Event Builder Status

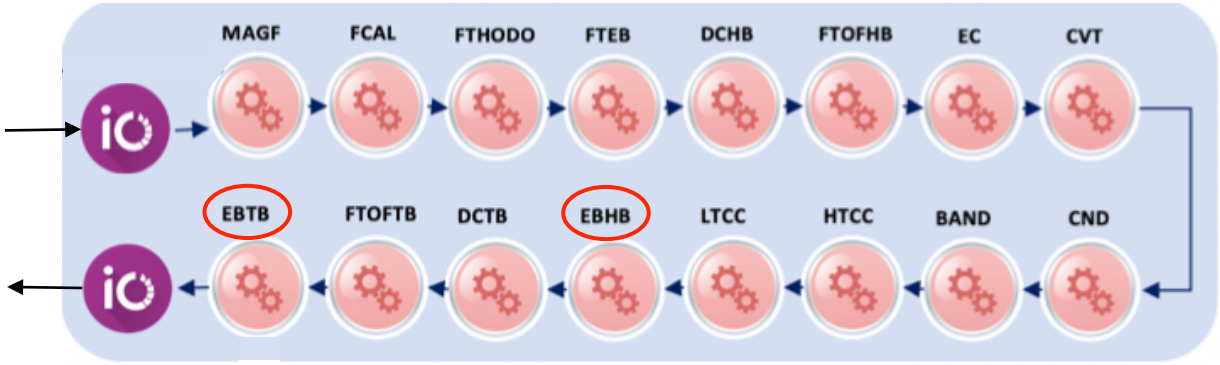
N. Baltzell

CLAS Collaboration Meeting

November 12, 2019

Overview

- The Event Builder is the last CLAS12 service run, after all detectors, and performs a few important tasks:
  - Geometrically associates detector responses into particles
  - Defines event start time(s) based on the best trigger particle candidate
  - Performs a basic, loose particle identification scheme
    - preserving info for easily tightening the criteria downstream
  - Retrieves/analyzes various event-based quantities
    - e.g. helicity state, scalers
  - Writes all info for physics analysis into “DST” HIPO banks
  - Documentation:
    - https://clasweb.jlab.org/wiki/index.php/CLAS12_EventBuilder



