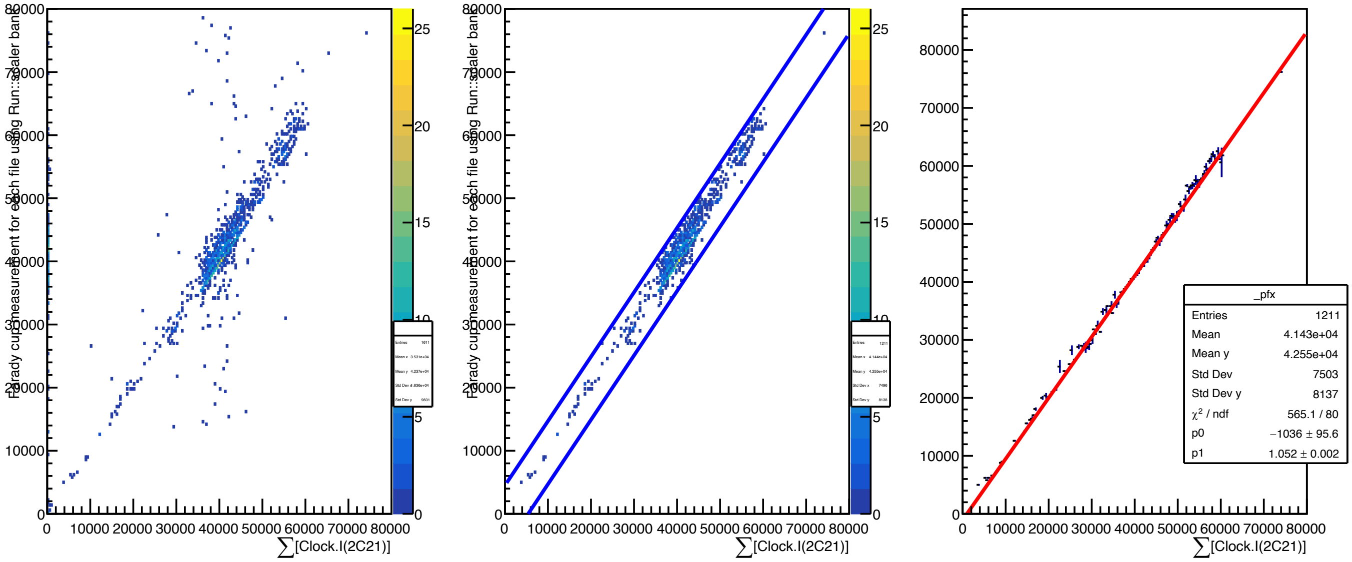

Fig: **Left:** 2D histogram of beamcharge from RUN::scaler bank in Y-axis vs clock.current[I2C21] in X-axis

Middle: Same 2D histogram as in left plot but only focusing in the region between 2 diagonal lines

Right: Profile histogram of middle plot is drawn and a 1D fit of profile is done.

The equation of diagonals in middle picture above are:

Top line1: $(Y - 4500)/X = 1.02$

Bottom line2: $(Y + 5500)/X = 1.02$

Parameters of profile fit:[also shown in stat box]

p0 = -1036. p1= 1.052

Region 3:

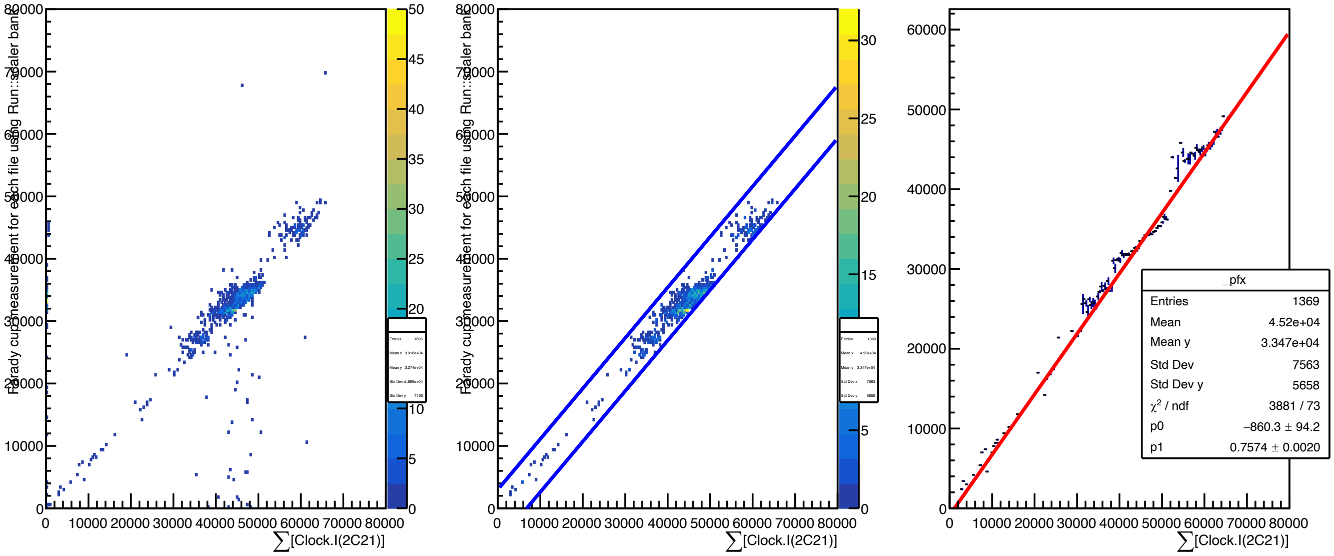

Fig**: Left:** 2D histogram of beamcharge from RUN::scaler bank in Y-axis vs clock.current[I2C21] in X-axis

Middle: Same 2D histogram as in left plot but only focusing in the region between 2 diagonal lines

Right: Profile histogram of middle plot is drawn and a 1D fit of profile is done.

The equation of diagonals in middle picture above are:

Top line1: $(Y - 3000)/X = 0.81$

Bottom line2: $(Y + 5500)/X = 0.81$

Parameters of profile fit:[also shown in stat box]

p0 = -860.3 p1= 0.756

Number of good electrons normalized with beamcharge

The aim of all the above procedures to extract beamcharge was to normalize the number of electrons with the beamcharge for each DST file in a run. After all, beamcharge extraction we plotted the quantity [number of electrons normalized by total beamcharge] for each DST file

