



LUNDS
UNIVERSITET



Positive Pion Photoproduction Near Threshold

Master Thesis Project-II, Lund 2012

Author:

Ayaz Khan

Department of Physics

Division of Nuclear Physics

Lund University

Supervisor:

Prof. Bo Jakobsson

Department of Physics

Division of Nuclear Physics

Lund University

Abstract

Pion photo production from the nucleon is an important process utilized to understand the structure of the nucleon and the fundamental dynamics of the nucleon system. The neutral pion production has been extensively studied resulting in valuable data sets. However, much less data is available for charged pion photo production at energies below 200 MeV . In order to get a reliable data set for π^+ , for the energy range from threshold to the Δ -resonance production, a photonuclear reaction ($\gamma + p \rightarrow n + \pi^+$) experiment was conducted at Max-lab. In the experiment a photon beam with energies from 169 to 180 MeV passed a circular collimator and hit a target (C, Ta). Charged Pions were produced well below threshold. The data was stored in Root-tree files and analyzed using C++ programming codes. The technique to identify π^+ in the experiment was based on probing the 26 ns $\pi\mu$ decay in Scintillator Range Telescopes. The differential cross section for π^+ as well as for protons was determined. The results are consistent with the theoretical predictions and measurements performed by experts at Max-lab.

Keywords: *Photonuclear reaction; π^+ production; Range telescope; Differential Cross Section*

Sammanfattning: Pionproduktion i foton-nukleon kollisioner är en betydelsefull process för att förstå nukleonens struktur och dynamik. Produktionen av neutrala pioner har studerats intensivt vilket gett flera set av data. Däremot finns det mycket mindre data för laddade pioner vid energier $< 200\text{ MeV}$. För att råda bot på detta har $\gamma + p \rightarrow n + \pi^+$ reaktionen studerats vid energier från tröskeln och upp till deltaresonansen. I det experiment som redovisas här har foton – kärn kollisioner i energier mellan 169 och 180 MeV studerats med strålmål av kol och tantalum. Pionerna produceras här väl under tröskeln. Data lagras i ROOT-filer och analyseras med C++ program. Tekniken för att identifiera π^+ i experimenten baseras på att identifiera 26 ns $\pi\mu$ sönderfallet i stackar av plastdetektorer. De differentiella tvärsnitten för både π^+ och p bestämdes. Resultaten är konsistenta med teoretiska beräkningar liksom med tidigare experiment.

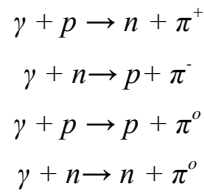
Nyckelord: *Fotonukleär reaction; π^+ produktion; plastdetektorer; differentiellt tvärsnitt*

Contents

1. Introduction	6
2. Experimental Setup (Max-lab).....	8
2.1. Electron gun and LINACs	8
2.2. MAX –I pulse stretcher ring	9
2.3. Nuclear physics cave.....	10
2.3.1. The Radiator	10
2.3.2. Tagging spectrometer.....	10
2.3.3. Bremsstrahlung Distribution	11
2.3.4. Electron-beam Dump	11
2.3.5. Focal Plane Detector Array	12
2.4. Photon Tagging	13
2.5. Photon beam and the collimator	14
3. Experiment <i>NP014</i>	15
3.1. Range telescope and Pion detection	16
3.2. Electronics set up and data acquisition system.....	17
4. Data Analysis.....	18
4.1. Proton Identification	19
4.2. Pion Identification.....	21
5. Results and Discussion	24
5.1. Double differential cross-section of protons	26
5.2. Double differential cross-section of pions	27
6. Conclusion.....	28
7. Acknowledgment	29
References.....	30
Appendices.....	31

1. Introduction

The pion is the lightest meson, made up of quark-antiquark pair [1]. It exists as π^+ , π^- or π^0 depending on its quark composition. The charged pions (π^\pm) have a mass of about $140 \text{ MeV}/c^2$ and mean life time of 26 ns . They decay mainly into muon/anti-muon and neutrino/anti-neutrino due to weak force. The π^0 has a slightly smaller mass of $135 \text{ MeV}/c^2$ and a much shorter lifetime of $8.40 \times 10^{-17} \text{ sec}$. It decays due to electromagnetic force and the main decay product is 2γ . Pions are naturally produced by the collision of cosmic rays with nuclei in the upper atmosphere [2]. Also one can produce pions in the laboratories by conducting different reactions like; Nucleon-Nucleon Reactions, Nucleon-Nucleus Reaction, Heavy- Ion Reactions and Photonuclear Reactions. The production of pions from the nucleon in a photonuclear reaction is an important process used to understand the structure of the nucleon and the underlying dynamics in the nucleon system [3]. This is one of the low energy processes that can be utilized to test and formulate *QCD-based* theories, e.g. the *QCD* theory about the strong nuclear force and the interaction between quarks and gluons. There are four kinds of reactions resulting pion photoproduction from nucleon;



The π^0 production from $\gamma + p \rightarrow p + \pi^0$ reaction has been extensively studied resulting about 1200 data sets for π^0 reactions. These data points cover the whole energy range from threshold to above the Δ -resonance. However, much less data is available for π^+ and π^- reactions with fewer than 50 data points at energies below 200 MeV .

In order to get data for the energy range from threshold to the Δ -resonance production, it is necessary to conduct $\gamma + p \rightarrow n + \pi^+$ reaction experiments. At Max-lab research facility, the photon can be tagged up to 200 MeV and hence is an ideal place for the positive pion photoproduction experiments [4]. In June, 2009, such an experiment (*see Experiment NP014*) was conducted at Max-lab. The π^+ were produced and detected by different detectors.

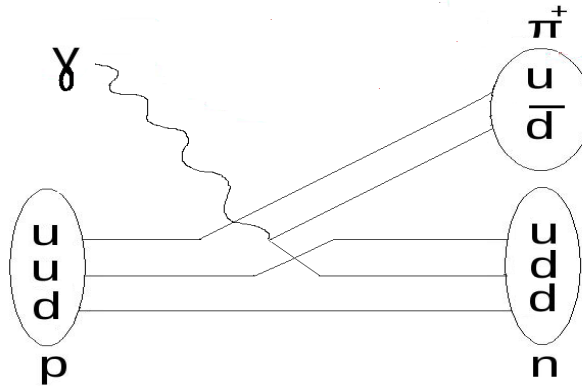


Fig 1. Feynman Diagram for the $\gamma + p \rightarrow \pi^+ + n$ process

The aim of this project was to determine the differential cross section of pions produced in the target material (carbon, Tantalum) and detected by Scintillator Range Telescopes.

The first part of this report describes the experimental set up (Max-lab), next comes the experiment (*NP014*) followed by the data analyses and results obtained. And the last part includes appendices containing the C^{++} codes used for the analysis.

2. Experimental Setup (Max-lab)

2.1. Electron gun and LINACs

The electrons are initially produced by an electron gun which is operating on the principle of thermionic emission. These electrons are emitted from the electron gun in short pulses. These pulses are 200 ns long (see Fig 2) having an energy of about 2 MeV and produced at a frequency of 10 Hz .

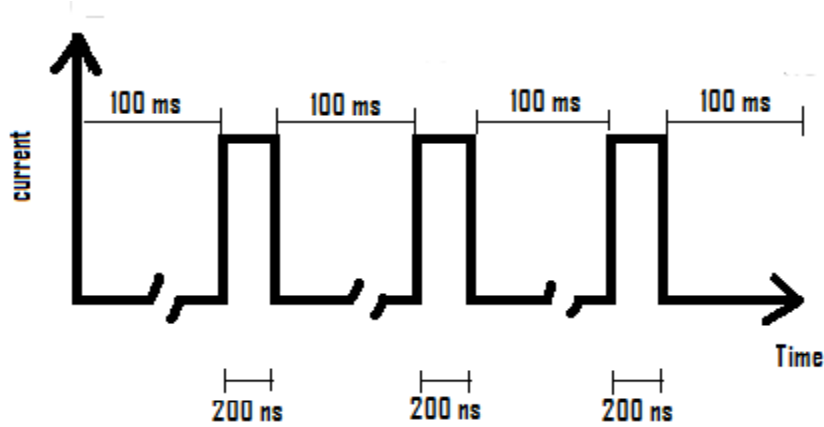


Fig 2. Electron pulses produced by the electron gun

These electron pulses enter a consecutive pair of two linear accelerators (LINACs). Each accelerator is 5.2 m long and has the capacity to increase the energy of the electrons up to $\sim 125\text{ MeV}$. Thus the two accelerators together can produce electron pulses with $\sim 250\text{ MeV}$ of energy.

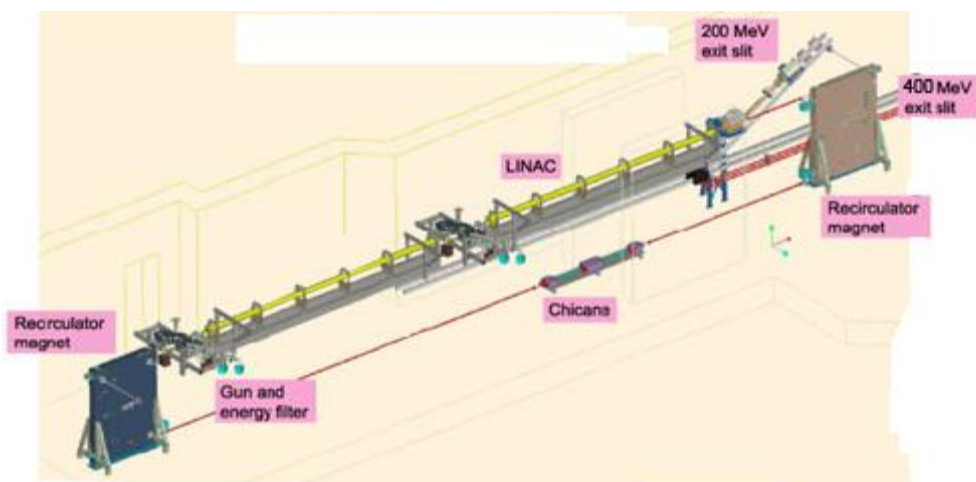


Fig 3. Overview of the electron gun and the two consecutive linear accelerators [5].

2.2. MAX –I pulse stretcher ring

Though, the electrons are accelerated to a usable energy by *LINACs*. However, the pulsed electron beam could lead to freeze up the detectors because of an overload of events and hence, is not suitable for data acquisition.

