

# STUDIO DELLE PARTICELLE ALFA DI UNA SORGENTE RADIOATTIVA PER MEZZO DI UNA CAMERA A IONIZZAZIONE

## Indicazioni preliminari

### Logbook elettronico

Nel computer della postazione è predefinita una directory DATI\_GRUPPI (/home/labo/DATI\_GRUPPI) all'interno della quale dovrete creare una sub-directory *gruppoN* (N = numero del vostro gruppo) dove salverete i dati grezzi e analizzati e i risultati principali (*logbook elettronico*).

*Vedete nei paragrafi successivi e in fondo a questo documento la lista delle informazioni da produrre durante l'esecuzione dell'esperimento.*

### Computer della postazione

Ogni postazione è dotata di un PC con sistema operativo Linux Ubuntu 20.04 LTS. *Alla fine del turno non spegnete il computer* (potete lasciare la sessione aperta o fare logoff).

### Analisi dati con ROOT

Ogni PC ha installato ROOT6, che viene anche utilizzato per alcuni moduli di acquisizione e analisi. In appendice trovate una lista di comandi utili per l'analisi degli istogrammi con ROOT. Si avvia da terminale dopo aver lanciato l'alias "root6" che definisce le variabili di ambiente necessarie.

## Materiale e strumentazione disponibile

- una camera a ionizzazione tipo Bragg cilindrica con un volume di gas di circa 100 mm  $\varnothing$  x 120 mm h , dotata di preamplificatore di carica, riempita con una miscela di Ar + CH<sub>4</sub> all' 1%.
- un crate NIM con vari moduli di elettronica tra cui:
  - o un amplificatore CANBERRA 2021 o simile
  - o un modulo HV (High Voltage) NHQ 112M pilotato in automatico da computer
- un PC con sistema operativo Linux Ubuntu 20-04 LTS
- un oscilloscopio digitale Picoscope 2204A impostato per acquisire a 12.5 MHz (Sampling Time 80 ns) sul canale A (canale B e AWG inattivi)
- una sorgente alfa tripla (Am-241, Np-237, Cm-244) inserita all'interno della camera di Bragg in prossimità del catodo (in basso)

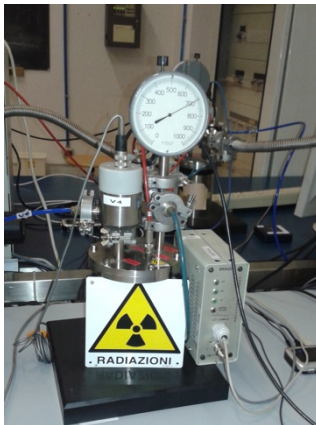


Figura 1: camera di Bragg (sinistra) e cassetto dell'elettronica (destra)

## Software

- **CONTROLLO PRESSIONE:** è un'applicazione per Arduino che serve per controllare la pressione della camera e l'intensità del campo elettrico. Si accede cliccando sull'icona con il disegno del manometro situata sul Desktop. Il programma regola automaticamente l'HV in modo da mantenere costante il campo ridotto E/p
  - o **il programma di solito**, quando vi loggate, **è già aperto**, in quanto per ripulire le camere da impurezze depositate e stabilizzare il flussaggio occorre attivare il circolo del gas almeno un paio d'ore prima.
  - o Se non lo fosse o dovesse bloccarsi per qualche ragione chiedere ai docenti per farlo ripartire
  - o Impostare la pressione voluta e premere enter. Il bottone verde Start si attiva e a questo punto potete premerlo per avviare effettivamente il flussaggio. Gli indicatori in blu mostrano i valori di Pressure e Voltage visti dall'arduino.
  - o Per modificare il valore premere Stop, inserire il nuovo valore e far ripartire.