Creating Particles

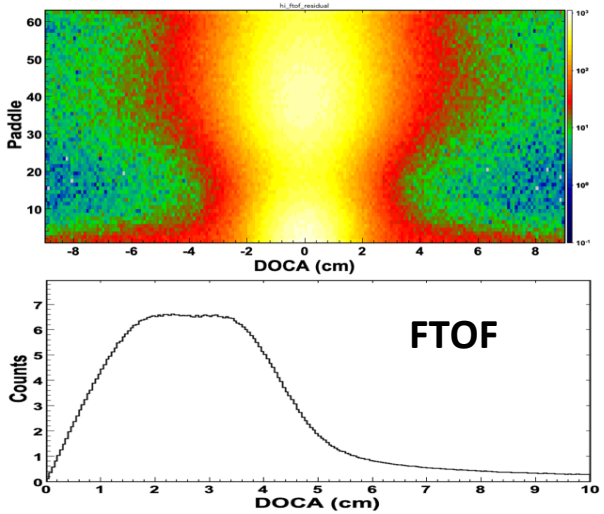
Charged Particles

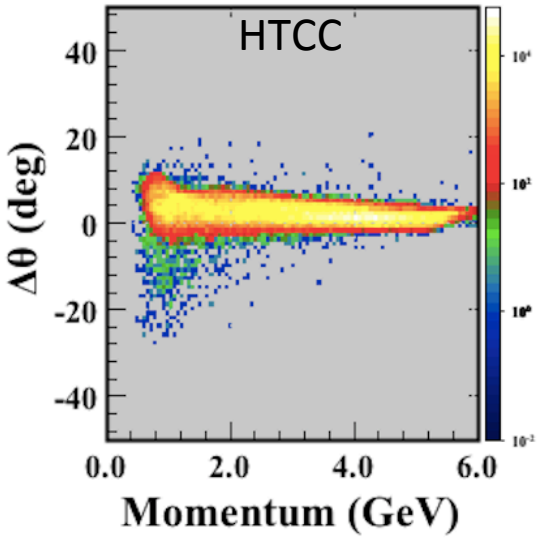
- seeded with tracks

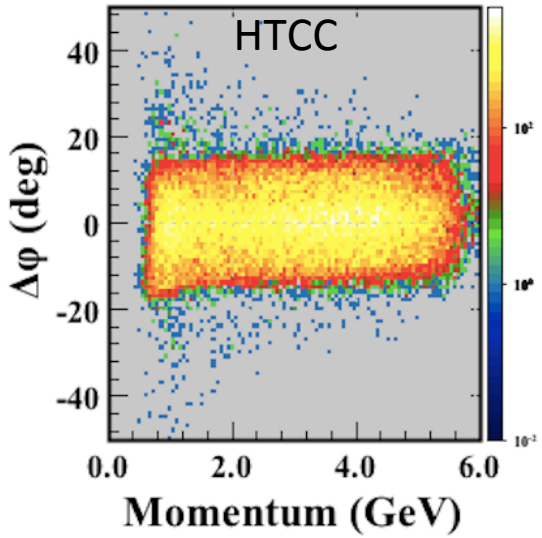
Neutral Particles

- seeded with trackless ECAL/CND

Associated detector responses are assigned based on geometric DOCA







Currently loose, flat detector-dependent cuts, with one hit per detector layer per track chosen, based on DOCA. Many-to-one, track-to-TOF relationships accommodate reconstruction of time-based ghost tracks during tracking development (and real TOF hit-sharing).

TODO: Add timing information in matching. Some analyses suggest that the coarse HTCC position resolution warrants special consideration, i.e. HTCC cluster-sharing between tracks, e.g. e^+e^- with small opening angles.

Event Start Time

Choose "Trigger Particle"

- FD track with FTOF hit
  - Highest energy e^- else e^+, if one exists
  - Else highest momentum track, assumed a π
- FT electron
  - In parallel, using "shadow" banks RECFT::Particle and RECFT::Event
  - FT electron with the closest timing match to an FD track (and all p/K/π mass hypotheses)

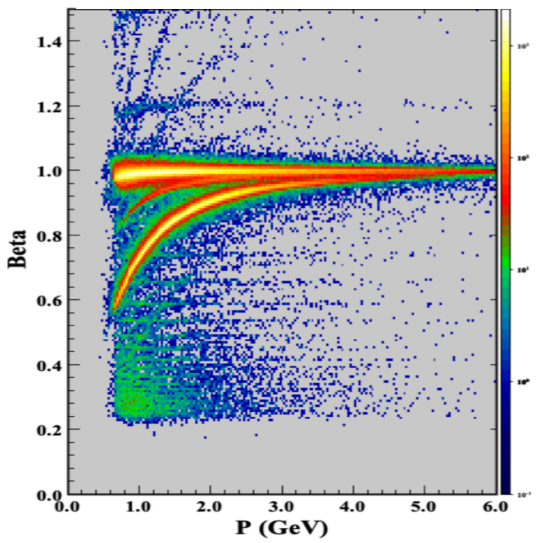
Construct its vertex time

- Based on path length, mass, momentum

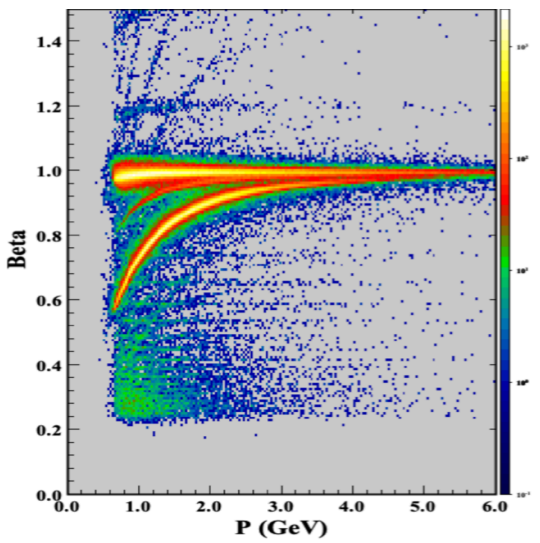
Use nearest RF bunch to assign start time

