E144 CCD Tracking Primer

Eric Prebys, Princeton University

Abstract

This document describes the E144 CCD tracking system, reconstruction, and DST format to the extent necessary for other E144 users to incorporate reconstructed CCD data into their analyses. A very brief overview of the CCD spectrometer and track reconstruction algorithm will be given, followed by a detailed description of the CCD DST Ntuple. Finally, some examples will be discussed.

1 The CCD Spectrometer

1.1 General

A conceptual top view of the end of the gamma line is shown in figure 1. A movable fork positioned converters of various thicknesses in to the E144 gamma line, about 10 m downstream of IP2, to convert some fraction of the scattered photons into e^+e^- pairs. Depending on the position of the fork, conversion rates could be tuned from a few 10^4 down to a few 10^3.

About 1 m downstream of the converter was the 5D36 analysis magnet. Because of the high energy of the photons (several GeV), the pairs were more-or-less collinear until reaching the magnet, at which point the positron would be bent one way and the electron the other. The magnet strength was adjustable up to a kick of about 250 MeV/c.

The planes of the spectrometer were arranged in two arms, one for electrons and one for positrons. Each arm had an "upstream" CCD plane about 1.5 m from the magnet center followed by a triplet of "downstream" CCD's about 1 m downstream. The three downstream CCD's were about 10 cm apart. The active area of each CCD was about 1.5 cm in the vertical direction by 2.6 cm in the horizontal direction. Readout was in the form of pixels, each 22.5 μm on a side. The inner edge of the upstream CCD's was about 8 mm from the beam center and that of the downstream triplet was about 12 mm from the beam center.

1.2 Coordinates

The coordinate system of the CCD system has the z axis running downstream along the nominal gamma line, with $z = 0$ being the front of the upstream vacuum box. The y axis is vertical and the x axis is horizontal, with positive x toward the South wall (down in the figure). Thus the even numbered CCD's are on the positive x

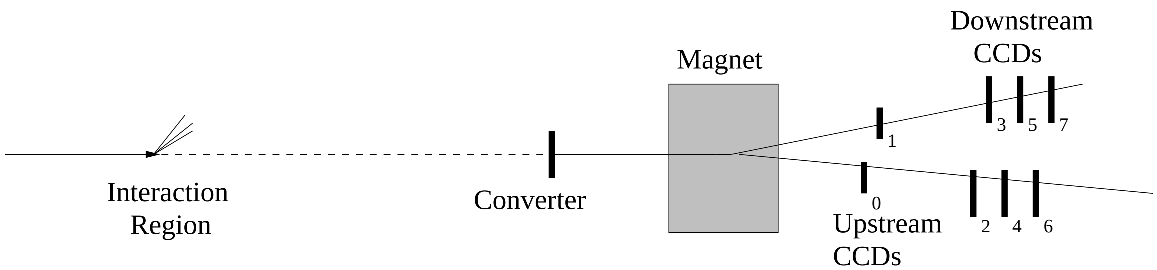

Figure 1: Conceptual view of E144 gamma line and CCD Spectrometer.

side and the odd CCD's are on the negative x side. Glenn has correctly pointed out that this is a left-handed coordinate system. Oops! I could change it's important to anyone.

Because of uncertainty in the direction of the magnetic field, tracks bent toward the positive x side have been arbitrarily designated "positrons" and those on the other side "electrons".

1.3 Running Modes

There were essentially two running modes of the CCD spectrometer.

- *N=2 Mode* - In this mode, the magnetic kick was set very high $\geq$ 200MeV/c. This would sweep out all but the highest momentum tracks, indicating high momentum photons. The kinematic edge of normal compton scattering corresponded to a point within the active region of the CCD's so this mode can be used to search for direct evidence of non-linear scattering. For this mode, fairly thick converters were used to generate lots of tracks per event (up to several hundred). This mode will be the most useful for the present offline analysis.

- *Pair Mode* - In this mode, the magnetic field was set to about 100 MeV/c to maximize acceptance for both the electron and positron from a pair, allowing offline reconstruction of actual photons. For this mode, the thinnest converters were used in an attempt to average 2 tracks per event. These data are probably not ready for use by the collaboration as a whole.

A table has been made of the running mode of the CCD spectrometer for each run. It will be discussed in the last section of this document.