In order to obtain a steady electron beam with lower continuous current rather than short pulses of high instantaneous current, the electron pulses coming from the accelerators are injected into *MAX –I* pulse stretcher ring (see Fig 4).

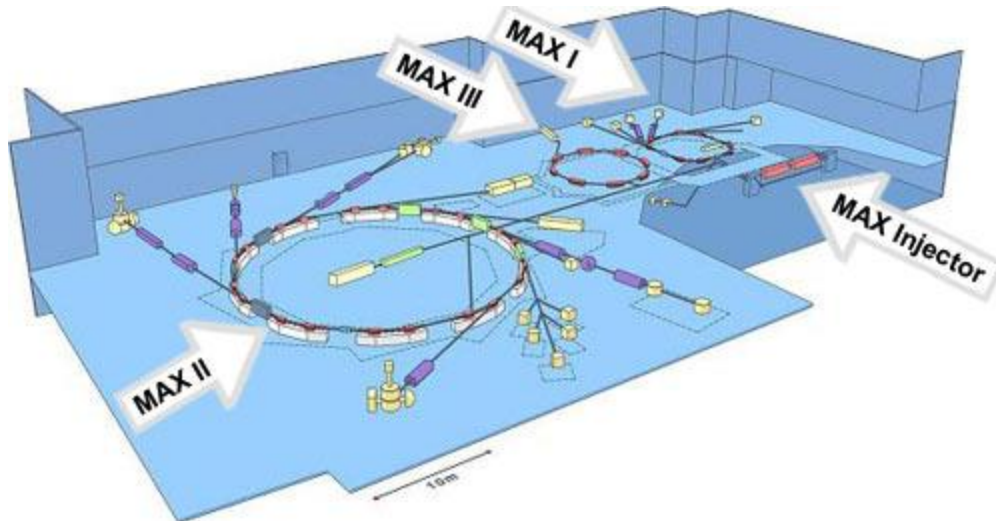


Fig 4. An overview of MAX-lab [6].

The *MAX I* pulse stretcher ring is 32.4 m in circumference and the electrons can circle the ring once in 108 ns while traveling almost at speed of light. Therefore, a single 200 ns electron pulse is injected to the *PSR* in approximately two turns.

From the *PSR*, the pulse is then slowly extracted in a period of 100 ms until the next pulse is reached from the injector [7]. This slow extraction results in an increase in pulse length by a factor of about 10^5 .i.e. from 100 ns to 100ms (see Fig 5), and thus an almost continuous –wave (CW) of electron is generated.

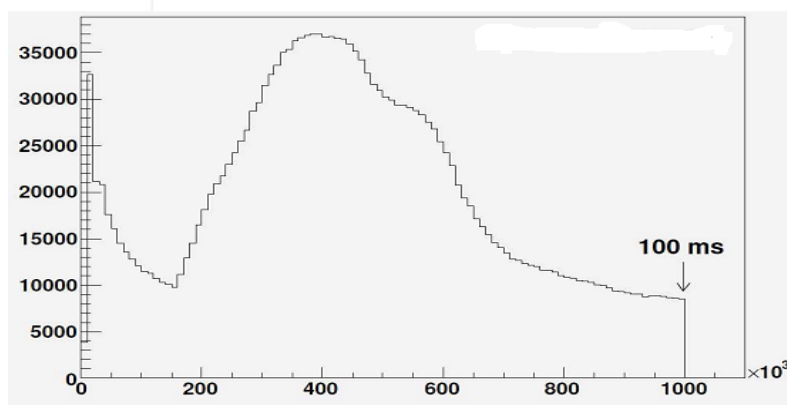


Fig 5. The time profile of the extracted beam from MAX I [8].

2.3. Nuclear physics cave

The stretched beam of electrons exiting the Max-I ring is transported in a beam line into the nuclear physics cave (see Fig 6). Different components dealing with the electron beam in nuclear physics area are described below.

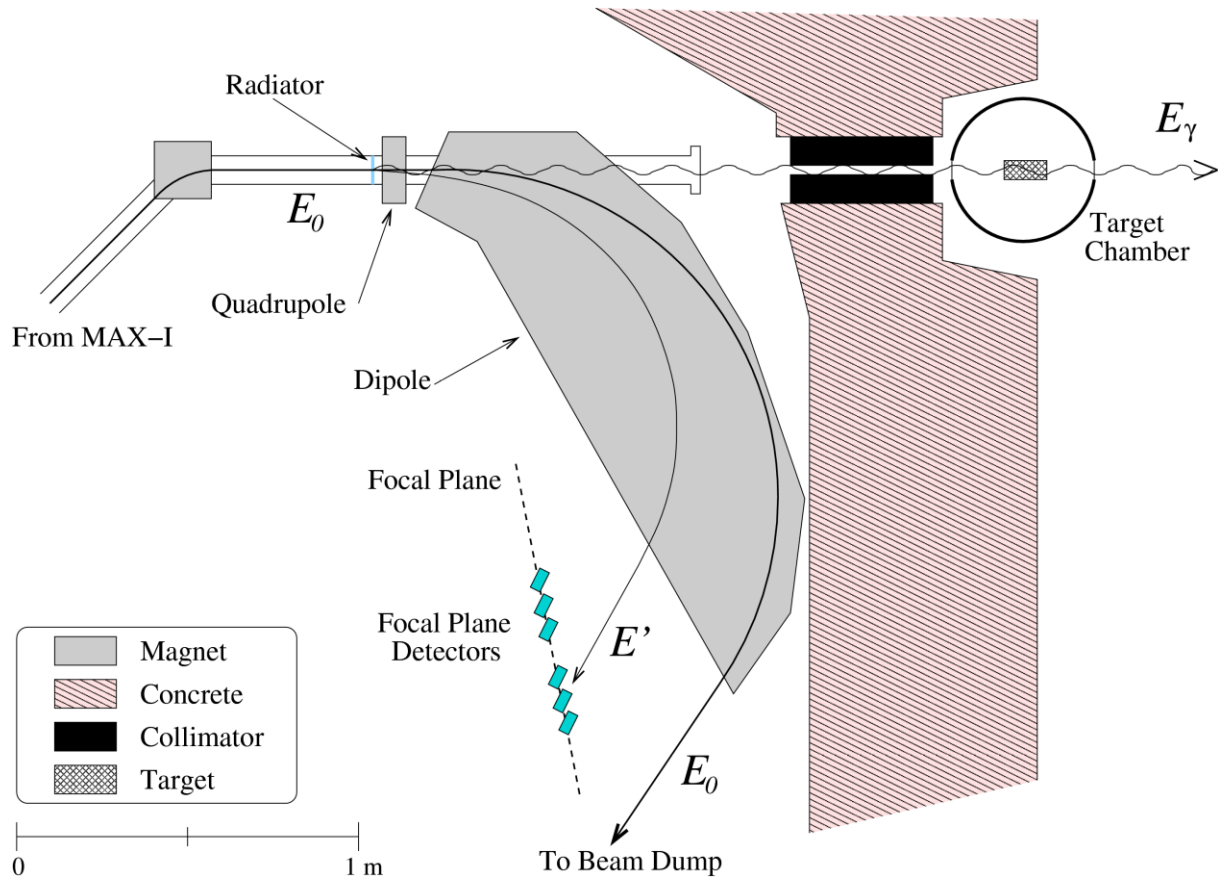


Fig 6. An overview of a typical radiator, tagging spectrometer, focal-plane detector array, and collimator setup [9].

2.3.1. The Radiator

In order to produce bremsstrahlung, the beam strikes a radiator, which is a thin conducting foil. Different materials of different thicknesses can be used as a radiator. However, commonly a $150/300 \mu\text{m}$ thick aluminium foil is utilized for this purpose. Some of the incident electrons on the foil are slowed down by electromagnetic interactions with the nuclei in the target, and send out bremsstrahlung [8].

2.3.2. Tagging spectrometer

A large dipole magnet, known as tagging spectrometer is placed after the radiator. The photons pass through magnet unaffected while the electrons experience a Lorentz force that makes them move in a curved path. The energy of the incoming electron " E_0 " is known from the electron beam energy, while

the energy of the decelerated electron “ E ” that emits photon is determined by the tagging spectrometer. The energy of the emitted photon is given by [8].

$$E_\gamma = E_o - E \quad (2.1)$$

Where E_o and E represent the energy of the incoming and recoling electron respectively.

2.3.3. Bremsstrahlung Distribution

Since, all electrons do not interact with the foil and hence all are not slowed down equally. Consequently a continuous distribution of bremsstrahlung photon energies is obtained, ranging from zero (no photon) up to the energy of the incoming electrons as shown in Fig 7 [8].

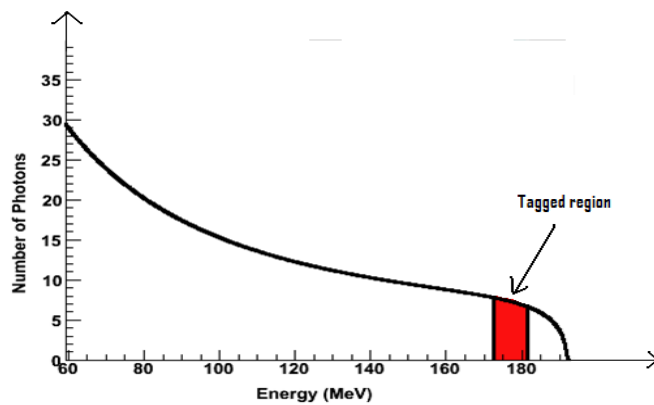


Fig 7. Bremsstrahlung spectrum; the number of photons as a function of their energy.

2.3.4. Electron-beam Dump

The beam of electrons that does not interact with radiator is bent by the Tagger into a shielded beam dump and collected by a Faraday cup [8]. (see Fig 8).