- *NEW: With a correction for z-vertex*
  - *this is done on a per-track basis, to compensate for systematic errors on z-vertex*

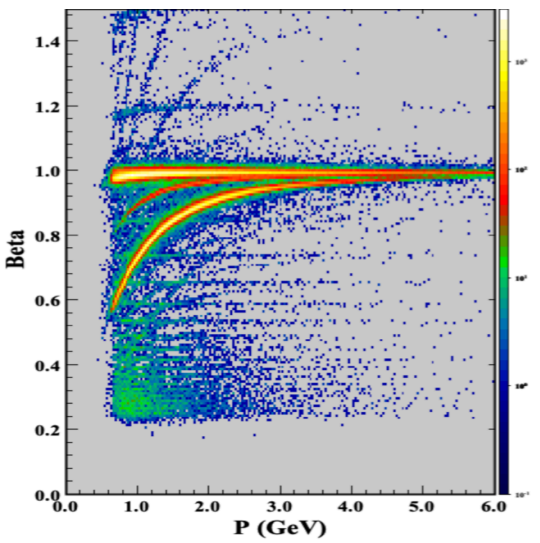
No Z-vertex correction



With trigger-particle vertex correction



With per-track vertex correction



Lepton and Neutral Identification

e^-e^+ in Forward Detectors

- Charged Track ECAL, HTCC, and FTOF hits
  - PCAL > 60 MeV and HTCC #phe > 2
- ECAL sampling fraction
  - ±5σ sampling fraction parameterized in momentum, with REC::Particle.chi2pid assigned as N_σ from nominal

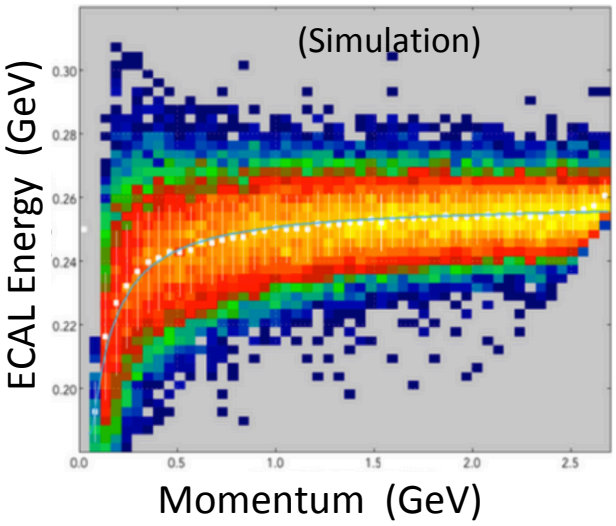
Neutrals

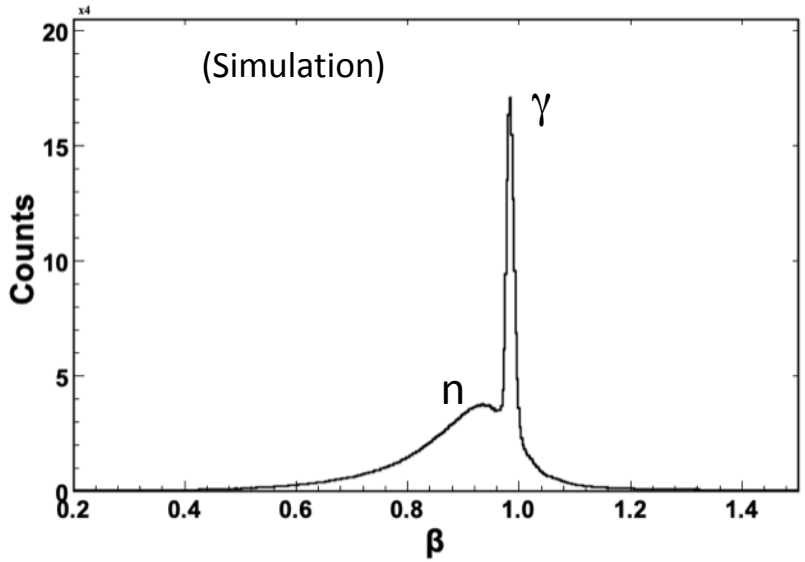
- Forward Detector
  - n/γ separated based on β
    - Photon energy calculated from ECAL sampling fraction
- Central Detector
  - neutron assigned for CND β<0.9

Forward Tagger

- e^-/γ separated based on hodoscope neutral veto
- Energies based on calorimeter

(neutron momenta assigned based on β)





All ECAL sampling fractions used above are in CCDB, with sector-dependence available, but needs to be updated by calibrators/rungroups based on final calibration.

