デジタルオシロスコープによるシンチレーション カウンターの高速サンプリング読み出しの研究

瓜生　奈都美　　福井　千尋

平成 25 年 3 月 7 日

目次

第１章　はじめに

1-1 実験目的

粒子の入射時に検出器が発する電気信号の信号電荷だけでなく、その時間発展をも測定することは、その検出過程とは独立的で重要な情報を与える。そこで、デジタルオシロスコープにより得られる時系列データをネットワークを通してパソコンで読み込む手順を確立した。さらに、放射線源が発するγ線が入射した場合の無機シンチレーターの発光を光電子増倍管で読み出し、その発光量および時定数を測定した。本実験では２種類の無機シンチレーターを使用した。 また、光電子増倍管の受光部窓ガラスで発生するチェレンコフ光との分離も議論する。

1-2 本論文の構成

本論文ではまず γ 線、使用する線源、シンチレーション光、チェレンコフ光などの原理について述べ、さらに実験装置の概要について説明を行う。さらに、データ収集の方法、データ解析の方法を示し、最後に得られた実験結果を記述するとともに、考察を加える。

第２章　原理

2-1 γ線

γ線は、励起状態の原子核が脱励起する際に放射する波長のごく短い電磁波である。
波長がごく短いために、透過力が強いことが特色の一つである。γ線は荷電粒子でないために、直接には物質を電離する作用がほとんどない。しかし、γ線は、2-1-2で述べるようなさまざまな過程を通じて物質から電子をたたき出すため、これがほかの荷電粒子と同様、物質中で電離あるいは励起によりエネルギーを損失する。
また、γ線の波長はごく短いため、粒子としての性質が強く出てくる。

2-1-1(a) 実験に用いた線源

本実験では ^{137}Cs 密封線源を用いた。

^{137}Cs は、半減期 30.0 年で、β^-崩壊により ^{137}Ba となる。この時、93.5%は準安定状態の ^{137m}Ba となり、その後エネルギー約 662keV の γ 線を放出し、^{137}Ba となる。残りの 6.5%は、直接基底状態の ^{137}Ba となる。

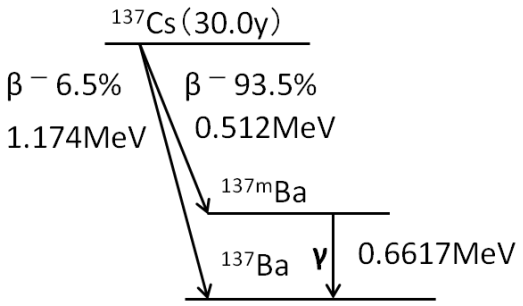


図 2-1：^{137} Cs の壊変図

図 2-2：使用した ^{137}Cs 密封線源

2-1-2 光電効果

光電効果とは、γ線が物質中に入射した際に、そのエネルギーの全てを軌道電子に与え、その結果この電子が原子外へ放出される現象である。

γ線の全エネルギーを E_γ　電子の束縛エネルギーを　I　とすると、
電子に与えられるエネルギーE は

$E=E_\gamma - I$

となる。

本実験においては、数百 keV のエネルギーを持つγ線について考えており、電子の束縛エネルギー　I は数 keV 程度で測定器のエネルギー分解能と比較しても十分小さい。よって、光電効果が生じればその波高は　E_γ　に比例するものと考える。

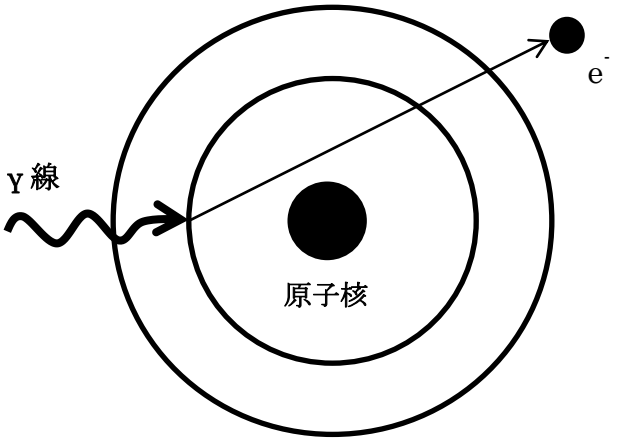


図 2-3：光電効果

2-1-3 コンプトン散乱

コンプトン散乱とは、自由電子によって物質中に入射したγ線が散乱を受ける現象である。
散乱前の光子のエネルギーを $h\nu(=E_\gamma)$　散乱された光子のエネルギーを $h\nu'$とすると、
電子に与えられるエネルギーE は

$E= E_\gamma - h\nu'$

となる。

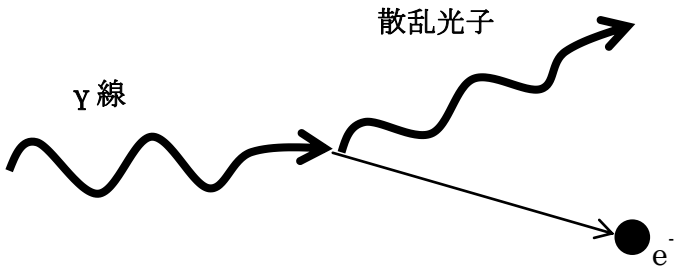


図 2-4：コンプトン散乱

2-2 シンチレーション光

荷電粒子が物質（シンチレーター）を通過する際、物質中の電子を励起状態にする。これが元に戻るときに放出する可視光（紫外線の場合もある）をシンチレーション光と呼ぶ。

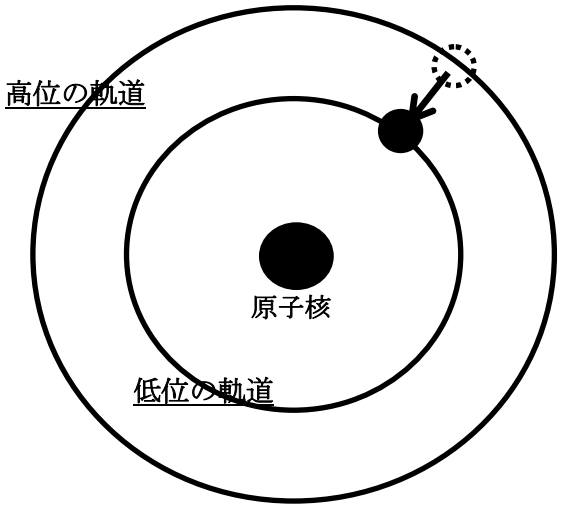


図 2-5：シンチレーション光

2-3 チェレンコフ光

荷電粒子が水やガラスのような透明な物質を通過する際、それが物質中での光の伝播する速さ（c/n）を超えると、荷電粒子の周囲の電磁場の波面が重なり、衝撃波が出る。これを、チェレンコフ光と呼ぶ。粒子の進行方向になす角を θ とすれば（図 2-5）

$$\cos\theta = \frac{1}{n\beta} \qquad (\beta = n/c, n = \text{屈折率}) \qquad (5)$$

