

# 経過報告

田中 隆己

2012年11月8日

## 1 解析

NEBULA の解析に自身をつけるために、先に Erel をお手軽に求めた。ここでいうお手軽とは、

- いろいろな consistency などを確認しない
- キャリブレーションなどは最小限
- 荷電粒子の momentum の微調整は neutron の beta を用いる
- event 数 (断面積) などの議論はまったく無視

等、到底外部には公開できないようなクオリティという意味。

### 1.1 calibration

上流の装置のキャリブレーションは行わず、HOD のキャリブレーションと DC の STC(単なる積分) を行った。

#### 1.1.1 HOD

近藤さん手法を用いて HOD の相対的な T と Q を合わせた。ここでは境界  $\pm 5\text{mm}$  を切り出して接続し、Be10 の磁場 sweep run を用いた。図 1.1 にキャリブレーション後の T と Q を示す。なお、x は FDC2 での位置となっている。Q の揺らぎについて近藤さんと議論したが、例えばモジュール自体の厚みが揺らいでいるのではないかという疑念が湧いている。これは次の機会に実測してみたいと思う。

#### 1.1.2 STC

単に TDC 分布の積分を行った。図 1.2, 1.31.4 に C(Be14, Be12+X) のときの TDC の分布、STC、残差分布を示す。

#### 1.1.3 DEBUG: HOD

smcom では HODP もデータを取っており、この存在を忘れていたことで、例えば HODQMAX の処理を間違えていた。

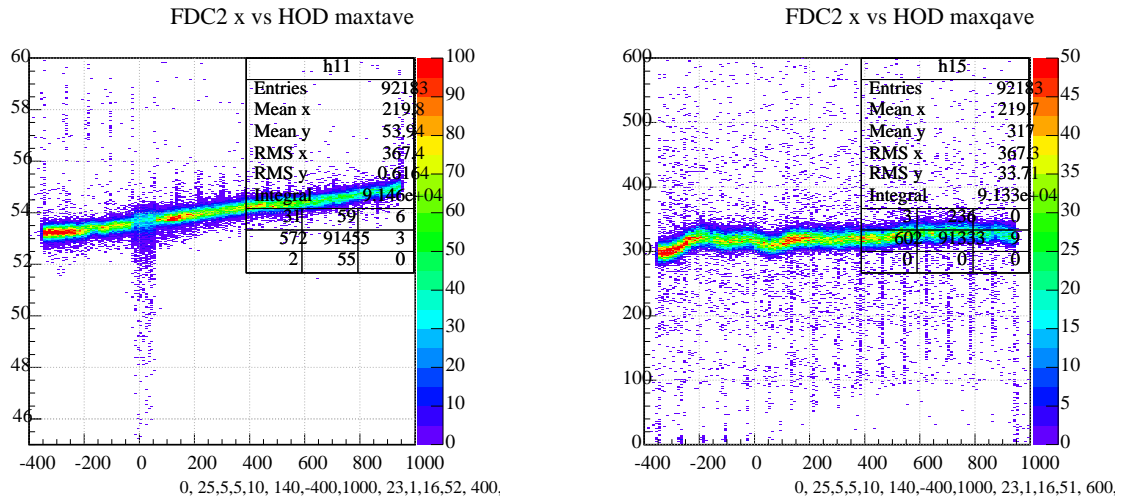


図 1.1 hod の T と Q の相対的な calibration。Q は一本のモジュールの中でもかなり揺らぎがある。

#### 1.1.4 DEBUG: FDC2

smcom ではケーブルの差し間違えをしていたらしく、dayone の parameter を使うと x の event が一部かけたように見える。なので、geometry 情報を正しくしてあげる必要がある。

#### 1.1.5 DEBUG: DSB

DSB trigger のときはやっぱりタイミングがシフトしている。理屈では T0 をすべての測定から引けばよいが、それも面倒なのでここでは DSB trigger の event は全て捨てた。