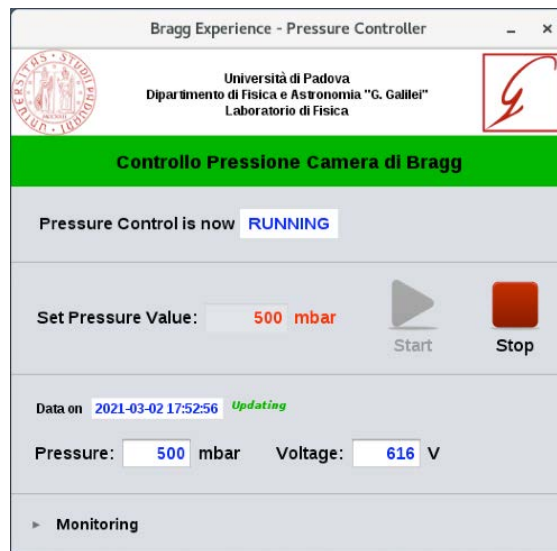


Figura 2: finestra di controllo della pressione

- **DAQ** : l'acquisizione dei dati gira sotto ROOT:
  - aprire un terminale
  - digitare l'alias "**root6**" per inizializzare il path e le variabili di ambiente
  - spostarsi sulla directory di lavoro **DATI\_GRUPPI/gruppoN**
  - far partire ROOT
  - al prompt caricare la libreria e creare la classe che gestisce l'acquisizione:  
**root[0] Pico2204AdaqGui()**
  - posizionare le finestre in modo da averle tutte accessibili tipo in figura 3