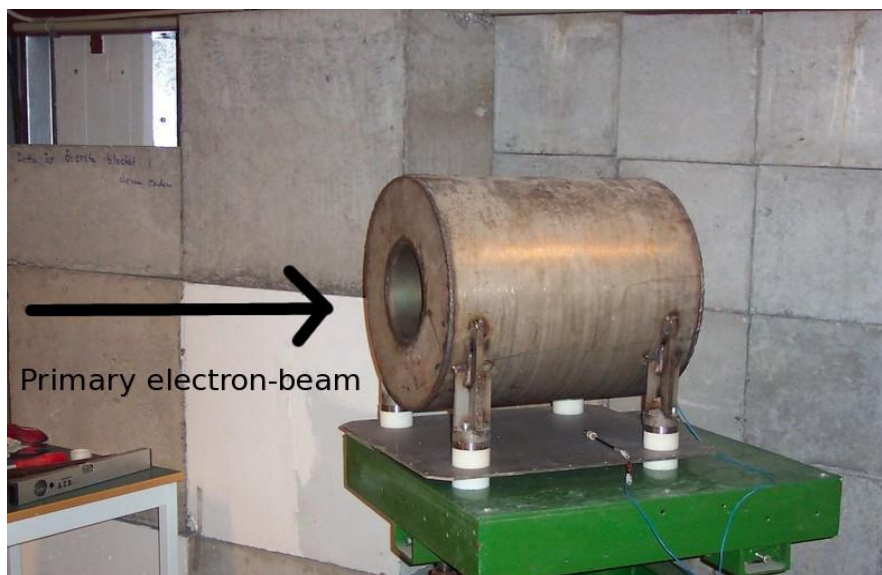


Fig 8. Photo of the unshielded electron beam dump. This dump is shielded by a concrete wall [7].

The radius of the path of electrons is determined by their energy; $r = \frac{\gamma m v}{qB}$ (2.2)

The electrons that have lost more energy in the radiator follow a path with a smaller radius as compared to the energetic electrons.

However, the electrons exit the radiator in a cone and consequently, even electrons with the same energy move in different directions. Therefore, the magnetic field of the spectrometer magnet is made in such a way that it focuses all electrons with the same energy towards a horizontal line outside the magnet. This line is known as the focal plane [8].

2.3.5. Focal Plane Detector Array

The path and energy of the electrons through the tagger magnet is determined by a hodoscope placed along the focal plane (see *Fig 9*).



Figure 9. Photo of the hodoscope. In the figure, the black sticks are the wrapped scintillators and light guides and the metallic tubes contain the *PMTs* [8].

The hodoscope is a detector array, used to detect the position of the electrons coming from the tagger spectrometer. It contains 64 plastic Scintillator detectors, which are aligned in two parallel rows and paired into 32 focal plane “channels” [7]. The two rows of detectors can be moved with respect to each other, and different overlaps can be obtained (*see Fig 10*).

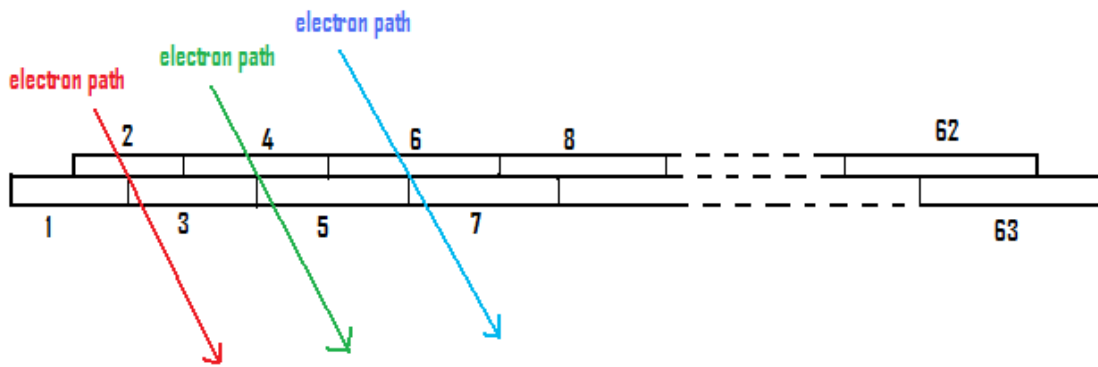


Fig 10. Focal plane channel

A signal is produced when an electron passes through two overlapping scintillators [8]. One problem with the scintillation detectors is that background radiation can strike a detector. Therefore these detectors are arranged in channel format and an event is processed only if the signal is produced simultaneously in two overlapping or adjacent scintillators depending upon which pair of detectors was hit by an electron. This suppresses the background radiation events which cannot produce coincident signals [7].

2.4. Photon Tagging

As discussed earlier, the energy of each bremsstrahlung photon can be determined by detecting the recoiling electron. A photon that corresponds to an electron whose energy has been determined by the spectrometer is known as “tagged” photon. In other words the photon has an “energy tag” and a “time tag”.

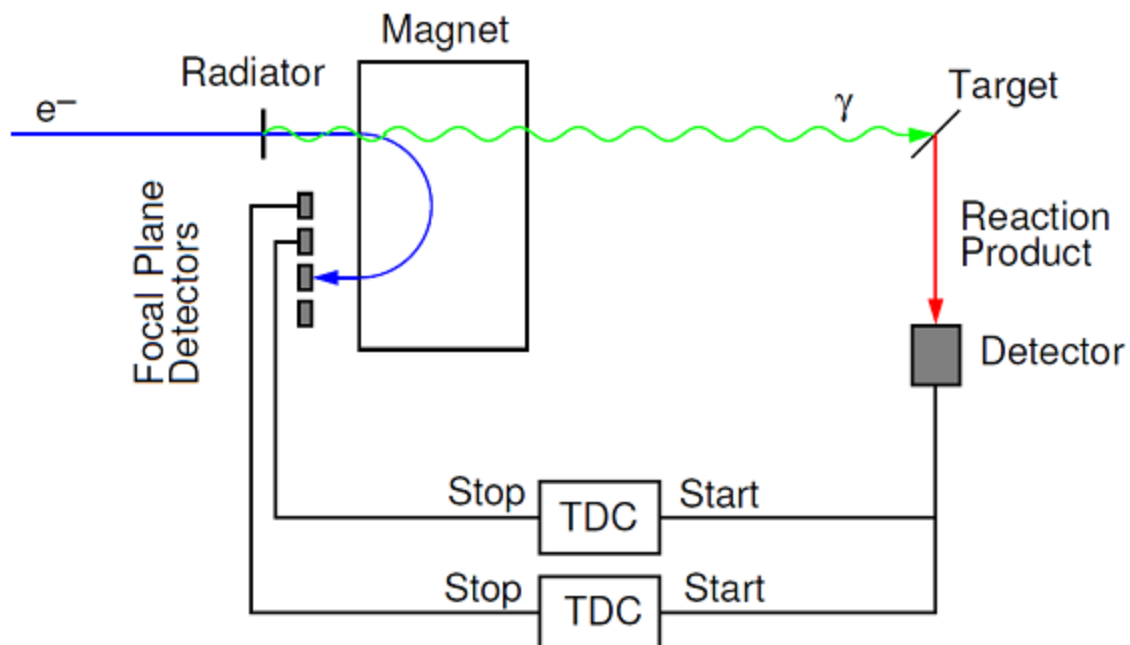


Fig 11. A schematic diagram of tagging of bremsstrahlung photons [8].

The photon tagging technique is schematically shown in Figure 11. When a reaction product is detected from a reaction induced by a photon, a start signal is sent to a set of TDCs (time-to-digital

converter clocks). Whereas the *TDCs* are stopped by signals from the detector channels in the hodoscope.

The electron that emitted the photon that caused a reaction in the target reaches the hodoscope within a certain time span. The time correlation between the detection of the electron and the detection of an event caused by its corresponding photon is used to correlate the photon to the right electron and hence determine the energy of the individual photons [8]. This process is known as “photon tagging”.

Long cables are used to delay the signals from the hodoscope so that they arrive after the start of the *TDCs*. Starting the *TDCs* by signals caused by a reaction product in the detector and not by an electron in the hodoscope is due to the fact that many photons do not react or even reach the target.

Therefore it is aimless to start the system for photons that do not participate in the experiment, especially because, it is impossible to determine their energy unless they cause a reaction in the target. Though each detector in the focal plane is connected to an individual *TDC*, however all of them start at the same time. If the *TDC* value of any of the focal plane channels lies in a certain time limit (known as prompt region) the start and stop signal can originate from a reacting photon and the corresponding electron, which makes it possible to determine the energy of that photon [8].

2.5. Photon beam and the collimator

The bremsstrahlung photons are emitted in a conical distribution and the target is placed several meters from the radiator [8]. Because of the large distance between the radiator and target, the spread of the cone could be so large that some the photons would miss the target. Hence, it is necessary to collimate the bremsstrahlung photons into a well defined beam.

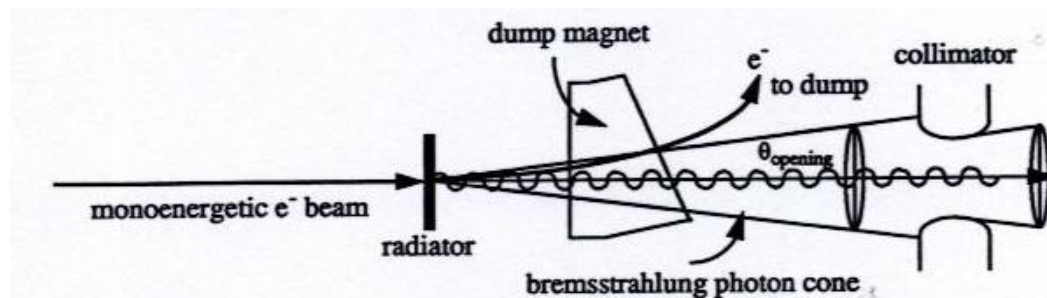


Fig 12. A sketch showing the bremsstrahlung-photon beam entering the collimator [10].

For this purpose, a 15 cm long collimator made of “heavimet” (tungsten alloy) having a hole diameter of 12 or 19 mm is placed behind the spectrometer magnet. The collimator is aligned with the beam axis. Consequently, only photon having a small angular distribution with respect to the beam axis will reach the target.

One issue with collimation of photon is that all photons do not pass through the hole but rather hit the collimator material and result in creation of different particles. To make sure that as few of these as possible hit the target, a strong magnet is used downstream from the collimator that deflects the charged particles created in the collimator. Further downstream, aligned with the first collimator, another collimator known as “scrubber” is placed that removes particles created in the collimator.

Hence, a well defined beam of photons is obtained and used for photonuclear experiments [8].

The target is placed in the beam of photons. Some of the photons will interact with the nuclei in the target material and results in creation of different particles. The particles that are created in the photonuclear reaction come out of the target and detected by one or more detectors.

3. Experiment NP014

The experiment NP014 (pion Photoproduction measurement) was conducted in June, 2009. A schematic diagram of the experimental setup is shown in Fig 13.