### 1.2 上流 PI

キャリブレーションは行っていないが BigRIPS チームの PI を信じて beam を選んだ。この gate の様子を図 1.7 に示す。

### 1.3 target hit

target は 30 なので BDC の中心が合っていると信じて図 1.6 の様に切った。

### 1.4 下流 PI

こちらは上流 PI を信じて適当に領域を選んだ。この gate の様子を図??に示す。解析速度向上のために下流 PI の gate は HOD の x vs t で生成している。

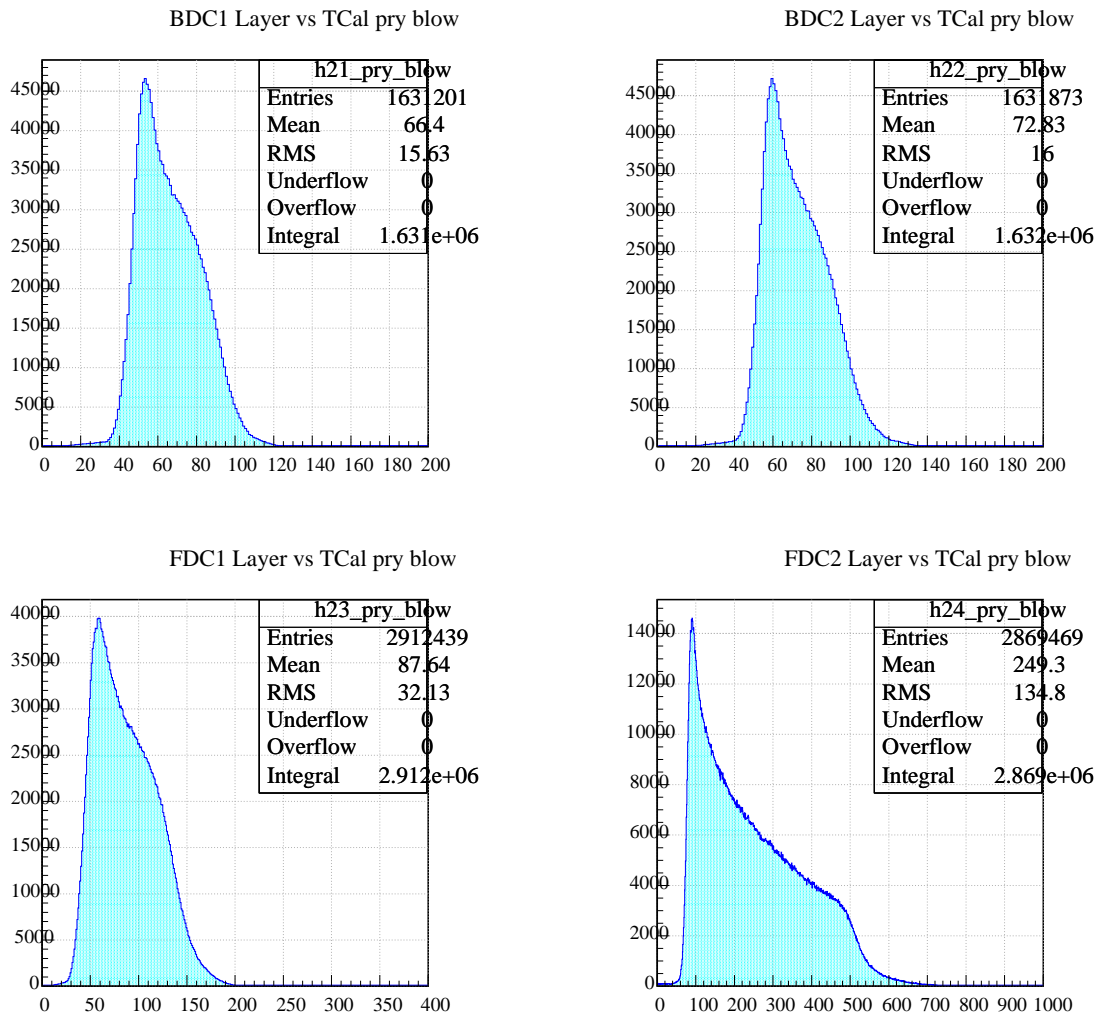


図 1.2 C(Be14,Be12+X) における traw

## 1.5 neutron の momentum

neutron の momentum は T の offset 以外に調整する必要はないので、target から来ていると思われる gamma 線の beta を用いて offset を微調した。このときの beta と momentum を図??に示す。