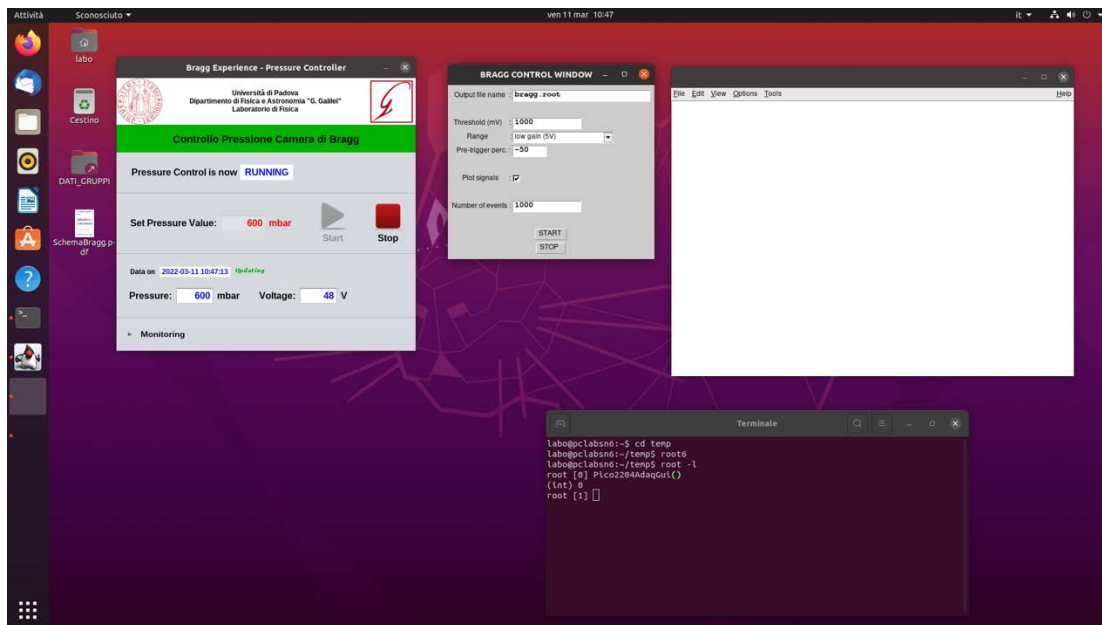


Figura 3: esempio di sessione si lavoro su un PC del laboratorio

## PRIMA PARTE: FUNZIONAMENTO DELL'APPARATO E MISURE PRLIMINARI

### Sequenza operazioni

- **Accensione elettronica e riconoscimento dei moduli**
  - camera di bragg + circuito gas + preamplificatore
  - Amplificatore-formatore (*connettori e controlli principali*)
    - Input polarity
    - Input/Output connectors
    - Coarse Gain / Fine Gain
    - Shaping Time
    - Pole-zero (trimmer)
  - Acquisitore
    - abilitato solo il channel A
  - HV
    - è abilitato solo l'ON/OFF
- **Operazioni sull'hardware**
  - Impostare la pressione a 600 mb e attendere la stabilizzazione.
  - Settare lo shaping time dell'amplificatore a 0.25 us
  - Inserire il segnale del preamplificatore nell'input dello shaping-amplifier e collegare l'uscita dell'amplificatore all'oscilloscopio
  - Visualizzare i segnali della camera, che dovrebbero mostrare la forma della curva di bragg invertita, e regolare l'amplificazione in modo che il picco di bragg capiti a circa 4V.
  - **Scattate una foto del segnale sull'oscilloscopio da inserire nel logbook elettronico**
  - Provare a variare lo Shaping Time: *cosa succede? dare una spiegazione della forma del segnale*
  - Tornare a  $ST=0.25$  us e inserire il segnale dell'amplificatore nell'ingresso A del Picoscope
- **Operazioni sul software**
  - Settare la trigger threshold tra 1V e 2V e selezionare la visualizzazione dei segnali (plot waveforms).
  - Visualizzare i segnali (*vedi punto 1 della sezione sull'analisi dati*) verificando che abbiamo le caratteristiche attese e prendere nota dell'altezza massima dei segnali buoni (picco di Bragg) : il massimo dovrebbe capitare tipicamente tra 160 e 220 circa. Se fosse significativamente più basso o più alto modificare l'amplificazione.

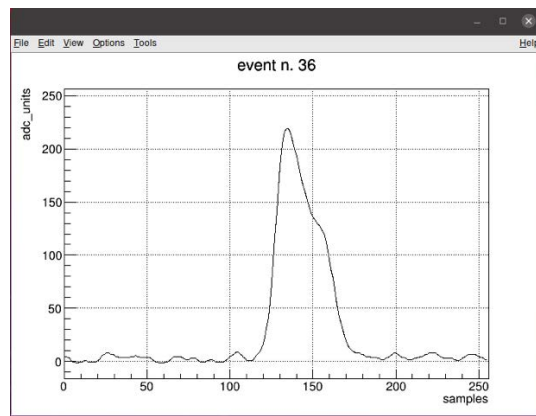


Figura 4: esempio di segnale

- Prendete un primo sample di prova di circa 500 eventi ed eseguite il punto 1 della sezione Analisi Dati 1 (controllo della forma dei segnali)
- Se tutto è corretto avviate un'acquisizione di alta statistica (~5000 eventi), sempre a 600 mb. Per questa acquisizione disabilitare il plot dei dati in modo da eliminare il tempo morto dovuto alla grafica.
- Nel frattempo aprite un secondo terminale e continuate la pre-analisi del primo file proseguendo dal punto 2.
- Appena completato il file ad alta statistica potete passare ad analizzare quello, rifacendo velocemente i plot del punto 2 successivi (che andranno poi salvati nel logbook)

## ANALISI DATI 1<sup>1</sup>

- Sono disponibili alcune macro per l'analisi dei dati in laboratorio:
  - [PlotSignals.C](#) : per visualizzare i segnali registrati
  - [AnaBragg.C](#) : per creare un'ntupla con le informazioni dei segnali analizzati singolarmente
- Le macro si trovano in [/home/studenti/macro](#)<sup>2</sup>

### 1. Controllo della forma dei segnali:

- a. Utilizzate la macro PlotSignals.C per disegnare un centinaio di segnali uno sopra l'altro e:

<sup>1</sup> In fondo alla scheda trovate una lista di comandi utili per l'analisi dei dati con ROOT

<sup>2</sup> Attenzione: non è la vostra \$HOME, il path è proprio [/home/studenti/macro](#) e parte dalla radice /.