2 Track Reconstruction

The raw CCD data is in the form of a list of "hits" from each plane of the spectrometer. Each hit is defined by a location (in CCD internal coordinates) and a pulse height in ADC counts. The first step of reconstruction is to read in the hits, optionally apply additional quality cuts (pulse-height cuts, noisy pixel elimination, etc). Then, because a single track may leave signal in adjacent pixels, an algorithm builds "clusters" out of contiguous groups of pixels. The position of the cluster is taken to be that of the pixel with the highest pulse height. Each CCD produces a list of clusters, most of which contain only on pixel.

Next, the position of each cluster is converted to a presice three-dimensional point, using a database which contains the position and orientation of each CCD[1]. Now the real work of track reconstruction can begin.

Because of the somewhat low signal to noise ratio of CCD's, there are a lot of noise hits in the spectrometer, which can produce false tracks. Luckily, the excellent spatial resolution of the CCD's can be used to reject most false combinations, if there are at least three hits on a track. Because of the difference in spacing of the upstream CCD relative to the other three, it is handled specially, ie this is not treated as a simple four-plane spectrometer.

The first part of the tracking procedure is to try to find tracks in the rear three planes of each arm. First, an imaginary line is drawn from each hit in the middle plane (CCD 4 or 5) to the nominal center of the magnet. This line is projected to the front rear plane (CCD 2 or 3) and the back rear plane (CCD 6 or 7). These planes are then searched for hits that fall within a certain distance of these projections in both x and y. If hits are found, and attempt is made to fit all possible three-hit combinations. Track candidates are kept if the fit has a χ^2 per degree of freedom below a certain cut. In the event of ambiguity, the track with the best χ^2 is kept.

In the next step, each three-hit track from the rear CCD planes is projected to the front CCD plane, and an attempt is made to find a matching hit. If a hit is found, a four-hit fit is done and the track is kept if it passes a χ^2 cut. Again, in the case of ambiguity, the track with the best χ^2 is kept. If no satisfactory hit is found in the front CCD, the rear tracks are still kept as three-hit tracks.

At this point all the hits which have been used in tracks are flagged as unavailable for further tracking. Then all remaining two-hit pairs in the rear CCD planes are projected to the front CCD, and an attempt is made to find a matching hit in the front CCD. As before, if a matching hit is found, a fit is done and tracks are kept on the basis of χ^2 cuts.

It's clear that this leaves is with three types of tracks, as is shown in figure 2:

[1] The production of this database goes beyond the scope of this document.

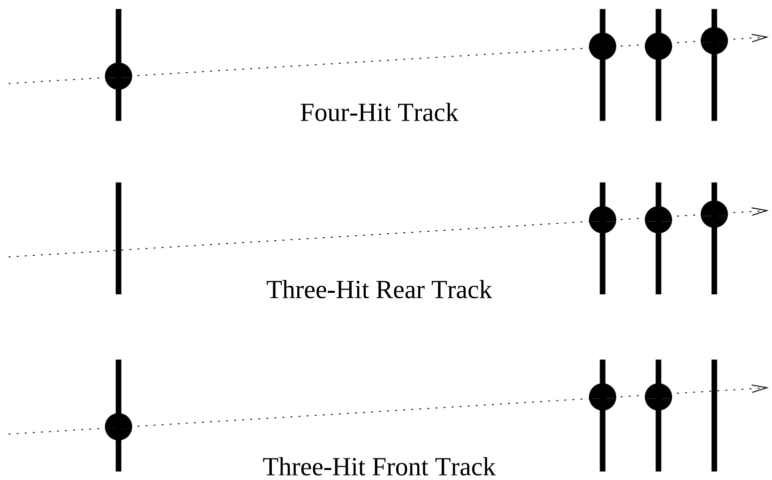

Figure 2: The three possible types of tracks in the CCD spectrometer.

- *Four-hit tracks* - These are clearly the best quality tracks both in terms of low fake probability and momentum resolution.
- *Three-hit rear tracks* - These are tracks with a hit in each of the rear CCD's, but no matching hit in the front CCD. They have a reasonably low fake rate, but the will not have the resolution of the four-hit tracks. Still they should be fine for the N=2 analysis.
- *Three-hit front tracks* - These are tracks with a hit in the front CCD and any two of the rear CCD's. They suffer from a high combinatoric fake rate, but should have resolution similar to that of the four-hit tracks. They will probably be useful for the pair analysis.

