Relazione di Spettroscopia

Camera di Bragg

Francesco Forcher

Università di Padova, Facoltà di Fisica
francesco.forcher@studenti.unipd.it
Matricola: 1073458

Enrico Lusiani

Università di Padova, Facoltà di Fisica
enrico.lusiani@studenti.unipd.it
Matricola: 1073300

Laura Buonincontri

Università di Padova, Facoltà di Fisica
laura.buonincontri@studenti.unipd.it
Matricola: 1073131

2 luglio 2016

Sommario

Studio della camera di Bragg, in particolare risoluzione energetica, misura del background e funzione range:pressione

Indice

I. Schema Circuiti

Da aggiungere, forse l'immagine che c'è anche nelle istruzioni?

II. Parte I

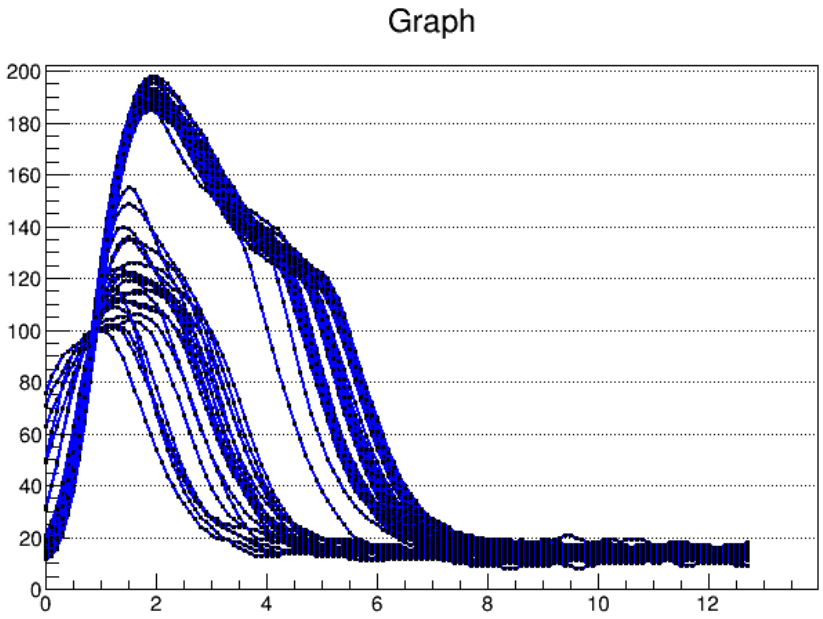
II.I Prima misura delle particelle alfa

Sono stati settati gli strumenti, a pressione 600 mb. Si è mantenuto lo Shaping Time a 0.25-0.5 μs, dopo aver osservato il comportamento. E' stata regolata l'amplificazione in modo da mantenere il picco attorno ai 3V. E' stato impostato il trigger in modo tale che fosse circa a metà altezza del picco sull'oscilloscopio.

Si è verificato che il numero di segnali spuri fosse inferiore al 30% del totale, facendo il grafico dei picchi e calcolando l'integrale dei segnali a bassa energia.

E' stato quindi acquisito il primo set di dati (circa 3000 eventi)

Grafico 1 Grafico segnali a 600mb



E' stato acquisito anche un set di dati con meno eventi per stimare la baseline. In particolare, è stato calcolato il centroide del picco della baseline, da inserire nella macro al posto del valore di default.

E' stato fatto il grafico degli integrali ed è stato calcolato il centroide del picco della baseline, da inserire nella macro al posto del valore di default.

L'analisi dei dati che segue è stata effettuata utilizzando la macro fornite dal laboratorio, modificando il limite dei campioni da integrare e inserendo il valore della baseline appena stimato. Il limite dei campioni a 600 mb è stato posto uguale a 90.