1. verificate che i massimi dei segnali buoni siano tutti più o meno attorno allo stesso valore (plot a sinistra)
2. verificate che si vedano 3 “fasce” corrispondenti ai 3 diversi tempi di frenamento delle 3 energie diverse delle sorgenti (plot a destra)
3. prendete nota della regione dove integrare il segnale e in quella dove calcolare il livello medio di fondo (*baseline*)

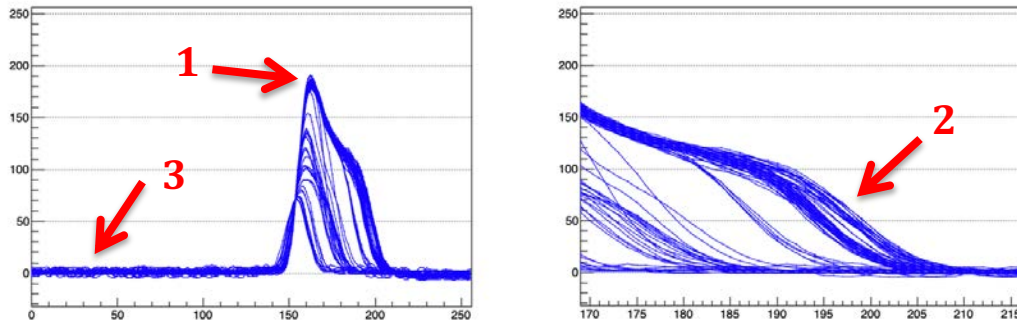


Figura 5: esempio di segnali disegnati con PlotSignals

- b. Salvare i grafici come figura (.gif, .png, ...) nella directory di lavoro (logbook elettronico): es: `root[1] csig->Print("segnali.gif");`

## 2. Verifica del corretto funzionamento della camera:

- a. Copiatevi la macro AnaBragg.C nella vostra directory di lavoro, in modo da poterla successivamente modificare.
- b. Utilizzate la macro per creare un'ntupla con i risultati dell'analisi evento per evento. La funzione principale ha la seguente forma :

```
int AnaBragg(const char *filename="", int blto=60, int intfrom=100, int intto=200,  
int npts=256, int nsig=0)
```

*filename* = nome del file dei dati acquisiti (.root)

*blto* = limite della regione su cui mediare la baseline

*intfrom, intto* = regione di integrazione

*npts* = numero dei punti campionati (256)

*nsig* = numero dei segnali da analizzare (0 = tutti)

`nt->Print();` per vedere il contenuto dell'ntupla

- c. La macro ha predefinito il **calcolo dell'integrale** e la **ricerca del massimo** senza sottrazione della baseline. Editate la macro per modificare i valori predefiniti adattandoli alle vostre misure.
- d. Analizzate il file con alta statistica degli eventi a 600 mb. Plottate l'integrale (energia) verificando di vedere 3 picchi ben distinti (Fig.6, plot a sinistra) e il massimo (picco di Bragg), dove dovrete trovare un singolo picco (plot a destra)



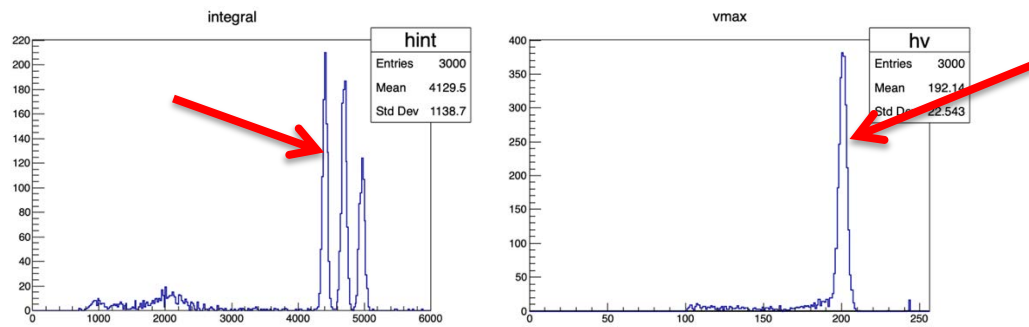


Figura 6: esempio di plot con AnaBragg : variabile integral (sinistra, vecchia sorgente) e variabile vmax (destra)

- e. Editate la macro per implementare il calcolo della baseline come media dei valori nella regione selezionata. Rifate i plot di integrale e vmax con la sottrazione della baseline evento per evento.
- f. Plottate le energie e i massimi in funzione del numero di evento per verificare la stabilità della presa dati: le bande dovrebbero essere approssimativamente orizzontali