## 1.6 fragment の momentum

FDC1 と FDC2 間の一次の matrix を磯部さんの書いたコードを用いて求めた。

ここで、delta は  $(p_{mean} - p)/p$  と定義している。この matrix より、

という式から実験で得られる x と a を用いて momentum を求めた。各粒子における matrix の値を以下に

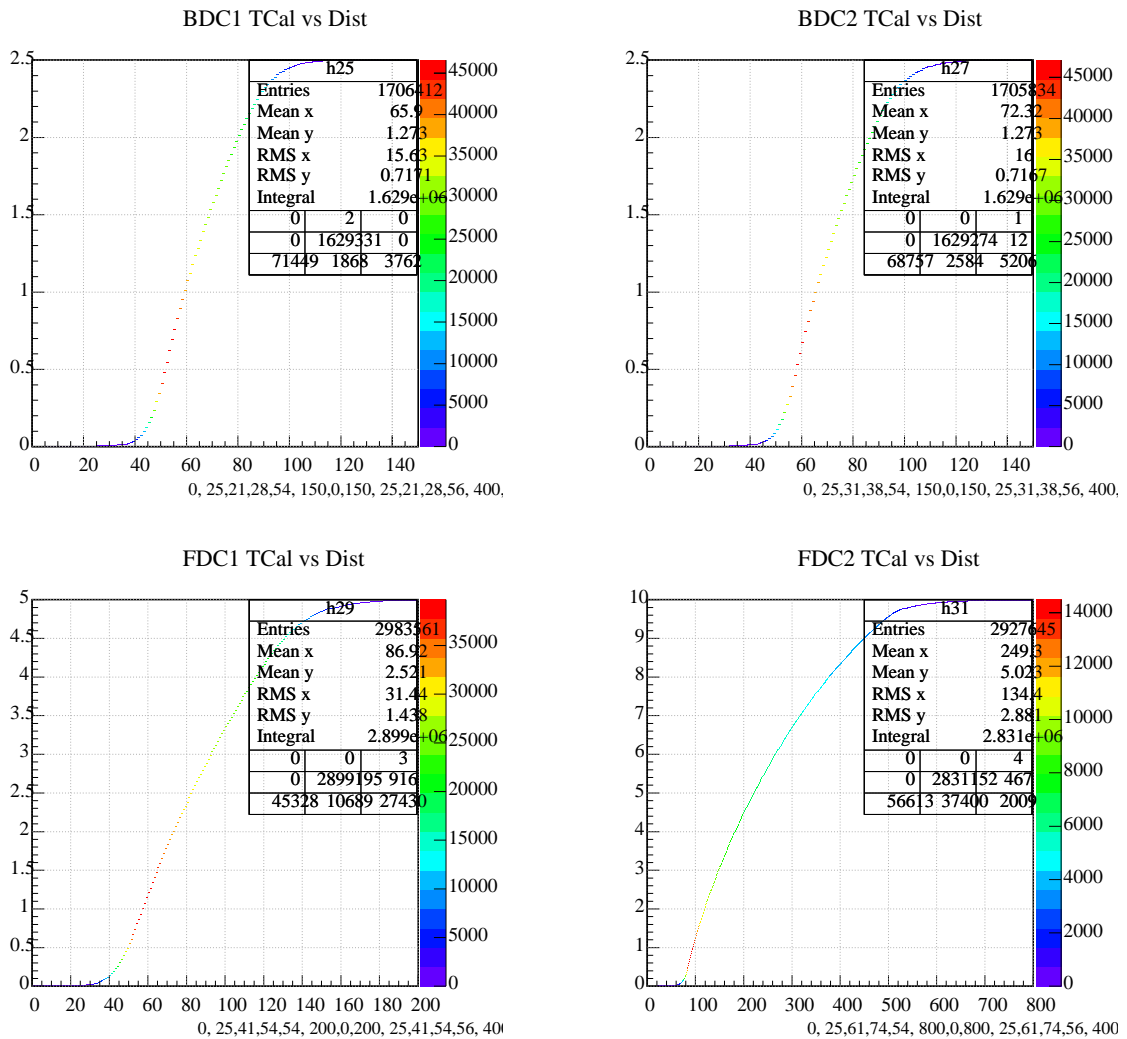


図 1.3 C(Be14,Be12+X) における STC

示す。

ちなみに、FDC2 での hit した位置のシミュレーションと実験の違いは、15C で-17mm, 14C で +1mm となった。

## 2 HIME 用 DAQ の構築

現在は 554 号室で HIME 用 DAQ の立ち上げを行っている。

### 2.1 DAQ の構成

- d01: event builder. 普通の PC に fedora14 32bit を入れただけ。
- d02: event sender(VME). VME ボードに PC と VME コントローラが乗っており、これに馬場さんが

