MarlinTPC
TPCCloudSimulation

Jason Abernathy

General Algorithm

1) Receive SimTrackerHits from Mokka
2) Convert energy deposits into electron clouds
    1) **26eV per electron**
3) Transport clouds of electrons to the readout electronics. Along the way:
    1) **Diffuse the clouds**
    2) **Amplify the clouds at the GEMs**
4) Deposit the clouds on the pads
5) Turn the electrons into a signal
6) Combine signals
7) Digitize signals

The Steering File

```
<execute>
    <processor name="MyGlobalFieldProcessor"/>
    <processor name="MyPrimaryCloudIonisationProcessor"/>
    <processor name="MyElectronCloudDriftGapProcessor"/>
    <processor name="MyElectronCloudGEM1Processor"/>
    <processor name="MyElectronCloudTransferGapProcessor"/>
    <processor name="MyElectronCloudGEM2Processor"/>
    <processor name="MyElectronCloudInductionGapProcessor"/>
    <processor name="MyElectronCloudChargeDepositProcessor"/>
    <processor name="MySignalShaperGaussianProcessor"/>
    <processor name="MySignalCombinerProcessor"/>
    <processor name="MySignalDigitisationProcessor"/>
    <processor name="MyLCIOOutputProcessor"/>
</execute>
```

PrimaryCloudIonisation

- Converts energy to electrons
  - **Perhaps this should be a processor parameter**
- Can choose to track only the primary particle
  - **Keeps some simulations simple**
- The clouds have:
  - **Mean position**
  - **Position std dev. in (x,y)**
  - **Time std dev. in z**
  - **A number of electrons**

ElectronCloudDrifter

- Can drift in homogeneous or non-homogeneous fields
  - **Uses RK45 stepper from the GSL**
  - **B-fields are set up using the field manager**
- Drift parameters:
  - **Drift velocity (in mm/us)**
  - **Gas diffusion (in um/sqrt(cm))**
  - **OmegaTau – gas parameter**
  - **Dimensions of TPC section**
    - Z limits, inner and outer radius
- One processor for each gas section of the TPC

ElectronCloudGEMAmplification

- Rips apart the clouds into single electrons
  - **Each electron has a chance to create a new cloud**
- Parameters:
  - **Collection efficiency**
  - **Extraction efficiency**
  - **Amplification**
  - **Defocussing**
  - **Width**
- Writes the amplification to the current event

ElectronCloudChargeDeposit

- Rips apart the clouds into single electrons
  - **Shoots a random position for each one**
  - **Uses GEAR (with modules) to add electrons to pads**
- Creates parameterized “ChargeSignals”
  - **An lcio::GenericObject**
  - **ChargeSignal contains**
    - Number of electrons
    - Pad number, cloud number, module id
      - Keeps the signals unique
    - Time variance of collected electrons

SignalShaperGaussian

- Uses the ChargeSignals and
  - **Spreads the electrons collected in a gaussian shape**
  - **Variance is provided by ChargeSignal**
- Creates the ChannelCorrection objects
- Works for now but...
  - **Will not work for multiple signals per pad**
  - **The height of multiple pulses are not directly proportional to the number of electrons collected (ie: non-linear)**
- Parameters:
  - **Gain, polarity, time resolution (bin width)**

SignalCombiner / Digitisation

- SignalCombiner
  - **Simply combines signals which should be in the same channel (TrackerData)**
- Digitisation
  - **Converts the TrackerData into TrackerRawData**
  - **Parameters:**
    - Bits per sample, pedestal value
  - **Creates conditions**
    - Pedestals
    - GenericADCElectronicsParameters

What Needs Work?

- Magnetic field information
  - **Difficult to use**
- Signal shaping
- Test signal combiner
  - **I've only simulated the primary particle**
- Stats:
  - **~200 deposits from primary track**
  - **~700k electron clouds hitting electronics**
  - **~900 pads with (6mm x 1mm)**
  - **~40s per event to simulate**
    - That's with a Debug build