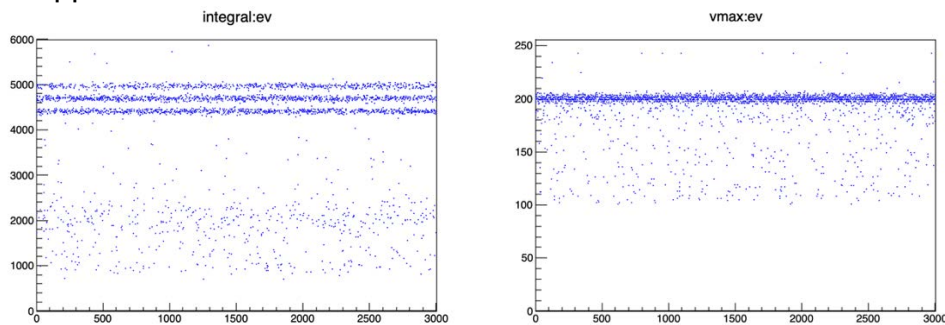
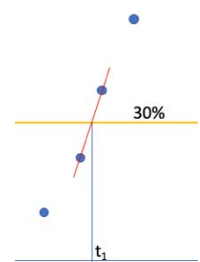


Figura 7: esempio di plot bidimensionali per controllo stabilità

- g. Plottate anche il valore del massimo vs. integrale
- h. Salvare tutti questi plot per i dati a 600 mb come figure (.gif, .png, ...) nella directory di lavoro (logbook elettronico)

### 3. Larghezza dei segnali

- a. Implementare nella macro AnaBragg.C la misura della larghezza temporale del segnale (campo *width* nell'ntupla), prendendo come riferimenti temporali i valori per i quali il segnale passa per il 30% del valore massimo (algoritmo semplice).
- b. Successivamente<sup>3</sup> si può migliorare l'algoritmo andando a determinare i tempi come intersezione tra la retta che definisce il livello del 30% e quella che passa tra il punto immediatamente sotto e quello immediatamente sopra detto livello (vedi schema a lato).



<sup>3</sup> Durante l'esperienza è sufficiente implementare l'algoritmo semplice. Il secondo è richiesto solo per il report e può essere implementato successivamente.

- c. plottare le larghezze e il bidimensionale larghezza vs. integrale verificandone la forte correlazione
- d. Salvare anche questi plot come figure (.gif, .png, ...) nella directory di lavoro (logbook elettronico)

## PARTE II – MISURE A DIVERSE PRESSIONI – MISURA DEI RANGE

Tutte queste misure andrebbero eseguite insieme, per questioni di omogeneità dei campioni di dati, in linea di massima il secondo giorno. In totale dovrebbero servire tra 1.5h e 2h circa.

- **Misura delle particelle alfa al variare della pressione**

- Acquisire un primo set di dati a 600 mb di sufficiente statistica (~2000 eventi)
  - verificare rapidamente con le macro già usate il giorno prima che tutto sia regolare
- prendere quindi altri 4 set di misure (~2000 eventi) a 650 mb, 550 mb, 500 mb e 450 mb
  - ogni misura richiede un tempo di 15-20min: è bene utilizzare questo tempo per procedere con l'analisi dati in parallelo. Questo permette anche di evidenziare subito eventuali anomalie nella presa dati e, se necessario, di ripetere la misura.
  - Per analizzare i dati in parallelo aprite un nuovo terminale e lavorate su quello. Si sconsiglia di fare analisi sul terminale dell'acquisizione, onde evitare rischi di perdere la misura in corso.
- Prendere infine 4 set di dati a statistica più bassa (~1000 eventi) intorno a 400 mb: 425 mb, 400 mb, 375 mb e 350 mb.



## ANALISI DATI 2

### 4. Plot di controllo

Durante la sessione di laboratorio:

- Per ogni dataset creare i plot di controllo (vedi punto 2 della prima parte) per essere sicuri che la presa dati non presenti problemi o anomalie.

