

# Spill timing in rapTORR based on the DMTPC files

Alexander Deisting for Team Waveform



22<sup>nd</sup> of February, 2019

Vienna Conference on Instrumentation

## Waveform analysis in rapTORR

- ▶ Relevant applications: Beam\_testApp or Waveform\_testApp in the A\_D\_C\_2\_C\_S\_P branch
- ▶ Based on the `dataproc::waveform::CSPWaveforms` class
  - ▶ The class is designed to run on waveforms of a digitiser channel
  - ▶ (All) interesting features (baseline, pulse amplitude, when the highest amplitude was reached, ...) are calculated for all waveforms
- ▶ The results of the waveform analysis for three anode channels is stored in TTree, together with information on the digitiser configuration (stored in `rapobj::digitiserChannelData`)
- ▶ Further analysis (e.g data cleaning, amplitude spectra) is done using this TTree

## Spill time analysis in rapTORR – 1/2

- ▶ Relevant application: Beam\_testApp in the A\_D\_C\_2\_C\_S\_P branch
- ▶ For the spill time analysis the BeamTriggerTime and the time of a waveform trigger are needed.

### Waveform trigger time

- ▶ Defined as the time where the highest (or lowest) amplitude value occurred
- ▶ Is the sum of a time recorded by the digitiser(?) (WaveformTriggerT0) as a unix time stamp and the time value for the largest (smallest) time bin in the waveform

### BeamTriggerTime

- ▶ Is essentially the waveform trigger time for the *beam spill is coming signal* recorded during the beam time
- ▶ Since the *beam spill is coming signal* is a NIM pulse, only trigger BeamTriggerTime are accepted as such when the amplitude is lower than  $-800$  mV

## Spill time analysis in rapTORR – 2/2

The BeamTriggerTimes are currently stored in the *rapobj::TPCWaveforms* as `std::vector<TTimeStamp>`:

```
void SetBeamTriggerTimeVector(std::vector<TTimeStamp>
    tBeamTrigger) { fBeamTriggerTimes = tBeamTrigger;
    return; };
std::vector<TTimeStamp> GetBeamTriggerTimeVector() const
    { return fBeamTriggerTimes; };
```

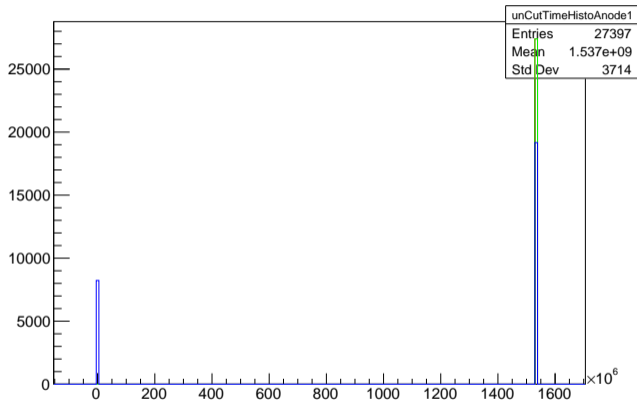
Right now this is done only in the `Beam_testApp` and not automatically by e.g the readout factory

In case everything goes well, the vector holds  $N$  times the same value (where  $N$  is the number of waveforms). Different cases already encountered:

- ▶ Two beam triggers arrive during the same exposure
- ▶ The beam trigger is not the first trigger (the `BeamTriggerTime` of all waveforms before is defaulted to zero)
- ▶ No beam trigger arrives (all waveforms' `BeamTriggerTime` is defaulted to zero)

# Spill structure analysis with the TPC

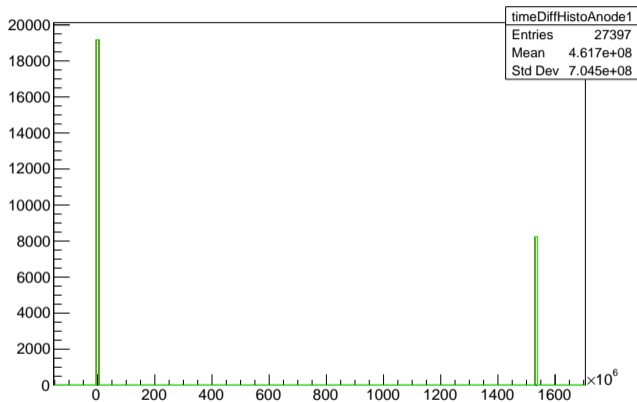
- ▶ Runs used: 1256073 to 1256085
- ▶ The time difference between the beam trigger time and the waveform trigger time is calculated using the information in the output TTree
- ▶ The remaining peak at zero seems to be the events, where the digitiser readout is triggered by the channel recording the *beam spill is coming* signal is coming