光を出す条件は、$\beta \geq 1/n$ である。光子の数は $\sin^2\theta$ に比例する。

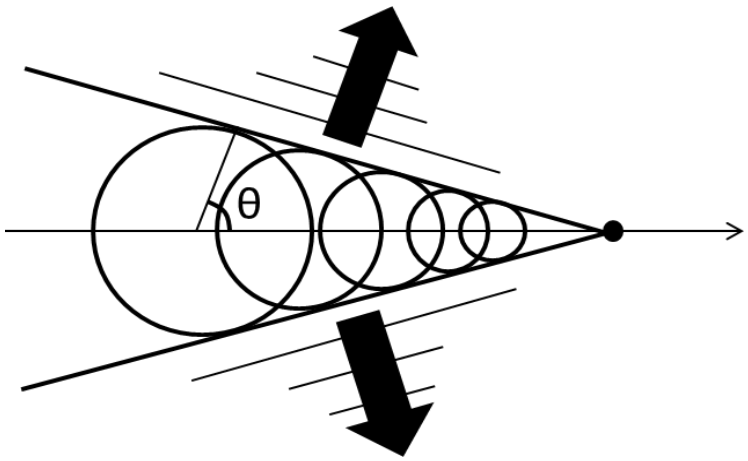


図 2-5：チェレンコフ光

第3章　実験装置の概要

本実験ではシンチレーターと光電子増倍管を用いてシンチレーションカウンターを製作した。また、デジタルオシロスコープにより得られる時系列データをネットワークを通してパソコンで読み込みを行った。シンチレーションカウンターとデジタルオシロスコープについて以下にまとめる。

3-1 シンチレーションカウンター

本実験では、まずシンチレーションカウンターを製作した。製作したシンチレーションカウンターを以下に示す。

シンチレーターと光電子増倍管の受光面は、光学グリースを用いて粘着する。シンチレーションカウンターには厚紙の蓋をかぶせ、さらに遮光布で覆い、不要な光が入らないようにする。

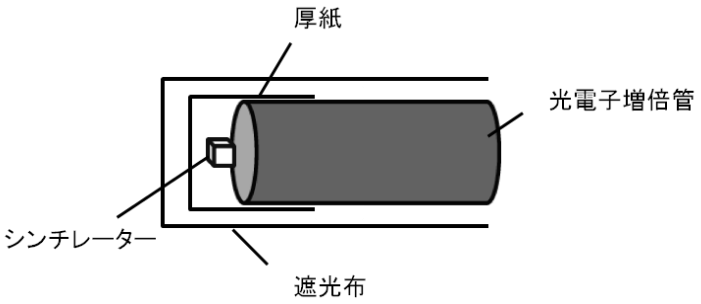


図 3-1：シンチレーションカウンター

3-1.1 シンチレーター

シンチレーション光を多く発生する物質をシンチレーターと呼び、大きくポリエチレン等を基本にした有機シンチレーターとイオン結晶、酸化物結晶、フッ化物結晶などの無機シンチレーターに大別される。本実験では後者のうち、Ceをドープした二種類の酸化物結晶シンチレーター(5mm×5mm角)を使用した。これらは発光元素のCeが、Ce^{3+}の5d準位から4f準位に遷移する際発光し、その強度が高いものである。詳しくは以下に述べる。

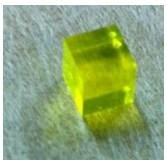
荷電粒子がシンチレーターを通過する際、物質中の電子を励起状態にし、これが基底状態に戻るときに光を放出する。これがシンチレーション光であるが、光の減衰時間は励起状態から脱励起する単位時間当たりの遷移確率の逆数になり、それをシンチレーターの時定数とも呼ぶ。

3-1.2(a) LSO（Ce添加ルテチウムシリコンオキサイド）

組成式Ce:Lu_2SiO_5

ルテチウムを含む酸化物シンチレーターは発光量が豊富なものが多く、LSOもその一種である。ただし、ルテチウムは通常^{175}Luであるが、その他に^{176}Luという放射性同位体が天然存在比は約2.6%で存在し、半減期が3.78×10^{10}年と長く、β崩壊するのでバックグラウンド源となるという欠点がある。

3-1.2(b) GAGG（Ce添加ガドリニウムアルミニウムガリウムガーネット）



組成式Ce:$Gd_3Ga_3Al_2O_{12}$

LSOがルテチウムを含むために検出体内部にバックグラウンド源を含んでしまう。そこでルテチウム以外の元素のみからなるシンチレーターとして開発された。

以下にLSOおよびGAGGの特性(公称値)を示す。

	発光量[photon/MeV]	発光波長[nm]	減衰時間[ns]
LSO	26000	420	40（公称値）
GAGG	60000	520	90（公称値）

表 3-1：使用したシンチレーターの特性

3-1.2 光電子増倍管

光電子増倍管(photomultiplier) とは、入射光子を多数個の電子に変換する真空管デバイスで、入射窓、光電面、電子増倍部より構成されている。その構造を以下の図に示す。

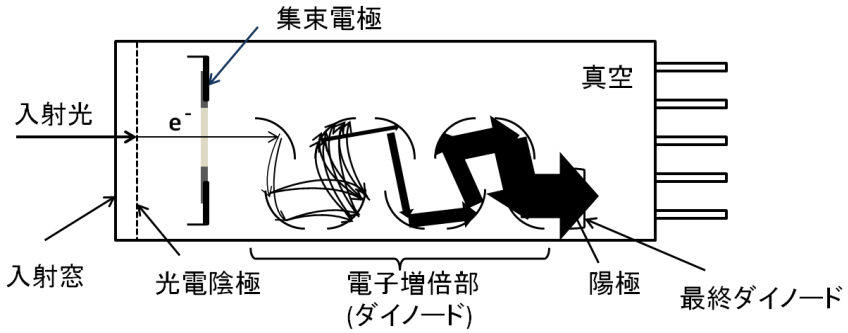


図 3-2：光電子増倍管の構造

光電子増倍管に入射した光は以下に示す過程を経て信号として出力される。

[1] ガラス窓を透過する。

[2] 光電面内の電子を放出し、真空中に光電子を放出する。

…入射窓ガラスの裏側（光電面）には光電効果を起こしやすい物質を塗布しており、光子を光電子に変換する。

[3] 光電子は集束電極により第一ダイノード上に誘導され、二次電子増倍された後、引き続き各ダイノードで二次電子放出を繰り返す。

…内部に何段か設けられている電極．ダイノード間に電位差をつけてあるので、電子がダイノードに当たるたびに新たに電子をたたき出すことによって、電気信号を大きく(=増幅) する．

[4] 最終ダイノードより放出された二次電子は陽極よりパルス信号として取り出される。

本実験では、浜松ホトニクス社の H7195UV を用いた。

3-2 デジタルオシロスコープ

アナログオシロスコープが化学的蛍光体を用いて波形を記憶しているのに対し，デジタルオシロスコープは電気的な手段で波形を記憶する。デジタルオシロスコープは，A/D変換器を用いて測定した電圧をデジタルデータに変換し，メモリに記憶させた後，画面上に波形として表示する。今日のデジタルオシロスコープは、LANインターフェースを装備し、これを経由してネットワークに接続されたコンピューターと交信可能である。本実験では、Agilent Technologies社のDSO5014A型を用い、ネットワーク入出力機能を用いて、パソコンへこのデジタル値を読み出した。
以下に、入力端子からディスプレイに表示されるまでの流れを示す。