Grafico 2 Grafico segnali baseline

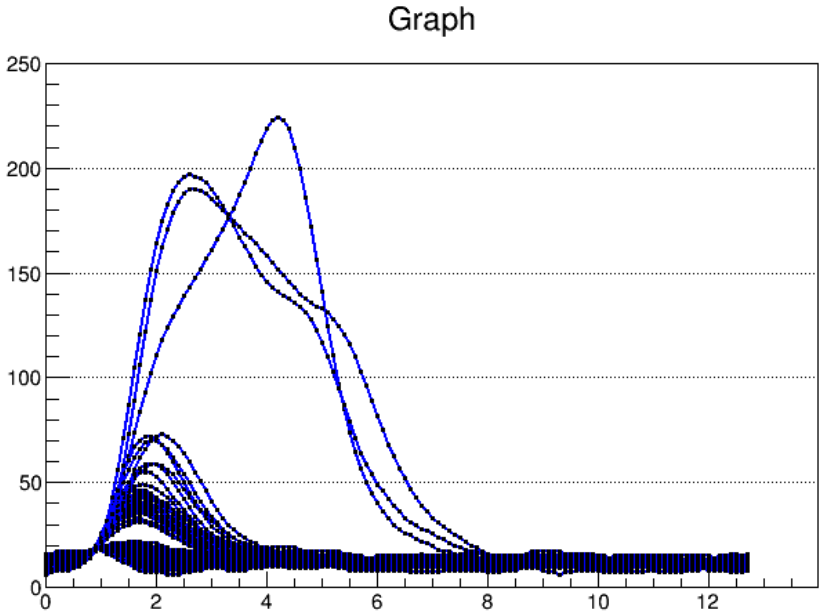


Grafico 3 Picco della baseline

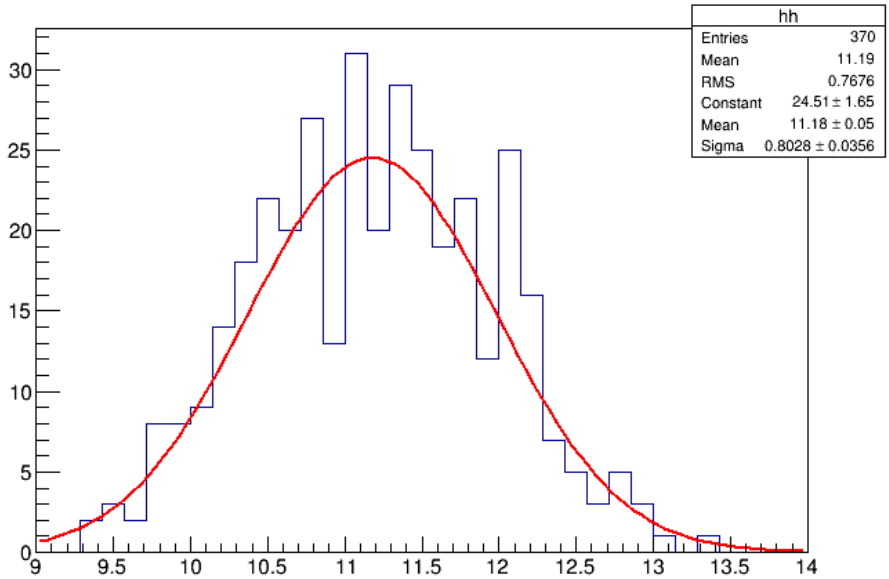


Grafico 4 Grafico integrale

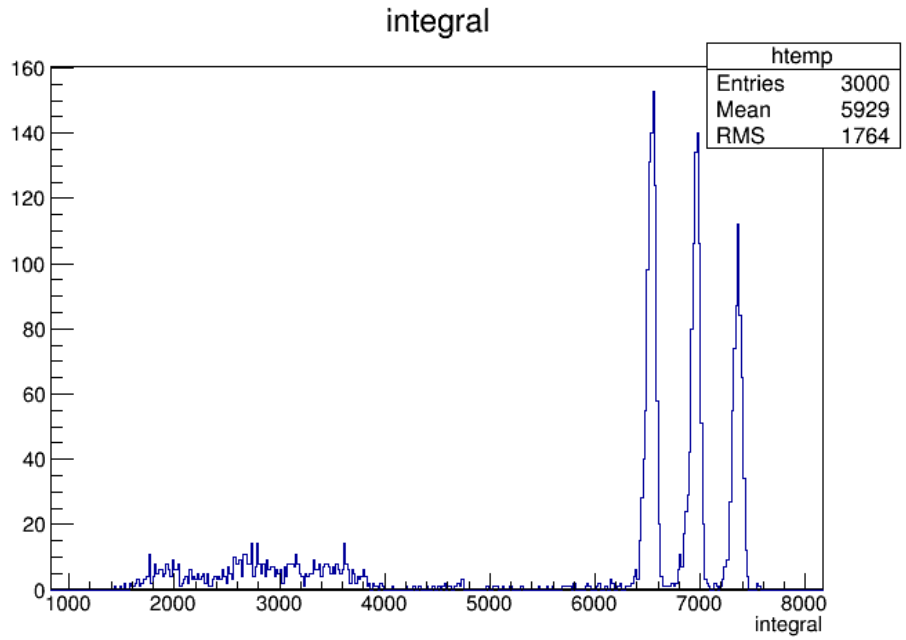


Grafico 5 Grafico integral:ev

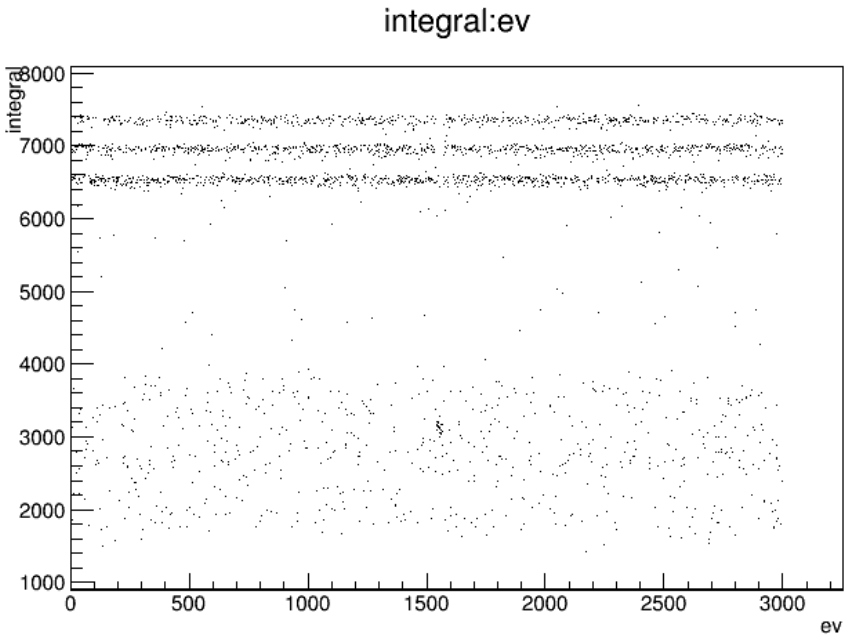


Grafico 6 Grafico integral:vmax

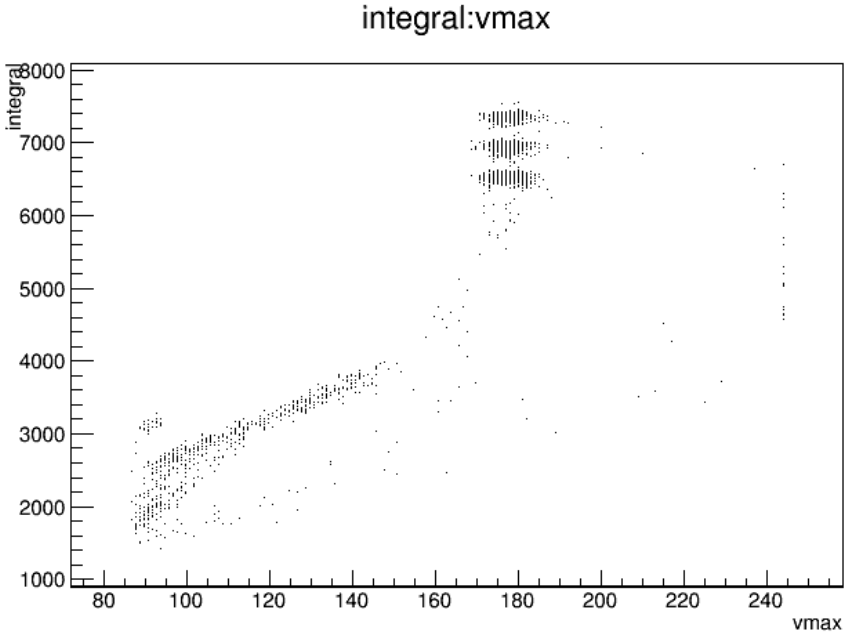


Grafico 7 Grafico vmax

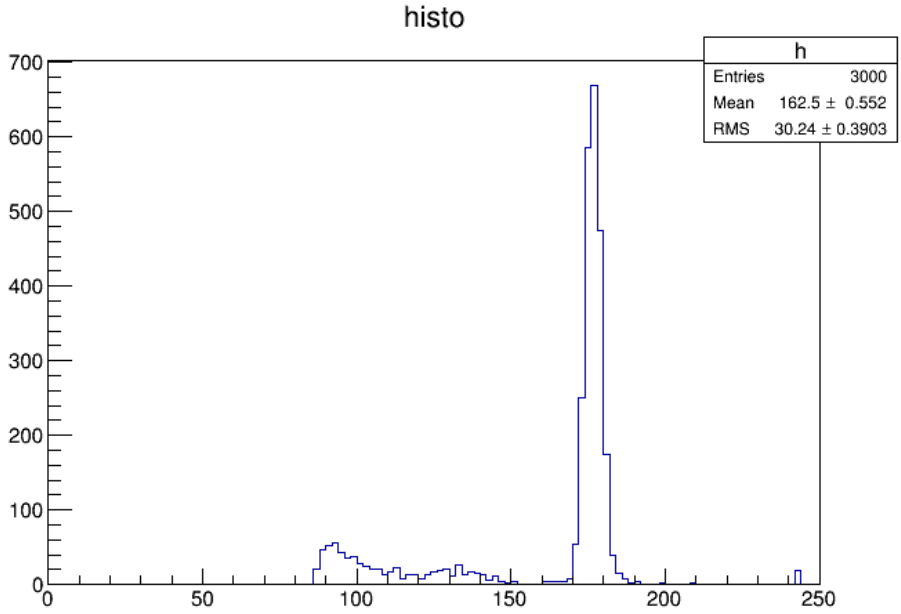
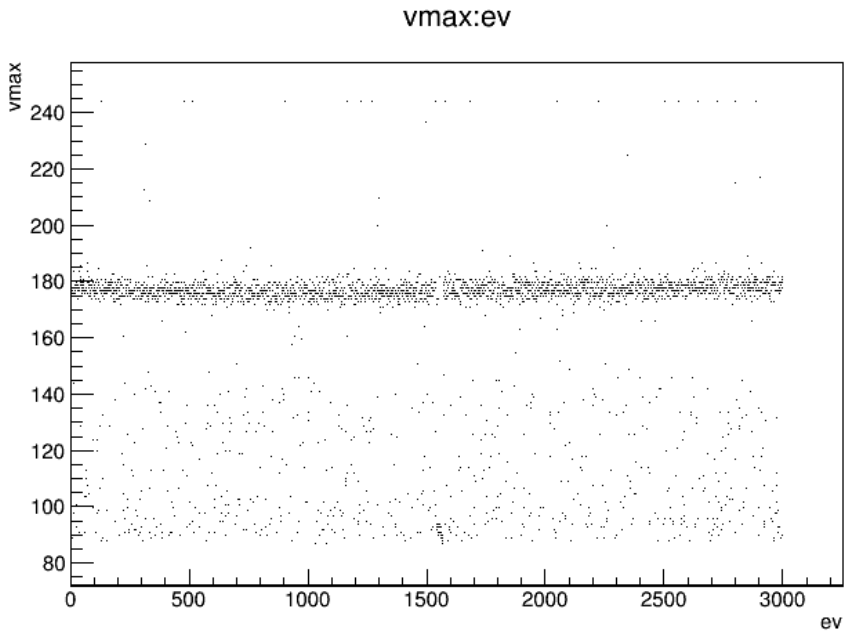


Grafico 8 Grafico vmax:ev



II.II Risoluzione energetica

Per il calcolo della risoluzione energetica, abbiamo per prima cosa trovata la relazione tra integrale ed energia, ipotizzandola lineare. Per fare ciò abbiamo calcolato l'energia teorica di ciscun picco, facendo la media delle energie dei decadimenti alfa, pesate sulla loro probabilità. Poi abbiamo calcolato l'integrale relativo a ciascun picco, come centroide del picco nell'istogramma degli integrali, e il suo errore, l'RMS del picco. Da questi dati abbiamo proceduto ad un interpolazione dell'integrale in funzione dell'energia, ricavando così la funzione energia:integrale. Usando la funzione, abbiamo riscalato l'asse delle ascisse nell'istogramma degli integrali, in modo che mostrasse l'energia. Da questo nuovo istogramma abbiamo ricavato la risoluzione energetica, misurando l'RMS dei picchi e usandola per la formula

TODO fatemela bene voi la formula qui, non so fare le equazioni

$$\mathrm{risoluzione} = \frac{2.335 \cdot \sigma_E}{E}$$

TODO fate voi anche qui

$$q = 19.6792 \pm 19.5559$$

$$m = 1.2469 \pm 0.00354684$$

Grafico 9 Grafico Energia:Integrale

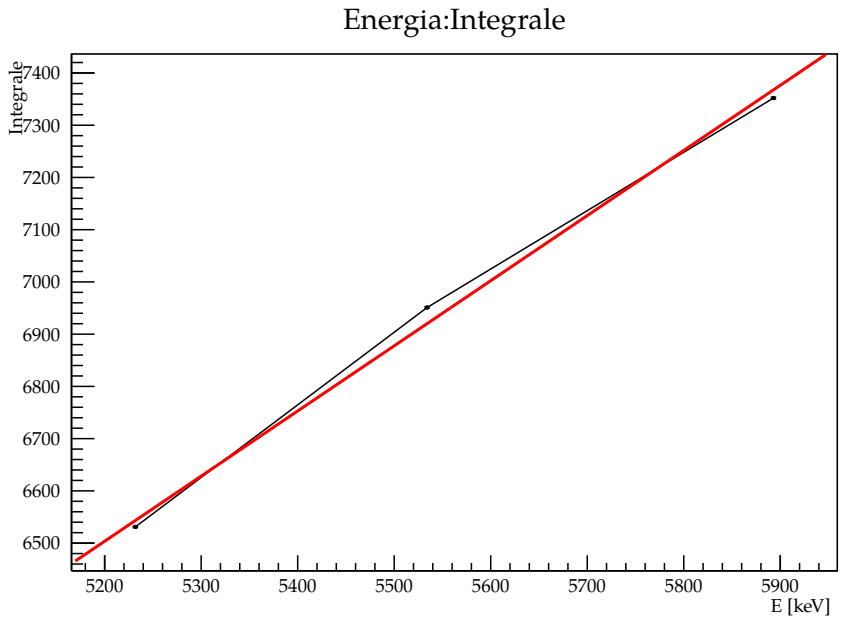
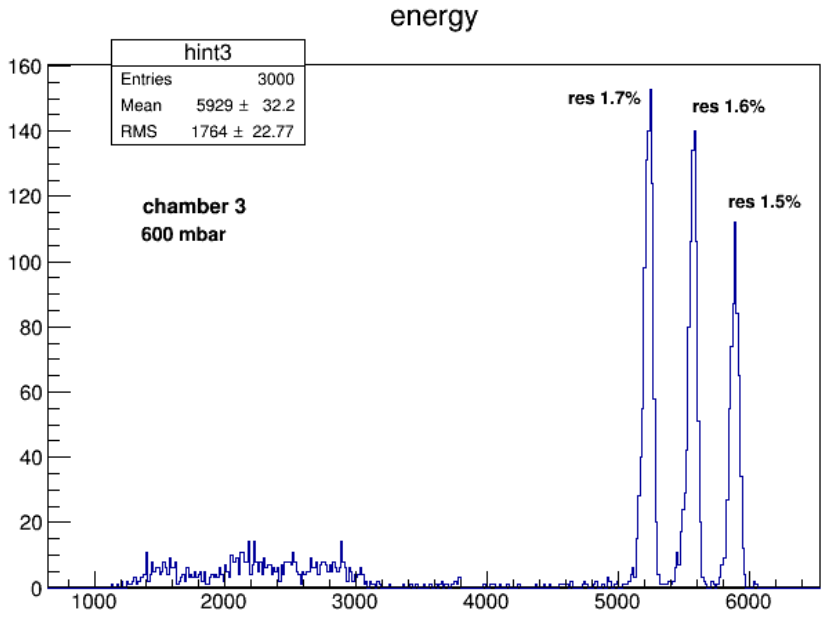


Grafico 10 Risoluzioni energetiche, grafico Energia:conteggio



III. Parte II

L'analisi dei dati che segue è stata effettuata utilizzando la macro fornite dal laboratorio, modificando il limite dei campioni e inserendo il valore della baseline stimato in precedenza.