### 5. Misure al variare della pressione

- a. Per ogni valore di pressione plottate il valore del massimo vs. integrale *Cosa succede per le misure a più bassa pressione? cercate un'interpretazione fisica per i risultati ottenuti.*
- e. Per le pressioni principali (quelle con 2000 misure) ricavate le posizioni dei centroidi delle alfa dell'Am-241 e i valori di  $v_{max}$  e plottateli in funzione della pressione
- f. *salvare i bidimensionali  $v_{max}$  vs. int e i plot dei centroidi e  $v_{max}$  come figure (.gif, .png, ...) nella directory di lavoro (logbook elettronico)*

### 6. Stima del range delle alfa alle diverse pressioni <sup>4</sup>

- g. Stimare in modo preciso la velocità di drift degli elettroni utilizzando le misure a bassa pressione: dall'analisi dei plot bidimensionali ( $v_{max}$  vs. *integral*), ( $v_{max}$  vs. *width*) e (*width* vs. *integral*) identificate le condizioni per selezionare al meglio le particelle che non vengono fermate completamente nel volume di gas e sbattono sull'anodo. Utilizzate questa selezione per ricavare in modo pulito il tempo che impiegano gli elettroni ad attraversare tutta la lunghezza della camera e quindi la velocità di drift.
- h. Calcolate i range alle pressioni maggiori per tutti e 3 gli isotopi.
- i. Verificate la legge di Bragg-Kleeman (corrispondenza pressione-range).
- j. *salvare il plot di Bragg-Kleeman per l'Am-241 come figura (.gif, .png, ...) nella directory di lavoro (logbook elettronico)*

### 7. Calibrazione e misura della risoluzione energetica per la pressione 600 mb<sup>4</sup>

- a. Determinate le posizioni dei picchi e associate le energie delle tre sorgenti.  
*Nota: ogni sorgente emette più alfa di diverse energie: cercare su internet (es. DDEP) le energie e intensità delle 2-3 righe principali. Attenzione alla differenza tra “ $\alpha$  transitions” e “ $\alpha$  emissions”...*

---

<sup>4</sup> Per l'analisi in laboratorio è sufficiente un'analisi semplice. Nel report rifarete l'analisi con maggior calma, verificando i risultati ottenuti in lab e magari migliorandoli.

*Un modo semplice di procedere consiste nel fittare con una gaussiana la spalla destra del picco (da poco prima del massimo fino alla base a dx) e farle corrispondere l'energia della riga più alta e intensa...*

- b. Rappresentate in grafico la corrispondenza tra posizione (a.u.) ed energia (plot a sinistra di fig. 8) determinando il coefficiente di conversione canale-energia e quindi la risoluzione energetica  $R = \text{FWHM}_E / E_\alpha$  per i 3 picchi, fornendo poi la media dei valori trovati.

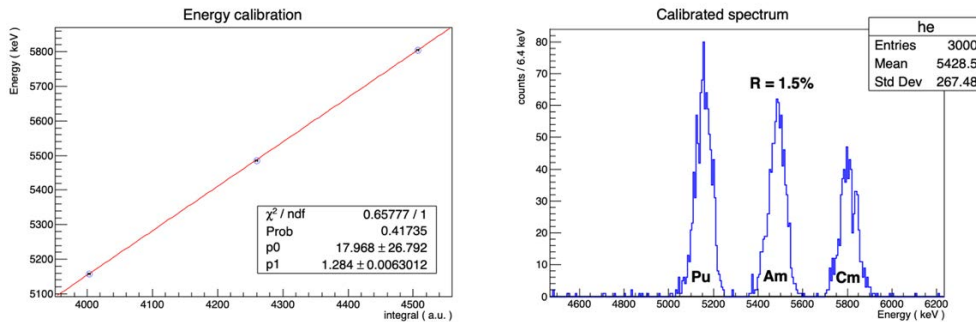


Figura 8: calibrazione in energia e plot ricalibrato (vecchia sorgente)

- c. salvare un fit di un picco come esempio e il plot di calibrazione (.gif, .png, ...) nella directory di lavoro (logbook elettronico). Salvare anche le risoluzioni calcolate in un file .txt.

## Logbook elettronico

Impacchettate la directory di lavoro con i plot richiesti e i dati grezzi e analizzati (root files) in un unico file da sottomettere su moodle (*logbook\_gruppoN\_bragg.tgz*) :

Esempio di impacchettamento con **tar**:

- posizionarsi nella directory DATI\_GRUPPI: `cd /home/labo/DATI_GRUPPI`
- impacchettare: `tar cvfz logbook_gruppo3_bragg.tgz Rossi`
- postare il file **logbook\_gruppo3\_bragg.tgz** su moodle.