・入力端子
→アッテネータ、アンプ

入力端子に加えられた入力信号は感度切換えのためのアッテネータを介してアンプへ導かれ、アンプで適当な大きさに増幅される。

これらが内蔵されているのでデジタルオシロスコープではレンジの切り替えが可能である。

→A/D変換器

加えられた信号を離散的な時間間隔でサンプルし、信号電圧をデジタル値(サンプルポイント)に変換する。波形を一定間隔でデジタル値に変換する過程をサンプリングと呼び、サンプリングするスピードをサンプル・レート（単位はS/s：サンプル・パー・セック）という。本実験ではサンプル・レートを2GS/sに設定した。

→取得メモリ

サンプリングされたデータは、順次取得メモリに記録される。サンプルポイントが波形ポイントとして保存される。波形ポイントが集まって波形レコードとなる。

→マイクロプロセッサ

取得メモリからディスプレイメモリへデータを転送する。この際、必要に応じて信号処理、表示の制御を行う。

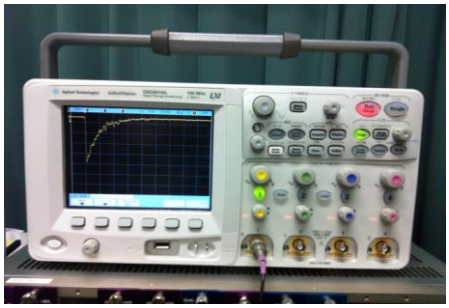
＊書き込みコントローラ

A/D変換および取得メモリへの書き込みの開始とトリガ回路からのトリガ信号を受けてのA/D変換および取得メモリへの書き込み停止を制御する。

書き込みコントローラは、その書き込みの停止のあと、マイクロプロセッサに取得の終了を知らせる。

→ディスプレイメモリ

ディスプレイメモリのデータはディスプレイコントローラによって表示装置上に波形の形で表示する。



第４章　プログラムの説明

前述のようにデジタルオシロスコープが2Gsamples/sec という高いサンプリングレートで取得した波形データは、LAN を経由してパソコンに読み出せる。今日では LXI と呼ばれる通信プロトコルに基づいてコマンドやデータの送受信を行うソフトウェアパッケージとして、VXI11 がある。

データ収集用プログラムは、VXI11 がサポートする関数をコールして、デジタルオシロスコープより波形データを得る。これをテキスト形式でディスク上に書き出したものを Raw Data（生データ）とし、オフラインの解析プログラム（付録 A）で処理して、シンチレーターの発光量及び時定数を求める。

4-1 データ収集用プログラム

4-1.1 ソースコードとアルゴリズム

```
#include <unistd.h>
#include "vxi11_user.h"

int main(int argc, char *argv[]){

  char* ofile = "wf.root";
  int nevt = 15000;
  int mode = 0;

  CLINK *clink;
  clink = new CLINK;
  static char* serverIP =   "10.0.1.103";

  if (vxi11_open_device(serverIP, clink) != 0){
    printf ("Couldn't open scope¥n");
    exit(1);
  }

  static char wf[10000];

  int     ret;

  ret=vxi11_send(clink, "channel1:disp on");
  ret=vxi11_send(clink, "timebase:scale 50ns"); // 50ns/div
  ret=vxi11_send(clink, "timebase:reference left");
```

対応するヘッダーファイルのインクルード

イベント数の設定

オシロスコープとの接続を張る

縦軸のスケールの設定

```
ret=vxi11_send(clink, "timebase:position 0."); //trig pos 10% from left

ret=vxi11_send(clink, "channel1:scale 2e-1"); // 200mV

ret=vxi11_send(clink, "waveform:format byte");
ret=vxi11_send(clink, "acq:type normal");
ret=vxi11_send(clink, "acq:count 1");
double yinc = vxi11_obtain_double_value(clink, ":WAV:YINC?");
double xinc = vxi11_obtain_double_value(clink, ":WAV:XINC?");
printf("%10.3e¥n",xinc);
printf("%10.3e¥n",yinc);
double hint = vxi11_obtain_double_value(clink, ":WAV:XINC?");
double range = vxi11_obtain_double_value(clink, ":TIM:RANGE?");
ret=vxi11_send(clink, "wav:poin:mode raw");
ret=vxi11_send(clink, "waveform:points 2000");
for (int iev=0 ; iev<nevt+1 ; iev++){
 ret=vxi11_send(clink, ":DIG chan1");
 ret=vxi11_send(clink, "wav:source chan1");
 ret=vxi11_send(clink, ":WAV:DATA?");
 long bytes_returned=vxi11_receive_data_block(clink, wf, 10000, 2000);
 if (iev==0) continue;

 printf("%d¥n", iev);
 printf("%d¥n", bytes_returned);
 for (int i=0 ; i<1000 ; i++){
  printf("%d ",(unsigned char)wf[i]);
  }
  printf("¥n");
   if (iev==0) continue;
  }

   vxi11_close_device(serverIP, clink);

}
```

横軸のスケールの設定

x ： x 軸データの時間間隔
y ： y 軸データの LSB 電圧

8bit の 2000 サンプル分のデータを PC が受け取る

イベント番号
サンプル数　を表示

１イベント当たり　1000 サンプルを表示
wf[i]　という 1000 個の配列をつくる

4-1.2 得られた Raw Data

```
5.000e-10
6.250e-03
1
1000
240 241 239 241 241 241 241 240 241 241 241 240 241 242 240 240 240
240 240 241 241241 242 240 240 241 241 240 241 242 240 242 240 241
240 241 241 241 241 240 241 241241 240 241 242 240 241 240 241 240
241 241 241 242 241 241 242 240 240 241 242 240 240 241 241 240 241
241 241 241 240 240 241 241 240 240 241 240 240 240 241 240 241 241
240 241 240 241 241 240 239 240 240 239 238 236 233 227 220 207 189
167 142 122 102 84 73 65 61 60 61 63 65 67 70 68 68 68 68 65 65 63 61
……


2
1000
238 238 239 239 240 239 241 241 240 241 241 240 239 240 240 240 239
238 239 237 239 239 239 239 240240 240 240 240 240 241 241 240 239
241 239 241 240 241 240 240 241 241 241 240 240 242 240 240 239 240
239 241 240 240 241 240 240 241 241 239 240 241 240 240 240 240 239
241 241 240 241 240 240 241 241 240 240 241 240 241 239 241 239 241
240 241 241 240 241 241 241 240 241 239 237 234 231 227 220 217 212
207 203 197 194 192 189 188 188 188 189 191 192 194 193 195 194 195
……
```

1 行目は、 x 軸データの時間間隔 0.5[ns/samples] を示す。

2 行目は、y 軸データの 1 当たりの電圧 $\frac{200\times8}{256} = 6.25[\mathrm{mV}]$ を示す。

3 行目はイベント番号、 4 行目はサンプリング数を示す。
5 行目から 8 bit のデータがサンプリング数の 1000 個続く。
その後は、イベント番号、サンプリング数、データが 1000 イベント分繰り返される。