III.I Misure a pressione 650mb

Grafico 11 Grafico segnali a 650mb

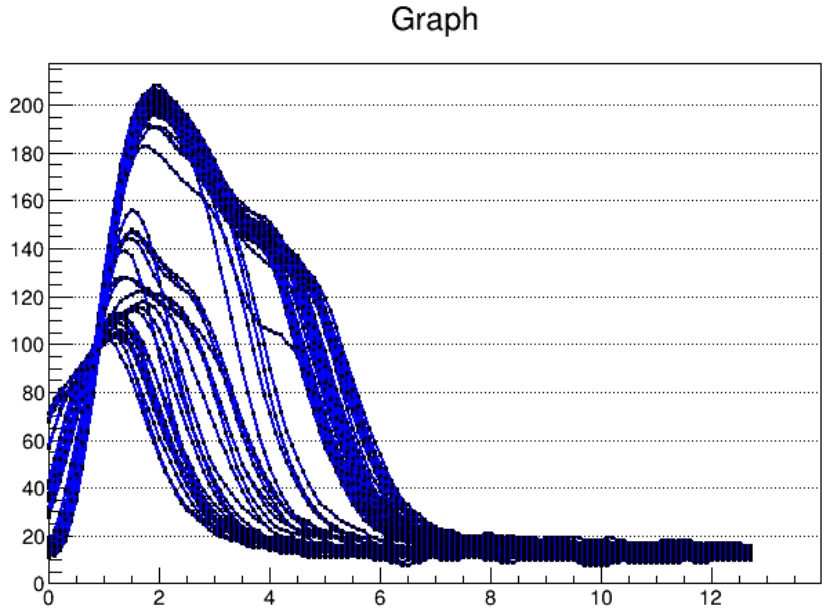


Grafico 12 Grafico integrale

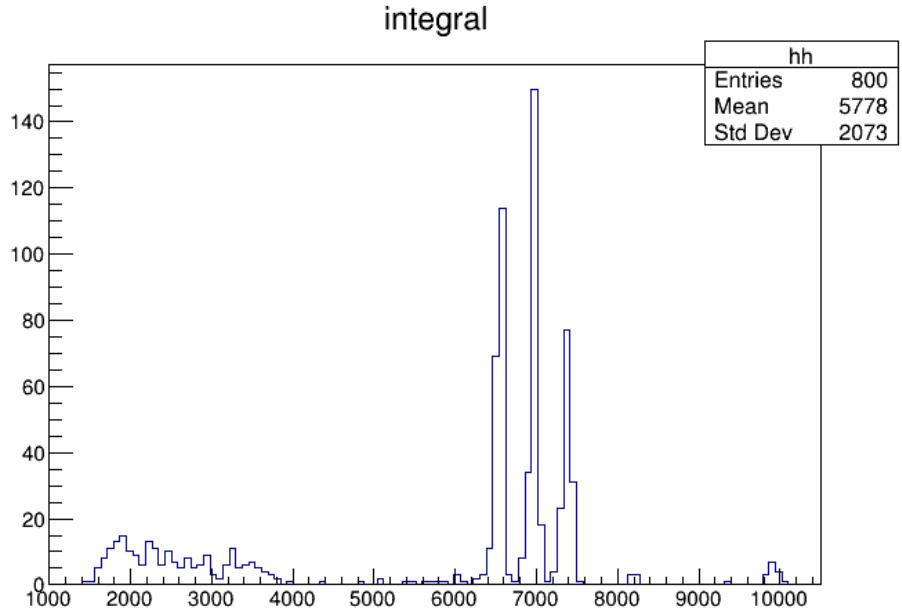


Grafico 13 Grafico integral:ev

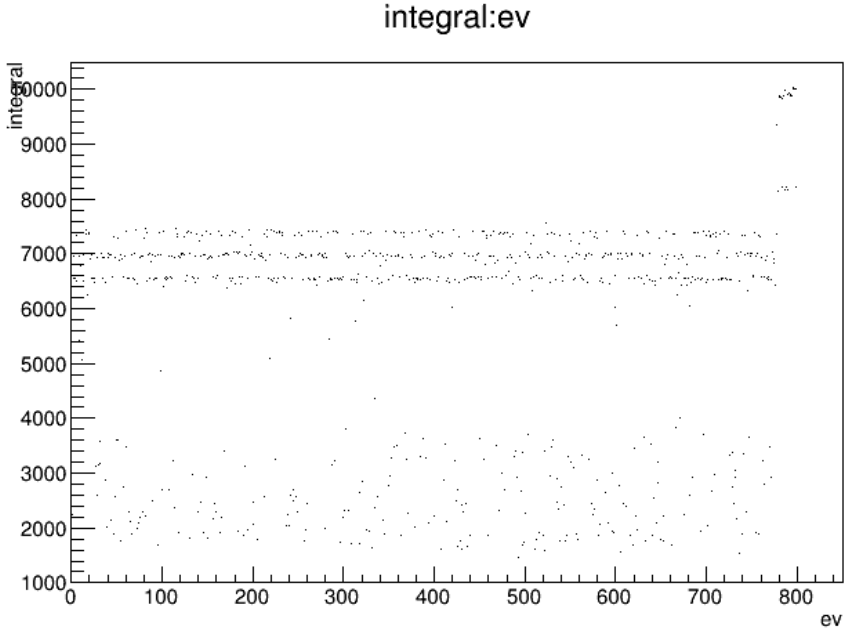


Grafico 14 Grafico integral:vmax

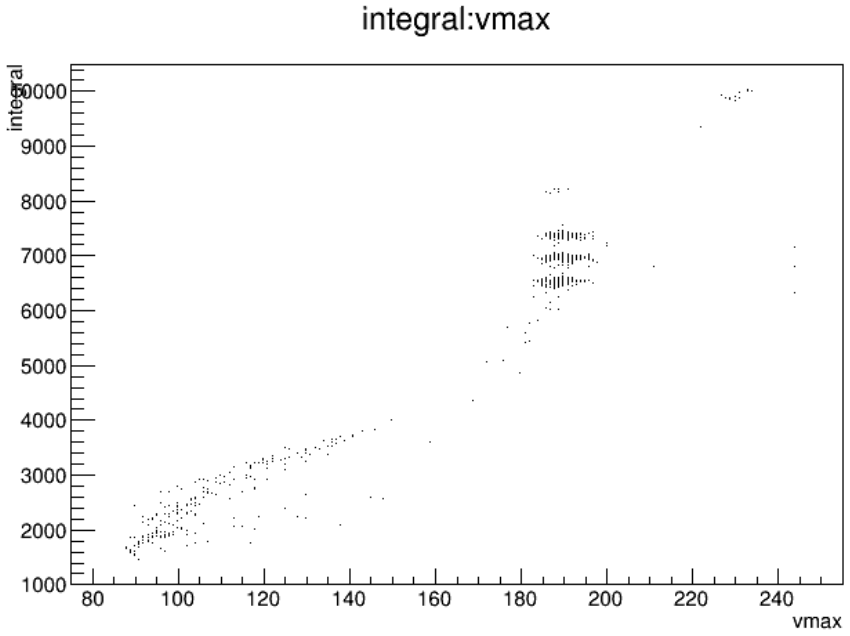


Grafico 15 Grafico vmax

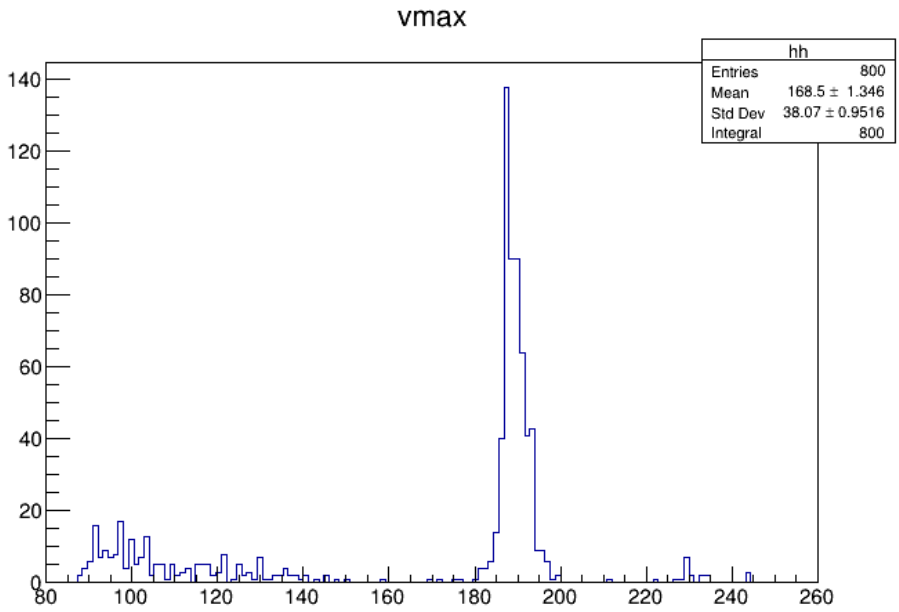
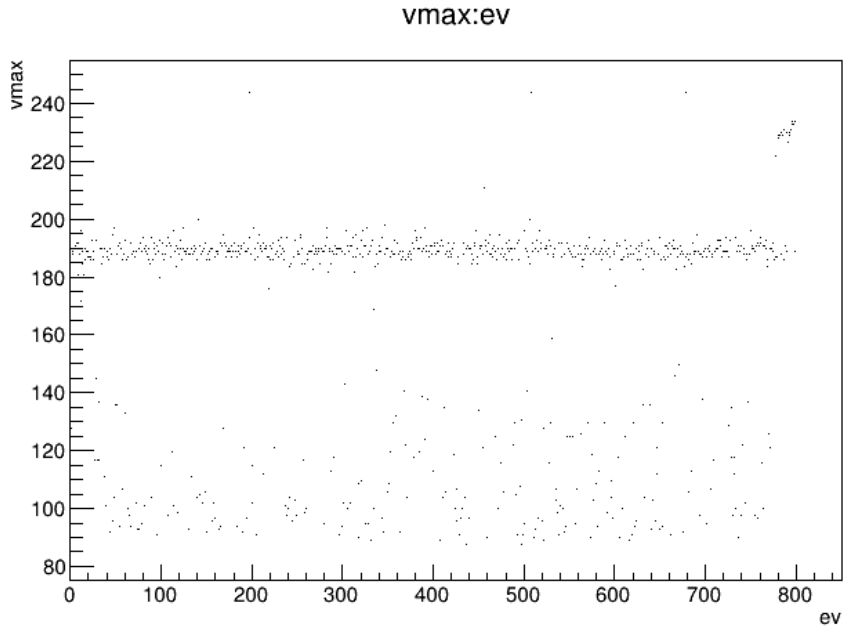