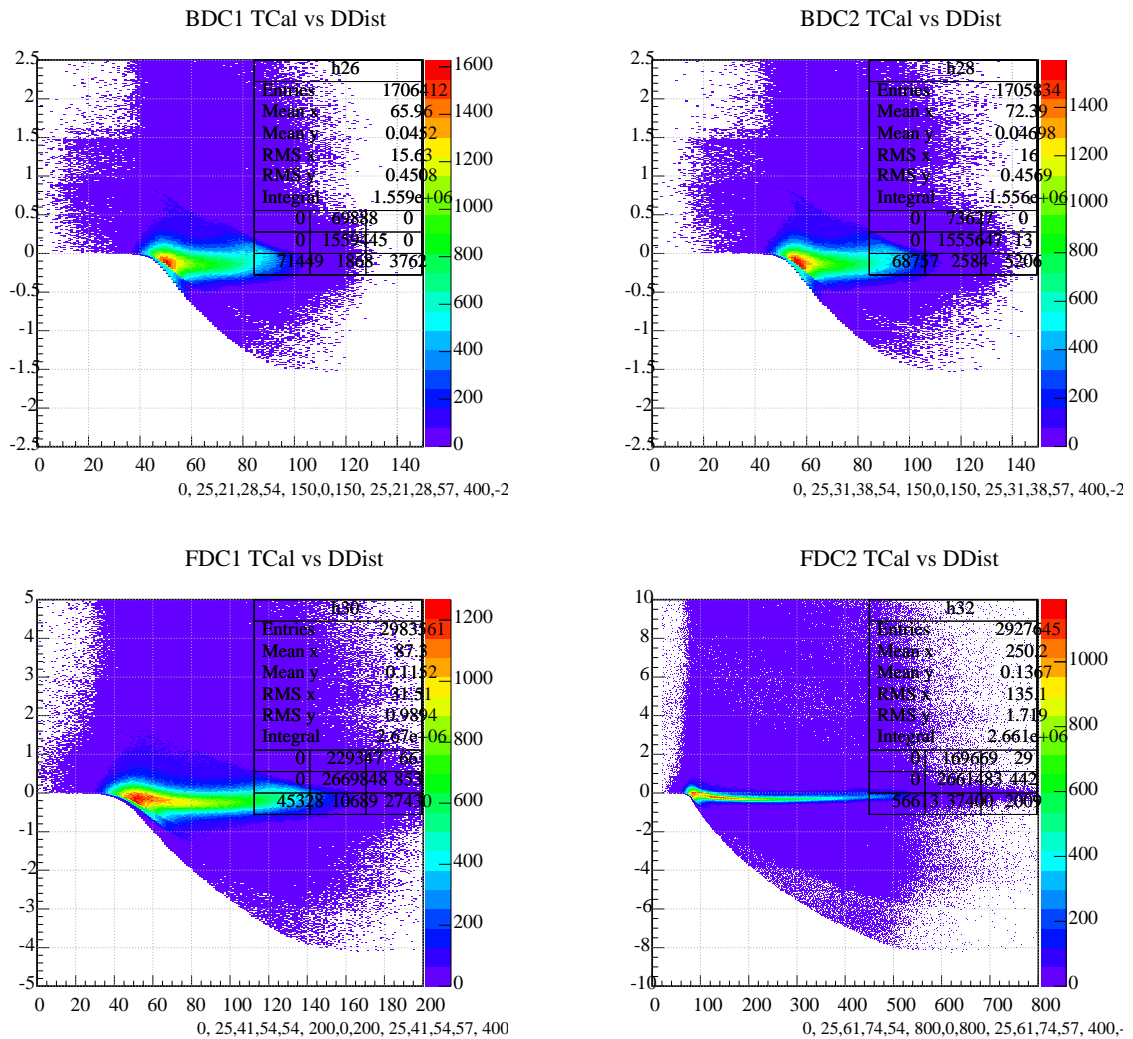


図 1.4 C(Be14,Be12+X) における DDIST

らもらった OS image を入れた

- d03: event sender(CAMAC). まだ何も用意していないが、CAMAC で一部データを取ることもなるかもしれない。

## 2.2 VME コントローラ (V7768) の front panel

勘違いして新しい規格の構造の物 (VME64x(向こうは VME64 と呼ぶ): IEEE 1101.1, VME64(向こうは VME32 と呼ぶ): standard vme) を買ってしまい、取り付けはできるものの、取り外しに苦労している。そのために、代理店に問い合わせた補修パーツを無料で手に入れた。