第５章　光量・時定数の測定

^{137}Cs 線源を用いて、2 種類のシンチレーターLSO,GAGG の光量と時定数の測定を以下に示すセットアップにより行った。

光電子増倍管に-1800V の HV をかけ、シンチレーターは光学用グリース（BC-630）のみで固定する。

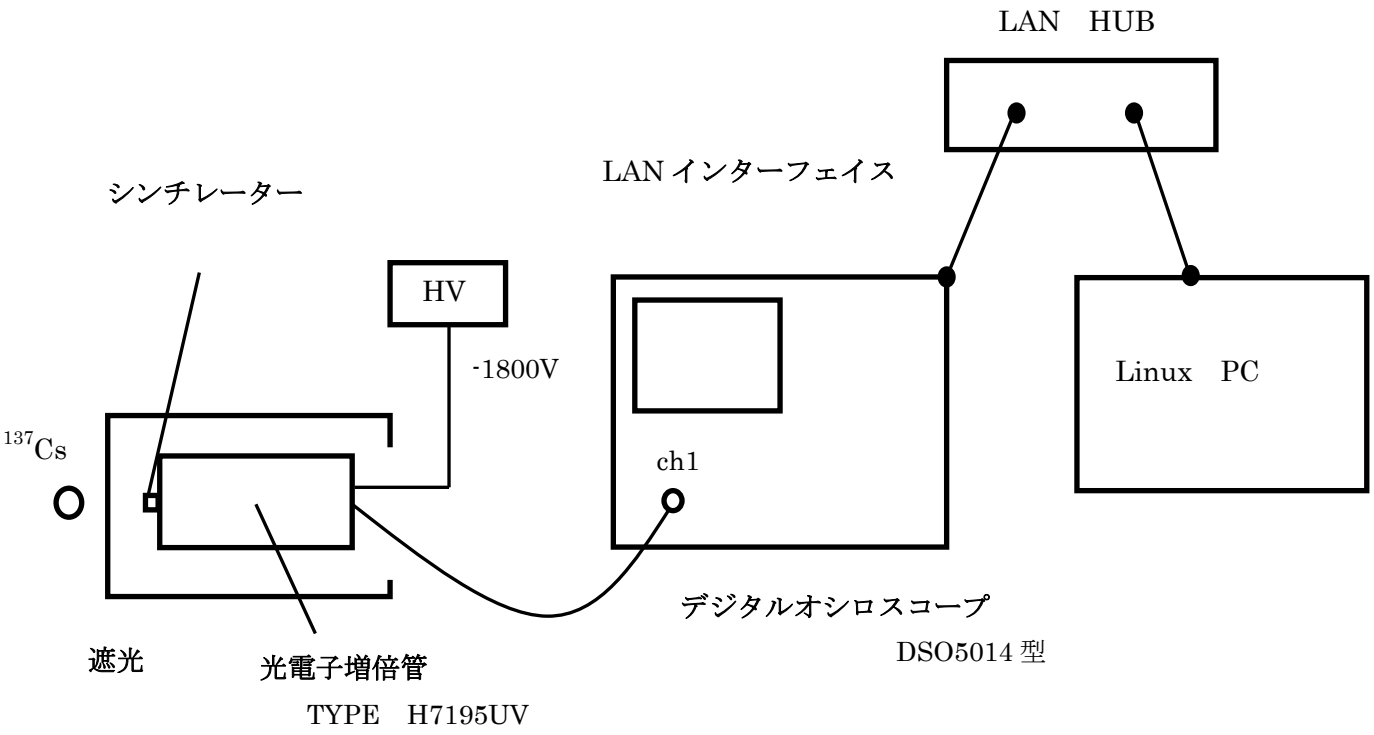


図 5-1：セットアップ図

手順

[1]データ収集用プログラムにより 15000 イベントのデータの書き出しを行い、Raw Data Ⅰを得る。

[2]解析用プログラム①により Raw Data Ⅰを文字列として読み込み、整数に直す。また、各イベントの積分値を計算する。

[3]積分値をヒストグラムで表すことにより、波高分布を得る。

[4]解析用プログラム②により積分値が 1000 以上のイベントのみの Raw DataⅡを作成する。

[5]解析用プログラム③によりパルスの数値データ(Raw DataⅡ)を平均する。

[6]平均したデータをプロットすることにより波形を得る。

[7]解析用プログラム④により[5]の数値の対数をとる。

[8][6]をプロットし、一次関数で Fitting する。

(解析用プログラムは付録参照)

5-1 得られた波高分布

5-1-1 クリスタルのバックグラウンド

線源をあてずに測定した結果得られた波高分布を示す。二種類のシンチレーターLSO,GAGG を使用したとき、図 5-2、図 5-3 が得られた。LSO を使用した場合の波高分布から、Lu の放射性同位体によるバックグラウンド源があることがわかる。これに対して GAGG 使用した場合はクリスタルにバックグラウンドがないのでパルスが非常に静かであることがわかる。

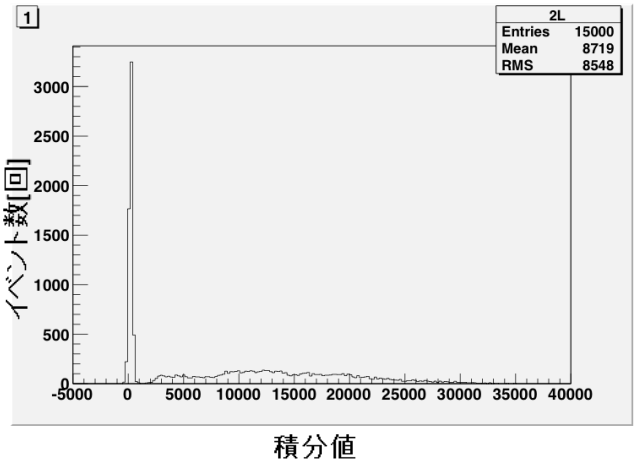


図 5-2：LSO を使用したときの波高分布

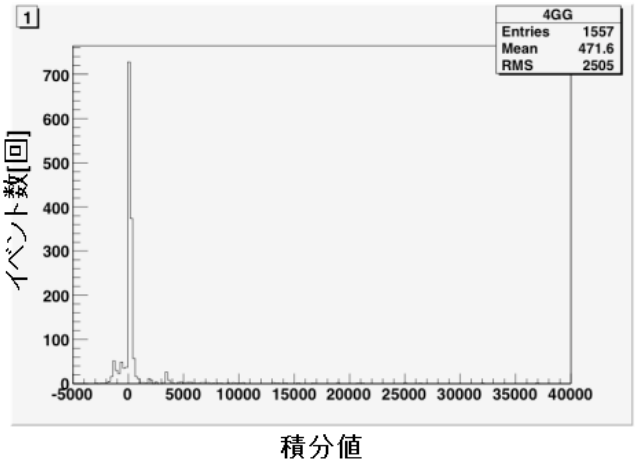


図 5-3：GAGG を使用したときの波高分布

5-1-2 チェレンコフ光、シンチレーション光

図 5-1 のようなセットアップにより、^{137}Cs 線源をあてた場合の波高分布を得ると、図 5-4、図 5-5 となった。図 5-4 で、積分値が 1000 以下のイベントは、γ線がシンチレーターを通らずに直接光電子増倍管の受光部窓ガラスに入射し、光電効果またはコンプトン散乱することにより発生したチェレンコフ光と考えられる。また、積分値が 1000~25000 のイベントはシンチレーション光によるものと考えられる。シンチレーション光によるもののうち、積分値が 1000~17000 のイベントはコンプトン散乱、17000~25000 のイベントは光電効果による波高分布で、光電効果によるもののイベント数が一番多いところを光電ピークと呼ぶ。また、チェレンコフ光によるものと考えた部分には、光電子増倍管自体のダークパルスも混じっている可能性がある。GAGG を使用した場合の波高分布も同様に考えられる。