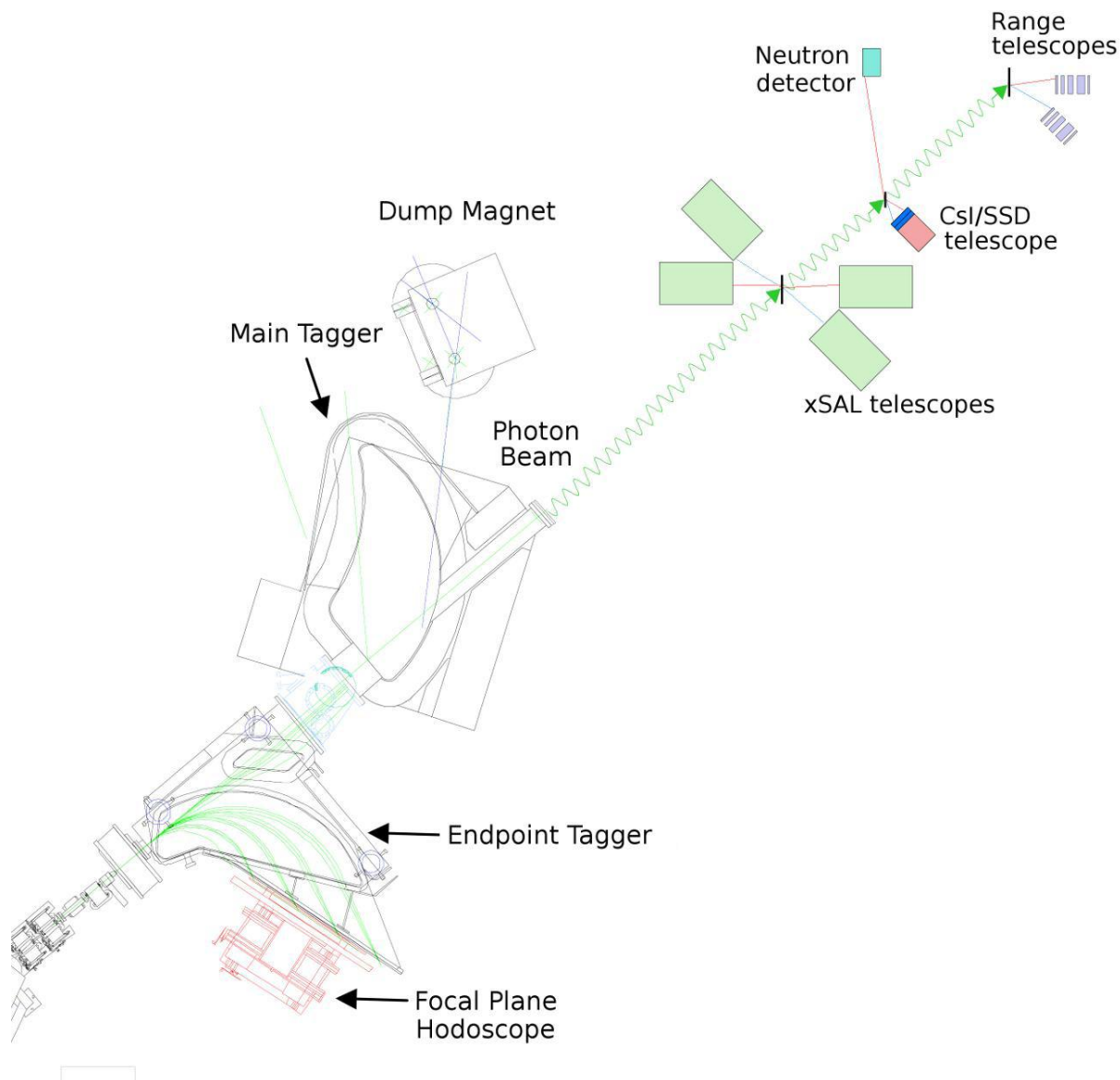


Fig 13. Experimental setup for the experiment NP014. (Courtesy of Dr. Brudvik)

As discussed in the previous section, the collimated beam of photons hit the target and the π^+ produced in the target materials were detected by *xSAL* Telescopes, *CsI/SSD* Telescope and Scintillator Range Telescopes. The solid target materials, used in the experiment were Methylene (CH_2), Carbon (C) and Tantalum (Ta). *Target empty* runs were made to measure the background rates. The stability of the photon beam was monitored by determining the tagging – efficiency on daily basis [11].

3.1. Range telescope and Pion detection

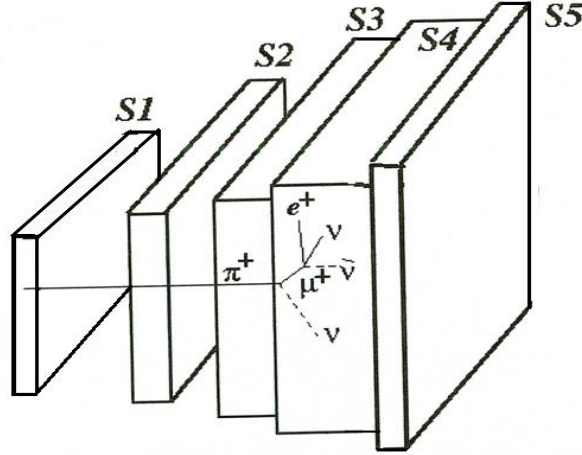


Fig 14. π^+ detection process in the range telescope.

Two plastic Scintillator telescopes (α/β) were used in angular positions $30^\circ/90^\circ$ and $60^\circ/120^\circ$ to detect pions in the experiment. Each telescope had five (NE102 plastic Scintillator) detectors, which were read out by (Philips XP2020) photomultipliers [12]. The parameters of the telescopes are given in Table 1.

Table 1. Proton and Pion energy bins (T_p, T_π) Using two(α/β) - range telescope

Detector	Thickness (mm)	Area α (mm ²)	Area β (mm ²)	T_p (MeV)	T_π (MeV)	Comment
S_1	5	60 x 60	70x70			Only ΔE
S_2	10	80 x 80	90x90	22 – 42	11.5 – 19.3	
S_3	20	90 x 90	100x100	42 – 66	19.3 – 30.2	
S_4	50	110 x 110	120x120	66 – 108	30.2 – 49.8	
S_5	5	130x130	140x140			Only Veto

The identification technique of the pion was based on probing the 26 ns $\pi\mu$ decay in the telescope (see Fig 14).

To detect Pion, one needs two energy signals. In this case, the low-energy limit was set by the thickness of S_1 . Taking into account the minor energy loss that pions can experience in the target

4. Data Analysis

Usually, different particles are identified by determining the energy loss of the particles in the detector material according to Bethe-Bloch formula.

$$-\frac{dE}{dx} = \frac{4\pi n z^2}{m_e c^2 \beta^2} \left(\frac{e^2}{4\pi\epsilon_0} \right)^2 \left[\ln \left(\frac{2m_e c^2 \beta^2}{I(1-\beta^2)} \right) - \beta^2 \right]$$

Where:

n is the density of the detector material

Z is the charge of the incident particle

m_e is the mass of electron

$\beta = v/c$ (v = Velocity of the incident particle and c = speed of light) and

I is the ionization energy of the stopping material

The energy loss of the particle depends on the charge and velocity of the particles as well as the nature of the detector material. Since the velocity of the particle depends both on the energy and mass of the particle, therefore different particles having the same energy but different mass would have different rate of energy loss [11]. (See for example protons and deuterons band in Fig 16).

In Scintillator Range Telescopes the particles were identified by looking at the energy deposited by them in a thin (ΔE) detector compared to the total energy (E) of the particles. The ΔE vs. E plots are called stopping power plots and reveal different bands for different types of particles as shown in Fig 16.

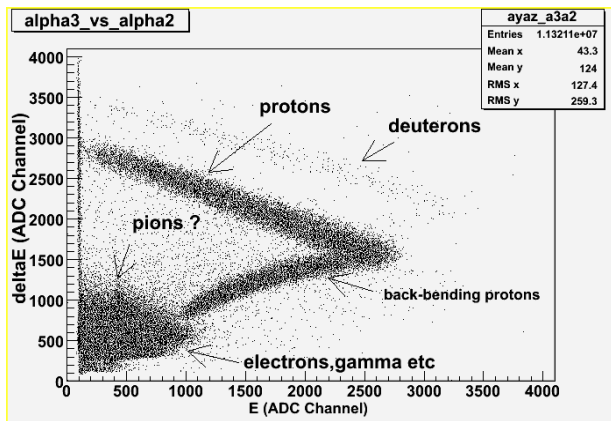


Fig 16. Example of ΔE - E plot for particles stopping in S_3 detector of the α telescope.

Such histograms were plotted for particles stopped in S_2 , S_3 and S_4 detectors of the α and β telescopes. Fig 16 represents a very clear proton band due to stopping in the third detector of the α telescope, however it also contains deuterons, pions, electrons and gamma etc along with back-bending protons. The back-bending protons are the ones that pass S_4 , i.e. these protons are not stopped by any of the stopping detectors S_2 , S_3 and S_4 and hence give two ΔE signals unlike to the protons that are stopped by any of the stopping detector and produce a ΔE and an E signal.

Because of the low statistics it was difficult to select pions directly from the raw data in the pion plots. Therefore it was decided to present first proton identification and proton cross section determination which is easier to follow and provides an important check of the tagger efficiency. For this purpose the following scheme was adopted.

4.1. Proton Identification

First, the bit-pattern conditions were used in order to define the stop detector (S_2 , S_3 or S_4) and get rid of back-bending protons in ΔE - E plots as shown in Fig 17.

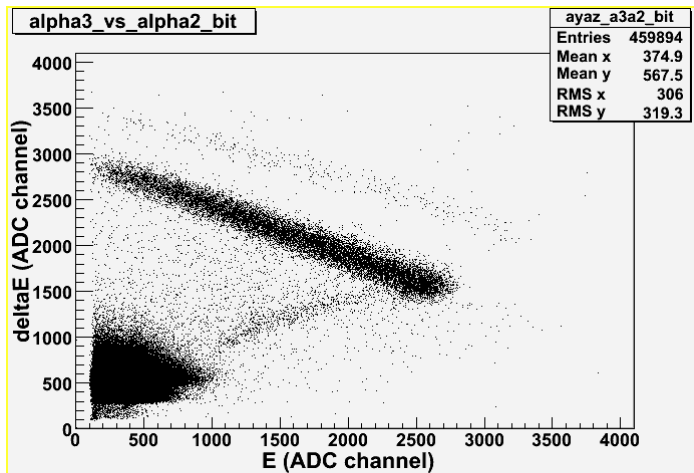


Fig 17. ΔE - E plot with bit-pattern conditions for particles stopping in S_3 detector of the α telescope.