It's important to comment that, because of the limits of the spatial resolution in projecting tracks from the rear CCD's to the front CCD, there will be many matching ambiguities at the front CCD in events with large numbers of tracks, making the reconstruction of meaningful four-hit tracks impossible. Therefore, for the high-multiplicity events used in the N=2 analysis, the front hits are not used, leaving only three-hit rear tracks.

3 CCD DST Ntuple

After the track reconstructor has run, the track information is stored to a column-wise ntuple. The NTUPLE has one entry per event, with variable arrays indexed to the number of tracks, thus the NTUPLE serves as a DST for the tracking information.

The general philosophy is to set the tracking cuts as loose as is practical to generate the DST, so the user can study the effects of various cuts during analysis.

The DST Ntuples are stored to files with names of the form ccdnnnnn.dst, where nnnnn is the run number. The files contain only the DST Ntuple, which is ID 5. The Ntuple contains one variable sized block for each event, which is called "EVENT".

3.1 Ntuple Elements

The EVENT block was created from a C program, where data were stored in the following structure:

```
#define MAXTRK 2000
#define MAX_CCD 8

struct ccddst_s {
  long irun;                /* Run number */
  long ievent;              /* Event number */
  float ptkick;             /* ptkick of the magnet */
  long iuse;                /* user longword (undefined) */
  float ruse;               /* user real word (undefined) */
  long nhit[MAX_CCD];       /* number of hits in each CCD */
  long nclust[MAX_CCD];     /* number of clusters in each CCD */
  long npoint[MAX_CCD];     /* number of points in each CCD */
  long ntrack;              /* number of tracks */
  long netrak;              /* number of electrons */
  long nptrak;              /* number of positrons */
  float pmom[MAXTRK][3];    /* momentum vector (x,y,z) */
  float ptot[MAXTRK];       /* Total momentum */
  float xmag[MAXTRK];       /* X position at magnet center */
  float ymag[MAXTRK];       /* Y position at magnet center */
  float zmag[MAXTRK];       /* Z position at magnet center */
  long iq[MAXTRK];          /* charge of track */
  long npfit[MAXTRK];       /* number of planes in the track fit */
  float chi2[MAXTRK];       /* chi2 of fit */
  float qmax[MAXTRK];       /* maximum charge of any hit on track */
  long itstat[MAXTRK];      /* track status word (defined in CCDData) */
  };
```

The size of the track arrays within the block was indexed to the "ntrack" variable. The significance of most of the entries should be clear from the comments. The most important ones will be discussed shortly. You'll notice that care has been taken to make the elements of the structure compatible with the elements of a FORTRAN COMMON block and indeed, these same names were used in the NTUPLE definition, with the usual allowances for the differences between FORTRAN and C array indexing conventions. The following is the corresponding COMMON block definition, produced automatically by the UWFUNC command within PAW:

```
*
*--   Ntuple Variable Declarations
*
      REAL PTKICK,RUSE,PMOM(3,2000),PTOT(2000),XMAG(2000),YMAG(2000)
     + ,ZMAG(2000),CHI2(2000),QMAX(2000)
      INTEGER IRUN,IEVENT,IUSE,NHIT(8),NCLUST(8),NPOINT(8),NTRACK,NETRAK
     + ,NPTRAK,IQ(2000),NPFIT(2000),ITSTAT(2000)
*
      COMMON /PAWCR4/ IRUN,IEVENT,PTKICK,IUSE,RUSE,NHIT,NCLUST,NPOINT
     + ,NTRACK,NETRAK,NPTRAK,PMOM,PTOT,XMAG,YMAG,ZMAG,IQ,NPFIT,CHI2,QMAX
     + ,ITSTAT
*
```

Some of the elements are essentially "expert-only". The most important for a general user are

Element	Definition
IRUN	Experimental run number.
IEVENT	Experimental event number. This and the run number can be used to match to main data stream.
PTKICK	Pt kick of the analysis magnet, in MeV/c. Only runs where the kick was 200 MeV/c or greater should be used for N=2 analysis.
NTRACK	Total number of tracks. This is the indexing variable for the variable arrays in the Ntuple.
NETRAK	Number of electrons.
NPTRAK	Number of positrons.
PTOT(1:NTRACK)	Total momentum of each track, in MeV/c.
XMAG(1:NTRACK)	Extrapolated X position of track at center of magnet.
YMAG(1:NTRACK)	Extrapolated Y position of track at center of magnet. This and XMAG are the primary quality cuts during analysis.
NPFIT(1:NTRACK)	Number of planes used in the track fit, could be 2-4, but should always be 3 for the N=2 analysis Ntuples.
CHI2(1:NTRACK)	chi-squared per degree of freedom for the track fit. Could be a quality cut, but use with care.
QMAX(1:NTRACK)	Maximum charge of any single hit on the track. An important quality cut earlier-on, but probably shouldn't be used by non-experts.
ITSTAT(1:NTRACK)	Encoded track information, described shortly.