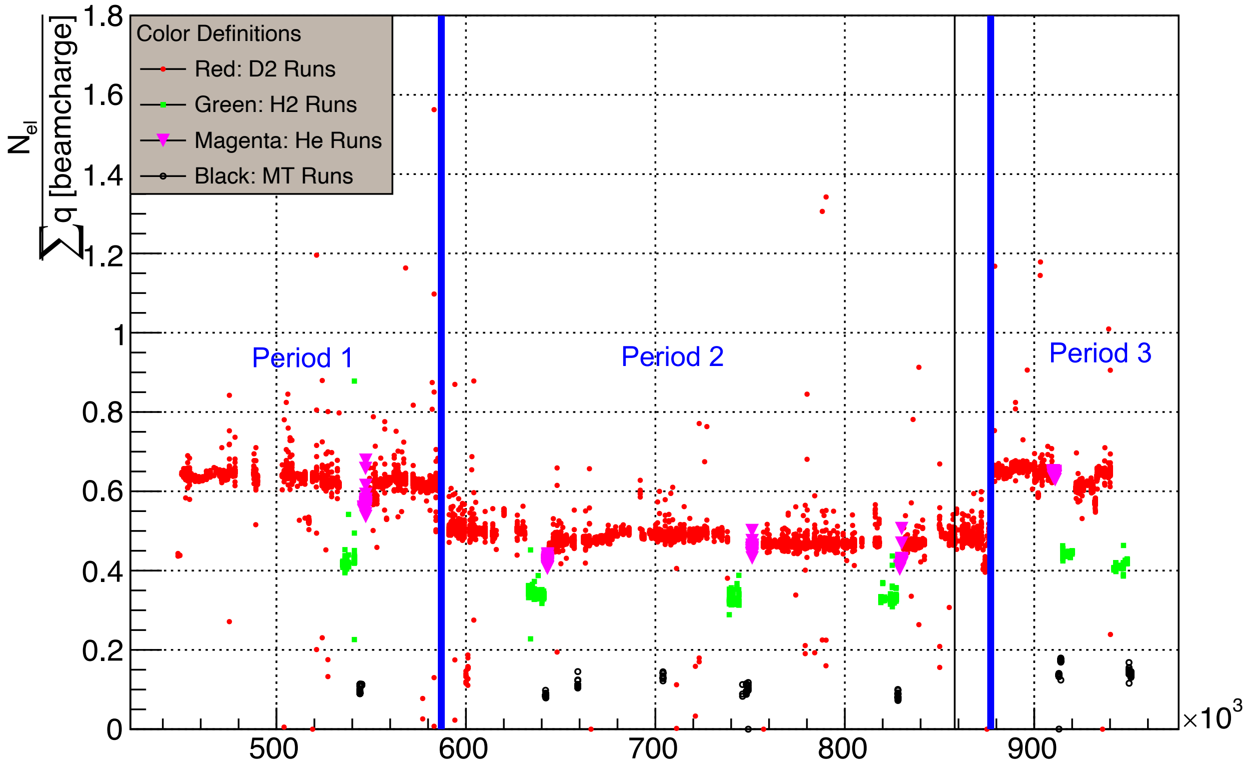

Fig: Number of electrons normalized by beamcharge is plotted for each DST file

We noticed three different periods in the plots above. Upon checking in the logbooks for any information that could have caused this, we found that at the beginning of period 2 or end of period 1 at Run 12579

https://logbooks.jlab.org/entry/3829393

- New RGF DAQ PROD66 and trigger *config v4_1.trg

Similarly, at end of period 2 or beginning of period 3 at Run 12878
https://logbooks.jlab.org/entry/3847392

- BCM recalibrated

For the three different periods that we observed in above plots we plotted 1D distribution of number of electrons normalized by beamcharge for each period.

Period 1

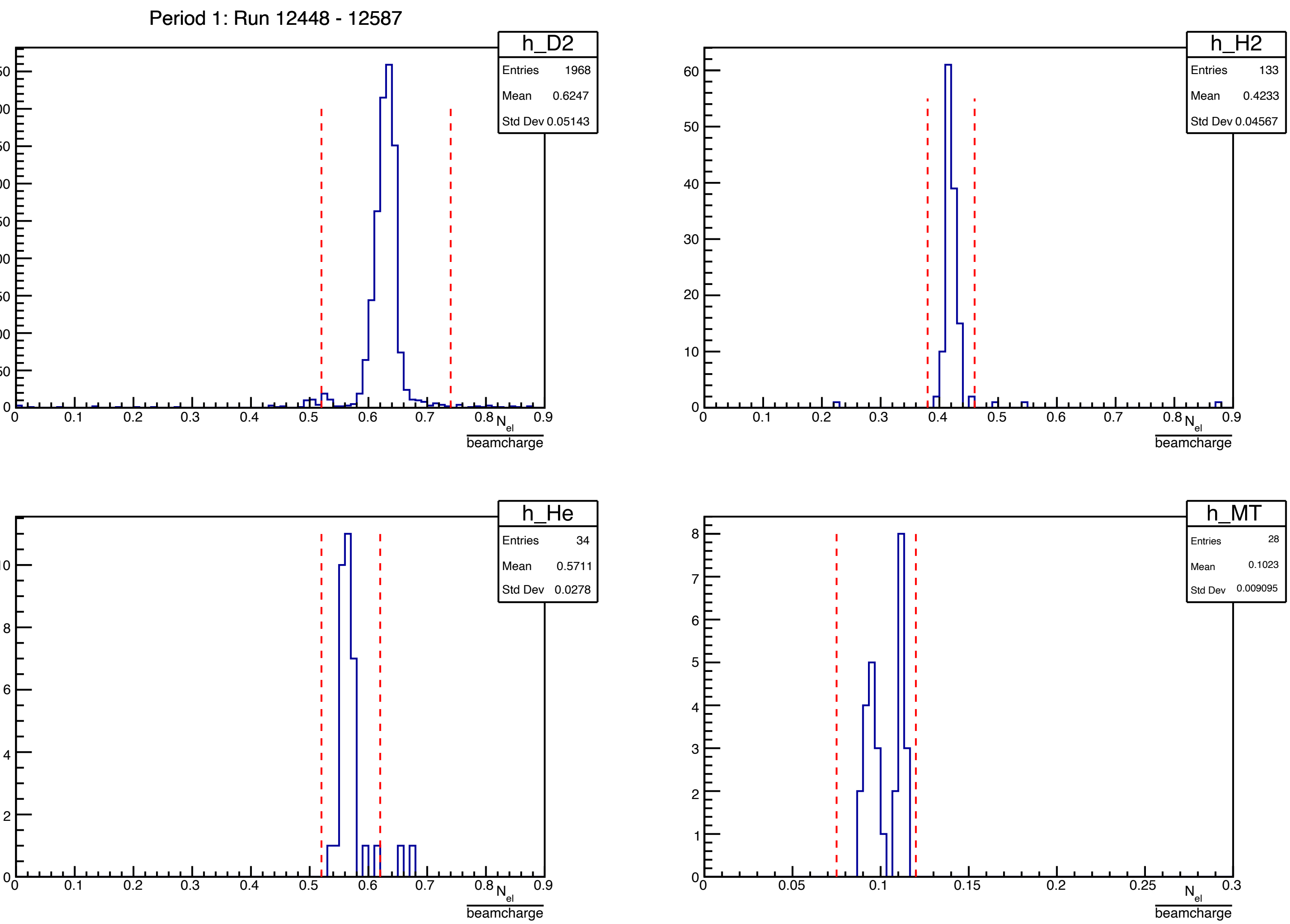

Fig: 1D distribution of number of electrons normalized by beamcharge for period-1 for each target type. The red vertical dotted line represents the cut that we applied for selection of good files.