Charged Hadron Identification

If a track fails e^-/e^+ identification, assume it's a charged hadron and assign its identity based on minimizing the time difference between event start time t_0 and vertex time.

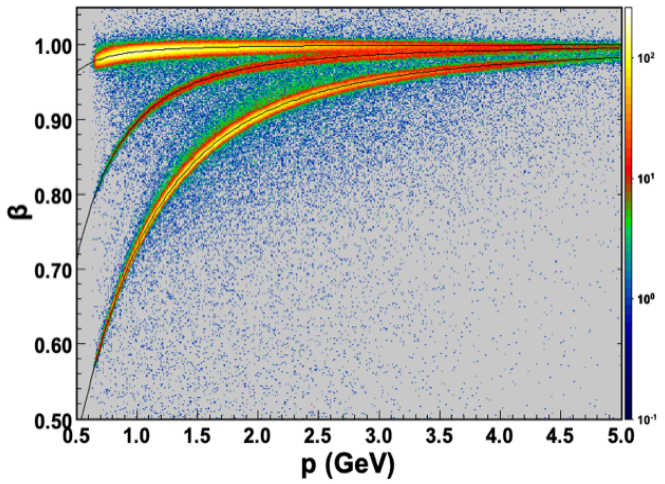
$$\Delta t_i = t_0 - \left[t_{FTOF} - \frac{L}{\beta_i(p)} \right], \quad i = \pi/K/p/d/...$$

- ~~If no TOF info, pid=0~~
- Vetoes from Cerenkov
  - #p.e. greater than 2 and below kaon threshold → reassign to π
- `REC::Particle.chi2pid`
  - a signed-N_σ (or a signed-χ) from nominal timing, based on σ per FTOF-paddle, so tightening the requirements can be done with a simple cut on chi2pid

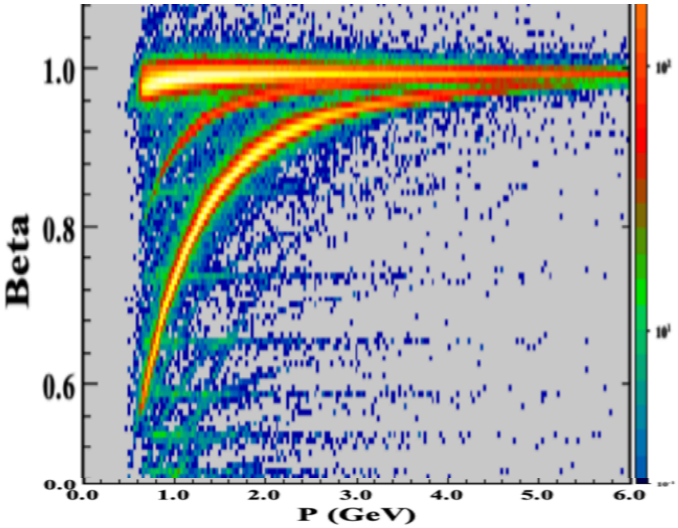
- *NEW: if no FTOF/CTOF available, fall back to ECAL/CND/etc.*
  - *check* `REC::Particle.status` *to easily see*

- Can be extended to more sophisticated scheme
- Will need updating from RICH

Simulation



Data



DSTs

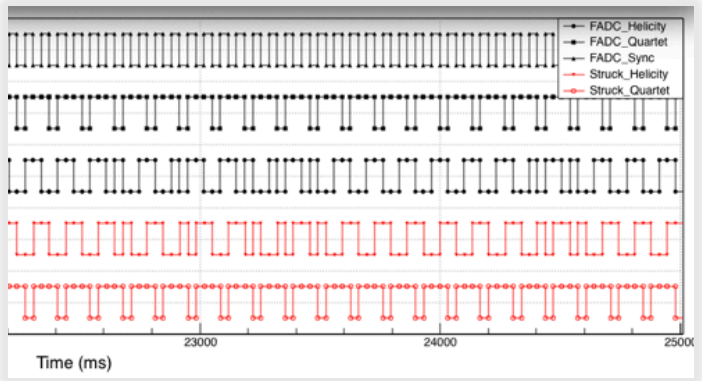
High level HIPO banks for physics analyses