In Fig 17, the back-bending protons are vanished but deuterons are still present as well as pions and electrons etc in the background. The next step was therefore to implement a cut into the histogram and get protons only (see Fig 18).

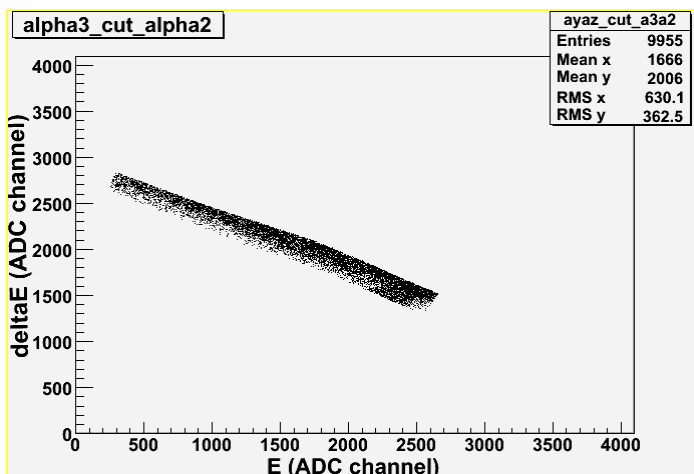


Fig 18. ΔE - E plot for protons stopping in S_3 detector of the α telescope after implementing a cut.

The protons obtained in this way were used to determine the tagger coincidence peak. Because of the different flight time, the tagger coincidence peak was determined for each stop detector individually (see Fig 19).

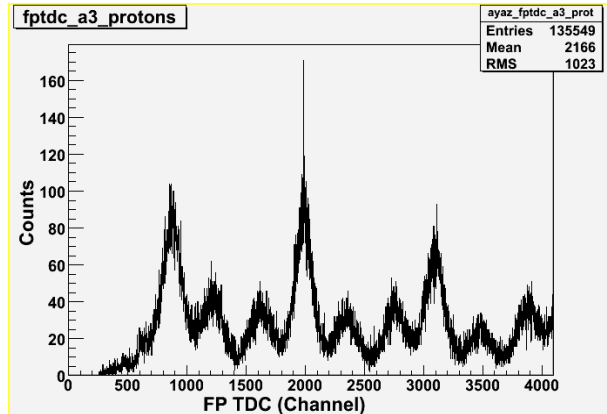


Fig 19. The TDC spectrum for tagger signals in coincidence with protons stopping in S_3 detector of the α telescope.

A gate is set over the tagger coincidence peak for each stop detector. The dynamical width (the difference in flight-time between the slowest and the fastest protons that stop in the detector) is of the order of $2 - 3$ ns. However, the detectors and electronics have also a contribution in the actual dispersion of the tagger spectrum and thus lead to uncertainty in the total dispersion.

In order to get the proper number of protons remaining after tagger background subtraction, a two-Gaussian fit was made to the tagger TDC spectrum (see Fig 21).

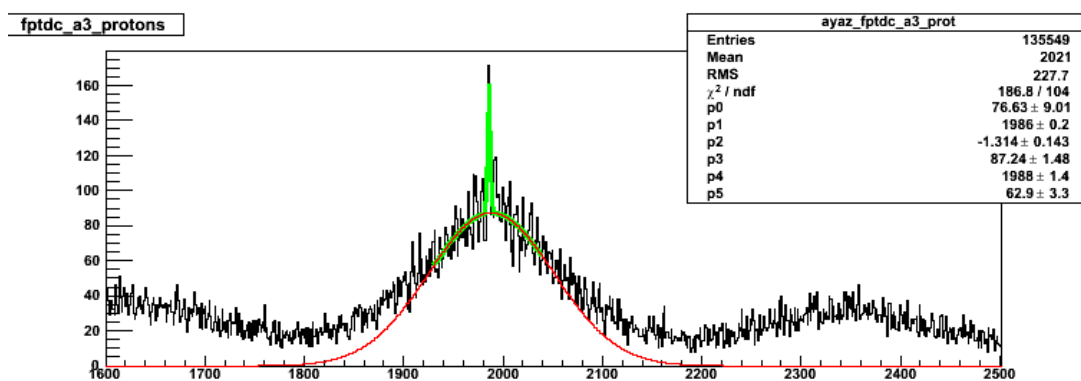


Fig 21. Two-Gaussian fit to a typical tagger spectrum for protons selected as described in the text.

The peak position, determined as above was then fixed. The wide distribution representing the background was determined from the region well outside the coincidence peak region. The narrow distribution showing the coincidence protons was determined from the central part only. This central part was then allowed to vary between 1 and 80 channels corresponding to $0.25 - 20$ ns and subtracted the background Gaussian from the signal i.e. the number of counts in each channel. Afterwards, the number of protons (N_p) obtained was plotted as a function of gate width (Δt) as shown in Fig 22.

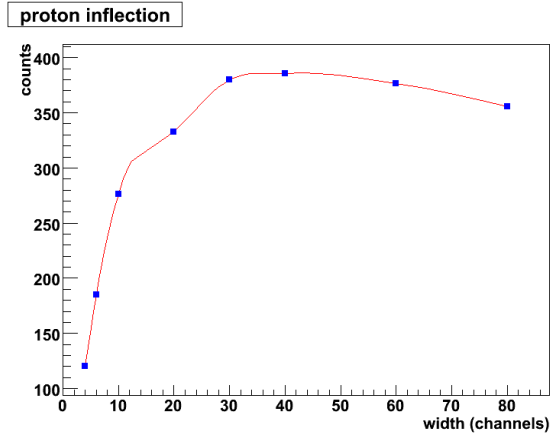


Fig 22. Example of N_p as a function of Δt in S_3 stopping detector of α telescope.

If the fit would be perfect, a constant level of N_p should be found when all true (signal) protons have been included [13]. If the N_p continues to increase or decrease, this is due to an imperfect background fit (which may be improved by an iterative process where e.g. the background constantly changed until N_p stays constant).

4.2. Pion Identification

As discussed in section 3.2, the data acquisition system recorded the energy deposited by the particles in E -detector for two different integration periods. The signal from the E -detector was split and sent to two separate QDCs with two individually adjustable gates (prompt and delay). All particles produced the same signal in both short and long gate QDCs except pions that produced an extra signal in the E -detector due to $\pi \rightarrow \mu$ decay (the muon has an energy of about 4.2MeV). This extra energy/signal was recorded by the long gated ADC. Thus pions can be identified by looking at this extra energy deposited by them in the prompt vs delay plots.

Therefore, first the prompt vs. delay histograms were plotted for each stopping detector (S_1 , S_2 , S_3) using the bit-pattern conditions as previously done for protons (see Fig 23).

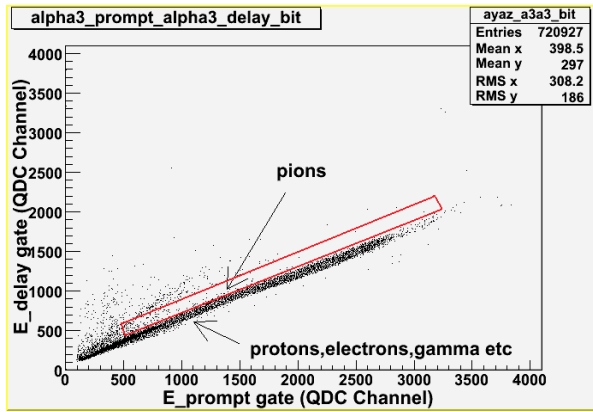


Fig 23. Example of Prompt vs. Delay plot for particles stopping in S_3 detector of the α telescope.

From Fig 23, it is clear that most of the particles lie along a narrow line corresponding to the same deposited energy in both prompt and delay-gate QDC s, whereas a small number of particles form a weaker parallel line above the main line. These particles have deposited extra energy (presumably from the 4.2MeV muon) in the delay gate compared the prompt gate [11] and hence were considered as pions.

The pions were then isolated by implementing a cut on Prompt vs. Delay histogram (see Fig 23). The cut was implemented close to the main band (consisting of protons, electrons etc) in order to avoid too many muons providing less than 4.2 MeV energy in the stopping detector. However, those points were excluded that were obviously shifted by much more energy than 4.2 MeV .

The pions selected by the above mentioned method were then plotted as stopping power ($\Delta E-E$) histogram as shown in Fig 24.

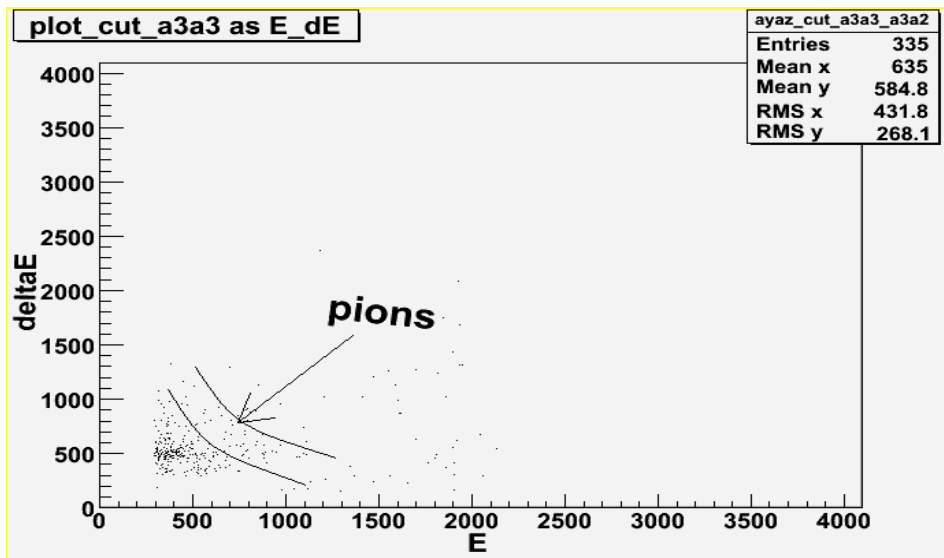


Fig 24. $\Delta E-E$ plot for points selected as π^+ in Fig 23.