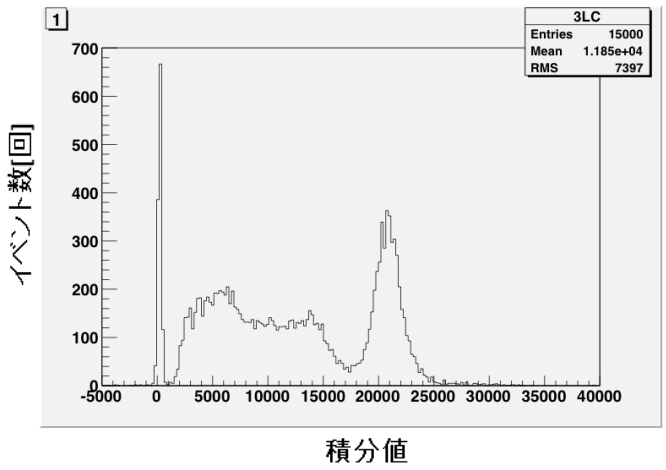


図 5-4：LSO を使用したときの波高分布

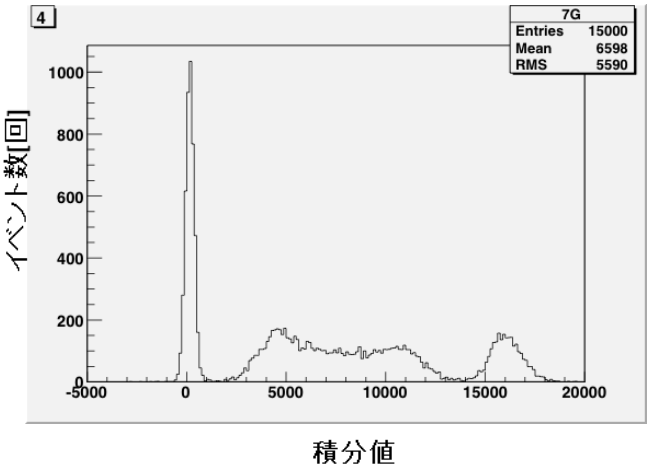


図 5-5：GAGG を使用したときの波高分布

5-2 光量の測定

波高分布を得た際と同一の光電子増倍管を用いて、印加電圧を同じにしたときに、オシロスコープのレンジを切り替えて測定した場合でもデータを正しく比較できるよう、光量を信号電荷に変換した。
信号電荷 Q[C]は電圧 V、抵抗 R、時間 t を用いると

$$Q = \frac{V}{R} t$$

により計算できる。
R は光電子増倍管のターミネータ 50[Ω]、t はサンプリングの時間間隔⊿t=0.5[ns]とし、V は以下の式で求められる。

$$V=\sum_i Vi$$
$$=C\sum_i\{h0 - h(i)\}$$

図 5-6 で示すように、本実験ではトリガーするタイミングを横軸フルスケールの 10%の位置(100 サンプル目)に設定したので、最初の 70 サンプルの平均をゼロ点(h_0)とし、i 番目にサンプルされた波形データを h(i)とする。また、波形データを電圧に換算する係数 C は 200mV/div の場合、1600mV が 8bit フルスケールに対応するため、6.25mV/LSB となる。

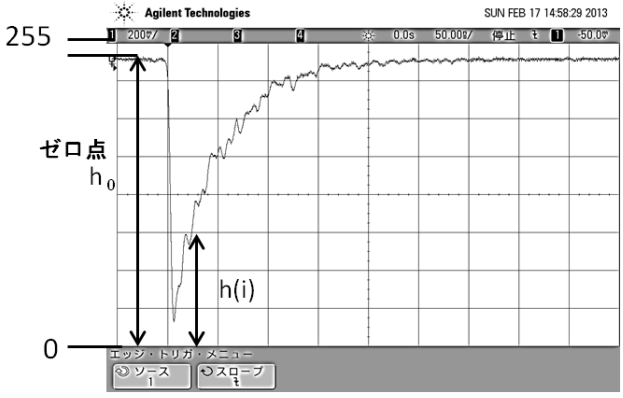


図 5-6：デジタルオシロスコープで得られた波形の例

5-2-1LSOの結果

LSOについて、図5-4の光電ピークの部分を拡大すると、図5-7のような波高分布を得た。光電ピークの積分値を20700と読み取り計算すると、1293.75[pC]と求められる。

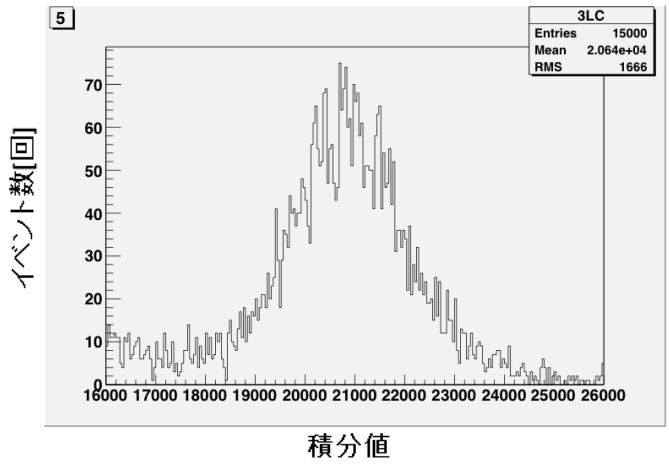


図 5-7：光電ピーク部分の波高分布

5-2-2GAGGの結果

GAGGについて、図5-5の光電ピークの部分を拡大すると、図5-8のような波高分布を得た。光電ピークの積分値を15870と読み取り計算すると、991.88[pC]と求められる。

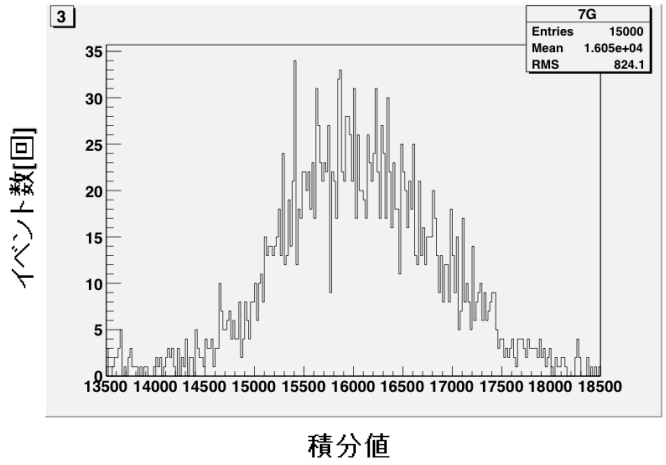


図 5-8：光電ピーク部分の波高分布

5-3 時定数の測定

5-3-1LSO の結果

手順に示した Raw Data Ⅱにある数値データを平均した。この波形データの時間発展を図 5-9 に示す。

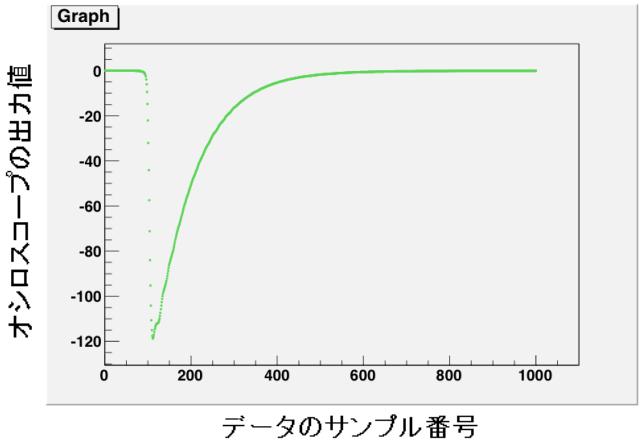


図 5-9：波形データの時間発展

立ち下がり部分が指数関数で e^{ax+b} と与えられるとき $\propto e^{-t/\tau}$ とみて時定数が求められる。サンプリング間隔は 0.5[ns]としているので t=0.5x[ns]より τ=0.5/a [ns]を計算する。
本実験ではオシロスコープの出力値の対数をとり、一次関数 y=ax+b で Fitting を行った。