Grafico 16 Grafico vmax:ev



III.II Misure a pressione 550mb

Grafico 17 Grafico segnali a 550mb

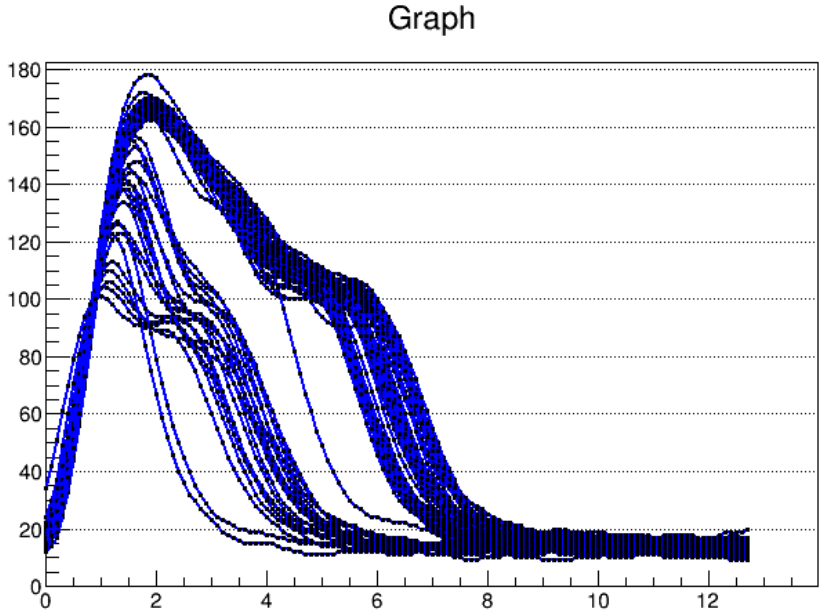


Grafico 18 Grafico integrale

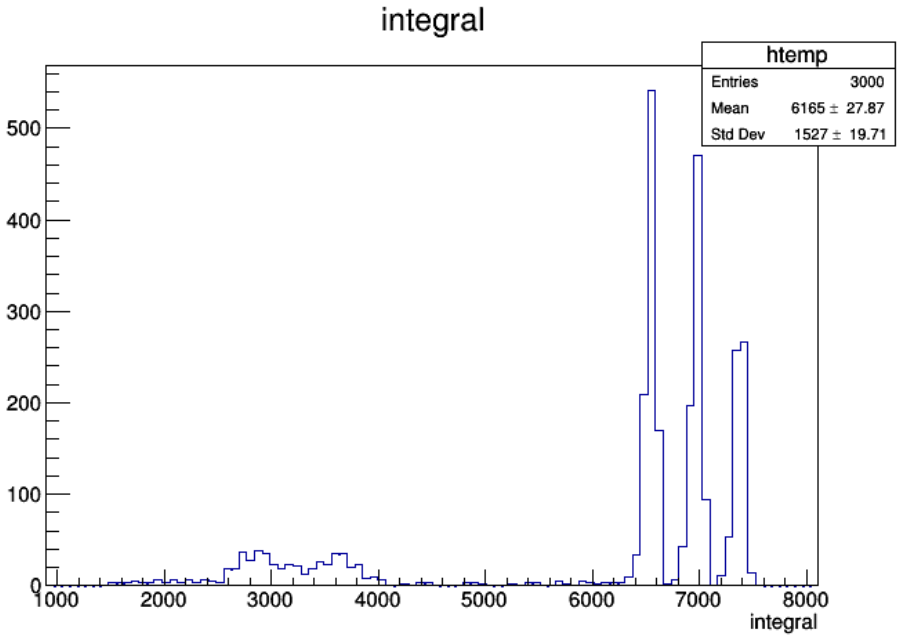


Grafico 19 Grafico integral:ev

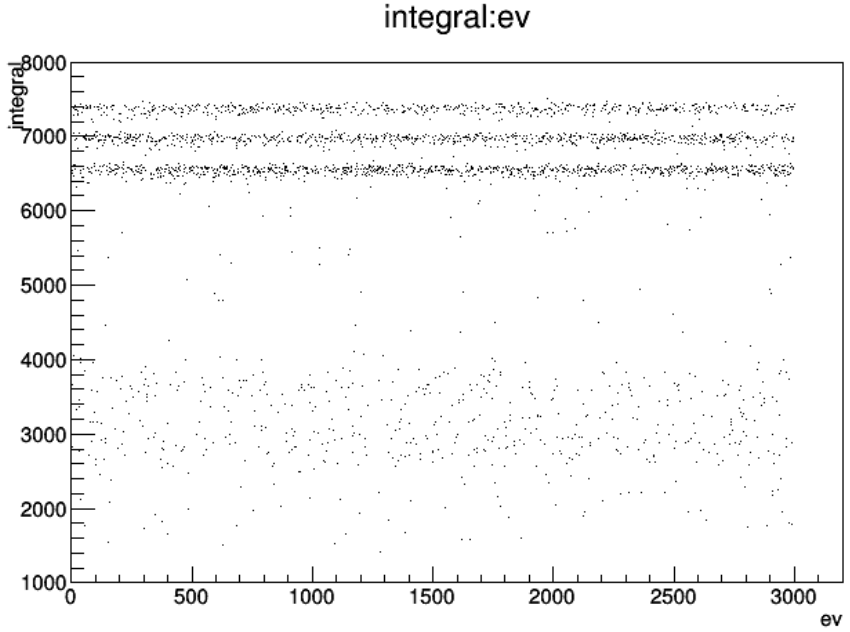


Grafico 20 Grafico integral:vmax

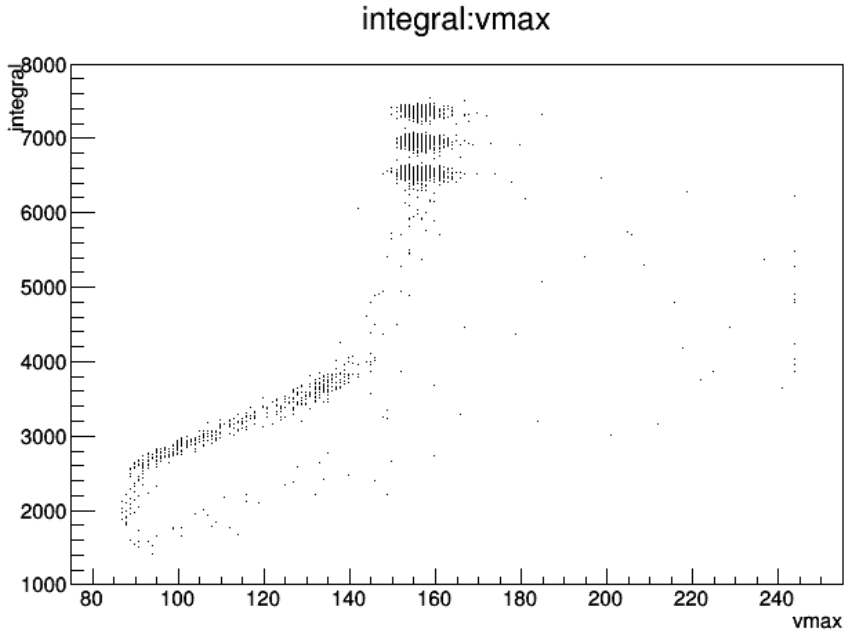


Grafico 21 Grafico vmax

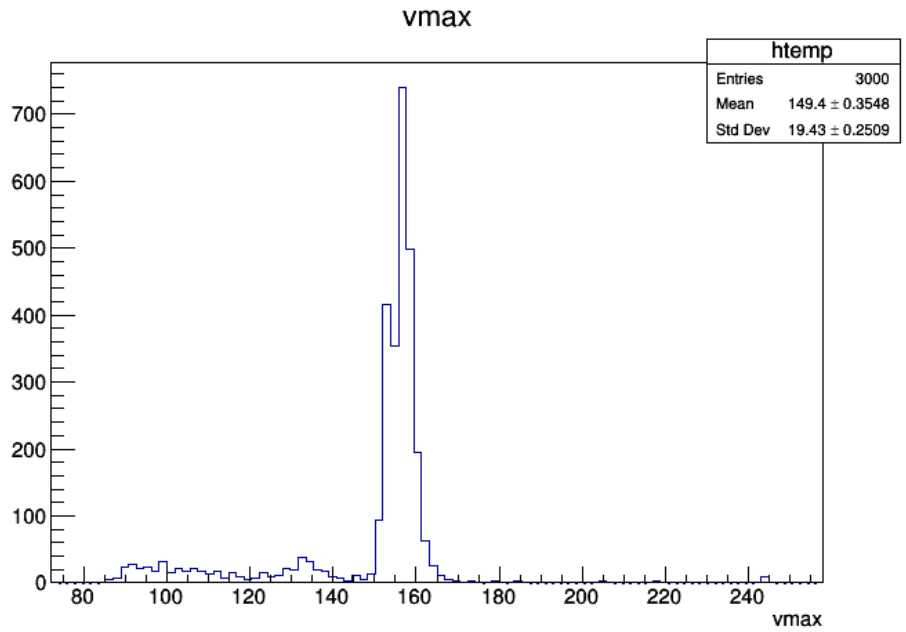
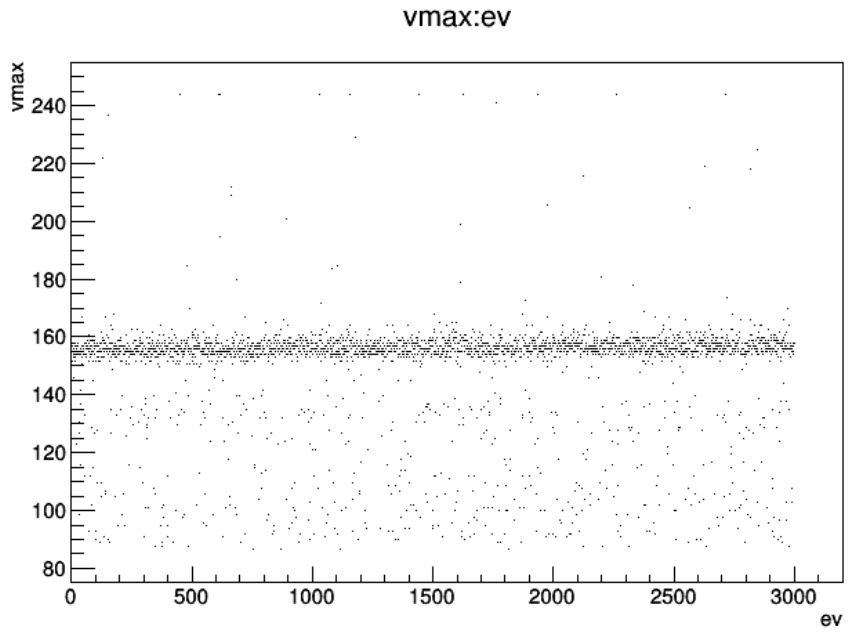