Fig 24 gives an example where statistics is on the verge of being able to define the pion band. However, there remains a small fraction of protons and a large number of electrons and gammas in the background. In order to get the proper number of pions the same procedure was repeated as that for

protons in the previous section. The pions were selected by using the bit-pattern condition, prompt vs. Delay condition and $\Delta E-E$ condition (see appendices for the C^{++} codes).

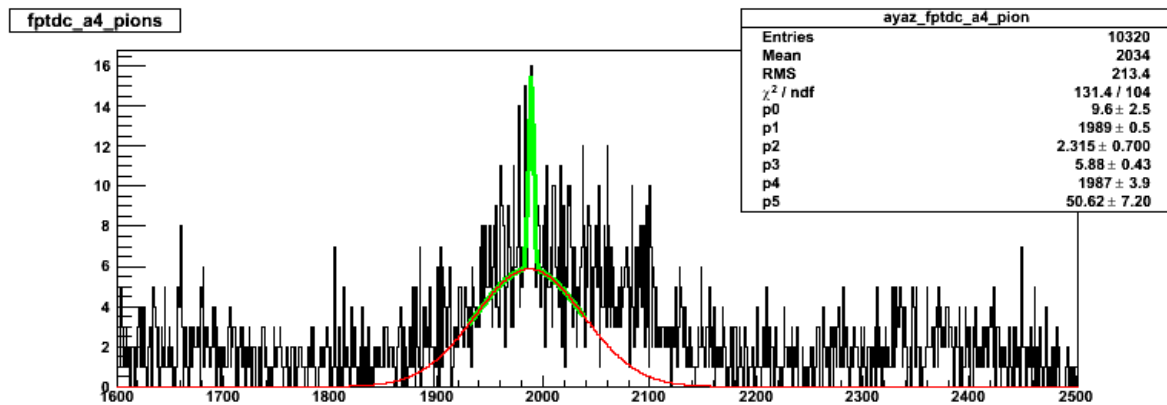


Fig 25. Example of TDC spectra for tagger signals in coincidence with pions stopping in detector S_4 of the α telescope.

The final number of pions (N_{π^+}) was again taken as the constant level of $N_{\pi^+}(\Delta t)$ function. Because of the low statics it is obvious that large errors are introduced by this method (see *Fig 25*) to determine the remaining background.

5. Results and Discussion

Cross section is a measure of the probability of a particular reaction occurring during the interaction of two particles. If a beam of particles is broader than the target, then one can consider the flux of the incident particles and define the differential cross section as the average fraction of particles through a solid angle ($d\Omega$) per unit time per unit flux [14].

$$\frac{d\sigma}{d\Omega}(E, \Omega) = \frac{1}{F} \frac{dN_s}{d\Omega} \quad (5.1)$$

where

F represents the flux of the incident particles

E stands for the energy of the incident particles

dN_s shows the number of scattered particles per unit time and

$d\Omega$ is the solid angle for the scattered particles

From eq (5.1), the total cross section (σ) is given by

$$\sigma(E) = \int d\sigma \frac{d\sigma}{d\Omega} \quad (5.2)$$

If a beam of X particles is incident on a target nucleus Y and produces reaction products a and b , the differential cross section in this case can be determine by

$$\frac{d\sigma}{d\Omega}(X + Y \rightarrow a + b) = \frac{N_a(E, \Omega)}{N_X(E) \rho_Y t_Y} \quad (5.3)$$

where

N_X is the number of the incident particles X with energy E

N_a is the number of produced particles a through solid angle $d\Omega$ and

ρ_Y, t_Y represent the density and thickness of the target nucleus respectively.

However, one may need other additional terms to account for detector efficiency and other measurement effects [11].

In this experiment one additional term (f_{eff}) was used as an efficiency factor and the double differential cross-section was determined by

$$\frac{d^2\sigma}{d\Omega dE} = \frac{N_p}{\Phi_o C_T \Omega \Delta E} f_{eff} \quad (5.4)$$

where

N_p is the number of identified protons (pions) falling in the peak of $t_{tag}-t_{trig}$ distribution remaining after (Gaussian) background subtraction is performed.

Φ_o is the total number of photons hitting the (experimental) target = total no of scaler counts \times tagger efficiency (0.24 in average)

$$\text{i.e. } \Phi_o = \sum_{i=1}^{32} N_{sc,i} \cdot F_{eff,i} \quad (5.5)$$

($N_{sc,i}$ is the number of hits in the i^{th} scaler and $F_{eff,i}$ is the efficiency of the i^{th} scaler)

$$\Rightarrow \Phi_o = 32 \cdot 18 \cdot 10^9 \cdot 0.24 = 138 \cdot 10^9 \quad \text{for carbon}$$

$$\text{and } \Phi_o = 32 \cdot 79.2 \cdot 10^9 \cdot 0.24 = 608 \times 10^9 \quad \text{for Tantalum}$$

C_T is the number of target atoms/cm²

$$\text{i.e. } C_T = x \cdot \rho \cdot \frac{N_{Av}}{A} \cdot \sin^{-1} \theta_T \quad (5.6)$$

[x is the target thickness (cm), ρ is the target density (g/cm³), $N_{Av} = 6.023 \times 10^{23}$ (atoms/g.atom), A is the mass number of the target and θ_T is the target angle ($\theta = 45^\circ$) to the beam ($\Theta_{lab} = 90^\circ$)]

$$\text{So } C_T = 0.05 \cdot 2.22 \cdot \frac{6.023 \times 10^{23}}{12} \cdot \sin^{-1}(45) = 7.88 \times 10^{25} \text{ atoms/cm}^2 \text{ for carbon \&}$$

$$C_T = 0.01 \cdot 1.65 \cdot \frac{6.023 \times 10^{23}}{181} \cdot \sin^{-1}(45) = 7.84 \times 10^{21} \text{ atoms/cm}^2 \quad \text{for Tantalum}$$

Ω is the effective solid angle = 65msr (for $\Theta = 90^\circ$)

ΔE is the energy bin for protons (pions) stopping in detector S_2 , S_3 and S_4 of α/β telescope (see section 3.1) and

f_{eff} is the efficiency factor for protons (pions).

In case of protons, the efficiency correction contains two factors i.e. $f_{eff} = f_1 \cdot f_2$

Where f_1 is the correction factor due to the reaction in the detector material and f_2 is used for the geometry of the detector. However, only loss due to reaction in the detector material is important. In a first approximation this factor could be calculated from the assumption of an infinite detector in which (CH, density 1.03g/cm³) the proton mean free path is ~ 4cm. An old reference [15] suggests that such an assumption is reasonable. Following this assumption, $f_{eff} = 1.07$ for S_2 , $f_{eff} = 1.21$ for S_3 and $f_{eff} = 1.38$ for S_4 detectors of the alpha telescope (analytical).

In case of pions the efficiency correction includes three factors i.e. $f_{eff} = f_1 \cdot f_2 \cdot f_3$. Where f_1 stands for the decay in flight, f_2 for the reaction in the detector material and f_3 for the geometrical shortcomings including losses of muons. The analytical calculations of these factors give

$f_{eff} = 3.39$ for S_2 , $f_{eff} = 3.04$ for S_3 and $f_{eff} = 2.47$ for S_4 detectors of the alpha telescope.

5.1. Double differential cross-section of protons

The double differential cross sections of protons were determined for Carbon and Tantalum according to eq (5.4) and given in Table 2 and Table 3 respectively.

Table 2. Double differential cross sections of protons for C^{12} data using α telescope ($\Theta_{lab} = 90^\circ$)

Det	ΔE (MeV)	N_p	C_T (atoms/cm ²)	Ω (msr)	Φ_o	f_{eff}	$d^2\sigma/d\Omega dE$ ($\mu\text{b}/\text{sr}\cdot\text{MeV}$)
S_2	22-42	500	7.88×10^{23}	65	1.38×10^9	1.07	0.38
S_3	42-66	380	7.88×10^{23}	65	1.38×10^9	1.21	0.31
S_4	66-108	455	7.88×10^{23}	65	1.38×10^9	1.38	0.21

Table 3. Double differential cross sections of protons for ^{181}Ta data using α telescope ($\Theta_{lab} = 90^\circ$)

Det	ΔE (MeV)	N_p	C_T (atoms/cm ²)	Ω (msr)	Φ_o	f_{eff}	$d^2\sigma/d\Omega dE$ ($\mu\text{b}/\text{sr}\cdot\text{MeV}$)
S_2	22-42	115	7.84×10^{21}	65	608×10^9	1.07	0.19
S_3	42-66	100	7.84×10^{21}	65	608×10^9	1.21	0.17
S_4	66-108	85	7.84×10^{21}	65	608×10^9	1.38	0.21

5.2. Double differential cross-section of pions

The values of differential cross sections of pions determined according to equation (5.4) are given in Table 4 and Table 5 for C^{12} and ^{181}Ta respectively.

Table 4. Double differential cross sections of pions for C^{12} data using α telescope ($\Theta_{lab} = 90^\circ$)

<i>Det</i>	ΔE (MeV)	N_π^+	C_T (atoms/cm ²)	Ω (msr)	Φ_o	f_{eff}	$d^2\sigma/d\Omega dE$ ($\mu b/sr \cdot MeV$)
S_2	11.5 – 19.3	13	7.88×10^{23}	65	1.38×10^9	3.39	0.082
S_3	19.3 – 30.2	2	7.88×10^{23}	65	1.38×10^9	3.04	0.010
S_4	30.2 – 49.8	0.2	7.88×10^{23}	65	1.38×10^9	2.47	0.0006

Table 5. Double differential cross sections of pions for ^{181}Ta data using α telescope ($\Theta_{lab} = 90^\circ$)

<i>Det</i>	ΔE (MeV)	N_π^+	C_T (atoms/cm ²)	Ω (msr)	Φ_o	f_{eff}	$d^2\sigma/d\Omega dE$ ($\mu b/sr \cdot MeV$)
S_2	11.5 – 19.3	11	7.84×10^{21}	65	608×10^9	3.39	1.5
S_3	19.3 – 30.2	1	7.84×10^{21}	65	608×10^9	3.04	0.06
S_4	30.2 – 49.8	0	7.84×10^{21}	65	608×10^9	2.47	0.003

6. Conclusion

The photonuclear reaction experiment (*NP014*) with photon energies from *169 to 180 MeV* was conducted at Max-lab in order to collect data for π^+ , for the energy range from threshold to the Δ -resonance. The target materials used in the experiment were carbon (*C*) and Tantalum (*Ta*). The pions (π^+) were produced well below threshold and detected by Scintillator Range Telescopes. The data was stored in Root-tree files and analyzed by using *C++* programming codes. The double differential cross sections both for protons and pions were determined. The results are consistent with the theoretical prediction of pion photoproduction and measurements performed by senior scientists at Max-lab.