Period 2

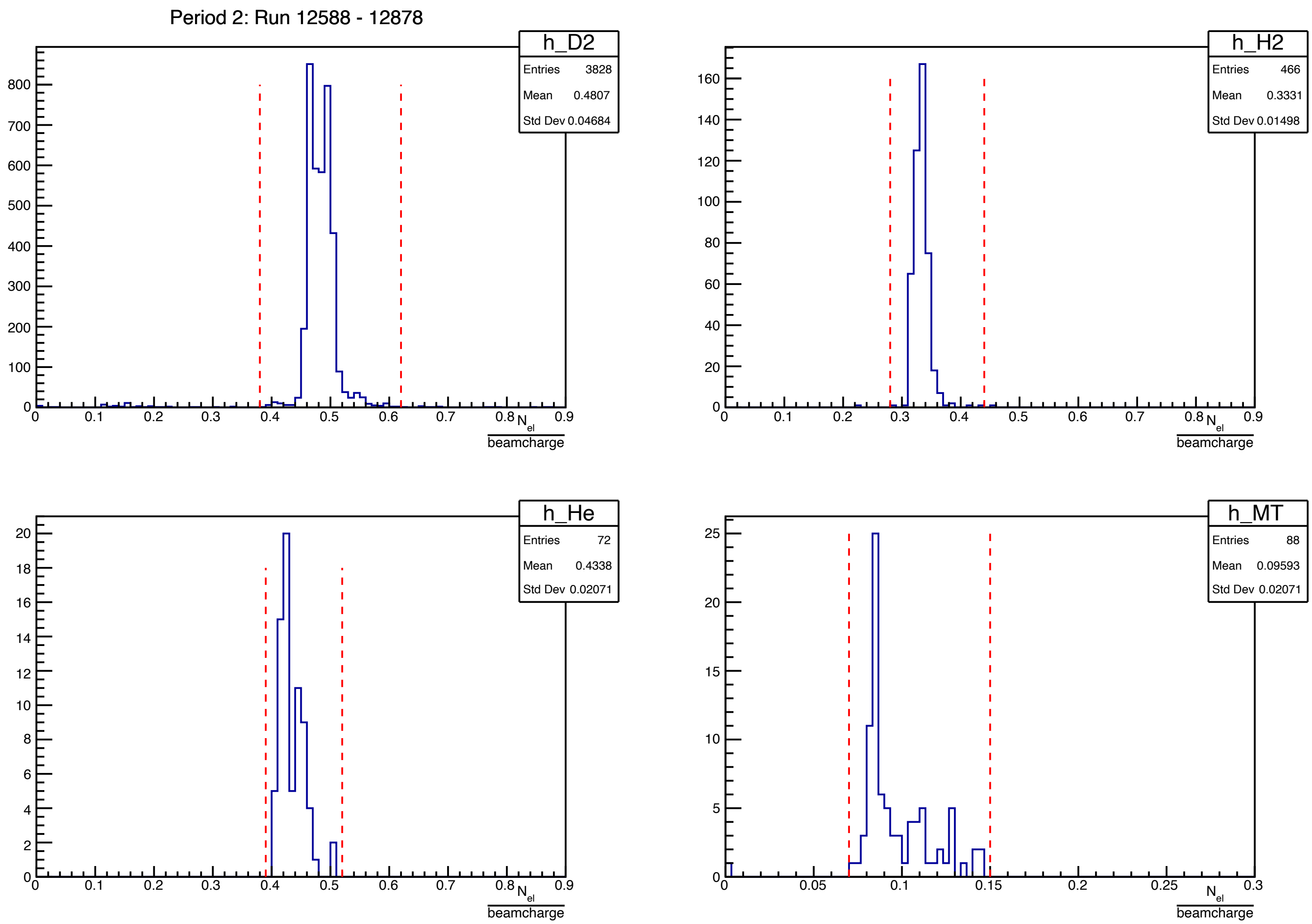

Fig: 1D distribution of number of electrons normalized by beamcharge for period-2 for each target type. The red vertical dotted line represents the cut that we applied for selection of good files.