Grafico 22 Grafico vmax:ev



III.III Misure a pressione 500mb

Grafico 23 Grafico segnali a 500mb

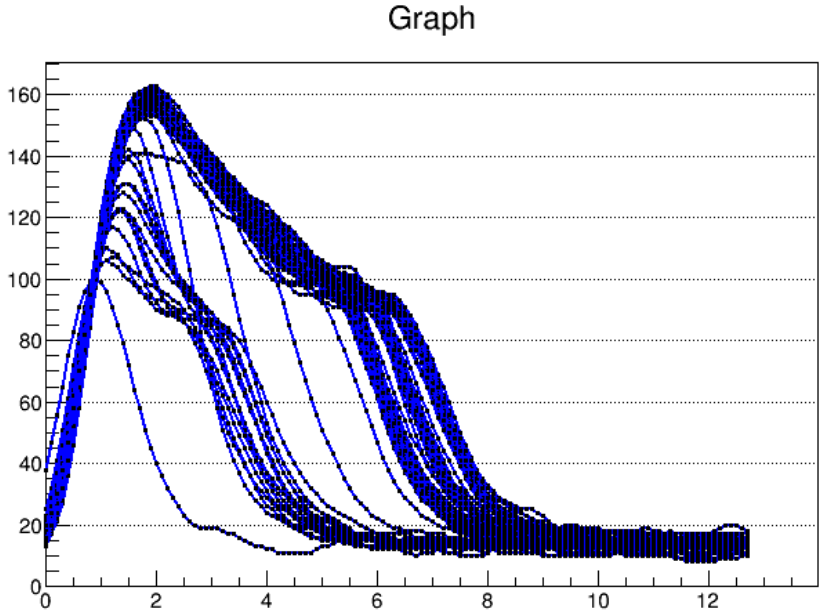


Grafico 24 Grafico integrale

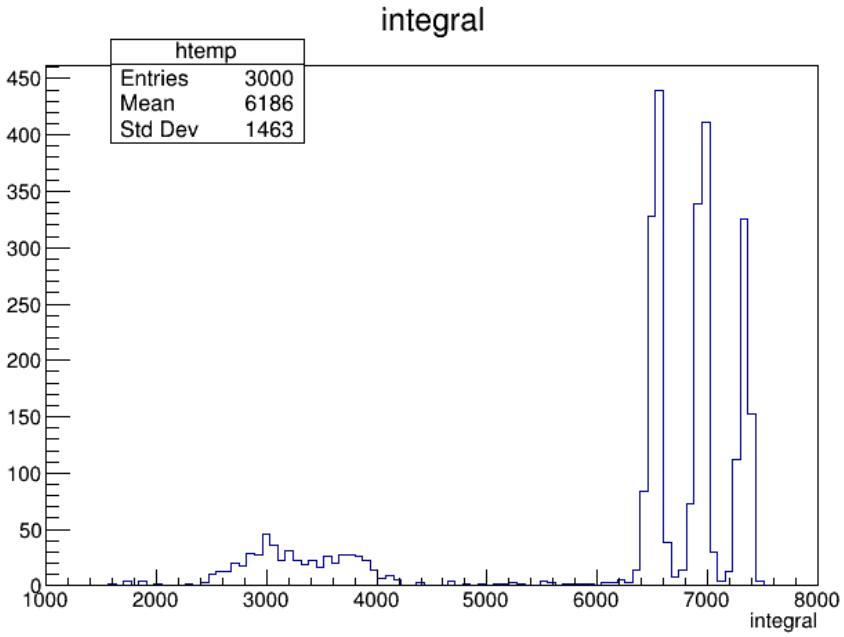


Grafico 25 Grafico integral:ev

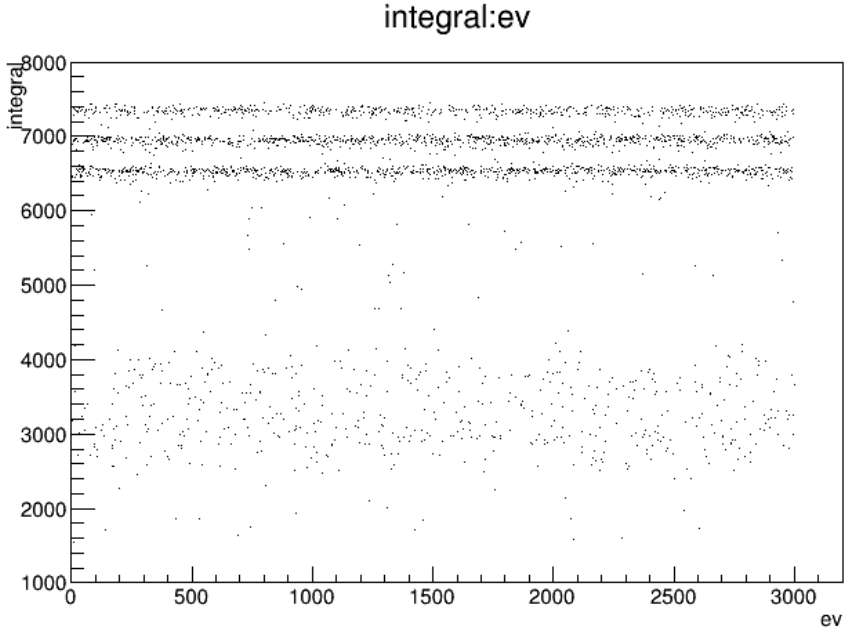


Grafico 26 Grafico integral:vmax

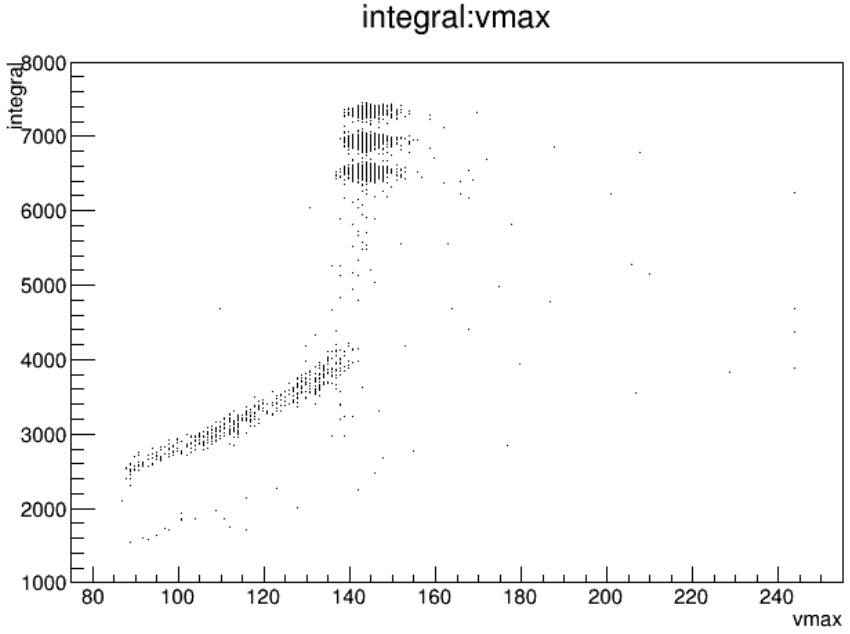


Grafico 27 Grafico vmax

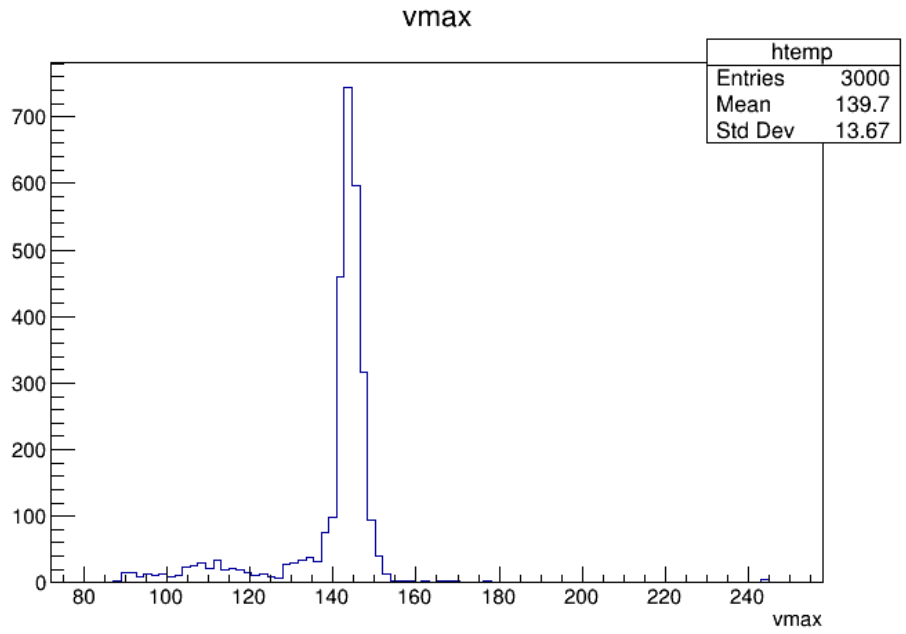
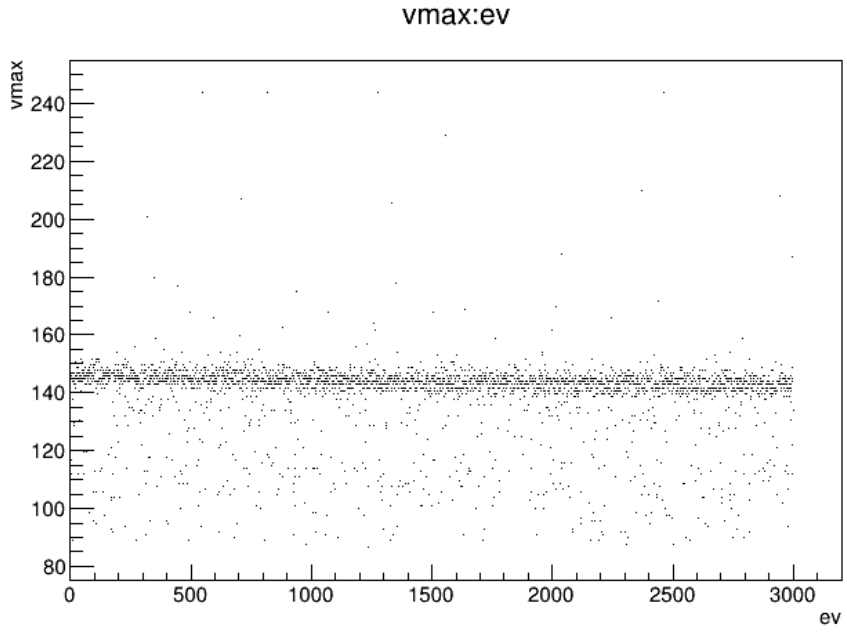