7. Acknowledgment

First of all the author wishes to express his profound gratitude to his supervisor, Dr. Bo Jakobsson for his invaluable assistance, support and guidance. The author is thankful to his co-supervisor Dr. Kevin Fissum for his kind instructions and suggestions for the analysis. The author expresses his grateful acknowledgement to Dr. Sugat Shende for his most valuable help in programing. The author wishes his special gratitude to Ulrika Forsberg for her unselfish and amiable help rendered throughout this study. The author extends his deepest gratitude to his former supervisor Dr. Kristina Stenström who made him capable of conducting research in a scientific environment. The author also expresses his thanks to Samiullah Khan who helped and supported him like a brother. Finally the author owes his special gratitude to his family members, particularly to his mother, whose unconditional love and prayers always helped him to achieve his goals and ambitions.

References

- [1] P. Haensel et al, (2007) Neutron Stars 1, Equation of state and structure. *Springer Science + Business Media, LLC.ISBN-10; 0-387-33543-9*
- [2] Y. Suzuki et al (2001) *Neutrino Oscillations and their origin. World Scientific Publishing Co.pte.Ltd.ISBN 981-02-4850-4*
- [3] R. Beck et al., Phys. Rev. Lett. 69, 1841 (1990).
- [4] G.V.O'Rielly et al,(2004). Charged Pion Photoproduction from threshold to the First Resonance Region. *A proposal submitted to Max-lab*
- [5] <https://www.maxlab.lu.se/node/263>
- [6] <http://www.maxlab.lu.se/maxlab/about/index.html>
- [7] M. Litwack, *Diploma thesis, Max-lab, Lund University (2010).*
- [8] U. Forsberg, *Diploma thesis, Max-lab, Lund University (2010).*
- [9] M. Karlsson, *PhD Thesis,Lund University (2005).*
- [10] K. Fissum, *PhD Thesis, University of Saskatchewan (1993).*
- [11] A.Colleen, *Diploma thesis, Max-lab, Lund University (2010).*
- [12] P. Golubev et al, *Nuclear Physics A 806,216-229(2008).*
- [13] Personal communication with Bo Jakobsson.
- [14] W.R.Leo.*Techniques for Nuclear and Particle Physics Experiments. Springer Verlag, New York, 2nd edition, 1994.*
- [15] L.H. Johnston et al., *Ire.Trans. Sci. (N3-5)(1958)5*

Appendices

Appendix 1

Appendix 1 includes the main C++ codes used for the analysis of the data. It should be noted that a number of histograms made during the analysis have been excluded because of the length of the program. Hence, it is possible that some codes might have been deleted while excluding unimportant things in the appendix. However, the author can provide the full analysis program upon request.

```
#include <stdio.h>
#include <stdlib.h>
#include "TFile.h"
#include "TTree.h"
#include "TROOT.h"
#include "TH1F.h"
#include "TH2F.h"
#include "TClonesArray.h"
#include "XEvent.h"
#include "ScalerEvent.h"
#include "XHist.h"
#include "ScalerHist.h"
#include "TStopwatch.h"
#include "TCutG.h"

//===== 12C data files=====
{
  "/home/ayaz/Desktop/jun09data/Data1095.root",
  "/home/ayaz/Desktop/jun09data/Data1096.root",
  "/home/ayaz/Desktop/jun09data/Data1097.root",
  "/home/ayaz/Desktop/jun09data/Data1098.root",
  "/home/ayaz/Desktop/jun09data/Data1099.root",
  "/home/ayaz/Desktop/jun09data/Data1117.root",
  "/home/ayaz/Desktop/jun09data/Data1118.root",
  "/home/ayaz/Desktop/jun09data/Data1119.root",
  "/home/ayaz/Desktop/jun09data/Data1120.root",
  "/home/ayaz/Desktop/jun09data/Data1121.root",

//===== 18F data files=====

  "/home/ayaz/Desktop/jun09data/Data1168.root",
  "/home/ayaz/Desktop/jun09data/Data1169.root",
  "/home/ayaz/Desktop/jun09data/Data1172.root",
  "/home/ayaz/Desktop/jun09data/Data1173.root",
  "/home/ayaz/Desktop/jun09data/Data1174.root",
  "/home/ayaz/Desktop/jun09data/Data1178.root",
  "/home/ayaz/Desktop/jun09data/Data1179.root",
  "/home/ayaz/Desktop/jun09data/Data1180.root",
  "/home/ayaz/Desktop/jun09data/Data1181.root",
};

int main()
{
  TROOT simple("simple", "offline");

  TStopwatch timer;
  timer.Start();
```

```

TFile *output = new TFile("rootfiles/myanalysis.root", "RECREATE");

output->mkdir("Cuts");
output->mkdir("Scalers");

XEvent *xevent = new XEvent();
TClonesArray *vq;
TClonesArray *vt5;
TClonesArray *tagtdc;
TClonesArray *tagtdc4;

TTree *TX;
TBranch *xbranch;

TTree *TS;
TBranch *scalerbranch;
ScalerEvent *scalevent = new ScalerEvent();

int filecount = 0;
int eventcount = 0;

//=====Open cut File=====

TFile
ayaz_cut1("/home/ayaz/Desktop/offlineAna/offlineAna/rootfiles/cuts/cuts_deltaE_E.root","open");
TFile
ayaz_cut2("/home/ayaz/Desktop/offlineAna/offlineAna/rootfiles/cuts/cuts_prompt_delay.root","open");

//=====

FILE * pFile;
pFile = fopen ("rootfiles/myanafile.txt","w");

//=====Create Histograms =====
char hname[256], htitle[256];
output->cd();
TH1F *bitpattern_0;
bitpattern_0 = new TH1F("vme bitpattern 0", "vme bitpattern 0", 16, -0.5, 15.5);

//=====Examples of histograms=====

TH2F *ayaz_a2a1_bit;
TH1F * ayaz_fptdc_a2_prot;
TH2F*ayaz_cut_a2a1;
TH2F*ayaz_cut_a2a2_a2a1;

ayaz_a2a1_bit = new TH2F ("ayaz_a2a1_bit", "ayaz_a2a1_bit", 410, 0, 4100, 410, 0, 4100);
ayaz_fptdc_a2_prot = new TH1F ("ayaz_fptdc_a2_prot", "fptdc_a2_protons", 4096, 0, 4096);
ayaz_cut_a2a1= new TH2F ("ayaz_cut_a2a1","ayaz_cut_a2a1 (cut a2a1)", 1024, 0,4096,1024,0,4096);

```



```

ayaz_cut_a2a2_a2a1= new TH2F("ayaz_cut_a2a2_a2a1","(plot_cut_a2a2 as
a2a1)",1024,0,4096,1024,0,4096);
//=====
output->cd("Scalers");
TH1F *hFPScaler;
hFPScaler = new TH1F("hFPScaler","FP Scaler", 64, -0.5, 63.5);
//=====Reading the cut file=====
TCutG *ayaz_cutEdE_a2a1;

ayaz_cutEdE_a2a1 = (TCutG*) ayaz_cut1.Get("ayaz_a2a1_cut");
//-----
UInt_t run = 0;
UInt_t event = 0;
UInt_t nentries = 0;

bool pattern[10];
bool a2a1prot,

Double_t qdc[255];
Double_t Rtdc[35];
Double_t oldvalue1, oldvalue2;
//=====Loop over runs : Begin=====

while(run < MAX_RUNS && DATAFILE[run][0])
{
TFile f(DATAFILE[run]);
fprintf( stderr, "Processing %s\n", DATAFILE[run]);

filecount++;
TS = (TTree*)f.Get("TS");
scalerbranch = TS->GetBranch("scalerbranch");
scalerbranch->SetAddress(&scalevent);

nentries = TS->GetEntries();
printf("Number of scaler entries %d\n", nentries);
for ( event = 0; event < nentries ; event++)
{
//if ( event%10000 == 0)
TS->GetEntry( event);
for( int i = 0; i < 32; i++) {
oldvalue1 = hFPScaler->GetBinContent(i+1);
hFPScaler->SetBinContent( i + 1, oldvalue1 + scalevent -
>Getvs0(i));
//printf("scaler value: %d \n", scalevent->Getvs1(i));
oldvalue2 = hFPScaler->GetBinContent(i+33);
hFPScaler->SetBinContent( i + 33, oldvalue2 + scalevent -
>Getvs1(i));
}
}
//===== End loop over events=====

TX = (TTree*)f.Get("TX");
xbranch = TX->GetBranch("xbranch");

xbranch->SetAddress(&xevent);

nentries = TX->GetEntries();
printf("Number of x entries %d\n", nentries);
for ( event = 0; event < nentries ; event++)
//for ( event = 0; event < 2 ; event++)

```

```

{
  if ( event%10000 == 0){
    printf("x entry No. %d\n", event);
    //fprintf (pFile, "x entry No. %d\n", event);
  }
  TX->GetEntry( event);

  eventcount++;

  //=====Reading QDC value =====
  for(int j = 0; j < 255; j++)
    qdc[j] = 0;

  vq = xevent->GetVqdc();
  int m = xevent->GetNVqdc();
  //printf("value of m:  %d \n", m);
  for ( int i = 0; i < m; i++){
    TVqdc *q = (TVqdc *)vq->operator[] (i);
    qdc[q->Index()] = q->Data();
    //printf("qdc %d = %d\n", q->Index(), q->Data());
  }

  //===== Reading RANGE TDC value =====
  for(int j = 0; j < 35; j++)
    Rtdc[j] = 0;

  vt5 = xevent->GetVtdc();
  int rt = xevent->GetNVtdc();
  //printf("value of rt :  %d \n", rt);
  for( int i = 0; i < rt; i++)
  {
    TVtdc *t5 = (TVtdc *)vt5->operator[] (i);
    Rtdc[t5->Index()] = t5->Data();
    //printf("Rtdc %d = %d\n", t5->Index(), t5->Data());
  }

  //===== Reading Bitpattern =====
  for ( Int_t i = 0; i < 10; i++){
    pattern[i] = false;
    if ( (xevent->GetVb0(0) >> i) & (0x0001 == 1) ){
      bitpattern_0 -> Fill( i);
      pattern[i] = true;
    }
  }

  //=====Filling histograms=====

  if(pattern[0] && pattern[1]&& !pattern[2] && !pattern[3] && !pattern[4])
    {ayaz_a2a1_bit -> Fill(qdc[65],qdc[64]);}

  //-----

  if (ayaz_cutEdE_a2a1-> IsInside(qdc[65],qdc[64]))
    {ayaz_cut_a2a1-> Fill (qdc[65],qdc[64]);}

  if (ayaz_cutpd_a2a2-> IsInside(qdc[65],qdc[97]))
    {ayaz_cut_a2a2_a2a1-> Fill (qdc[65],qdc[64]);}