Note that only the total momentum is listed as useful. This is because the individual elements are not, at present, very meaningful. As mentioned before, because the high track multiplicities in the N=2 study sample, the front CCD planes are not used in the reconstruction, so NPFIT will always be 3.

For individual tracks, the most useful indicators of quality are XMAG and YMAG. Because the converted e^+e^- pairs are nearly collinear at these energies, they do not separate until they get to the center of the magnet. Thus, the real tracks in the spectrometer should project back to the same point in the center of the magnet. To save time, the track reconstructor requires *all* track candidates to come from approximately the direction of the magnet center by using "roads" to select hit pairs, but there is no further bias; that is, given two sets of hits which fall in the road, there is no favor given to the one which points closer to the magnet center, at the reconstructor level.

It's important to remember that because of beam steering, the location of the gamma beam moved from run to run, so the cuts on XMAG and YMAG may have to be changed for different sets of runs.

3.2 The Track Status Word

The track status word (ITSTAT above) has encoded information about the track. It's not particularly useful for the N=2 analysis, but it is described here for the sake of completeness.

This word is used to "keep track" of the progress of the track reconstruction, so some of the bits are not particularly useful offline. Each bit is defined in the original C++ reconstruction code, and I have also given each a FORTRAN-compatible designation.

C++	**FORTRAN**	**Value (Hex)**	**Definition**
has_hits	ITSHHT	00000001	Track candidate has hits after quality cuts (always set if track makes it to DST).
has_upstream_hit	ITSUSH	00000002	Track has an hit in the upstream CCD plane.
has_fit	ITSFIT	00000004	Fit has been performed on track (always set for DST tracks).
fit_planes	ITSPLN	00000100	Start of encoded information about which planes were used in fit (described below).
fit_error	ITSERR	00010000	An error occured during fitting (should never be set for DST tracks).

All of these are single bit flags, with the exception of the "fit_planes" entry. This parameter indicates that starting position of the four-bit word indicating which planes were used in the fit, with planes being numbered from upstream to downstream. The encoding is to set the bits in the status word corresponding to

$$ITSPLN * (IUSE0 + IUSE1 * 2 + IUSE2 * 4 + IUSE3 * 8)$$

where IUSEn is set to 1 if plane n was used in the fit. For example, if planes 0,2, and 3 were used in the fit, the bits corresponding to (Hex) 00000D00 would be set in the status word. For the N=2 analysis set, bits (Hex) 00000E00 (planes 1,2, and 3) should always be set.

4 Ntuples, Supporting Files, and Examples

The CCD archive available via anonymous ftp from pupbcf.hep.princeton.edu (128.112.100.22). This document can is stored as /pub/e144/ccddst/doc/ccddst.ps. The DST Ntuple files are in /pub/e144/dst.

In the directory /pub/e144/ccddst is an example kumac file ccdxmp.kumac, which takes as its arguments the names of the E144 Ntuple file and the corresponding CCD Ntuple file. It calls the COMIS file ccdxmp.f, which is found in the same directory.

The example "streams" the two Ntuples together, using event numbers to match blocks of data. The details should be clear from the comments. The real work of streaming in the CCD information is done by the GETCCD subroutine. It must be remembered that most events don't have CCD data. CCD data is only present on the 1/2 Hz trigger, and is missing from about 10% of those.

This example generates 6 histograms:

100 X position of all tracks, projected back to the center of the magnet (in mm).

101 Y position of all tracks, projected back to the center of the magnet (in mm).

105 X position of tracks which have a cut on their Y position.

106 Y position of tracks which have a cut on their X position.

110 Momentum of tracks passing both X and Y cuts (in GeV).

120 Number of tracks passing X and Y cuts versus the EC37 signal.

This particular example is set up to work with the z05 format E144 Ntuples, but the GETCCD subroutine should be fairly portable, either to another COMIS routine or to a standalone program.