Grafico 28 Grafico vmax:ev



III.IV Misure a pressione 450mb

Grafico 29 Grafico segnali a 450mb

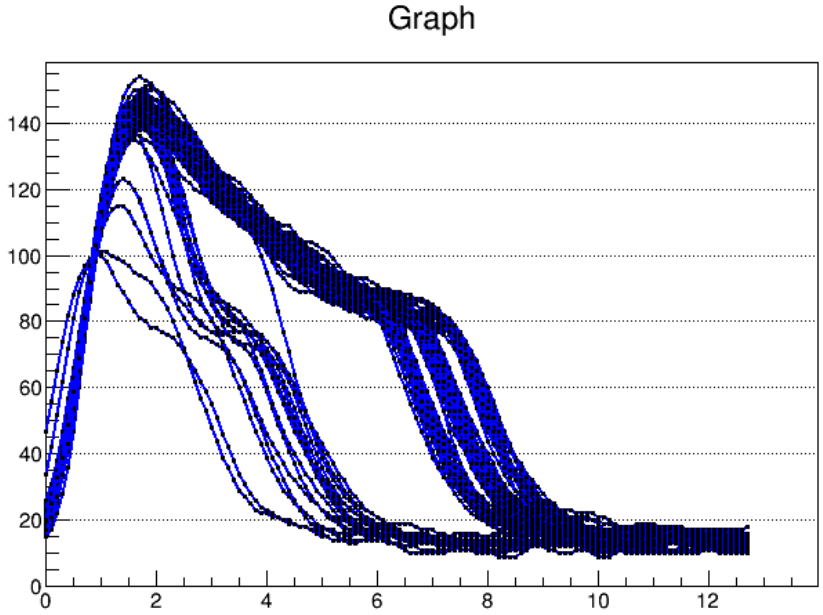


Grafico 30 Grafico integrale

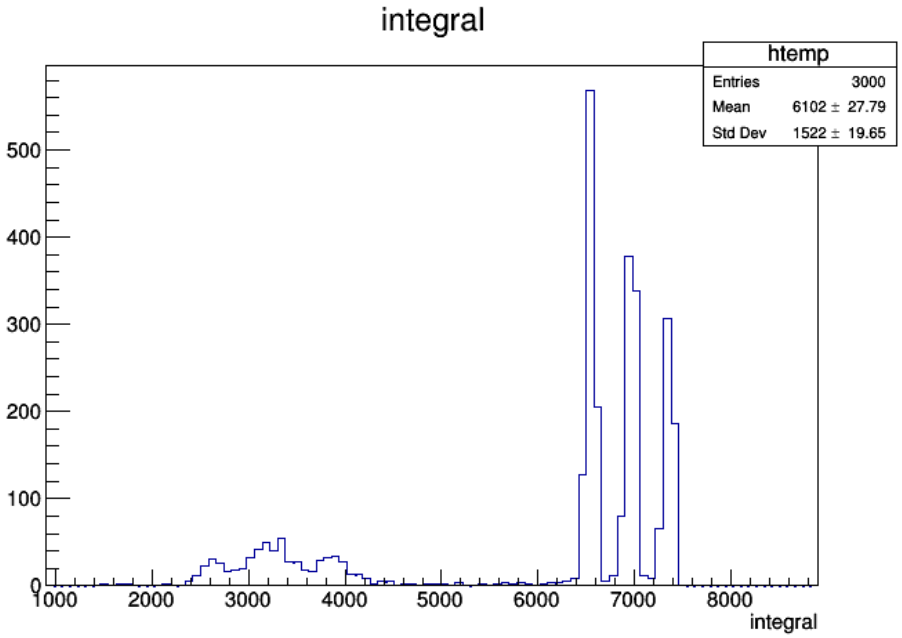


Grafico 31 Grafico integral:ev

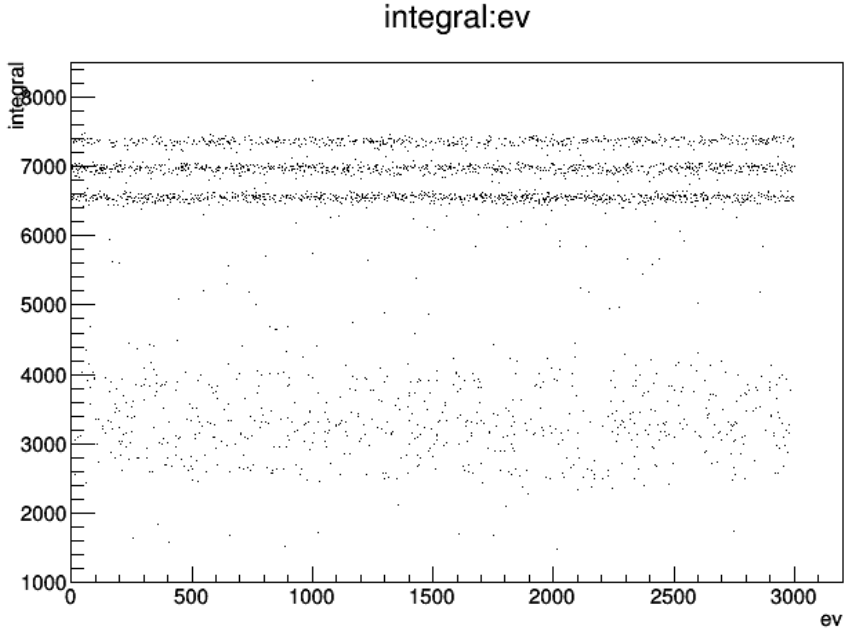


Grafico 32 Grafico integral:vmax

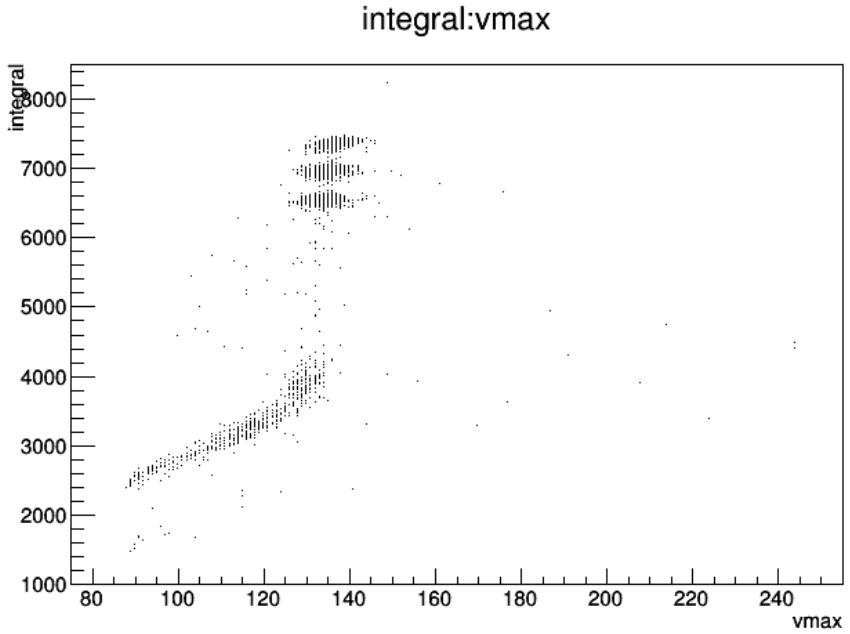


Grafico 33 Grafico vmax

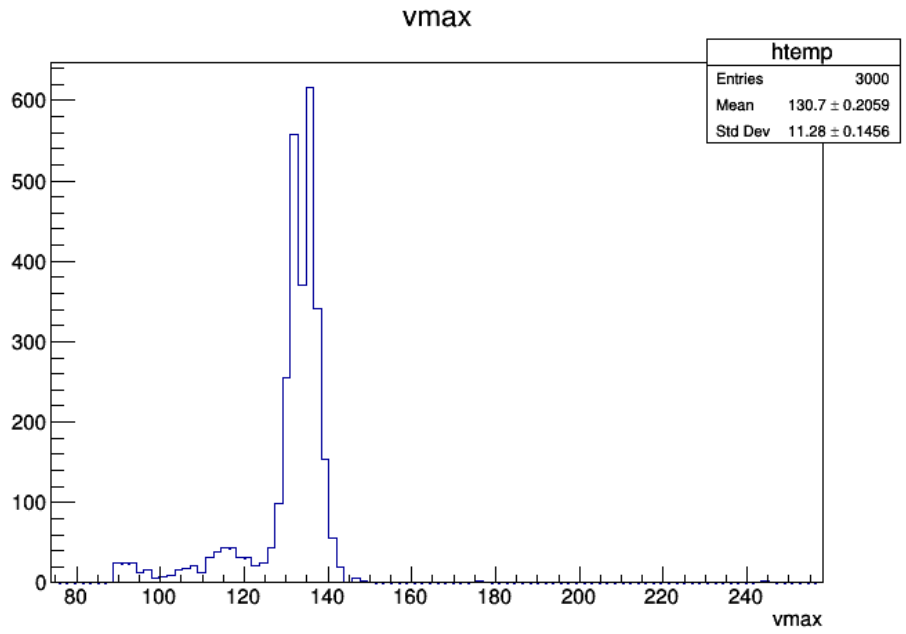
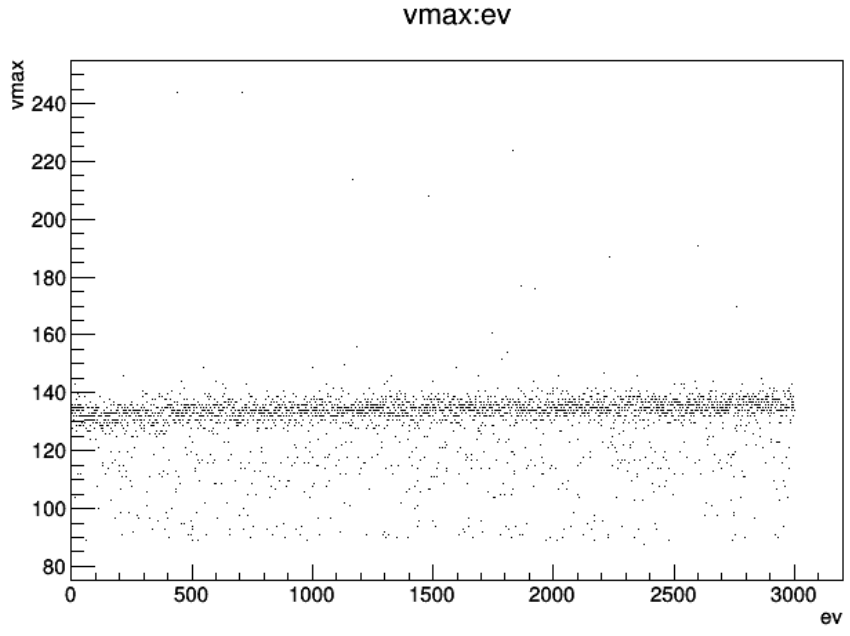


Grafico 34 Grafico vmax:ev



III.V Misure a pressione 400mb

Grafico 35 Grafico segnali a 400mb

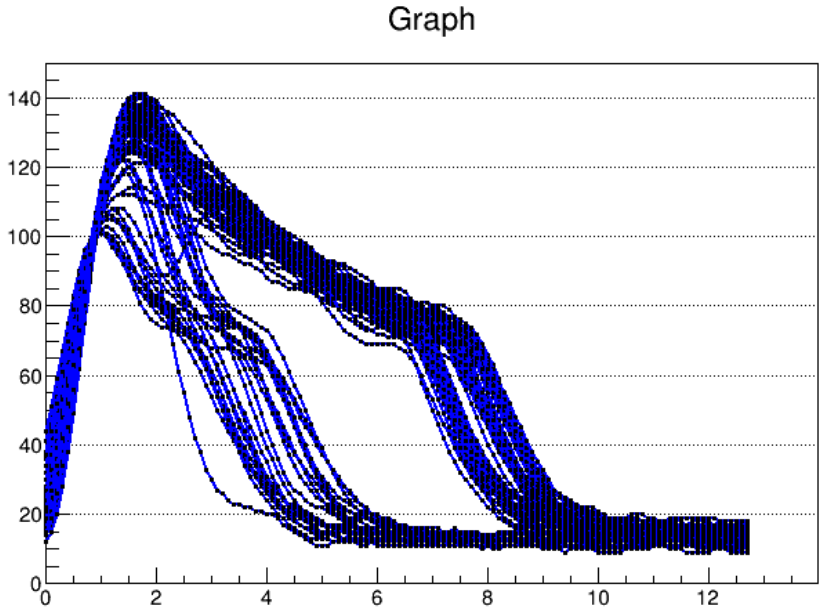


Grafico 36 Grafico integrale

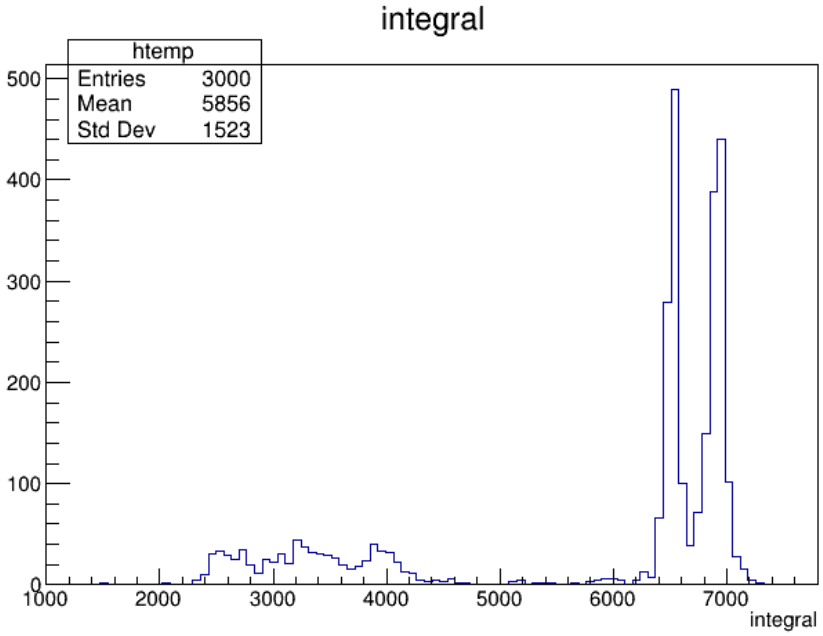


Grafico 37 Grafico integral:ev

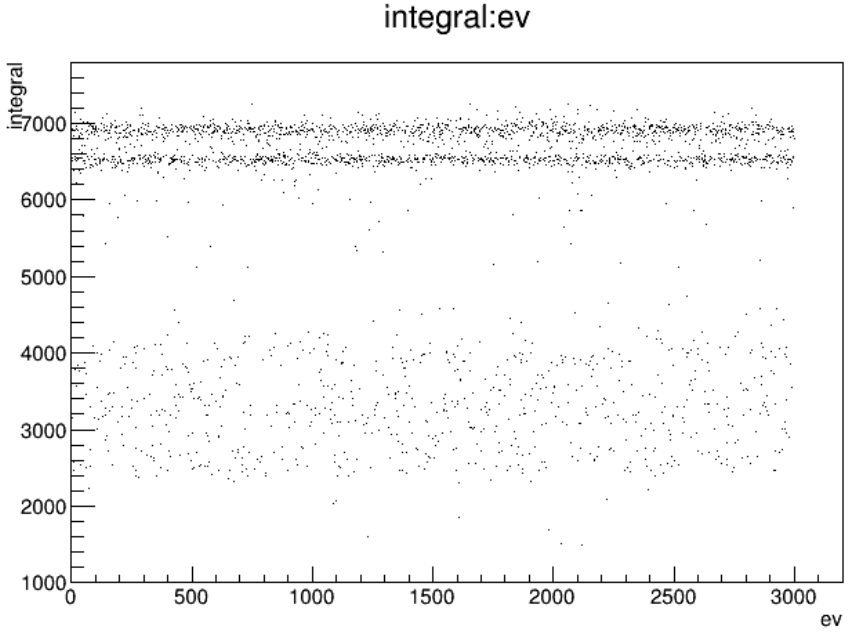


Grafico 38 Grafico integral:vmax

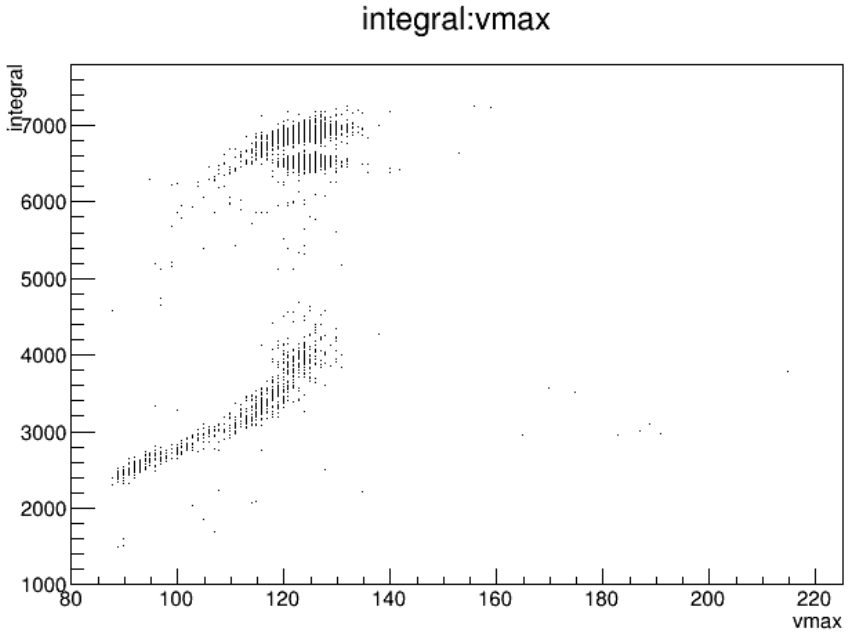


Grafico 39 Grafico vmax

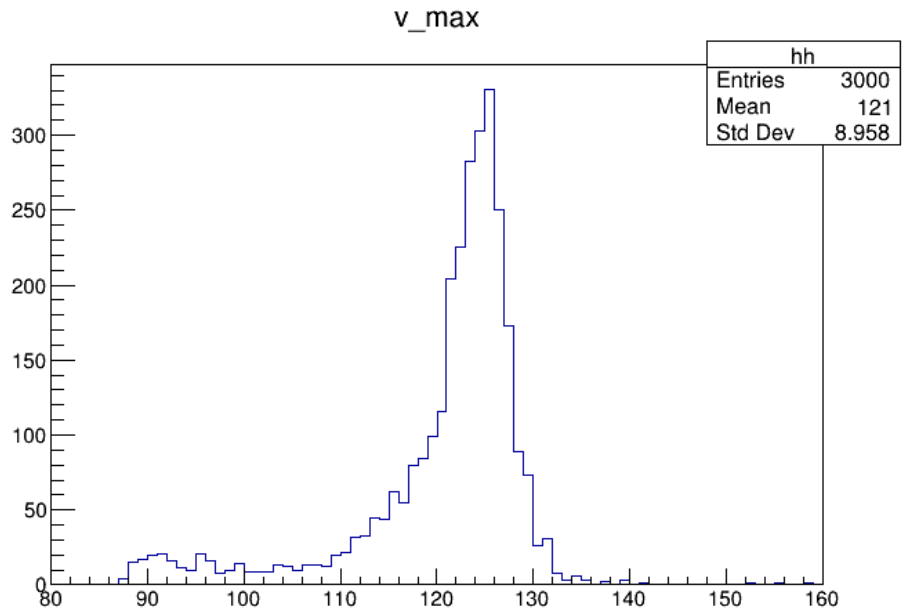
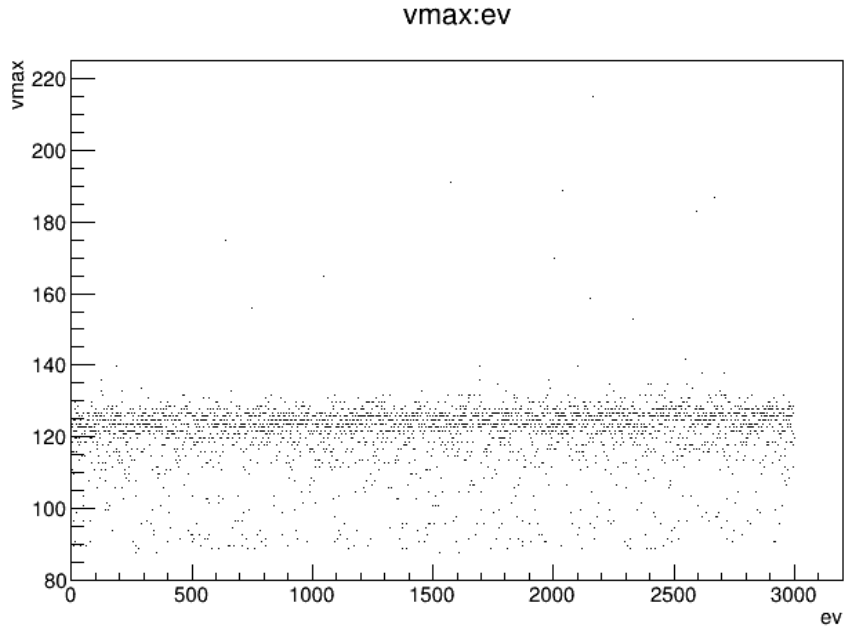