```

```

//=====Implement the cut=====

    a2alprot = false;
    tagtdc = xevent->GetTagger();
    int nt = xevent->GetNTagg();
    double fptdc = 0;

    for ( int i = 0; i < nt; i++){
        TTagg *t = (TTagg *)tagtdc->operator[](i);
        fptdc = t->Data() - fpOffset[t->Index()] + 2000;
//=====Filling fptdc histograms=====

    if (ayaz_cutEdE_a2a1-> IsInside(qdc[65], qdc[64]))

        {ayaz_fptdc_a2_prot -> Fill (fptdc);}

//=====
        f.Close();
        run++;
    } // End loop over runs

output->Write();
output->Close();

fclose (pFile);
delete output;
delete xevent;

timer.Stop();
Double_t rtime = timer.RealTime();
Double_t ctime = timer.CpuTime();

printf("Total %d data files processed \n", filecount);
printf("Total %d events processed \n", eventcount);
// if ((ctime >= 0.0) && (ctime <= 60.0) )
printf("RealTime=%f seconds, CpuTime=%f seconds\n", rtime, ctime);
//else
//if (((ctime / 60) >= 0.0) && ((ctime / 60) <= 60.0) )
printf("RealTime=%f minutes, CpuTime=%f minutes\n", rtime / 60, ctime /
60);
//else
printf("RealTime=%f hours, CpuTime=%f hours\n", rtime / 3600, ctime /
3600);

    return 0;
}

```

Appendix 2

Here come the codes for implementing cuts into ΔE -E histograms. The same procedure was done for implanting cuts on prompt.vs delay histograms.

```
void create_cut_proton()
{
    gROOT->SetStyle("Plain");

    TFile
f_in_ayaz("/home/ayaz/Desktop/offlineAna/offlineAna/rootfiles/myanalysis.ro
ot","open");

    TFile
f_out_ayaz("/home/ayaz/Desktop/offlineAna/offlineAna/rootfiles/cuts/cuts_de
ltaE_E.root","recreate");

    gStyle -> SetPalette(1);
    TH2F *ayaz_a2a1      = (TH2F*) f_in_ayaz.Get("ayaz_a2a1_bit");
    TH2F *ayaz_a3a2      = (TH2F*) f_in_ayaz.Get("ayaz_a3a2_bit");
    TH2F *ayaz_a4a3      = (TH2F*) f_in_ayaz.Get("ayaz_a4a3_bit");
    ayaz_a2a1 -> Draw("colz");
    gPad -> Update();
    TCutG *cutg = (TCutG*)gPad->WaitPrimitive("CUTG","CutG");
    cutg->SetName("ayaz_a2a1_cut");
    cutg->Write();
    ayaz_a3a2 -> Draw("colz");
    gPad -> Update();
    TCutG *cutg = (TCutG*)gPad->WaitPrimitive("CUTG","CutG");
    cutg->SetName("ayaz_a3a2_cut");
    cutg->Write();
    ayaz_a4a3 -> Draw("colz");
    gPad -> Update();
    TCutG *cutg = (TCutG*)gPad->WaitPrimitive("CUTG","CutG");
    cutg->SetName("ayaz_a4a3_cut");
    cutg->Write();

    f_out_ayaz.Close();
}
```

Appedix 3

These are the cods used to make the double Gaussian fit and consequently determine the number of protons (pions) in the tagger peak.

```
#include "TROOT.h"
#include "TCanvas.h"
#include "TStyle.h"
#include "TH1.h"
#include "TMath.h"
#include "TRandom.h"
#include "TPad.h"
#include "TF1.h"
#include "TF2.h"
#include "TGraph.h"
#include "TMultiLayerPerceptron.h"

void myana_proton_inflection2_alpha()
{
    gROOT -> SetStyle("Plain");
    gStyle->SetOptFit(1);

    // TFile f_in("/media/alpha/myanalysis.root","open");

    TFile
    f_in("/home/ayaz/Desktop/offlineAna/offlineAna/myanalysis.root","open");

    TCanvas *MyC = new TCanvas("MyC","Test Canvas",10, 10, 800, 600);

    //MyC->SetFillColor(2);
    MyC->GetFrame()->SetFillColor(21);
    MyC->GetFrame()->SetBorderSize(6);
    MyC->GetFrame()->SetBorderMode(-1);

    gStyle -> SetPalette(1);
    MyC->Divide(1,2);

    TH1F *ayaz_fptdc_a2_prot      = (TH1F*) f_in.Get("ayaz_fptdc_alpha2");

    TH1F *temp = fptdc_alpha2 -> Clone();
    //temp -> Rebin(2);
    //temp -> Scale(0.5);

    MyC -> cd(1);

    temp -> SetAxisRange(1600, 2500,"X");
    //temp -> SetAxisRange(2100, 2600,"X");

    TF1 *fsignal = new TF1("fsignal", "gaus", 1975, 1990);
    TF1 *fbackground = new TF1("fbackground", "gaus", 1800, 2100);
    TF1 *fspectrum = new TF1("fspectrum", "gaus(0)+gaus(3)", 1950,2050);

    double par1[3], par2[3];
    double par_bg2[5],par_bg3[5];
    //fspectrum -> SetParameters(1,1,1,1,1,1);
    fsignal -> SetLineColor(kRed);
    fbackground -> SetLineColor(kBlue);
    fspectrum-> SetLineColor(kGreen);
```

```

//fsignal -> SetParameters(1,1982);
//fsignal -> SetParameters(2,3);

temp -> Fit(fsignal, "", "", 1980, 1995);
fsignal -> GetParameters(par1);
temp -> Fit(fbackground,"", "", 1940, 2010);
fbackground -> GetParameters(par2);

double par_spect2[6];
fspectrum
SetParameters(par1[0],par1[1],par1[2],par2[0],par2[1],par2[2]);

temp -> Fit(fspectrum, "", "", 1930, 2040);
temp -> DrawCopy();

double par_spect[16];
fspectrum -> GetParameters(par_spect);

fbackground -> SetParameters(par_spect[3],par_spect[4],par_spect[5]);

TH1F h("h","",4096,0,4096);
h.Eval(fbackground);
//h.Eval(fspectrum);
h.SetLineColor(2);
h.DrawCopy("same");

MyC -> cd(2);

TH1F signal("signal","",4096,0,4096);

for (int i=0; i < 4096 ; i++)
{
    double x1 = temp ->GetBinContent(i);
    double x2 = h.GetBinContent(i);
    double y = x1 - x2;
    signal.Fill(i,y);
}

//TGraph *newsig = new TGraph(4096);
TH1F *newsig = new TH1F("newsig","",4096,0,4096);

for (int i=1800; i < 2300 ; i++)
{
    double newx = temp ->GetBinContent(i) ;
    double newy = h.GetBinContent(i) ;
    double newcount = newx - newy;

    newsig -> Fill(i,newcount);
    //sig -> SetPoint(i, i, sign);
}
newsig -> DrawCopy();

double prot_pos = par_spect[1];//par1[1];

cout << "proton position = " << prot_pos << endl;

cout <<"80 channels(" <<prot_pos-40 <<"-"<< prot_pos+40<<"); Integral =
"<<newsig->Integral(prot_pos-40,prot_pos+40) << endl;

```

```

cout <<"60 channels(" <<prot_pos-30 <<"-"<< prot_pos+30<<"); Integral =
"<<newsig->Integral(prot_pos-30,prot_pos+30) << endl;
cout <<"40 channels(" <<prot_pos-20 <<"-"<< prot_pos+20<<"); Integral =
"<<newsig->Integral(prot_pos-20,prot_pos+20) << endl;
cout <<"30 channels(" <<prot_pos-15 <<"-"<< prot_pos+15<<"); Integral =
"<<newsig->Integral(prot_pos-15,prot_pos+15) << endl;
cout <<"20 channels(" <<prot_pos-10 <<"-"<< prot_pos+10<<"); Integral =
"<<newsig->Integral(prot_pos-10,prot_pos+10) << endl;
cout <<"10 channels(" <<prot_pos-5 <<"-"<< prot_pos+5<<"); Integral =
"<<newsig->Integral(prot_pos-5,prot_pos+5) << endl;
cout <<"6 channels(" <<prot_pos-3 <<"-"<< prot_pos+3<<"); Integral =
"<<newsig->Integral(prot_pos-3,prot_pos+3) << endl;
cout <<"4 channels(" <<prot_pos-2 <<"-"<< prot_pos+2<<"); Integral =
"<<newsig->Integral(prot_pos-2,prot_pos+2) << endl;

```

```

TCanvas *MyC2 = new TCanvas("MyC2","proton inflection",10, 10, 800, 600);

```

```

double width[8] = {4, 6, 10, 20, 30, 40, 60, 80};
double proton_y[8] = {newsig->Integral(prot_pos-2,prot_pos+2),
newsig->Integral(prot_pos-3,prot_pos+3),
newsig->Integral(prot_pos-5,prot_pos+5),
newsig->Integral(prot_pos-10,prot_pos+10),
newsig->Integral(prot_pos-15,prot_pos+15),
newsig->Integral(prot_pos-20,prot_pos+20),
newsig->Integral(prot_pos-30,prot_pos+30),
newsig->Integral(prot_pos-40,prot_pos+40)};

```

```

gr = new TGraph(8, width, proton_y);
gr->SetLineColor(2);
//gr->SetLineWidth(4);
gr->SetMarkerColor(4);
gr->SetMarkerStyle(21);
gr->SetTitle("proton inflection");
gr->GetXaxis()->SetTitle("width (channels)");
gr->GetYaxis()->SetTitle("counts");
gr -> Draw("ACP");

```

```

}

```