Period 3

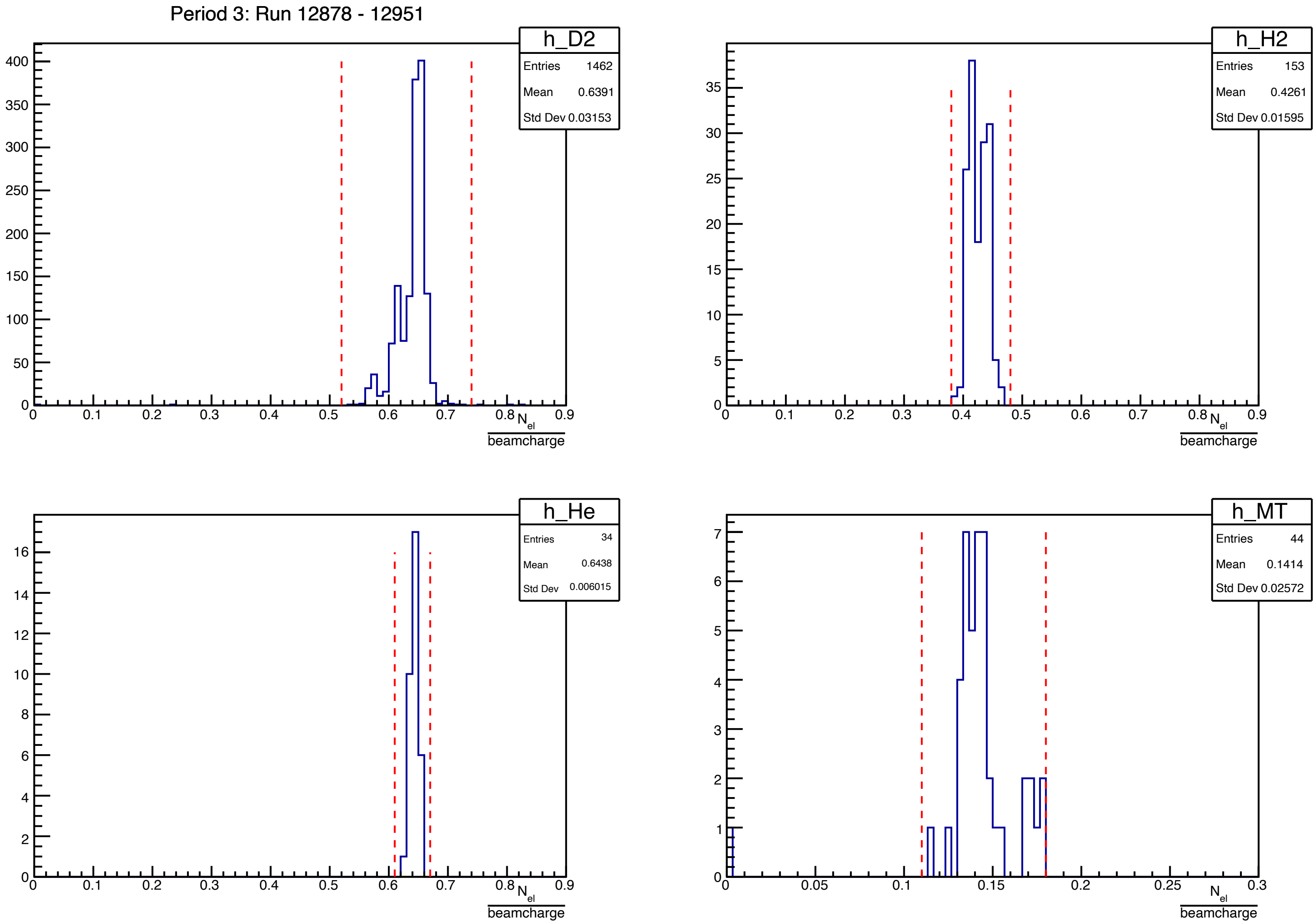

Fig: 1D distribution of number of electrons normalized by beamcharge for period-3 for each target type. The red vertical dotted line represents the cut that we applied for selection of good files.

Hence, after all these procedures we finally had beam charge information for each DST files in all runs and we made a final table of all runs with:

- Run Number
- File Number
- Beamcharge
- Number of good electrons
- Flag showing whether the file is good or bad based on the cut of number of electrons normalized with beamcharge.