Grafico 40 Grafico vmax:ev



III.VI Misure a pressione 380mb

Grafico 41 Grafico segnali a 380mb

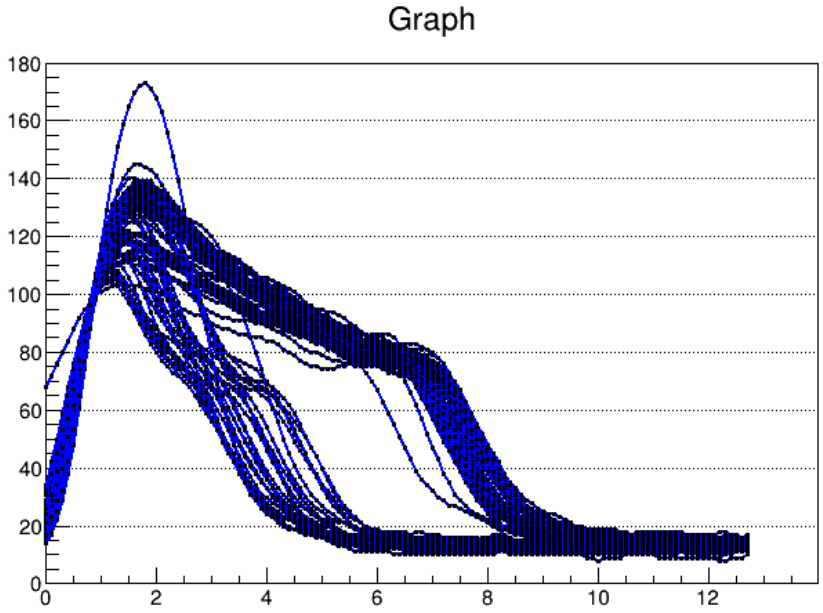


Grafico 42 Grafico integrale

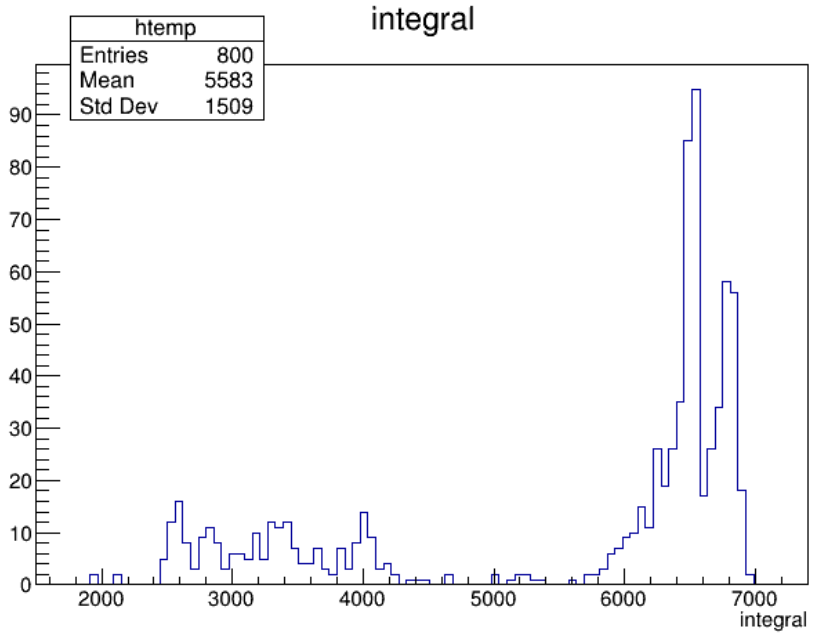


Grafico 43 Grafico integral:ev

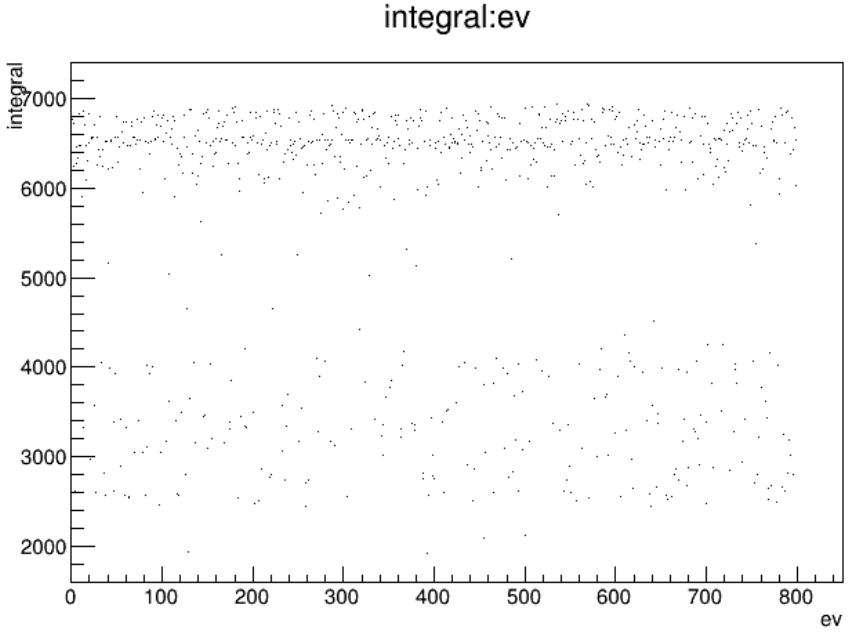


Grafico 44 Grafico integral:vmax

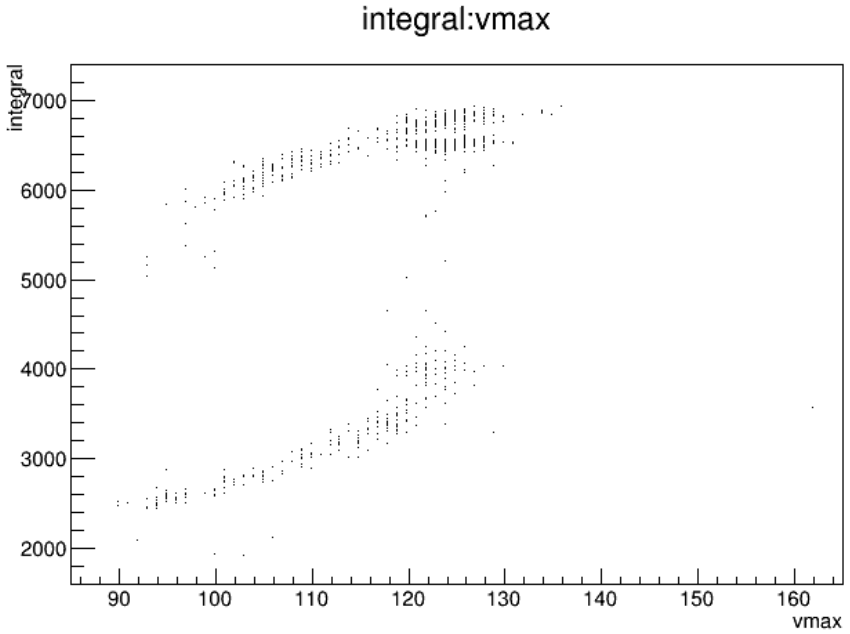


Grafico 45 Grafico vmax

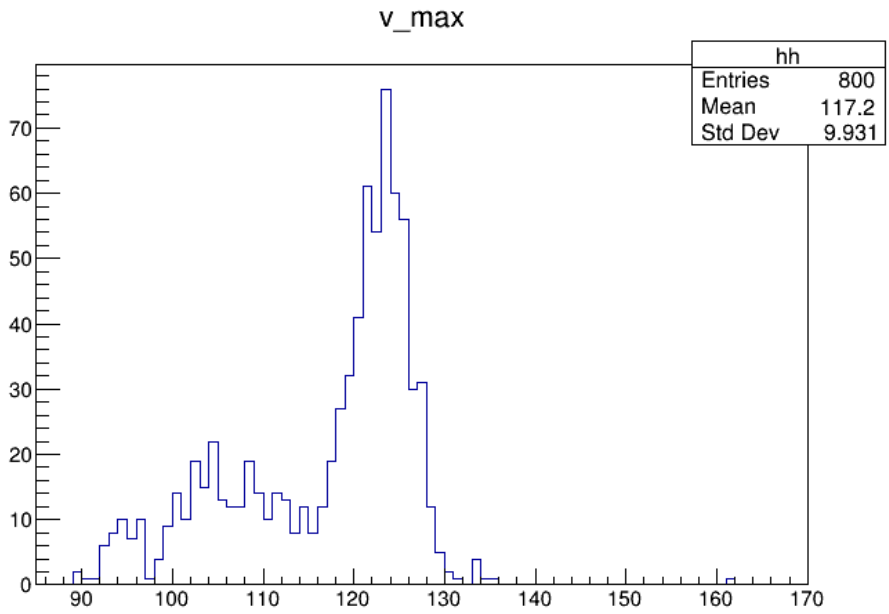
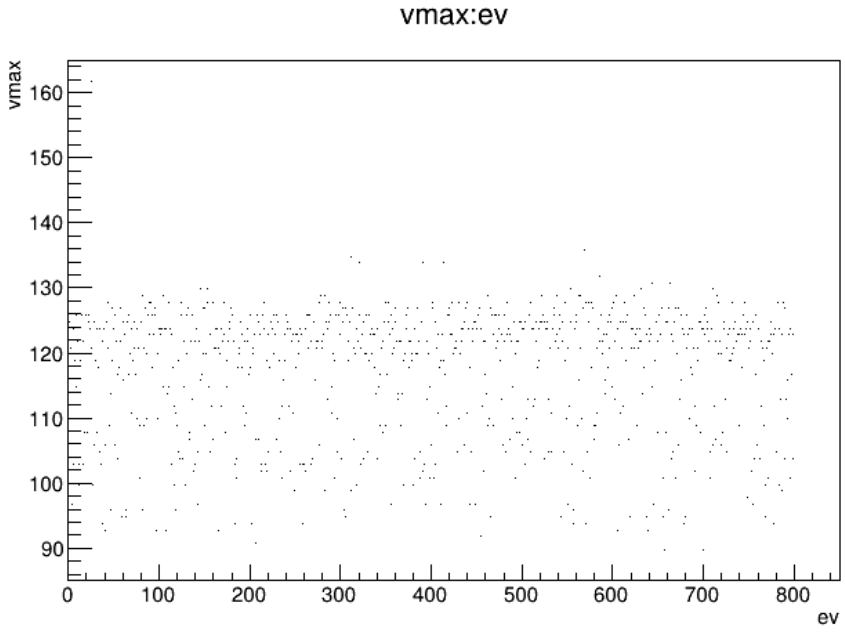


Grafico 46 Grafico vmax:ev



III.VII Curve in funzione della pressione

Come si può vedere dai grafici TODO INSERIRE LABEL QUI, l'integrale dei picchi, ovvero l'energia della particella alfa, rimane costante al variare della pressione, almeno finche non si arriva a pressioni troppo basse. Questo cambiamento a pressioni basse è dovuto al fatto che a bassa pressione le particelle riescono ad arrivare oltre la lunghezza della camera, e la restante carica non viene più rivelata. Addirittura, il terzo picco sparisce a basse pressioni, confondendosi con il secondo a causa dell'energia mancante.

Nel grafico di vmax invece si può chiaramente notare un andamento lineare (TODO perché?).