- Most DST bank names are prefixed by “REC”, in `event.json`
  - https://github.com/JeffersonLab/clas12-offline-software/blob/master/etc/bankdefs/hipo4/event.json
  - `REC::Event`
    - run/event #, event time, trigger bits, helicity, etc
  - `REC::Particle`
    - pid, charge, momentum, etc
  - `REC::“ResponseType”`
    - e.g. `Calorimeter, Scintillator, Cherenkov, Track`
    - hit/cluster energies, positions, times, shapes
    - contains `pindex` link to its particle
    - *Note, this only contains responses associated with particles*
- Also keep RUN::*, RAW::scaler banks, RAW::epics, and some helicity banks
- *NEW: addition of `REC::Particle.vt`, per-particle start time (RF- and vz-corrected)*

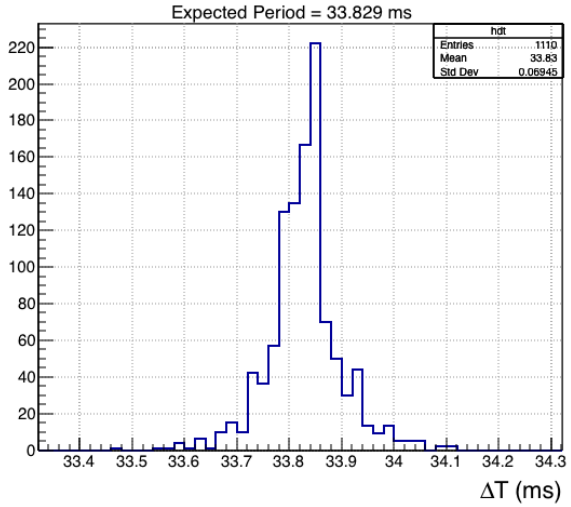
See documentation and release notes:
https://clasweb.jlab.org/wiki/index.php/CLAS12_DSTs
https://github.com/JeffersonLab/clas12-offline-software/releases

Delayed Helicity (1)

- Accelerator delivers 4 helicity signals:
  - helicity, pair sync, pattern sync, settle
  - can be used for integrity checking, some needed for using pseudo-random generator



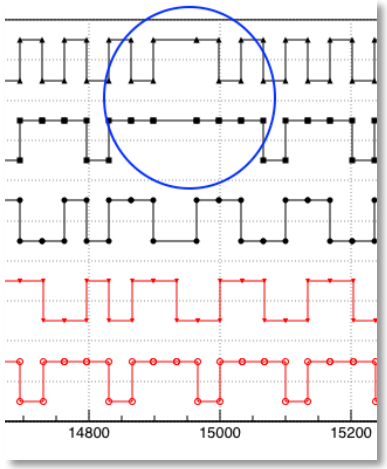
- We read and record them all, in two types of hardware, for redundancy and cross-checking
  - FADC: readout on every CODA trigger
  - Scaler: counts latching on helicity states
  - They always agree with each other when easily comparable, but each has their own problems. Scaler turns out not very useful for event-by-event helicity, but still good for beam-charge-asymmetry. FADC has some missed channel readouts, only in low-state, correctible.