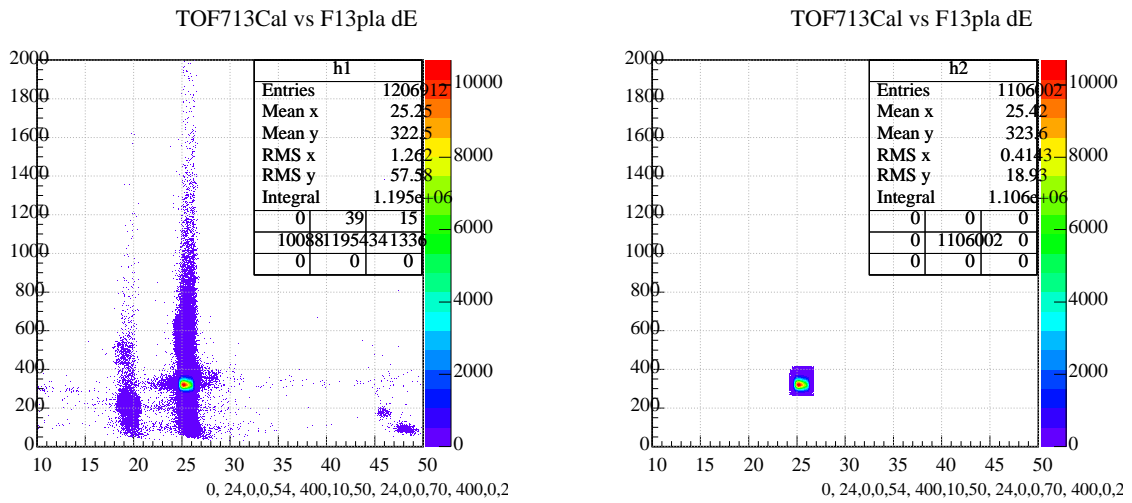


図 1.5 C(Be14,Be12+X) における upstream の PID

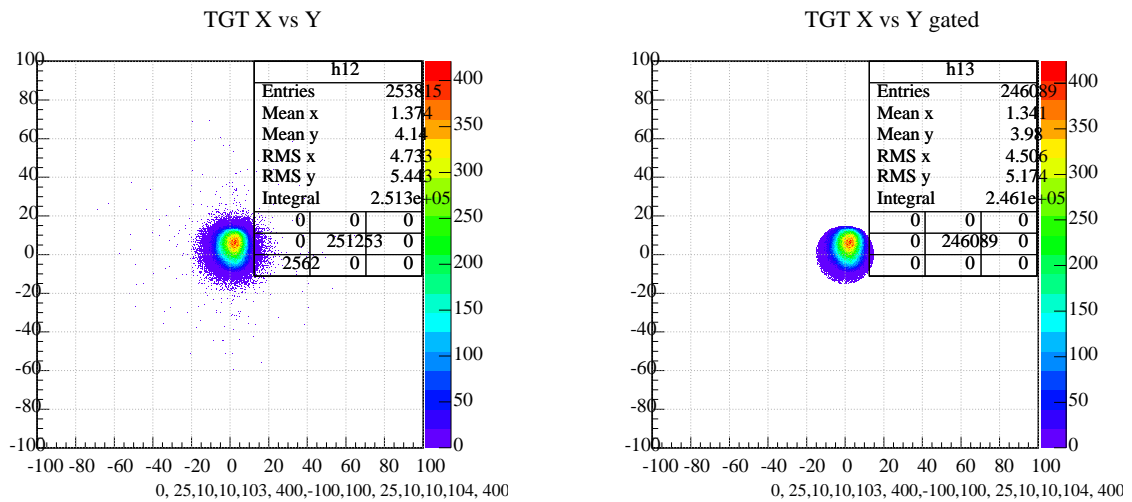


図 1.6 C(Be14,Be12+X) における target の像

## 2.3 VME コントローラ (V7768) への OS のインストール

具体的な手順などの詳細は web( ryuki/iroiro/daq/) に書いたので参照ください。結構苦労しました。

BIOS 的に 8GB までの CF でしか boot できない模様。ただし、馬場さんが 8GB では無理だったと言っているので 4GB が安全かも。