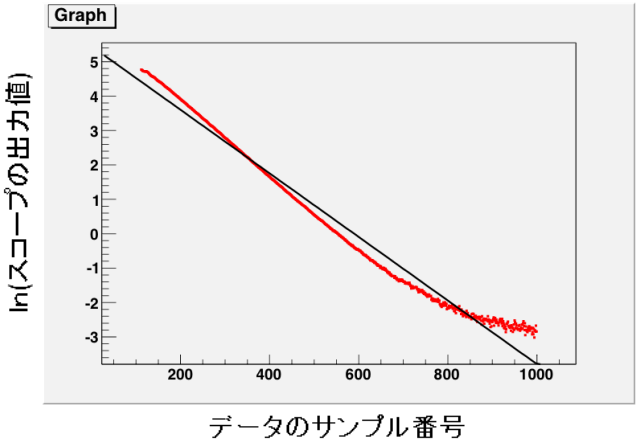


図 5-10：一次関数で Fit した結果

すると図 5-10 に示すように、直線では Fit が良くない。そこで速い成分が支配的な領域と遅い成分が支配的な領域に分けて時定数を得ることにした。

まず、速い成分が支配的な領域として、データのサンプル番号 150～400 の範囲で Fit した結果を図 5-11 に示す。

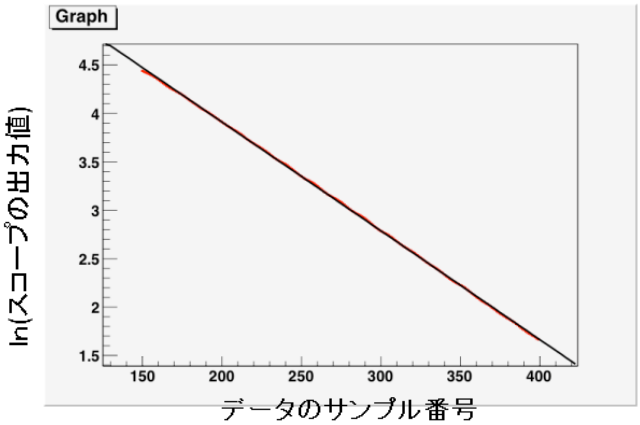


図 5-11：図 5-10 のサンプル番号 150～400 で Fit した結果

Fit した結果

a=-0.00112402

と得られたので

τ=44.48≈45[ns]

と求まった。

また、遅い成分が支配的な領域として、データのサンプル番号 550～750 の範囲で Fit した結果を図 5-12 に示す。

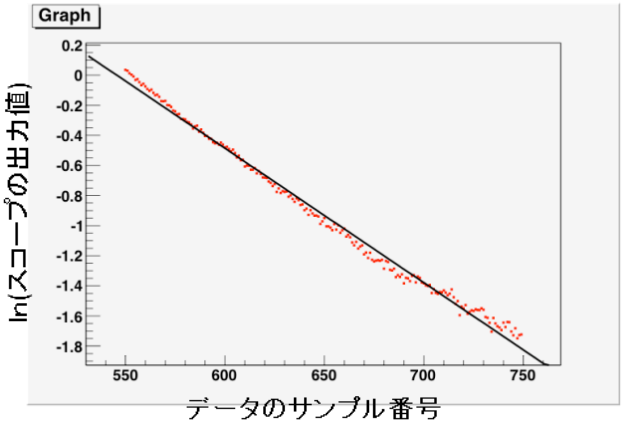


図 5-12：図 5-10 のサンプル番号 550～750 で Fit した結果

Fit した結果

a =-0.00893018

と得られたので

τ=55.99≈56[ns]

と求まった。

5-3-2GAGG の結果

LSO と同様に、Raw Data Ⅱ にある数値データを平均した。この波形データの時間発展を図 5-13 に示す。また、オシロスコープの出力値の対数をとり、一次関数 y=ax+b で Fitting を行った結果を図 5-14 に示す。

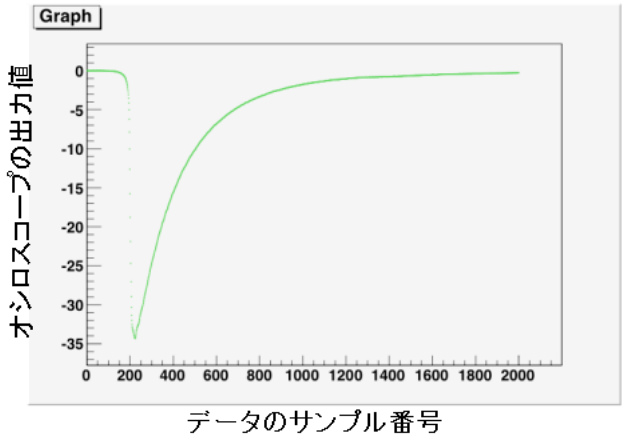


図 5-13：波形データの時間発展

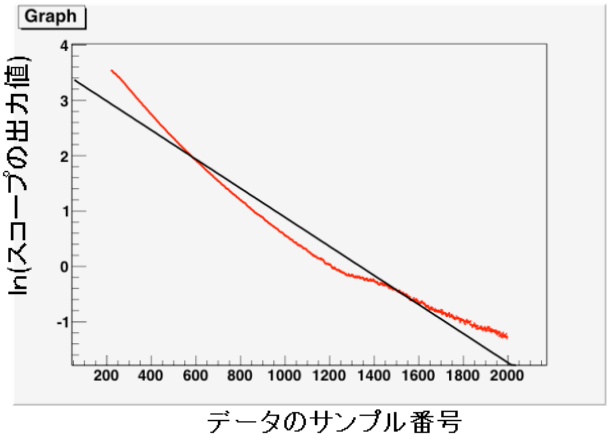


図 5-14：一次関数で Fit した結果

LSO と同様、直線では Fit が良くないので、速い成分が支配的な領域と遅い成分が支配的な領域に分けて時定数を得ることにした。

まず、速い成分が支配的な領域として、データのサンプル番号 150～600 の範囲で Fit した結果を図 5-15 に示す。

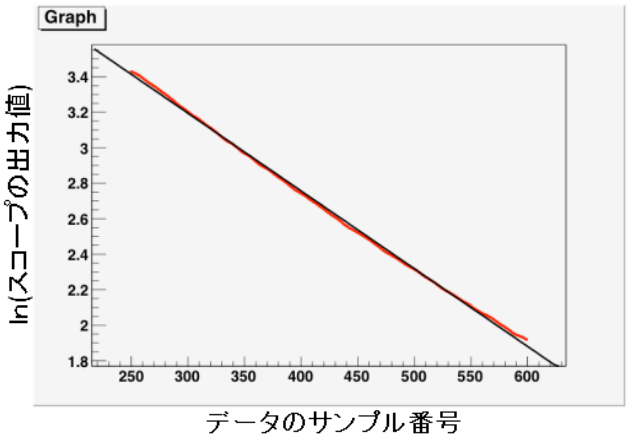


図 5-15：図 5-14 のサンプル番号 250～600 で Fit した結果

Fitting の結果

a =-0.00438095

と得られたので

τ=114.13≈114[ns]