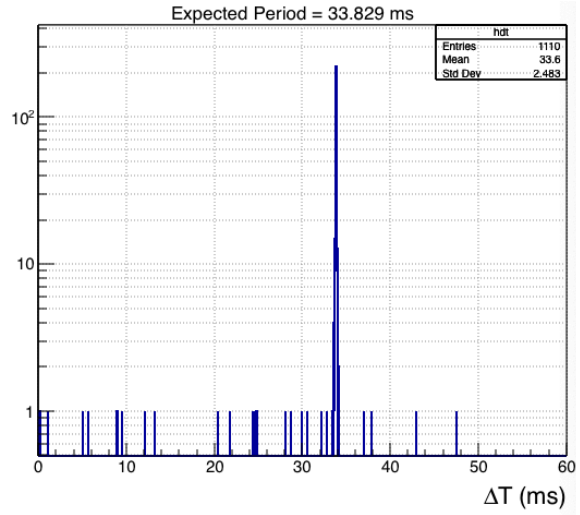


- Reporting Styles
  - Direct Reporting (Spring 2018 only)
    - Easy, just read live helicity state from FADC, no significant need for the other signals, corrections, integrity checks
  - Delayed Reporting
    - Fall 2018
      - *No online correction available! Need an offline correction …*
    - Spring 2019 and later
      - Online correction available in every event in `HEL::online`

Delayed Helicity (2)

- For an offline correction, we start by tracking state changes during serial decoding, and populating tag=1 HIPO HEL`::flip` bank with each state change.
- Analysis written in coatjava to read flip banks, on-the-fly during analysis and provide event-based helicity
  - must address missed readouts due to DAQ deadtime, and FADC partial readouts
  - multiple techniques implemented and compared, e.g.
  - simple walk forward
    - requires only 8-states, with integrity check along the way
  - pseudo-random generator option
    - requires 30-state initialization, and clock synchronization depending on usage. This is similar to the online correction available as of 2019.
    - allows crossing deadtime gaps, and going all the way to the end of file
  - can consider an “uber” correction to combine the best of all, meanwhile it seems unecessary and the simple walking is sufficient and robust
- Half-Wave Plate
  - Extracted from EPICS archive and populated in CCDB and used in analysis to correct for it
  - In all cases in non-raw HIPO banks, we store both HWP-corrected and raw helicity for validation, with +1/-1/0, for positive/negative/UDF
- Validation
  - Performed across ~100 of files across few runs during development
  - And with help from RG-B and comparison with their online version across many runs
  - Checks out very good, agrees with online correction when available, allows recovery of some cases where online correciton unavabilable, couple issues optimized since 6.3.1
- TODO:
  - standardize post-processing workflow for run-groups, or push towards analysis working with shared software libraries (avoiding post-processing)





Documentation

Linked from the main software documentation wiki:
https://clasweb.jlab.org/wiki/index.php/CLAS12_Software_Center

https://clasweb.jlab.org/wiki/index.php/CLAS12_EventBuilder

Contents [hide]

1 Overview
2 Output Banks
3 Beam Helicity
4 Beam Radio Frequency
5 Particle Creation
  5.1 Charged Particles
  5.2 Neutral Particles
  5.3 Forward Tagger
6 Event Start Time
7 Particle Identification
  7.1 Electron/Positron
  7.2 Charged Hadrons
    7.2.1 Cherenkov Vetoes
  7.3 RICH
  7.4 Neutrals
8 CLAS Calibration Database
  8.1 ECAL Sampling Fraction Parameterization

https://clasweb.jlab.org/wiki/index.php/CLAS12_DSTs

Contents [hide]

1 Bank List
2 Inter-Bank Linking
3 Detector Identifiers
4 Bank Details
  4.1 RUN::config
  4.2 REC::Event
  4.3 REC::Particle
  4.4 REC::"Response"
    4.4.1 REC::Traj
    4.4.2 Reverse Indexing
5 Non-DST Detector Banks

Please check it out when you have questions/concerns about DST format and Event Builder, and give feedback!

Summary

Updates since June Meeting

- Z-vertex correction for event start time, per-particle, used for all timing-based PID
- Charged hadron β calculated (and PID assigned) from non-TOF detectors if TOF unavailable
- Improved delayed helicity correction
  - based on analyzing tag=1 helicity events, multiple methods implemented and compared, required changing user API
  - associated methods to provide timestamp based lookup of tag=1 scaler events
- Doing FT-based start time (and writing RECFT::Particle) even if electron in FD

TODO

- Near Term
  - Switch to trajectory banks for path lengths
    - Did the groundwork for EB, ready on a branch
    - Requires all other reconstruction services and calibration suites to follow suit
  - Incorporate BAND, pending final specs
  - Implement upgraded CD neutral identification by CND group
    - requires EB/PID switch all to CTOF clusters
- Further Future
  - add timing to geometric matching
  - incorporate RICH
  - alternative/better identification schemes