一方、boot できてしまえば 4GB 以上の CF でも扱える (例えば CD で boot する等)。容量の大きなシステムを組みたければ CF は boot のみに用いて、usb メモリにシステムを入れることで何とか実現できる。ただし、外付け DVD ドライブで読み込めたのは puppy linux だけだったので、fedora や ubuntu を入れたければあらかじめ CF に手で install する必要がある。

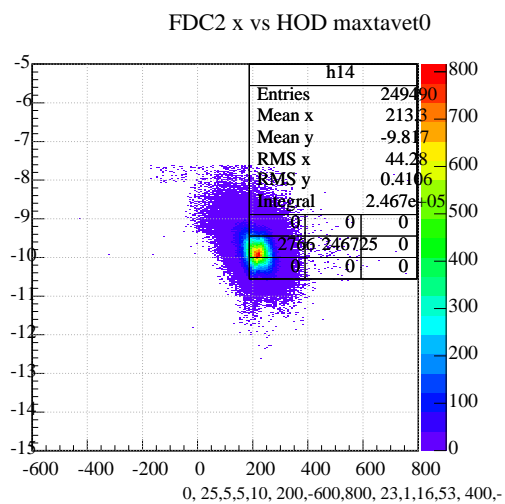
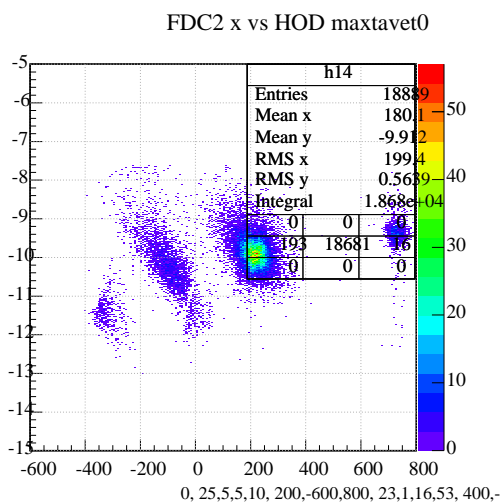


図 1.7 C(Be14,Be12+X)における upstream の PID