Beam trigger and waveform trigger times, no cuts

# Spill structure analysis with the TPC

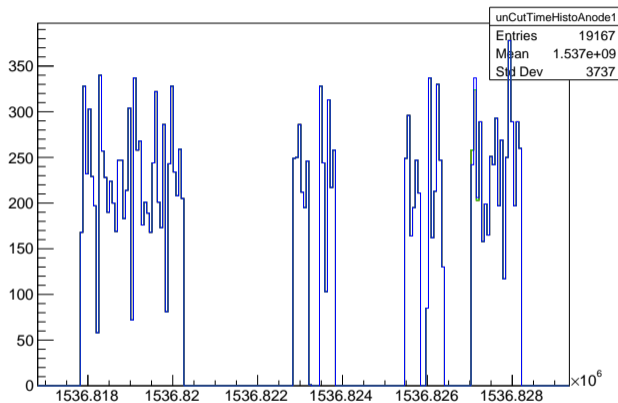
- ▶ Runs used: 1256073 to 1256085
- ▶ The time difference between the beam trigger time and the waveform trigger time is calculated using the information in the output TTree
- ▶ The remaining peak at zero seems to be the events, where the digitiser readout is triggered by the channel recording the *beam spill is coming* signal is coming



Time difference, no cuts

# Spill structure analysis with the TPC

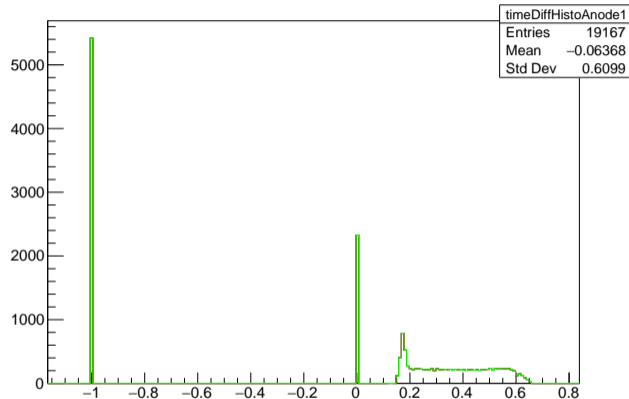
- ▶ Runs used: 1256073 to 1256085
- ▶ The time difference between the beam trigger time and the waveform trigger time is calculated using the information in the output TTree
- ▶ The remaining peak at zero seems to be the events, where the digitiser readout is triggered by the channel recording the *beam spill is coming* signal is coming



Beam trigger and waveform trigger times,  
BeamTriggerTimeSeconds != 0

# Spill structure analysis with the TPC

- ▶ Runs used: 1256073 to 1256085
- ▶ The time difference between the beam trigger time and the waveform trigger time is calculated using the information in the output TTree
- ▶ The remaining peak at zero seems to be the events, where the digitiser readout is triggered by the channel recording the *beam spill is coming* signal is coming

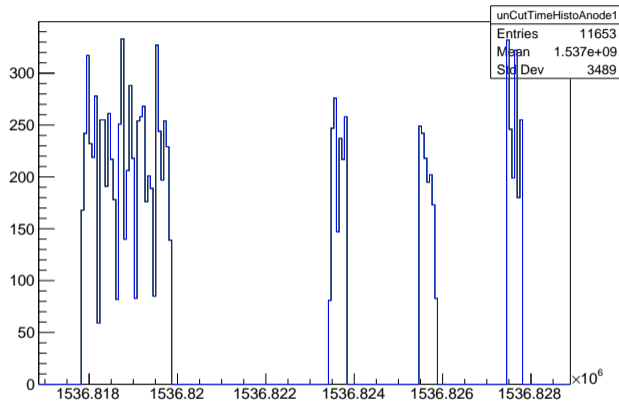


Time difference, no cuts, BeamTriggerTimeSeconds  
!= 0



# Spill structure analysis with the TPC

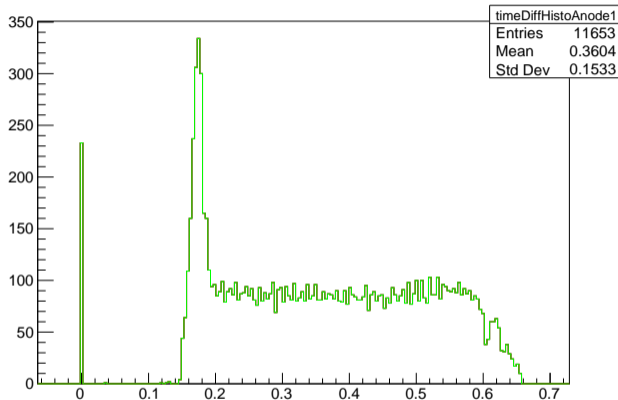
- ▶ Runs used: 1256073 to 1256085
- ▶ The time difference between the beam trigger time and the waveform trigger time is calculated using the information in the output TTree
- ▶ The remaining peak at zero seems to be the events, where the digitiser readout is triggered by the channel recording the *beam spill is coming* signal is coming



Beam trigger and waveform trigger times,  
`BeamTriggerTimeSeconds != 0 &&`  
`WaveformTriggerT0NanoSec != 0`

# Spill structure analysis with the TPC

- ▶ Runs used: 1256073 to 1256085
- ▶ The time difference between the beam trigger time and the waveform trigger time is calculated using the information in the output TTree
- ▶ The remaining peak at zero seems to be the events, where the digitiser readout is triggered by the channel recording the *beam spill is coming* signal is coming



Time difference, no cuts, `BeamTriggerTimeSeconds != 0 && WaveformTriggerTONanoSec != 0`

- ▶ Check that the falling edge of the *beam spill is coming signal* is used and not just the minimum time
- ▶ Include the calculation of the spill trigger times into the readout factory?

# Backup