Grafico 47 Andamento integrale del picco 1 in funzione della pressione [mb]

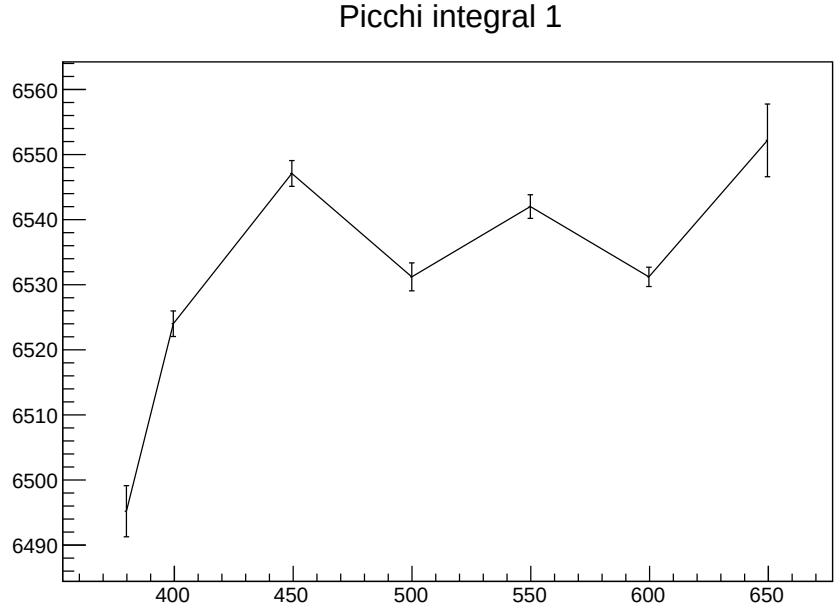


Grafico 48 Andamento integrale del picco 2 in funzione della pressione [mb]

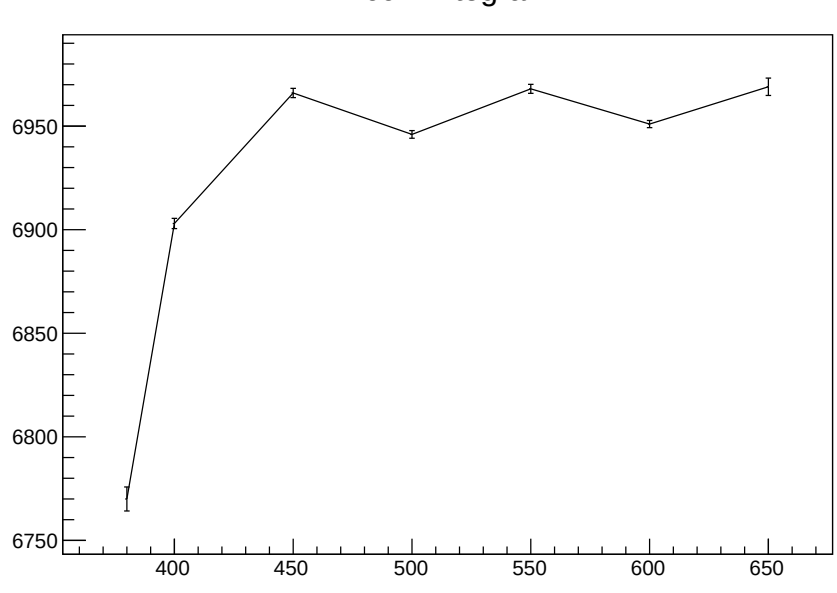


Grafico 49 Andamento integrale del picco 3 in funzione della pressione [mb]

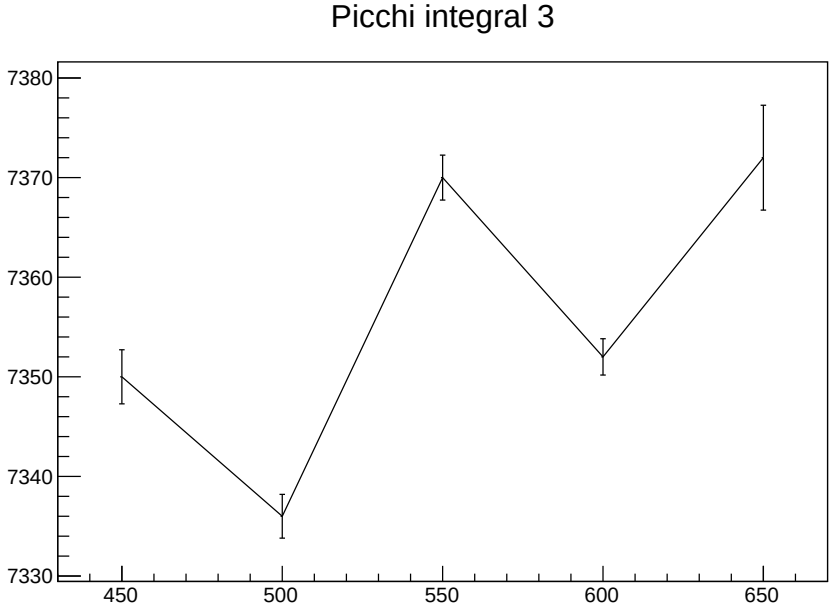
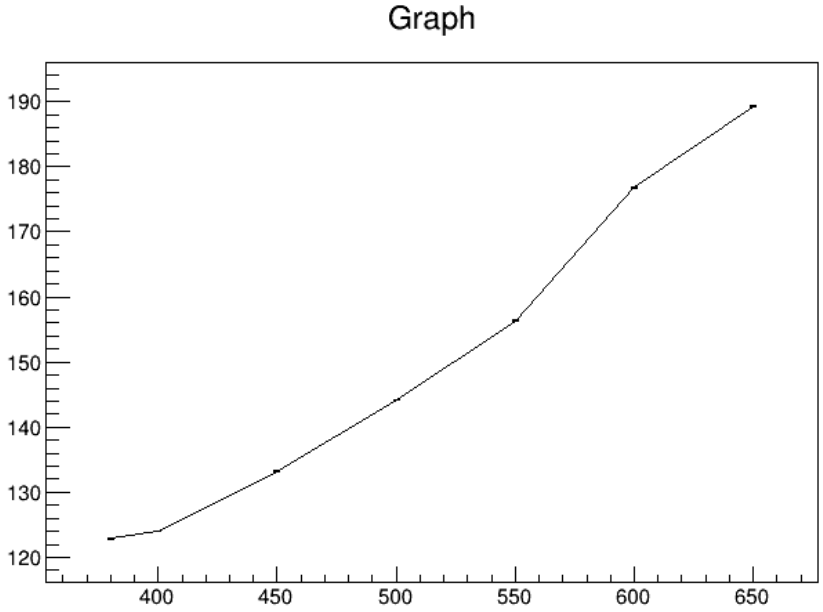


Grafico 50 Andamento integrale di vmax [V] in funzione della pressione [mb]



IV. Tabelle

Non credo ce ne siano

Magari le tabelle dei decadimenti?

V. Discussioni e conclusioni

Da fare

VI. CODICE

É presentata qua la parte fondamentale del codice in c++ usato per i calcoli numerici.

```
#include <Riostream.h>
#include <stdlib.h>
#include <TROOT.h>
#include <TSystem.h>
#include "TNtuple.h"
#include "TFile.h"
#include "TTree.h"
#include "TCanvas.h"
#include "TGraph.h"
#include "TGraphErrors.h"
#include "TF1.h"

struct bragg_signal {
  short int s[128];
};

int AnaBraggWidth(char *filename, int intto=80, float blfix=13, int
    nsig=0) {

  bragg_signal signal;

  TFile *fin=new TFile(filename);
  if (!fin->IsOpen()) {
    std::cout << "file not found! " << std::endl;
    return -1;
  }

  TTree *tree = (TTree*)fin->Get("bragg");
  if (!tree) {
    std::cout << "Bragg tree not found! " << std::endl;
    return -2;
  }

  TBranch *br = tree->GetBranch("signals");
  if (!br) {
    std::cout << "Signal branch not found! " << std::endl;
    return -3;
  }

  br->SetAddress(&signal);
  int nev = br->GetEntries();
  std::cout << "Number of events in file : " << nev << std::endl;

  // ANALIZZA EVENTO x EVENTO

  // altri parametri iniziali
```

```
46  float thr_frac = 0.4; // soglia rispetto al vmax per il calcolo
      della larghezza
47  int intfrom = 0;// regione di integrazione da 0 a intto
48  if (intto>128) intto=128;
49  int blfrom = 108, blto = 128; // regione per il calcolo della
      baseline
50
51
52  float bl; // baseline evento x evento
53  float integral;
54  float vmax; // massimo relativo alla bl
55  float width; // larghezza dei segnali
56
57
58  char outfilename[200];
59  strcpy(outfilename,"anabragg_");
60  char *cc=strrchr(filename,'/');
61  if (cc) {cc++; strcat(outfilename,cc);}
62  else strcat(outfilename,filename);
63
64  TFile *fout=new TFile(outfilename,"RECREATE"); // output file
65
66  TNtuple *nt=new
      TNtuple("nt","","ev:vmax:width:integral:baseline");
67
68  int maxev=nev;
69  if (nsig && nsig<nev) maxev=nsig;
70
71  // LOOP SUGLI EVENTI
72  for (int i=0; i<maxev; i++) {
73
74    // recupera l'evento
75    br->GetEntry(i);
76
77    // inizializza a zero
78    bl=0;
79    integral=0;
80    vmax=0;
81    width=0;
82
83    // calcolo baseline
84    for (int j=blfrom; j<blto; j++)
85      bl += signal.s[j]; bl /= (blto-blfrom);
86
87    // calcolo integrali e vmax
88    for (int j=intfrom; j<intto; j++) {
89      integral += (signal.s[j] - blfix);
90      if ( (signal.s[j] - blfix) > vmax ) vmax = (signal.s[j] -
      blfix);
91    }
92
93    // CALCOLO DELLA LARGHEZZA DEL SEGNALE A UNA CERTA PERCENTUALE
```

```
    DEL VMAX
94    // ...
95    float ratio = 0.4;
96    for (int j = 0; j < 128; ++j)
97    {
98        if (signal.s[j] > ratio*vmax) ++width;
99    }
100   /* alternativamente
101   int start = 0;
102   int end = 0;
103   for (int j = 0; j < 128; ++j){
104     if (!start && signal.s[j] > ratio*vmax) ++width;
105     if (start && !end && signal.s[j] > ratio*vmax) ++width;
106   }*/
107
108
109     nt->Fill(i,vmax,width,integral,bl);
110   }
111   std::cout << maxev << " events analyzed..." << std::endl;
112
113   fout->Write();
114   fout->Close();
115
116   fin->Close();
117
118   new TFile(outfilename); // riapre il file dei risultati
119
120   return 0;
121 }
```

../src/AnaBraggWidth.C

```
1  #include <Riostream.h>
2  #include <stdlib.h>
3  #include <TROOT.h>
4  #include <TSystem.h>
5  #include "TNtuple.h"
6  #include "TFile.h"
7  #include "TTree.h"
8  #include "TCanvas.h"
9  #include "TGraph.h"
10 #include "TGraphErrors.h"
11 #include "TF1.h"
12
13 struct bragg_signal {
14   short int s[128];
15 };
16
17 int plotSignal(bragg_signal sig, int same) {
18
19   float x[128]; for (int i=0; i<128; i++) x[i]=i*0.1;
20   float y[128]; for (int i=0; i<128; i++) y[i]=sig.s[i];
```

```
21    TGraph *g = new TGraph(128,x,y); // crea il grafico
22    g->SetMarkerStyle(7); // imposta alcuni attributi
23    g->SetLineColor(4);
24    g->SetLineWidth(2);
25
26    TCanvas *csig = (TCanvas*)gROOT->FindObject("csig"); // cerca
        l'oggetto "csig" (canvas)
27    if (!csig) {
28      csig=new TCanvas("csig"); // se non c'e' la crea nuova
29      csig->SetGridy();
30      g->Draw("apl"); // disegna il grafico e anche il frame con gli
        assi
31    }
32    else {
33      csig->cd(); // se c'e' si posiziona sulla canvas "csig"
34      if (same)
35        g->Draw("pl"); // disegna nel frame esistente
36      else {
37        csig->Clear();
38        g->Draw("apl"); // disegna in un nuovo frame
39        gSystem->Sleep(200); // aspetta 200 ms
40      }
41    }
42    csig->Modified(); // aggiorna la canvas
43    csig->Update();
44    gSystem->ProcessEvents(); // aggiorna la grafica
45
46    return 0;
47  }
48
49  int PlotSignals(char *filename, int plfrom=0, int plto=100, int
      same=1) {
50
51    bragg_signal signal;
52
53    TFile *fin=new TFile(filename);
54    if (!fin->IsOpen()) {
55      std::cout << "file not found! " << std::endl;
56      return -1;
57    }
58
59    TTree *tree = (TTree*)fin->Get("bragg");
60    if (!tree) {
61      std::cout << "Bragg tree not found! " << std::endl;
62      return -2;
63    }
64
65    TBranch *br = tree->GetBranch("signals");
66    if (!br) {
67      std::cout << "Signal branch not found! " << std::endl;
68      return -3;
69    }
```

```
70
71    br->SetAddress(&signal);
72    int nev = br->GetEntries();
73    std::cout << "Number of events in file : " << nev << std::endl;
74
75    for (int i=plfrom; i<plto; i++) {
76      br->GetEntry(i);
77      plotSignal(signal,same);
78    }
79
80    return 0;
81  }
```

../src/PlotSignals.C