と求まった。

また、遅い成分が支配的な領域として、データのサンプル番号 1400～2000 の範囲で Fit した結果を図 5-16 に示す。

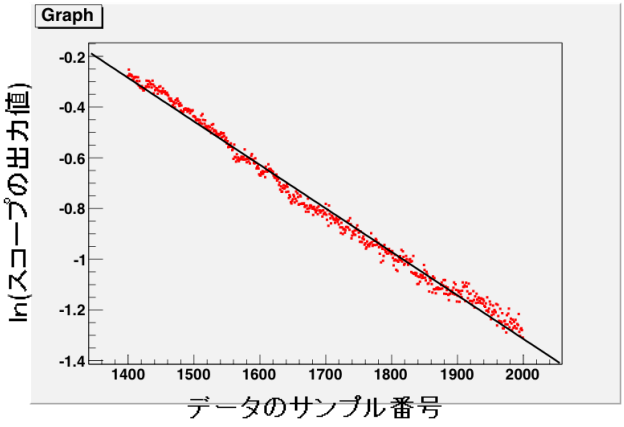


図 5-16：図 5-14 のサンプル番号 1400～2000 で Fit した結果

Fitting の結果

a =-0.00171525

と得られたので

τ=291.50≈292[ns]

と求まった。

第 6 章　まとめ

- デジタルオシロスコープの波形サンプリングデータを LAN 経由でパソコンに読み出すことができた。
- 最大 2Gsamples/sec のサンプリングによりチェレンコフ光（光電子増倍管のダークパルスの場合もある）とシンチレーション光を明瞭に分離できた。
- 光量を計算すると、GAGG よりも LSO の方 30%程度多いという結果になった。
- 時定数を計算すると、GAGG(100ns)よりも LSO (50ns)の方が減衰時間が短くなった。
- LSO は Lu によるバックグラウンドがあるのに対して GAGG は無いので信号が見やすい。

6-1 考察

・時定数を求める際に、部分的に範囲を区切って、２つの範囲での直線の Fitting により時定数を求めたが、理想的には指数関数の和の形で Fitting する関数を用意するべきである。

・表 3-1 から LSO より GAGG の方が発光量は多いが、光量測定の結果 LSO は 1300pC、GAGG は 1000pC と得られ、LSO の方が多かった。しかし、使用した光電子増倍管はバイアルカリ光電面で、シンチレーターの特性は表 3-1 より LSO と GAGG の発光波長はそれぞれ 420ns，520ns で表 6-1 を見ると LSO の発光波長では量子効率が大きいので、本実験の測定結果は大きくずれていないと考えられる。

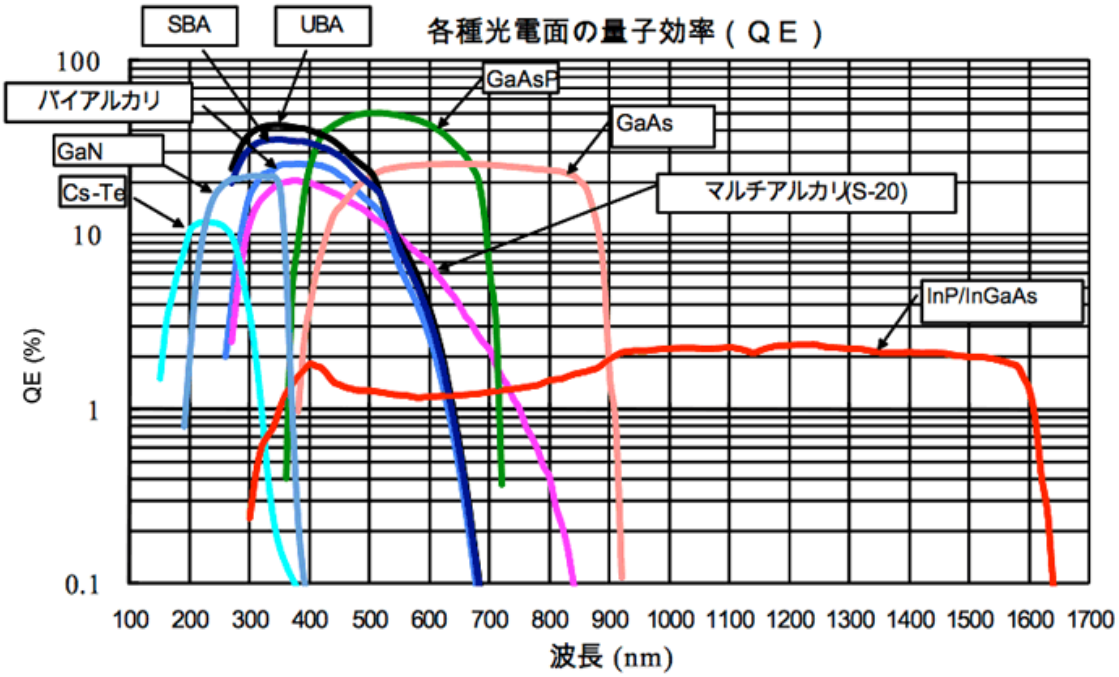


表 6-1：光電子増倍管の各種光電面の量子効率

6-2 参考文献

・奈良女子大学理学部物理科学科　2012 年度卒業生
栗林和加 立川晶絵 峰村さつき
「2012 年度卒業論文　ポジトロニウム消滅ガンマ線対の検出と時間分解能の研究」
・奈良女子大学理学部物理科学科　2006 年度卒業生
玉木智子　平井珠生
「2006 年度卒業論文　石英ガラスで発生するチェレンコフ光を用いた荷電粒子の検出」
・浜松ホトニクス株式会社
光電子増倍管のカタログ

6-3 謝辞

本研究を進めるにあたり、指導教官である宮林先生をはじめ、林井先生、諸先輩方には丁寧かつ熱心な指導を賜りました。ここに感謝の意を表します。本当にありがとうございました。

付録 解析用プログラム

波高分布や波形を得るために、解析用のプログラムを使って、得られた Raw Data を読み込み、計算を行う。

データ解析用プログラム①

Raw Data を文字列として読み込み、整数に変換する。トリガーするタイミングは時間のスケールの 1division(データ 100 個分)に設定した。よって安定していると考えられる 70 個目までのサンプルで平均値を求め、それをゼロ点とする。ゼロ点から、71 個目から 1000 個目までのサンプルの値をそれぞれ引いたものを積算して、積分値を求める。

```
#include <stdio.h>
#include <iostream.h>
#include <stdlib.h>

char header1[40], header2[40], buffer[10];
int iev, nbytes; int i;
int m;
float x,y;
int wf[1000];

FILE* fp;
FILE* fpout;

int main(int argc, char *argv[]){
  if( argc < 3 ) std::cout<<"./read_test input_file output_file."<<std::endl;

  fp = fopen(argv[1],"r");

  fpout = fopen(argv[2],"w");

  fscanf(fp,"%s",header1);
  fscanf(fp,"%s",header2);
  printf("%s %s¥n",header1,header2);
```

raw data のファイルをオープンする

output file をオープンする

x,y の表示

ここからイベントループ
(15000 イベント繰り返し)

```
   while(fscanf(fp,"%s",buffer)!=EOF){
     iev=atoi(buffer); printf("%d¥n",iev);

     fscanf(fp,"%s",buffer); nbytes=atoi(buffer);
     printf("%d¥n",nbytes);

     for(i=0; i<nbytes; i++){
       fscanf(fp,"%s",buffer); wf[i]=atoi(buffer);
}

      double sum = 0.0;
      double ave = 0.0;
      const int nped = 70;
      for(i=0; i<nped; i++){
         sum = sum + (double)wf[i];
      }
      ave = sum / 70.0;
         std::cout<<"ave"<<ave<<std::endl;

      double sum2 = 0.0;

       for(i=nped; i<nbytes; i++){
           sum2 = sum2 + (ave - (double)wf[i]);
            }

            fprintf(fpout,"%f¥n",sum2);
   }
}
```

イベント番号、サンプル数の表示

1000 のサンプルデータを整数に変換

70 までのサンプルでゼロ点の計算
ave として表示

ゼロ点からそれぞれの値を引いて
積分値 sum を求める

データ解析用プログラム②

波高分布を見ると、シンチレーション光によるものは積分値が 1000 以上のイベントだと考えられる。そこで、Raw Data のうち 1000 以上のイベントのデータのみをあつめたファイルを作成する。

```
#include <stdio.h>
#include <iostream.h>
#include <stdlib.h>

char header1[40], header2[40], buffer[10];
int iev, nbytes; int i; int m;
float x,y;
int wf[1000];

FILE* fp;
FILE* fpout;

int main(int argc, char *argv[]){
  if( argc < 3 ) std::cout<<"./read_test input_file output_file."<<std::endl;

  fp = fopen(argv[1],"r");

  fpout = fopen(argv[2],"w");

  fscanf(fp,"%s",header1);
  fscanf(fp,"%s",header2);

  while(fscanf(fp,"%s",buffer)!=EOF){
      iev=atoi(buffer);

      fscanf(fp,"%s",buffer); nbytes=atoi(buffer);

       for(i=0; i<nbytes; i++){
      fscanf(fp,"%s",buffer); wf[i]=atoi(buffer);

    }
```

ここからイベントループ

イベント番号
サンプル数
サンプリング数 1000 のデータを整数に変換

```
        double sum = 0.0;
        double ave = 0.0;
        const int nped = 70;
        for(i=0; i<nped; i++){
           sum = sum + (double)wf[i];
        }
        ave = sum / 70.0;

        double sum2 = 0.0;
           for(i=nped; i<nbytes; i++){
              sum2 = sum2 + (ave - (double)wf[i]);
               }

               if(sum2 > 1000.0 ){
printf("%f¥n",sum2);
               fprintf(fpout,"%d¥n",iev);
               for(i=0; i<nbytes; i++){
                   if(i==(nbytes-1)){
                     fprintf(fpout,"%d¥n",wf[i]);
                  }else{
                     fprintf(fpout,"%d ",wf[i]);
                  }
                 }
              }
    }
}
```

積分値の大きさが 1000 以上の
シンチレーション光によるものだけを
必要なイベント数として
fprintf で表示。
また、その時の積分値を
printf で表示。

データ解析用プログラム③

時定数を求めるために、パルスの平均を求める。解析プログラム②で得た Raw Data について 1000 サンプルの数値データのそれぞれを平均する。このときゼロ点補正を行い、波形データの時間発展は ROOT でプロットする。

```
#include <stdio.h>
#include <iostream.h>
#include <stdlib.h>

char header1[40], header2[40], buffer[10];
int iev, nbytes; int i;
int s;
double m; double mm;
float x,y;
int wf[1000];
int wg[1000];
double wh[1000];
FILE* fp;

int main(int argc, char *argv[]){
    if( argc < 2 ) std::cout<<"./read_test input_file output_file."<<std::endl;

    fp = fopen(argv[1],"r");

  m = 0.0;
  s = 1000;
  for(i=0; i<s; i++){
    wg[i]=0;
    wh[i]=0.0;
  }
```

wg[i]、wh[i]
に初期値０を与える

ここからイベントループ

```
  while(fscanf(fp,"%s",buffer) !=EOF){
        iev=atoi(buffer);

      m = m + 1.0;
     for(i=0; i<s; i++){
```

足し上げたイベント数をｍで数える

wg[i]に　全イベント分の wf[i]の値を足していく

```
        fscanf(fp,"%s",buffer); wf[i]=atoi(buffer);
        wg[i]=wg[i]+wf[i];
      }
     }

   for(i=0; i<s; i++){
      wh[i]=wg[i]/m;
       }

   double sum = 0.0;
   double ave = 0.0;
   mm = 0.0;
   for(i=0; i<70; i++){
      sum=sum+wh[i];
   }
       ave=sum/70.0;

   for(i=0; i<s; i++){
      wh[i]=wh[i]- ave;
      mm = mm + 1.0;
      printf("%f %f¥n",mm,wh[i]);
   }

}
```

イベント数ｍで割って
サンプルしたデータの平均
を求める

最初の 70 サンプルでゼロ点を求め、
ゼロ点を 0 として補正した
1000 サンプルを表示

また、その時に
ROOT でプロットするために
ｍｍでサンプル番号を数え
ｘ、ｙ　として表示する

解析用プログラム④

直線を Fit させて時定数を求める。プロットした波形の縦軸であるオシロスコープの出力値の立ち下がり部分について対数をとり、直線として ROOT を用いて一次関数で Fitting を行った。また、立ち下がりが速い成分と遅い成分の和になっていると考えられるので横軸のサンプル番号の範囲を変えて速い成分が支配的な領域と遅い成分が支配的な領域についても同様に Fitting を行った。

```
#include <stdio.h>
#include <iostream.h>
#include <stdlib.h>
#include <math.h>

char header1[40], header2[40], buffer[10];
int iev, nbytes; int i;
int s;
double m; double mm;
int wf[1000];
int wg[1000];
double wh[1000];
double y[1000];
FILE* fp;

int main(int argc, char *argv[]){
  if( argc < 2 ) std::cout<<"./read_test input_file output_file."<<std::endl;

  fp = fopen(argv[1],"r");



  s = 1000;
  m = 0.0;
  mm = 111.0;
  for(i=0; i<s; i++){
    wg[i]=0;
    wh[i]=0.0;
  }
```

ここからイベントループ

```
  while(fscanf(fp,"%s",buffer) !=EOF){
```

```
      m = m + 1.0;
      for(i=0; i<s; i++){
       fscanf(fp,"%s",buffer); wf[i]=atoi(buffer);
       wg[i]=wg[i]+wf[i];
     }
    }

  for(i=0; i<s; i++){
     wh[i]=wg[i]/m;
  }

  double sum = 0.0;
  double ave = 0.0;
  for(i=0; i<70; i++){
     sum=sum+wh[i];
  }
      ave=sum/70.0;

  for(i=112; i<1000; i++){
     wh[i]= wh[i]-ave;
     y[i]= log(-wh[i]);
    mm = mm + 1.0;
        printf("%f %f¥n",mm,y[i]);
        }

}
```

プロットのピーク以降の立下り部分について、平均のイベントのゼロ点補正をした wh[i]の対数をとり y[i]　で表示。
この時もプロットと同様に ROOT の計算用にｍｍを用いて
ｘ、ｙ　表示とする