## Analisi dati per il report

Nel report ripetete le analisi proposte in laboratorio, ricontrollando ed eventualmente migliorando gli algoritmi, ed estendete le misure della parte 2 anche agli altri isotopi.

## Comandi utili in ROOT per l'analisi dei dati

Qui di seguito alcuni esempi di righe di comando utili per l'analisi dei dati con ROOT

- **PlotSignals:**

- `PlotSignals("bragg500.root", 0, 50);` // plotta i primi 50 segnali uno sopra l'altro
- La scala verticale viene definita sulla base dell'ampiezza del primo segnale. Se è troppo piccola:
  - cliccare col tasto destro sulla scala verticale (yaxis) e selezionare `SetRangeUser`
  - introdurre gli estremi del range che si vuole visualizzare

- **AnaBragg:**

- `AnaBragg("bragg500.root", 100, 120, 180);` // carica il file, analizza i segnali integrando dal sample 120 a 180 e usando i primi 100 sample per calcolare la baseline.

### Istogrammi 1D

- `nt->Draw("integral");` // disegna il plot della variabile `integral` determinando range e binning in modo automatico.
- `nt->Draw("integral","integral>3000");` // disegna il plot della variabile `integral` selezionando solo gli eventi con `integral > 3000`. Range e binning sempre automatici.
- `nt->Draw("integral">>hint(200,3000,5000));` // disegna il plot di `integral` in un istogramma custom di nome "hint" con 200 bin e range da 3000 a 5000
- `nt->Draw("width","vmax<100 && integral>3000");` // disegna la larghezza `width` per eventi selezionati con `vmax` minore di 100 AND `integral > 3000`. La sintassi è quella del C++.

### Istogrammi 2D

- `nt->Draw("integral:ev"); nt->Draw("vmax:integral");` // bidimensionali y (prima variabile) vs. x (seconda variabile, dopo i due punti). In entrambi i casi la scelta di range e binning è automatica
- `nt->Draw("vmax:integral">>hint2d(200,3000,5000,256,0,256));` // bidimensionale con binning e range fissati. Notare che nella definizione di `hint2d` i primi 3 campi si riferiscono alla x (quindi all'`integral`) e i secondi 3 alla y (`vmax`)
- `nt->Draw("vmax:integral">>hint(200,3000,5000,256,0,256)","","colz");` // come sopra, ma con l'opzione grafica "colz", molto utile per visualizzare in modo chiaro queste distribuzioni bidimensionali. Notare il campo centrale "" corrispondente a una stringa vuota di selezione.

### Calibrazione assi

- `nt->Draw("integral">>hint(200,3000,5000));` // disegnare il plot della variabile di interesse in un istogramma custom, in cui decidete voi binning e range
- Nella finestra dei comandi dovreste vedere il vostro istogramma nella lista degli oggetti. Digitate `.ls` e invio. Esempio:

```
root [6] .ls
TFile**      anabragg_bragg500.root
TFile*       anabragg_bragg500.root
OBJ: TNtuple nt      : 0 at: 0x7fad46ed68a0
OBJ: TH1F    hint    integral : 0 at: 0x7fad4456b330
OBJ: TH1F    hwidth  width : 0 at: 0x7fad4831f5a0
OBJ: TH2F    h2      integral:vmax : 0 at: 0x7fad450ee800
KEY: TNtuple nt;1
root [7]
```

- Potete anche digitare il nome dell'istogramma e dare invio. Esempio:

```
root [7] hint
(TH1F *) 0x7fad4456b330
root [8]
```

- Se non avete messaggi di errore significa che ROOT riconosce la scrittura "hint" come indirizzo dell'oggetto corrispondente e potete utilizzarlo per tutte le operazioni associate all'oggetto.
- `hint->GetXaxis()->Set(200, da, a);` // ricalibra l'asse X modificando il range dei 200 bin che ora andrà da "da" (corrispondente al valore 3000 grezzo) ad "a" (corrispondente a 5000).

Disegni e grafici utili sulla camera:

