MUON TRACK RECONSTRUCTION FROM CATHODE STRIP READOUT OF STREAMER TUBE PLANES IN CHORUS MUON SPECTROMETER

by

N. Ziya PERDAHÇI

B.S. in Physics, Boğaziçi University, 1993

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science
in
Physics

Boğaziçi University

1996

MUON TRACK RECONSTRUCTION FROM CATHODE STRIP READOUT OF STREAMER TUBE PLANES IN CHORUS MUON SPECTROMETER

APPROVED BY

Prof. Dr. Engin Arık
(Thesis Supervisor) .....

Prof. Dr. Avadis Hacınlıyan .....

Prof. Dr. Emine Rızaoğlu .....

DATE OF APPROVAL : JANUARY 12, 1996

Acknowledgements

I would like to thank my thesis advisor, Prof. Engin Arık and the spokesperson of CHORUS Prof. Klaus Winter for creating the possibility of participating in an international collaboration.

I also would like to thank the Muon Spectrometer Group for their help and cooporation, in particular I need to mention Dr. Arif Mailov, who introduced me into detector physics and data processing.

Without a good friendship the days at CERN would have been quite difficult. Especillay I am grateful to Joëlle Hérin, Rui Manuel Melo da Silva Ferreira and Erhan Pesen for their invaluable friendship and for all the activities at and outside CERN.

I also would like to thank my friends outside the collaboration, in particular, Taylan Akdoğan for his friendship and for his help in developing the tracking algorithm, and N.Gökhan Ünel and İnanç Birol for their never ending support.

My grateful thanks are due to Beverly Friend for his help in learning the analog electronics.

I am indebted to TÜBİTAK and Centre for Turkish-Balkan Physics Research and Applications for their support during my stay at CERN.

Abstract

We have studied several methods to achieve the best spatial resolution and the highest track reconstruction efficiency using the limited streamer tube planes with external pickup strips orthogonal to the signal wires and the analog readout of the CHORUS Muon Spectrometer. The analysis has been performed with real data of 100 GeV beam muons and cosmic muons at CERN Labrotary in Geneva. Using a combined algorithm of common bias level subtraction and an empirical function to determine the center of the induced charge distribution on the 18 mm wide strips, we have obtained a spatial resolution of about 3 mm.

Özet

CHORUS Müon Spektrometresi'nde yer alan kıvılcım tüpü detektörlerinde harici olarak sinyal tellerine dik yerleştirilmiş, analog elektronikle okunan katot şeritlerinden elde edilebilecek en iyi uzay çözünürlüğü ve detektörden geçen müonların izledikleri yolların en yüksek verimle tespit edilmesine yönelik çeşitli metotlar üzerinde çalıştık. Yaptığımız bütün analizler gerçek deney verileri olan 100 GeV'lik müonlar ve kozmik müonlar kullanılarak Cenevre'deki CERN Laboratuarında gerçekleştirildi. 18 mm genişliğindeki şeritler üzerinde endüklenen elektrik yük dağılımının merkezini bulabilmek için ortak gürültü seviyesi çıkartımı ve ampirik bir formülün birlikte kullanıldığı bir yordam ile yaklaşık 3 mm'lik bir uzay çözünürlüğü elde ettik.

Contents

List of Figures

List of Tables

List of Symbols

PH_{ij}: Raw data pulse height (in ADC counts) on the j^{th} strip in the i^{th} plane

$\overline{PH_{ij}}$: Mean noise on the j^{th} strip in the i^{th} plane

σ_i: Error of the COG calculation

P_{ij}: Pulse height of the j^{th} strip in the i^{th} plane

1. INTRODUCTION

1.1 Neutrinos in the Standard Model

In the Standard Model all matter is made out of three kinds of elemantary particles. These are leptons, quarks, and the gauge particles (or mediators). There are six leptons, classified according to their charge (Q), and the three conserved lepton numbers, which are electron number (L_e), muon number (L_μ), and tau number (L_τ). Leptons form three families (or generations):

$$\begin{pmatrix} e^- \\ \nu_e \end{pmatrix} \begin{pmatrix} \mu^- \\ \nu_\mu \end{pmatrix} \begin{pmatrix} \tau^- \\ \nu_\tau \end{pmatrix}.$$

Neutrinos (ν_e, ν_μ, ν_τ) are neutral leptons, which only take part in weak interactions. In the Standard Model neutrinos are massless and their interaction is described by the elektroweak theory. There are also six antileptons, with all signs reversed. According to experimental observations, parity violation in weak interactions seem to be maximal, as a consequence of which all observed neutrinos are left-handed, and all antineutrinos are right-handed. The tau neutrino has not been observed directly yet, but given the requirements of the electroweak theory and the properties of the tau lepton, it must exist.

1.2 Beyond the Standard Model: Neutrino Oscillations

If at least one of the neutrino flavours has a mass different from zero, then the weak eigenstates ν_α ($\alpha = e, \mu, \tau$) produced in a weak interaction (e.g. an inverse beta decay or a weak decay) will in general be a linear combination of the mass eigenstates ν_i ($i = 1, 2, 3$):

$$|\nu_\alpha >= \sum_i U_{\alpha i} |\nu_i > \qquad (1..1)$$

where U is a unitary mixing matrix. If U were diagonal, then the neutrino flavours would not mix, i.e. no oscillation would occur.

Since neutrinos are free particles after they are produced, the time evolution of neutrino flavours with momentum $\vec{p}$ created as ν_α at time t=0 is given by (expressed in natural units: $\hbar = c = 1$)

$$|\nu_\alpha(t)>=\sum_i U_{\alpha i} e^{-iE_i t}|\nu_i(0)>, \tag{1..2}$$

where $E_i = \sqrt{p^2 + {m_i}^2}$ are the corresponding energies of the mass eigenstates.

Let us take the special case of $\nu_\mu - \nu_\tau$ mixing. In this case we have

$$U = \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\theta & sin\theta \\ 0 & -sin\theta & cos\theta \end{pmatrix} \tag{1..3}$$

where θ is the $\nu_2 \leftrightarrow \nu_3$ mixing angle. Given that at $t = 0$ we have

$$|\nu_\mu(0)>=(-sin\theta|\nu_2> +cos\theta|\nu_3>)\,, \tag{1..4}$$

then at a later time

$$|\nu_\tau(t)>=(-sin\theta e^{-iE_2 t}|\nu_2> +cos\theta e^{-iE_3 t}|\nu_3>)\,. \tag{1..5}$$

One can express the probability that an initial beam of ν_μ will contain some ν_τ at time t as

$$P_{\mu\to\tau} = |<\nu_\tau(t)|\nu_\mu(0)>|^2 \tag{1..6}$$

Since $|\nu_i>$'s are orthonormal states we have,

$$P_{\mu\to\tau} = |-e^{iE_2 t}sin\theta cos\theta + e^{iE_3 t}sin\theta cos\theta|^2 = sin^2 2\theta_{\mu\tau} sin^2\frac{(E_2 - E_3)}{2}t\,. \tag{1..7}$$

This shows that the mixing probability is an oscillatory function of time. In the high energy limit we can approximate $(E_2 - E_3)$ with

$$(E_2 - E_3) \approx p\left(1 + \frac{m_2^2}{2p^2}\right) - p\left(1 + \frac{m_3^2}{2p^2}\right) = \frac{\Delta m^2}{2p}\,, \qquad (1..8)$$

where $\Delta m^2 = {m_2}^2 - {m_3}^2$. This allows us to write the oscillation probability as

$$P_{\mu\to\tau} \simeq sin^2 2\theta_{\mu\tau} sin^2\left[\frac{\Delta m^2}{4p}t\right] = sin^2 2\theta_{\mu\tau} sin^2\left[1.27\Delta m^2(eV^2)\frac{L(m)}{E(MeV)}\right]. \qquad (1..9)$$

It is obvious that one would have a nonvanishing probability of oscillation only if $\theta_{\mu\tau}$ and Δm^2 are different from zero. The $\frac{L}{E}$ value for CHORUS is about 0.02 m/MeV [1].

1.3 Neutrino Oscillations and CHORUS

Present data from solar and atmospheric neutrino experiments favour the hyphothesis of neutrino oscillations. All solar neutrino experiments find less ν_e than the theoretical predictions [2],[3],[4],[5]. As for the atmospheric neutrino experiments, two of them measure a ratio ν_μ/ν_e smaller than expected from theoretical calculations [6],[7],[8]. Nevertheless, this interpretation requires further confirmation from experiments. A complementary approach is to look for neutrino oscillations in laboratory experiments, where the experimental conditions, in particular the composition, energy, and the flux of the neutrino beam are under control.

The CHORUS Experiment is one of these experiments, which was set up in the CERN-SPS Wide Band Muon Neutrino Beam to search for $\nu_\mu \leftrightarrow \nu_\tau$ oscillations. It has taken data in 1994 and 1995. CHORUS seeks to observe the very short path of the τ lepton originating from a ν_τ induced interaction in the emulsion target. At SPS energies, the mean decay length of τ is about 0.1 mm. The muon track from the τ decay is reconstructed in the Muon Spectrometer and extrapolated back to the emulsion target to identify the τ. In order to reduce the number of events to be scanned, loose kinematical cuts are also applied.

One of the tracking sections of the CHORUS Muon Spectrometer is the limited streamer tube planes with analog pickup strips readout. The analog system was set up right before the neutrino data taking, but there remained the problem of determining its essential physical characteristics, which are vital for extracting useful information out of the inevitable analog electronics *garbage*. For this purpose we first determined the shape of the particle and noise clusters. Then came the study of the effect of the ground connection layout on the noise. For the ground connection having the minimum *noise probability* (explained in section 4.1.3), we obtained the optimum cluster size in terms of track residuals and reconstruction efficiency. This study also supplied us with the form of the induced charge on the strips. In the end, we developed an algorithm, which combined the two mostly used methods to calculate the best estimate of the avalanche position along the sense wires, that gave us a spatial resolution of about 3 mm for the entire detector. In our analysis we used 100 GeV beam muons from the CERN-SPS and cosmic muons.

2. OVERVIEW OF THE CHORUS EXPERIMENT

2.1 Neutrino Beam

CHORUS Detector utilises the West Area Neutrino Facility (the Wideband Neutrino Beam) of CERN. The basic principle is to first produce secondary pions and kaons from high-energy proton collisions with a fixed target. In CERN, the super proton synchrotron (SPS) accelerates protons to 450 GeV with a cycle of 14.4 seconds. Protons are extracted in two 6 msec long spills separated by 2.5 seconds. They are focused onto a 3 mm beryllium target, which consists of six rods of 10 cm length each and separated by a free space of 10 cm. With this double pulsing 10^{13} protons per spill or $2\text{x}10^{13}$ protons per cycle can be delivered to the target without damaging the Be rods. The energy of the incident protons is partly converted into the production of pions and kaons, and the subsequent weak decays of these mesons yield muons and the desired neutrinos. Mesons produced at large angles are absorbed by a collimator while the remainig ones are first focused by a radial magnetic field of a magnet (horn) and then reflected by a magnet (reflector) according to their electric charge. The most important decay modes are:

$$\pi^+ \rightarrow \mu^+\nu_\mu \qquad (2..1)$$

$$K^+ \rightarrow \mu^+\nu_\mu \qquad (2..2)$$

$$K^+ \rightarrow \mu^+\nu_\mu\pi^\circ \qquad (2..3)$$

$$K^+ \rightarrow e^+\nu_e\pi^\circ. \qquad (2..4)$$

The fourth decay mode indicates that the beam is slightly contaminated by electron neutrinos ($\sim 1\%$). A background of $\bar{\nu}_\mu$ in the ν_μ beam is also present due to mesons of negative charge which manage to escape the horn and the reflector. The prompt ν_τ content of the SPS beam is virtually zero. The mean energy of the ν_μ beam is around 30 GeV and the CHORUS Detector is located approximately 800 meters away from the Be target.

2.2 CHORUS Detector

The CHORUS experiment is a ν_τ appearance experiment, i.e., it searches for the appearance of ν_τ's in the CERN-SPS beam. It is designed to detect the inclusive reaction

$$\nu_\tau N \rightarrow \tau^- X \tag{2..5}$$

with three subsequent decay modes:

$$\tau^- \rightarrow \mu^- \bar{\nu}_\mu \nu_\tau \qquad (B.R. = 17.8\%) \tag{2..6}$$

$$\tau^- \rightarrow h^-(n\pi^\circ)\nu_\tau \qquad (B.R. = 50.3\%) \tag{2..7}$$

$$\tau^- \rightarrow (\pi^+\pi^-\pi^-)(n\pi^\circ)\nu_\tau \qquad (B.R. = 13.8\%) \tag{2..8}$$

in a background of ν_μ induced charged and neutral current events. X in reaction 2..5 denotes the hadron final state, and h^- denotes a negatively charged hadron.

The conceptual design of the experiment is based on the detection of the characteristic decay topology of the short lived tau lepton and on kinematical constraints based on the missing transverse momentum p_T of the undetected ν_τ in tau decays.

Since the lifetime of the tau lepton is only $3\text{x}10^{-13}$ sec, the direct observation of the tau lepton and the detection of the decay kink requires a vertex detector of extremely good spatial resolution, for which the best candidate is the nuclear emulsion with a spatial resolution of about 1 μm.

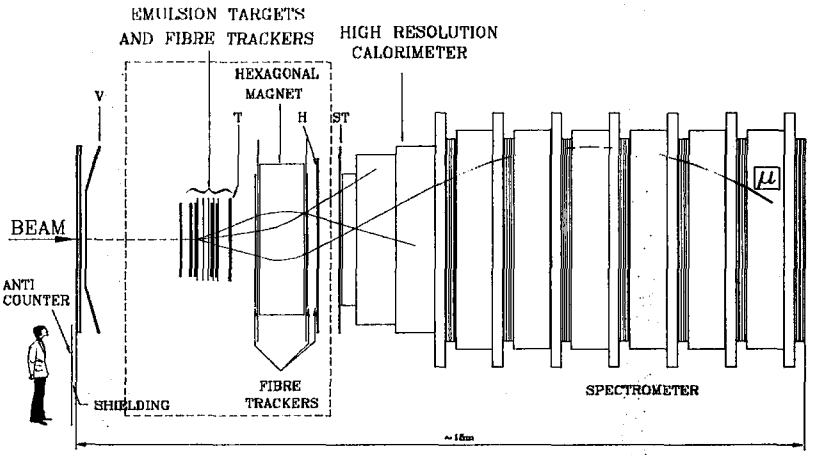

Figure 1: The CHORUS Detector

An electronic detector (Figure 1) composed of arrays of scintillating fibres, a magnetic spectrometer, a high resolution calorimeter and a muon spectrometer is used to select events with a negative muon or a negative pion and missing transverse momentum and locates the events to be scanned in the emulsion by extrapolating the tracks back to the exit point at the emulsion. The large number of ν_μ induced charged current or neutral current interactions which have to be scanned if no distinction from the ν_τ induced reaction were possible, is reduced by selecting candidates for the ν_τ reaction on the basis of their different kinematics. A very brief description of the different parts of the CHORUS Detector is given in the following sections.

2.2.1 Emulsion Target and Target Trackers

The emulsion target has a total volume of 230 litres and a total mass of 800 kg. It has two segments. Each segment has two stacks of 1.44 x 1.44 m^2 surface area and 2.75 cm thickness. The stacks are subdivided into 8 sectors of 0.71 x 0.36 m^2 surface area, each of which has 25 emulsion layers. This configuration permits fast semiautomatic and automatic scanning.

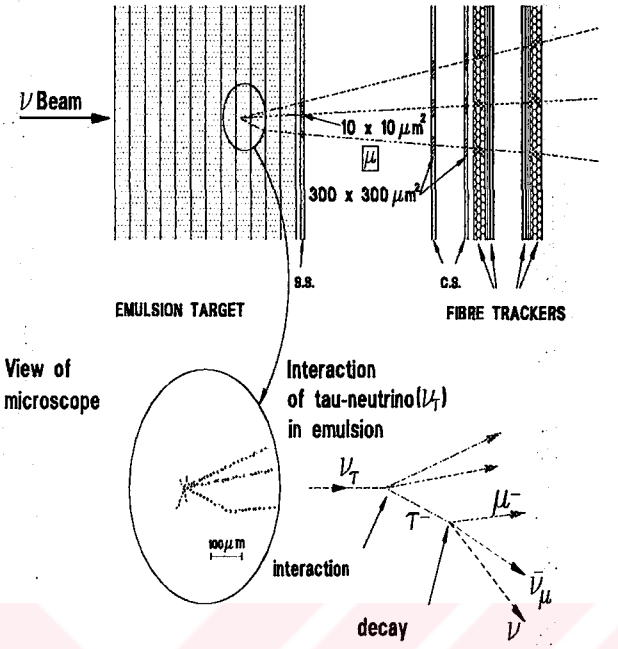

Figure 2: The Emulsion Target and The Target Trackers

The emulsion targets are followed by scintillating fibre planes of 500 μm diameter (target trackers, 7 layers staggered). Between these target trackers and the bulk emulsion there are three auxiliary emulsion sheets mounted (Figure 2). The tracks measured with the fibre trackers predict the track position in two adjacent special sheets within an area of $(500\mu\text{m})^2$ and provide an angular resolution of $\delta\theta \sim 3$ mrad.

2.2.2 Hexagonal Magnet and Diamond Trackers

An air core magnet is located between the target region and the calorimeter to determine the charge and the momentum of particles before they enter the calorimeter. It consists of six equal-sided triangles with 1.5 m sides, 0.75 m depth. Windings of thin aluminium sheet cover all faces of the triangles, producing a homogeneous field in each triangular section parallel to the outer side (Figure 3). The field stregth is 0.12 Tesla, without any radial dependence.

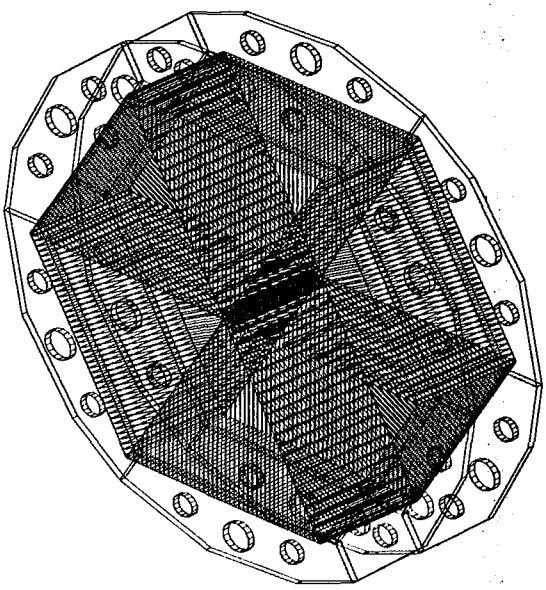

Figure 3: The Hexagonal Magnet

Three scintillating fibre trackers of hexagonal shape (diamond fibre trackers) are placed one upstream and two downstream of a hexagonally shaped magnet. The last target tracker plane and the first diamond tracker plane provide a high precision angle measurement before the magnet. The deflection in the magnet is measured with two hexagonal trackers, downstream of the magnet.

Combining the errors of the direction measurements before and after the magnet one obtains a total measurement error of

$$(\frac{\Delta p}{p})_{measurement} = 1.6\% \cdot p\ (GeV/c).$$

Multiple scattering gives a momentum independent contribution of $(\Delta p/p)_{m.sc.} = 16\%$. Combining these two errors quadratically one obtains a momentum resolution varying between 16 % (p = 2 GeV/c) and 23 % (p = 10 GeV/c), sufficient to determine the sign of the charge with more than 8 standard deviations for particles up to 10 GeV/c.

2.2.3 Calorimeter

The CHORUS calorimeter is located between the hexagonal magnet and the spectrometer, consisting of three sectors with decreasing granularity, called electromagnetic

(EM), hadronic one (HAD1) and hadronic two (HAD2) (Figure 4). The first sector is aimed at the measurement of the energy of the electromagnetic component of the event, the first two provide the measurement of the hadronic component of the events and the third completes the latter measurement acting as a tail catcher. The first two sectors are made out of scintillating fibres of 1 mm diameter, embedded in lead sheets (spaghetti calorimeter), whereas the third sector is made of lead and scintillation strips (sandwich technique). The calorimeter consists of vertical and horizontal modules, with limited streamer tube planes in between. These tracking detectors are required to connect the tracks in the muon spectrometer following the calorimeter to those reconstructed by the scintillating fibre tracker in front of the calorimeter. The main purpose of the calorimeter is to distinguish the reactions (II.2), (II.3), (II.4) from ν_μ induced, charged

$$\nu_\mu N \rightarrow \mu^- X, \tag{2..9}$$

and neutral current

$$\nu_\mu N \rightarrow \nu_\mu X \tag{2..10}$$

interactions. The energy flow vorector of the whole hadron final state is determined by measuring the center of gravity of its energy flow in the calorimeter and connecting it to the vertex position measured by the fibre trackers.

The energy resolution of the calorimeter is $\sigma(E_e)/E \cong 13\%$ (GeV) for electrons and $\sigma(E_\pi)/E \cong 30\%/E(\text{GeV})$ for pions.

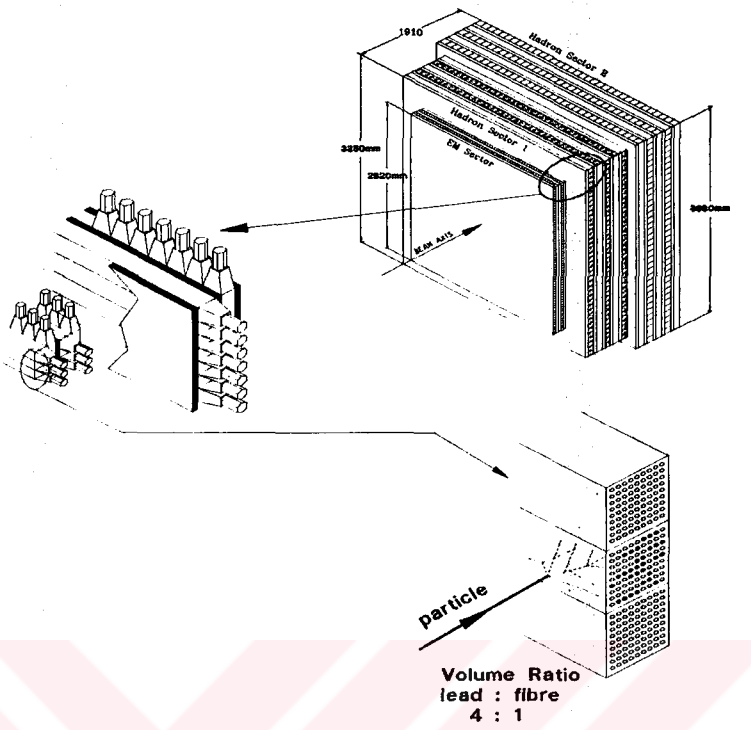

Figure 4: The High Resolution Calorimeter

2.2.4 Trigger

The trigger system of the CHORUS Detector is designed to select all neutrino induced events in the emulsion target and to reject background events from cosmic rays, muons in the neutrino beam, and neutrino interactions outside the target.

The neutrino interaction trigger is using information from three hodoscope modules, one close to the emulsion target (E plane), one close to the last fibre tracker of the target region (T plane), and one just in front of the calorimeter (H plane).

A veto system (V) is positioned 2 m in front of the target region and consists of two planes of 20 cm wide scintillators oriented vertically, and staggered by one-half width. An additional plane of 32 scintillators vetoes events from the concrete floor.

2.2.5 Muon Spectrometer

The muon spectrometer consists of six circularly magnetised iron modules of 375 cm diameter and seven tracking sections, one in front, five in the gaps between the iron modules, and one behind. Each magnet module is constructed of twenty iron plates of 2.5 cm thickness. Coils are running through a central hole of 8 cm diameter to the outer

radius; the magnetic field in the iron is close to 2 Tesla. The gaps are interleaved with 8 mm thick scintillation counters, with strips of 15 cm width, oriented, alternatively, horizontally and vertically. They are coupled to photo multipliers in groups of 5 plates.

The tracking sections consist of one drift chamber and eight limited streamer tube planes. Each drift chamber module is equipped with three planes of sense wires separated by 6 cm and oriented at 0°, +60°, −60° with respect to the horizontal. The streamer tubes have a wire spacing of 1 cm and a cathode strip pitch of 2.1 cm. Drift time measurements are made for groups of eight wires and give a resolution of 1 mm.

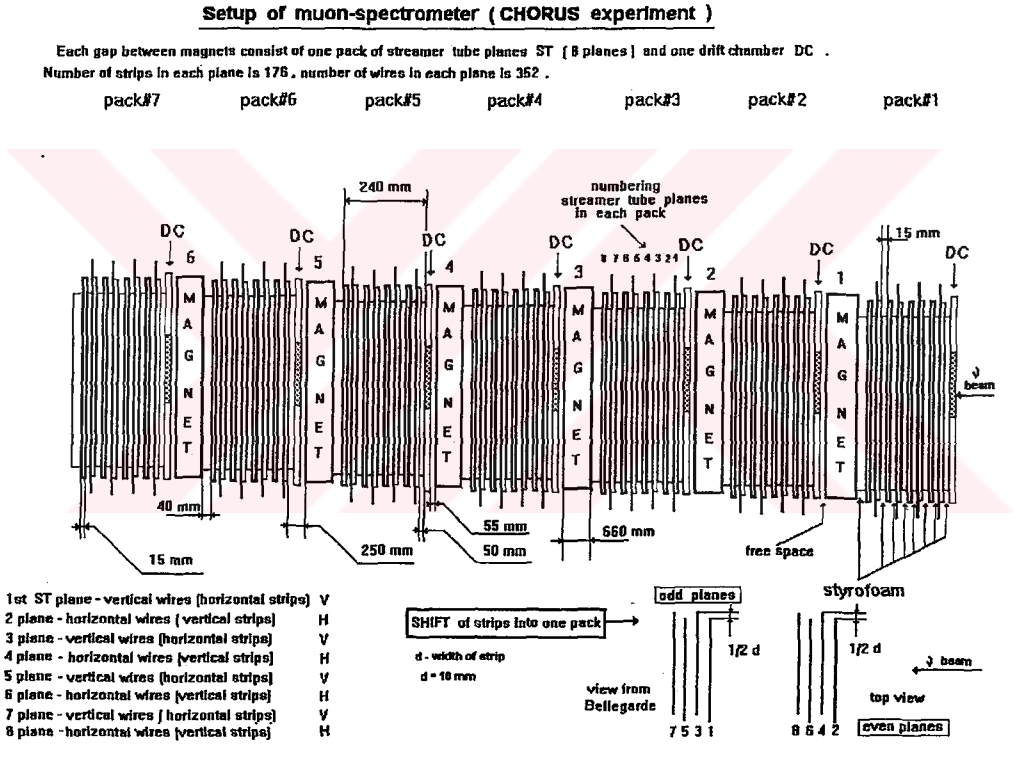

Figure 5: A Deatiled Picture of the Muon Spectrometer

Each tracking section thus gives 19 coordinate measurements which are combined and determine a space vector for each track. From a fit to these space vectors along the muon track the momentum is determined. The resolution, $\Delta p/p$, depends on the number of modules traversed; a 20 GeV/c muon traversing all six modules is measured with 15% resolution; if it is traversing only one module the resolution is 30%.

The scintillation counters in the gaps of the iron magnets have four functions. They are providing trigger signals for penetrating tracks and are complementing the measurement of hadron energy leaking from the preceding hadron calorimeter. They also

give the range of muons stopping in one of the modules, and they measure the energy loss by bremsstrahlung in the iron modules.

2.2.5.1 Streamer Tube System of the Muon Spectrometer

The Mechanical Construction The plastic streamer tubes installed in CHORUS muon spectrometer are of the Iarocci type [9]. They consist of 1 mm thick extruded PVC (being the easiest to shape) profiles with eight open cells, each of 9 x 9 mm^2 inner dimensions with a silver-plated Cu-Be wire of 100 μm diameter at its center. The 8-tube structure is closed by a PVC cover sheet (0.5 mm thick), placed on top of the open profile. Two 8-cell profiles, closed with two such covers, are mounted in one gas-tight extruded PVC envelope, closed by PVC end-caps. The gas, wire and cathode connections are only on one end-cap. This forms one single chamber unit. To form a standart streamer plane, 22 of those chambers are glued onto an Al sheet of 3.7 x 3.7 m^2 area and 1 mm thickness. The centering of the wire over the whole length is ensured by inserting wire holders made of polyethylene in every 47.5 cm. The position of the wire holders is shifted along the wire by 10-15 cm with respect to the foregoing plane with the same wire direction assuring a uniform detection efficiency for traversing muons. The inner surfaces of both the profile and the cover are coated with a layer of high resistivity. This allows the detection of the streamer with pickup strips, placed on the outside of the PVC envelope that contains the streamer cells. The conducting layer of the profiles is produced by graphite paint; its resistivity varies between 20 $k\Omega/cm^2$ and 20 $M\Omega/cm^2$. The covers are coated with a mixture of carbon black in PVDC. Its resistivity, of about 1 $M\Omega/cm^2$, varies only a few per cent. The gas mixture chosen is argon:isobutane with a ratio of 1:2.4. The working voltage is about 4.2 kV, which is applied to the conducting layer on the cell walls of each tube.

The analog strips face the covers, whose uniform resistivity guarantees a constant transparency for streamer signals. The strip plane consists of 50 μm thick Al strips of 18 mm width, glued on one side of a 1 mm thick PVC sheet. The other side of the sheet is covered with 50 μm thick Al foil, which acts as the ground electrode and shield. The analog strips are oriented perpendicularly to the wires with a pitch of 21 mm. There are 176 strips on each plane and 9240 strips are read in total (Figure 6).

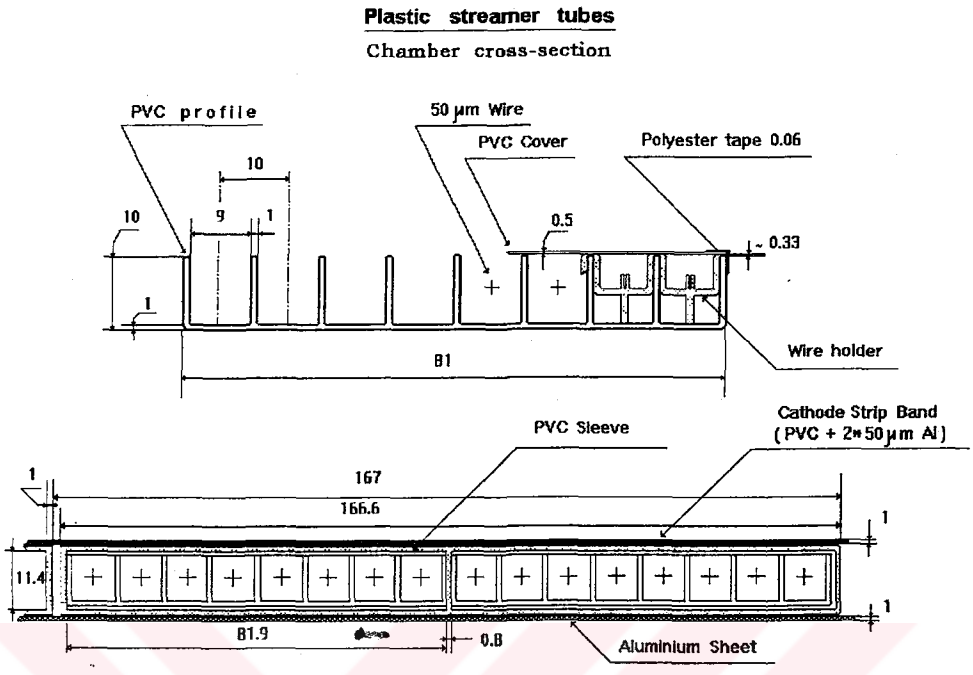

Figure 6: PVC Open Profile

The analog electronic system of the pickup strips The analog system [10] measures the charge induced on the pick up strips when ionizing particles traverse the streamer tubes. Due to the lack of spare electronic parts it was not possible to equip each of the 9856 strips with its own electronic channel. Therefore it was decided that the first and the last 22 strips of the first and the second plane of each pack not to be read out. So, we have a total of 9240 electronic channels.

In designing the analog electronics, each pickup strip was conceived as a transmission line [11] without loss with a characteristic impedance of

$$Z_0 = \sqrt{\frac{L}{C}},$$

with L and C (per unit length) are given by

$$L = \frac{\mu_o d}{w}, \quad C = \frac{\epsilon_{PVC} w}{d}$$

for a parallel plate geometry. Here w is the thickness (18 mm) of the strips and d is the distance between strips and the Al foil. A straightforward calculation gives $Z_0 \sim 11$ Ω. The analog electronic readout chain starts with transistor cards which are directly attached to the strips. A (pnp) transistor at the end of each cathode strip, biased to match the strip impedance, converts the induced voltage signal to a current at its collector. The second step is the analog card. The current is connected via a twisted pair flat cable to a pulse transformer at the input of the analog card. This current signal is then integrated, amplified, and fed to a peak detector that holds the maximum signal from the integrator that is proportional to the input charge. Next comes the logic card. The output of the peak detector is held for 1.6 μs by the logic card unless an event trigger arrives; in this case the signal is held until the end of the analog-to-digital converter (ADC) conversion cycle. To obtain the required dynamic range (about 1000), the output of the peak detector is digitized twice by an 8−bit flash-ADC of 1.2 μs conversion time by switching the reference voltage from 1 V to 5 V, giving a high and low sensitive conversion and an effective dynamic range of 1250 channels (more than 10 bits). The high-sensitivity conversion is adjusted such that the charge of the streamer of one minimum ionizing particle corresponds, on the average to about 40 ADC counts; one ADC count is equivalent to a charge of about 0.5 pC.

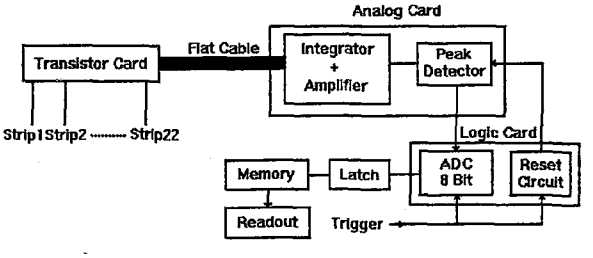

Figure 7: Readout Electronics

2.3 Sensitivity of the CHORUS Detector to $\nu_\mu \leftrightarrow \nu_\tau$ Oscillation

The sensitivity range for the mixing angle $sin^2 2\theta_{\mu\tau}$ depends on the accumulated statistics and the understanding of the possible background sources. In the hypothetical case that all incident ν_μ oscillate into ν_τ, $N_\tau^{max} = \int \sigma_\tau dE_\nu (dN_\nu/dE_\nu)$ events are expected,

where σ_τ is the cross section for the reaction $\nu_\tau N \rightarrow \tau^- X$. The integrated ν_μ spectrum is $N_\mu = \int \sigma_\mu dE_\nu (dN_\nu/dE_\nu)$. These two integrals have been determined by scaling the CHARM II results with the help of Monte Carlo simulation yielding $N_\tau^{max}/N_\mu = 0.51$ [12]. For two years of neutrino beam time (2×220 days) the expected number of ν_μ induced charged current interactions is $N_\mu \approx 5\text{x}10^5$, which yields the maximum number of ν_τ interactions as $N_\tau^{max} \approx 2.55 \times 10^5$. The corresponding number of background events for the three decay modes given in section 2.2 is about 1.70. In the limit of large Δm^2 ($\Delta m^2 \gg \frac{E}{L}$) oscillations will be washed out i.e. $sin^2(1.27\Delta m^2 \frac{L}{E}) \approx \frac{1}{2}$ and the number of observed τ^- decays will satisfy

$$N_\tau^{obs} \leq \frac{1}{2} N_\tau^{max} \epsilon_\tau^{tot} sin^2 2\theta_{\mu\tau} \ , \qquad (2..11)$$

where $\epsilon_\tau^{tot} \approx 0.05$ is the total efficiency of detecting a ν_τ charged current interaction. If we assume $\nu_\mu \leftrightarrow \nu_\tau$ oscillations occuring just below the present limits ($\Delta m^2 > 40 eV^2$ and $sin^2 2\theta = 5 \times 10^{-3}$), we expect to have

$$N_\tau^{obs} \leq \frac{1}{2}(2.55 \times 10^5)(0.05)(5 \times 10^{-3}) \approx 63 \ . \qquad (2..12)$$

The sensitivity for the $\nu_\mu \leftrightarrow \nu_\tau$ oscillation mixing parameter $sin^2 2\theta_{\mu\tau}$ can be calculated by setting $N_\tau^{obs} \approx 1.70$ (i.e. the expected number of background events) :

$$sin^2 2\theta_{\mu\tau} \geq \frac{2 \cdot 1.70}{\epsilon_\tau^{tot} \cdot N_\tau^{max}} \approx 2.6 \times 10^{-4} \ . \qquad (2..13)$$

The regions of sensitivity of CHORUS and the previous experiments are shown in Figure 8.

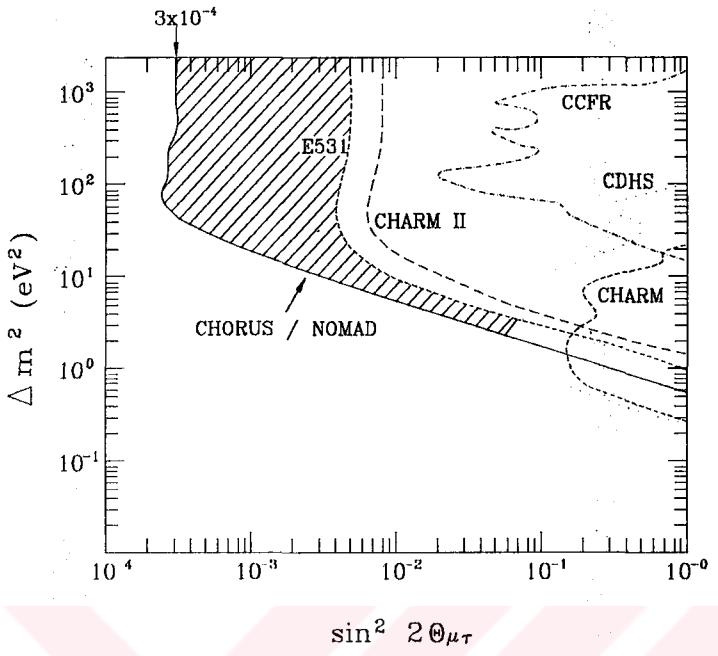

Figure 8: Exclusion plot for $\nu_\mu \leftrightarrow \nu_\tau$ oscillation parameters for the CHORUS Detector and previous experiments

3. REVIEW OF PICKUP STRIP DETECTORS

3.1 Introduction

The development of plastic streamer tubes started in Frascati in 1976. The first tubes were 2-3 cm in diameter, with 100μm thick wires mounted at the center. They were operated in the limited streamer mode, by using a gas mixture of argon and isobutane. A small plastic stremer tube detector with external pickup strips was operated in 1978 as the inner detector in the $\gamma\gamma 2$ experiment at Adone, the Frascati storage ring. It must be emphasized that streamer tubes were used in an experiment before the exact nature of their operation mode was clearly understood. After occasional observations which can be traced back to 1970 [13], the first systematic study of the streamer mode was performed by Charpak et al. [14]. At that time it was interpreted as a limited streamer mode, i.e. a discharge surrounding the wire and propagating along it, with natural self-quenching within $\sim$1 cm. Later on the streamer nature of discharge was established by Charpak et al. [15] and Fisher et al. [16], where it was observed that the centroid of charge generated by the process was definitely outside the center of wire, suggesting a propagation from the wire towards the cathode. Subsequently there was direct visual assessment by pictures of the self quenching streamer nature of the process by Alekseev et al. [17], and later by Atac et al. in [18] and [19], who managed to obtain detailed pictures of single and double streamers by means of image intensifiers. The photographs, which Atac et al. took, gave full support to the previous observations, namely that the fast and large electronic pulses are produced by those streamers which start from the wire and extend toward the cathode with a length from 1.5 mm to about 3 mm depending on the applied voltage. The width of the streamers was measured to be between 150-200 μm. They also observed that streamers grew in the direction of the initial avalanche.

The first large scale application of small streamer tubes was the implementation of the CHARM detector (1979) [20]. In 1980 a Frascati-Milano-Torino-CERN collaboration started the construction of the Mont Blanc proton decay detector, entirely based on the use of plastic streamer tubes with x-y strip readout. The plastic streamer tubes later used in CHARM II [21] and CHORUS are of this type.

3.2 Plastic Streamer Tubes

In general, a limited streamer tube is a gas discharge amplifier constructed from an anode wire, usually > 40 μm (100 in our case), raised to a high voltage in a quenching gas, typically a mixture of argon and hydrocarbon (isobutane in our case) or CO_2 quenching components, with a cathode placed 5 to 15 mm (7.57 mm in our case) from the wire. The limited streamer mode of operation is observed primarily in chambers constructed from thick sense wires and operated with heavily quenched gases (1:2.4 in our case).

3.2.1 Properties of the Limited Streamer Mode

In conventional gas detectors, that is, for some gas mixtures and pressures, there are basically four modes of operation depending on the high voltage [22]. These are ionization, proportional, limited proportional, and Geiger-Müller in the order of increasing voltage. Unconventional gas multiplication phenomena occurs under certain conditions. In a recent study Frank E. Taylor presented the following properties for this mode, applicable to Iarocci type tubes [23].

As the voltage is increased from the proportional region, there is an almost discontinuous fast transition [19] to a mode of operation, i.e. the limited streamer mode, where the charge Q collected on the anode wire is a linear function of the voltage. The collected output charge Q in this mode is only weakly dependent on the initial ionization [24], although the transition voltage does appear to decrease with increasing input charge. One can summarize this property by the following relation:

$$Q = \beta(V - V_T)\,, \tag{3..1}$$

where β is a constant for a given wire diameter, and the parametric threshold voltage, V_T depends on the gas composition and the wire diameter. Let this property given in equation (3..1) be Property 1.

The second important characteristic of the limited streamer mechanism is that the slope $\beta = dQ/dV$ in equation (3..1) is independent of the gas composition. This was

verified by [25] for various Ar-CO_2 gas mixtures and by [26] for other gasses. Let this property of lack of dependence of the slope $\beta = dQ/dV$ on the gas composition be Property 2.

In addition to these two properties there are also properties of the limited streamer mechanism which depend on the anode wire diameter. It has been observed that, within experimental erros, the ratio of the voltages for wires of different thickness yielding the same charge was not dependent on the collected charge. Since the charge integrated on the anode wire is a linear function of the high voltage as given by (3..1), the constancy of this voltage ratio may be stated as:

$$\beta_1 V_1 = \beta_2 V_2 \ , \tag{3..2}$$

for any pair of wire diameters, d_1 and d_2, at any voltage including V_T. Let this behaviour expressed by (3..2) be Property 3.

The fourth property of limited streamer tubes is again related to the constancy of the ratio of the voltages pronounced in Property 3. Taylor collected the data from the measurements given in the literature and found out that this ratio was very close to 0.87 for the tubes of the same outer dimensions and different anode wire thicknesses. This number has the following significance. The electric field from the anode wire which controls the avalanche is given by:

$$E(r) = \frac{1}{r} \frac{V}{ln(b/a)}. \tag{3..3}$$

Here a is the diameter of the sense wire and b is the distance from the wire to the wall. In this equation ln(b/a) is the only place where the dimensions of the limited streamer tube enter. And 0.87 happens to be the theoretical value one obtains if one puts the different values for the wire diameters into the ratio:

$$\xi = \frac{ln(b/a_1)}{ln(b/a_2)}. \tag{3..4}$$

This ln(b/a) scaling property given by equation (3..4) constitutes the last property, which is Property 4.

3.2.2 Cathode Material

If the gas mixture is not sufficiently quenching, ultra violet photons emitted from the tip of the streamer can extract electrons from the cathode (for which the details are given in section 3.3), which drift to the wire and trigger a new streamer just outside the dead region of the primary streamer. These secondary streamers appear as delayed pulses at the readout. Due to this mechanism, the cathode material also has an influence on the choice of the tube geometry. In comparing, for example, a graphite cathode with an aluminum one E. Iarocci [27] observed that aluminum cathode requires a higher ultra violet absorption by the gas mixture to overcome afterpulse generation. This makes graphite a better choice. The graphite coated PVC also has the advantage that it does not create problems concerning the gas purity in a long term operation.

3.2.3 Pickup Strips

The large signals available in the streamer mode allow the operation of strips as terminated transmission lines [28]. The attenuation length of such strip lines for induced streamer pulses is above 20 m, including the dissipative effect of the adjacent resistive cathode layer. One can therefore arrange large layers (100 m^2) covered by plastic streamer tubes with external pickup strips.

3.2.4 Resistive Cathode Transparency

The basic geometry of a resistive cathode detector with strip readout is shown in Figure 9a. The figure actually represents one half of the cross section of the detector. The electrical connections are shown in Figure 9b. The strips are grounded via the input resistance R_s of the readout circuit. The anode wire is also grounded and the resistive cathode is connected to a d.c. high voltage supply (-4.2 kV in our case).

The equivalent circuit with all the relevant parameters to simulate a streamer pulse generation on wires and strips is shown in Figure 9c. The cathode is simulated by resistive elements R coupled to the strips by capacitive elements C. The charge induced

on the cathode by the positive ions is simulated by a current source array connected between the anode wire and the cathode. During the very short time in which negative image charge is induced on the cathode wall, the current charges up the cathode-strip capacitors C flowing directly through the strip load R_s. As soon as the positive ions become neutralized, the capacitors C start to discharge through the resistances R. One can tune the resistivity of the cathode so as to get a sufficiently long discharge time, which will allow an undistorted pulse transmission through the pickup strips (thereby the detector is said to be transparent to the streamer). In our detector the discharge time is about 1.6 μs. The minimum resistivity for transparency varies in a wide range between 1 kΩ and 1 MΩ per square [27].

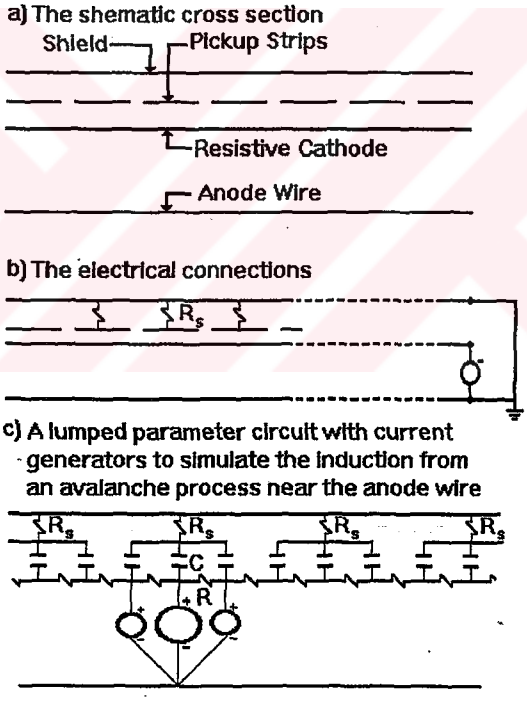

Figure 9: Transparency of the resistive cover

3.3 Formation of the Self Quenching Streamers

The first noticeable model of self quenching streamer formation was suggested by Atac et al. in [18] and [19]. According to them, during the primary ionization created by the traversing charged particle, the space-charge field near the anode wire may be large enough to partially cancel the applied electric field. As a result, the electrons are cooled

and a radiative recombination of Argon ions (A^+) and electrons (e^-) may occur:

$$A^+ + e^- \rightarrow A^* + h\nu,$$

where A* represents an excited Argon atom. What happens usually is that the photons reaching outside of the space-charge cloud are simply absorbed by the quenching gas. Some photons, however, may be energetic enough to ionize the quenching gas molecules, thereby creating ionization electrons. A few of these electrons might be produced very near the space-charge cloud (say 50-100 μm) and then drift back and multiply at the tip of positive ion cone where the electric field is the highest, thus triggering a streamer. This process continues until the last of such energetic photons creates an electron at the streamer front.

The second model that is worth mentioning is due to Taylor [23]. This model is based on a straightforward application of the Townsend discharge theory and electrostatics, and is able to account for the above mentioned four properties of the limited streamers in a qualitative way. In a limited streamer tube the electrons from the primary ionization are accelerated towards the anode in an electric field given by equation (3..3). According to the Townsend theory of gas discharge, the number of electrons, n, increases exponentially by the differential equation

$$dn = n\alpha(r)dr, \tag{3..5}$$

where α(r) is the Townsend coefficient which is a function of the radius r. This coefficient is given by

$$\alpha(r) = Ape^{-Bp/E(r)}, \tag{3..6}$$

where A and B are constants depending on the gas and p is the gas pressure. The total number of electrons collected on the anode wire is found by integrating equation (3..5)

:

$$n = n_0 exp \int_a^{r_0} \alpha(r) dr, \qquad (3..7)$$

where the integral is over the radial distance from the surface of the wire to the radius r_0 of the initial ionization, and n_0 is the initial number of electrons. From equation (3..7), it is clear that the final number of electrons collected on the anode is proportional to the initial number of electrons from the primary ionization. Using typical values of the Townsend coefficients it can be easily calculated that the gas multiplication is of the order of 10^7 for the primary ionization electrons. Since the electrons are more mobile than the ions by a factor of about 10^3, the positive ions are practically stationary during the collection time. Thus such a great multiplication factor in the vicinity of the anode wire brings in large space charge effects: The positive ion cloud left behind forms an electric field which opposes the original electric field of the anode wire. This, in turn, affects the electron avalanche rate through the Townsend equations described above. Taylor asserts that the limited streamer mechanism is a consequence of a delicate balance of the electron avalanche in an electric field, which is dynamically adjusted by the avalanche itself.

3.4 Streamer Tubes as a Calorimeter in CHARM II

The streamer tube system of the CHORUS experiment was lent by the CHARM II Collaboration [10]. The test results on the performance of the streamer tubes can be found in detail in [21] and in [29]. In CHARM II the streamer tubes were used in a target calorimeter to measure the vertex position, the hit multiplicity in the planes following the vertex, and the shower width, as well as its direction and energy.

According to ref [21], the measurements with the pick up strips on a prototype plane showed that for a hit centred on a strip, the two neighbouring strips each contain about 30% of the charge of the central strip and an interpolation by a Gaussian function was used to compute the impact coordinate of a traversing particle, in which a constant background level was subtracted from the strips.

4. EXPERIMENTAL RESULTS

4.1 The Analysis Procedures

For analysis, the X9 testbeam of the SPS at CERN and cosmic muons were used. The SPS accelerates protons up to 450 GeV. When the protons reach their maximum energy, they are extracted and sent to a fixed target. The muon beam is formed from the secondary particles produced. There are two extractions per cycle (spills). The two spills are separated by 2.8 seconds and the total cycle has a duration of 14.4 seconds. The energy of the negative testbeam muons was 100 GeV. In both types of data, we had all the magnets of the spectrometer demagnetised, so that hit and cluster informations can be introduced from all planes in a straight line track fitting. Analog signals from the pickup strip planes contain information about the strip numbers which are hit and the corresponding pulse heights in ADC counts.

- Analog signal measurements from strips are inherently more sensitive to noise. Therefore, a detailed study of the ground connection layout and its effects on the noise level is needed.
- We must have a very clear definition of what an analog cluster has to be.
- Since the gain of the charge amplifiers on analog cards may differ, the gain of the amplifiers should be calibrated by test pulses.

In the following analysis we will need the definition of an analog cluster. A preliminary definition could be that a cluster is a group of adjacent strips with nonzero pulse heights.

4.1.1 Study of the Pickup Noise

Noise can be defined as any signal acquired after an event trigger which is not related to any streamer discharge. Possible sources of noise can be categorised as follows:

- Pickup on the cathode strips,
- Thermal noise generated in the charge amplifiers on analog cards,
- Pickup in the signal cables (flat, twisted pair) between transistor and analog cards.

It must be emphasized here that the ADC converters located on the logic cards do not enter into the above list, because they are pedestal free [11]. The measured r.m.s. amplifier noise with a set up which included only an analog card and the necessary power supplies was about 8 mV which is equivalent to approximately 2 ADC counts with a high sensitivity conversion. This noise level might seem quite reasonable. Nevertheless, we measure a steady r.m.s. noise of about 16 mV (about 4 ADC counts with high sensitivity) with peak to peak values of as high as 80 mV at the peak detector output (right before the ADC converters) on randomly chosen channels. This difference is to be attributed to the other noise sources in the foregoing items.

4.1.2 Particle Clusters versus Noise Clusters

We have written a code which tries to distinguish the noise clusters from the clusters belonging to muons using the following criteria:

- Consider planes with single digital hits, which will guarantee that we will only be dealing with single particle clusters.
- Assuming that clusters belonging to these single hits should contain the maximum pulse heights of the planes, find these maxima on all single digital hit planes. This simple assumption is based on the fact that no pickup noise could be as high as the one generated by a streamer.
- Check the preceding assumption by comparing the place of the maximum pulse height and the single digital hit.
- In order to avoid the edge effects the first and the last strip of planes are excluded in all analysis.

After having confirmed that the clusters really belong to hits, a least squares fit is performed in both projections, y and z, separately in the following way:

First, we compute from the pulse height distribution of a cluster, the impact coordinate of the traversing particle. The center of gravity (COG) is calculated for each cluster by weighting each strip position y_{ij} in one projection (vertical strips) and z_{ij} (horizontal strips) in the other with its ADC charge using the expressions:

$$y_i = \frac{\sum_j PH_{ij} y_{ij}}{\sum_j PH_{ij}}, \quad (4..1)$$

$$z_i = \frac{\sum_j PH_{ij} z_{ij}}{\sum_j PH_{ij}}, \quad (4..2)$$

where PH_{ij} is the nonzero pulse height (in ADC counts) on strip number j in the i'th plane. Then a least squares fit is performed independently for each projection provided that there are at least ten points, (x_{ij},y_{ij}), (x_{ij},z_{ij}), i$\geq$10, in a given projection. The goodness of the fit is controlled by the standart chi square minimization technique i.e.,

$$\chi_y^2 = \sum_i \frac{[y_i - ax_i - b]^2}{\sigma_i^2}, \quad (4..3)$$

$$\chi_z^2 = \sum_i \frac{[z_i - ax_i - b]^2}{\sigma_i^2}, \quad (4..4)$$

where the error values are calculated by propagation of errors assuming a box distribution for the strips:

$$\sigma_i = \frac{1}{\sum_j PH_{ij}} \sigma_{strip} \sqrt{\sum_j (PH_{ij})^2}, \quad (4..5)$$

$$\sigma_{strip} = \frac{1.8}{\sqrt{12}}, \quad (4..6)$$

in which σ_{strip} is given in cm. At this point we require that the integral probability of the χ^2 of the fit $P_\chi(\chi^2;\nu)$ is greater than 5%, which indicates that the fit is just acceptable [30]. Here the factor $\nu = i - 2$ is the numbers of degrees of freedom for fitting i data points with two parameters a and b of a straight line. If this condition is not satisfied, then a new fit is performed after distances of the points to the fitted track are calculated and the one which is the farthest from the track is rejected. This filtering procedure is repeated until the fit is within the required probability range. In the end, we are left only with the planes used in the succesful fit and looking at these planes all the analog hits which are far away ($\sim$ half a meter) from the fitted tracks are recorded as noise including zero ADC values.

Let $\overline{PH_{ij}}$ denote the mean noise on the i'th strip in the j'th plane. Then the following expression is used to calculate the mean noise on a strip:

$$\overline{PH_{ij}} = \frac{1}{N} \sum_{The\ whole\ RUN} PH_{ij} \qquad (4..7)$$

where N is the number of times a given plane is used in successful fits.

Figure 1 shows the cluster length distribution of noise on all strips and the pulse height distribution of noise with zero ADC counts being suppressed so that the others can easily be seen. We notice that single-strip clusters constitute about 55% of all the noise clusters whereas looking at the single hit clusters on muon tracks, we see that the contribution of the single strip clusters is only 1.5% (Figure 11). This implies that, with a little sacrifice, it is possible to discard half of the noise clusters, which is very important if one attempts to make an independent fit in which only the analog information is used.

The spike visible in the sixteenth channel in Figure 10 is not a characteristic of noise. It is because of the half-flash conversion technique used in the 8-bit ADC's and it was also observed in CHARMII (known as module-16 problem) [11].

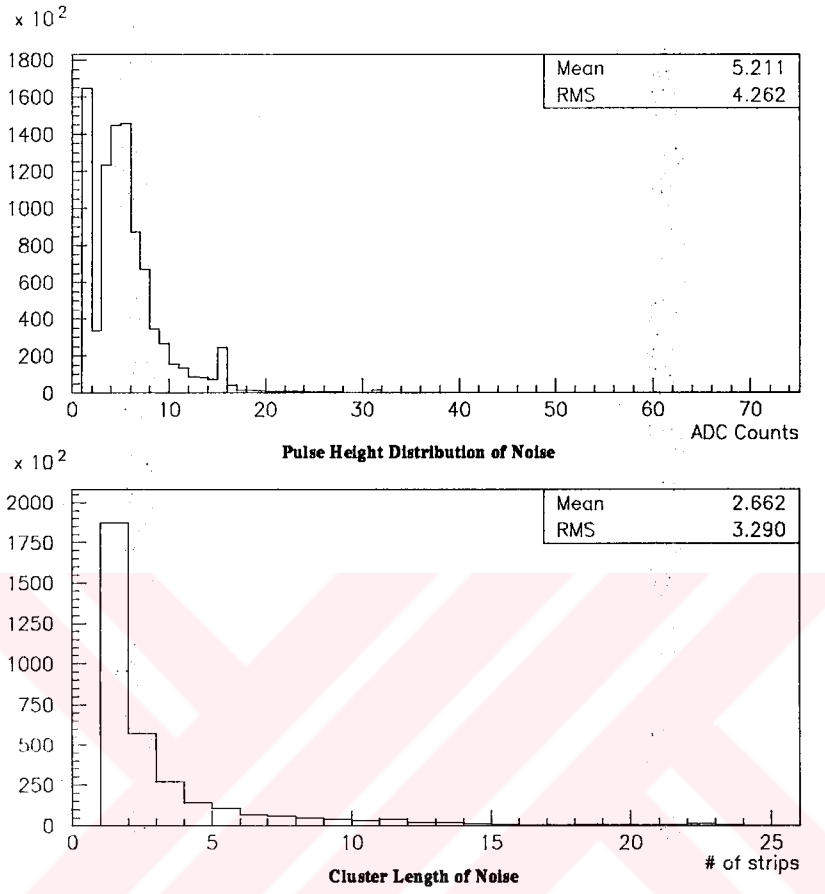

Figure 10: Noise Distributions

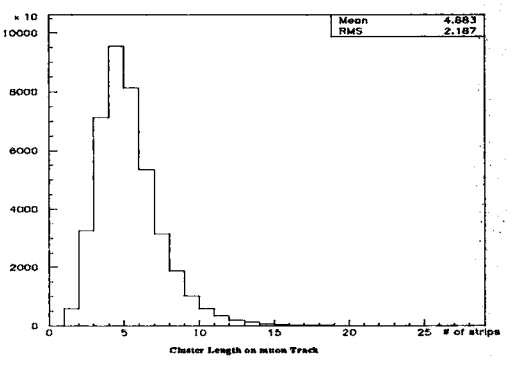

Figure 11: On track single hit cluster length distribution, 100 GeV muons

What is shown in Figure 12 is the mean noise on strips indicated as horizontal bars along with their r.m.s. values for the eight planes of the arbitrarily chosen fifth gap. It is clear that the mean noise on a strip is typically of the order of 0.1 ADC counts, but the r.m.s. values are quite significant. This means that although strips are generally quiet, they sometimes give signals of the order of five ADC counts. This corresponds to about 20 mV, which is not very big as compared to a typical signal of 1 V from a minimum ionizing particle. Nevertheless, this suggests a correction to the raw data, which will be discussed shortly.

It is also very instructive to compare the total pulse height distributions of noise clusters with the single hit clusters on muon tracks. These distributions (Figure 13) suggest that, except for the single strip clusters, noise and single hit clusters can be distinguished from each other to a great extent if the following empirical relation is used:

$$Total\ pulse\ height \geq (cluster\ length - 1)\ \text{x}\ 8 \qquad (4..8)$$

This formula includes all the single hit clusters on muon tracks and helps to exclude most of the clusters off muon tracks.

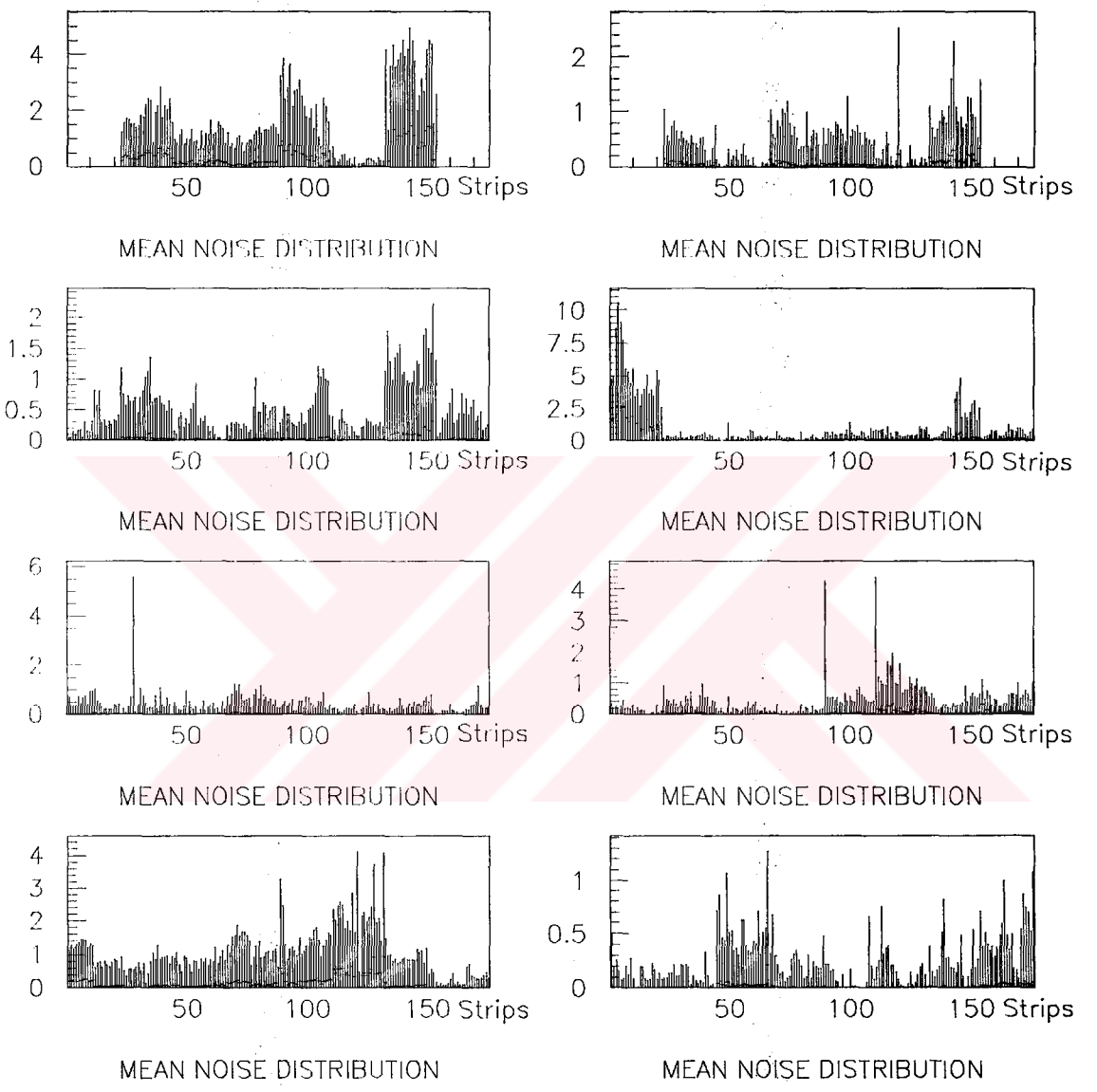

Figure 12: Mean noise on strips

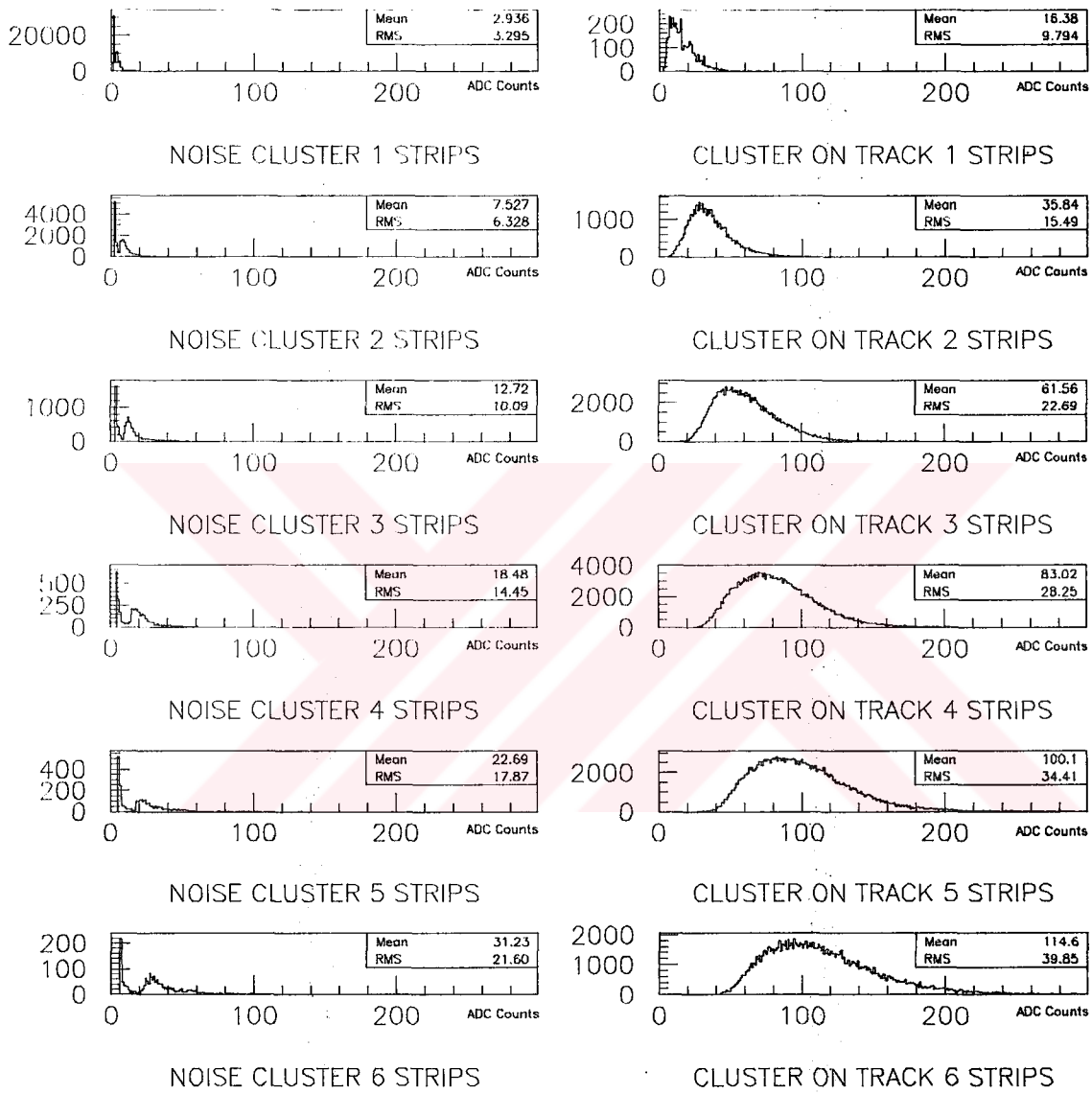

Figure 13: Total Cluster Pulse Height Distributions for Various Cluster Lengths

4.1.3 Study of the Effect of the Screening Ground on the Noise

There are three possible ground connections for the screening ground of the flat cable that carries the current from transistor cards to analog cards:

- Screen cable can be connected to the solid earth on the analog cards,
- Screen cable can be connected to the floating earth on the analog cards,
- It might be left hanging loose.

Assuming that pickup noise is a random process for the whole detector, one can calculate the probability of havig noise on any strip by

$$Prob = \frac{total\ \#\ of\ times\ strips\ which\ had\ noise}{total\ \#\ of\ strips\ in\ the\ noise\ region}\ . \qquad (4..9)$$

The results for the three cases are given in Table 1. The noise probability is reduced by a factor of nearly two in the case of floating earth connection. Therefore this type of connection was used during the whole neutrino data taking period.

Screening cable connected to	Probability
nowhere	2.0 %
solid earth	1.8 %
floating earth	1.0 %

Table 1: Noise Probability

4.2 Determination of the Cluster Size

By cluster size we mean the total number of strips used in COG calculation. Up to now the data analysis went through without any cut. That is, whatever comes as raw data is kept as it is and used for COG calculation. However, considering the noise level in the system one has to put a limit to the cluster size so as to extract the best physical information possible. To do this, we start from the strip having the maximum pulse height, which we call the central strip, and then proceed to the left and right of

this central strip. On the way to the left and right, we compare the pulse heights of the strips and always keep the strip which has a higher pulse height until the desired number of strips is reached. It is obvious that following this procedure one will always end up with the strips having higher pulse heights than the ones which are left out. Basically, there are two points that are crucial to the determination of the cluster size.

- The total number of muons reconstructed in both projections and,
- The number of planes participated in these successfull fits.

Below we tabulated the reconstruction efficieny for various choices of cluster size.

Cluster Size	Muons reconstructed	Planes in fit Y	Planes in fit Z
no cut	5408	14.49	15.06
10	5682	14.51	15.07
9	5816	14.57	15.11
8	6103	14.60	15.14
7	6493	14.71	15.24
6	7087	14.89	15.39
5	7861	15.30	15.82
4	8848	15.82	16.42
3	9734	16.34	17.08

Table 2: Reconstruction Efficiency, 100 GeV beam muons

These figures clearly illustrate that as the cluster size gets smaller the reconstruction efficiency gets bigger and bigger. We stop at the length of three which gave the best results.

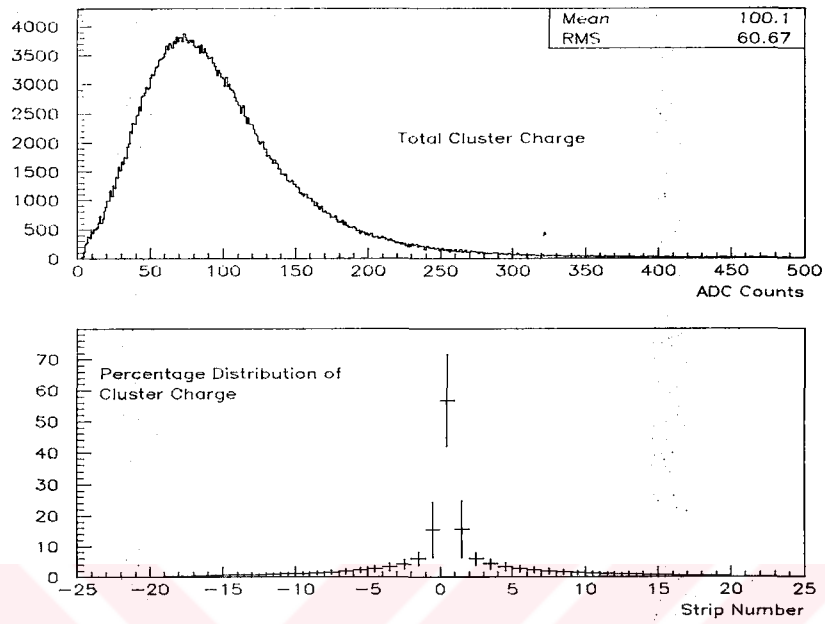

Figure 14: Total Charge Distribution of Single Hit Clusters on muon Track

This can be explained by looking at the total charge distribution of clusters and percentage contribution of pulse heights of the cluster members. On the average the total cluster charge is about 100 ADC counts, 80% of which is shared among the center of the cluster plus two of its nearest neighbours. From Fig.3, it is clearly seen that the average contribution of the second strip, to the right or to the left of the center of the cluster, to the total charge of a cluster is already below 6% with an r.m.s value of about 2% at the raw data level. Looking at the total charge distribution of clusters (Figure 14) one can easily calculate that this corresponds to about 8 ADC counts. But we have already seen that this value is very close to the rms noise intrinsic in the system. So, beyond the first strip, noise starts to predominate over physical signals. Therefore we have adopted the definition that only the strip with the biggest pulse height and its two nearest neighbours will form a cluster.

4.3 The Induced Charge Distribution

We can utilize clusters used in the successful fits to get the induced charge distribution on the strips. Although there are many models of the induced charge distribution in the literature for various kinds of analog strip detectors, it is much better to find it experimentally. Here it is important to note that the most accurate information about the position of streamers is obtained from reconstructed trajectories of muons. We obtain the induced charge distribution as a two dimensional scatter plot of the difference between the coordinates of strips in clusters and the impact coordinate of traversing muons versus the normalized pulse heights on strips. What is shown in Figure 15 is the induced charge distribution for different cluster sizes. We note that as the cluster size gets bigger the central shape of the induced charge distribution remains the same. The additional part is only a tail of a certain length which accompanies this central distribution. We attribute this long tail to the left and right of the central region to noise. Therefore we will only be interested in the central part when we attempt to parametrize the induced charge distribution in section 4.5

It is also interesting to look at the normalised pulse height distribution of noise clusters. Since these signals are not related to any streames and therefore do not lie on muon tracks we take the geometrical center of the noise clusters as their reference points. Figure 16 shows such distributions for various cluster sizes. We see a constant distribution on all strips, which indicates that the probability of picking up noise is the same for all strips. This observation forms the basis for subtracting a bias level in COG calculation, which will be discussed in section 4.4

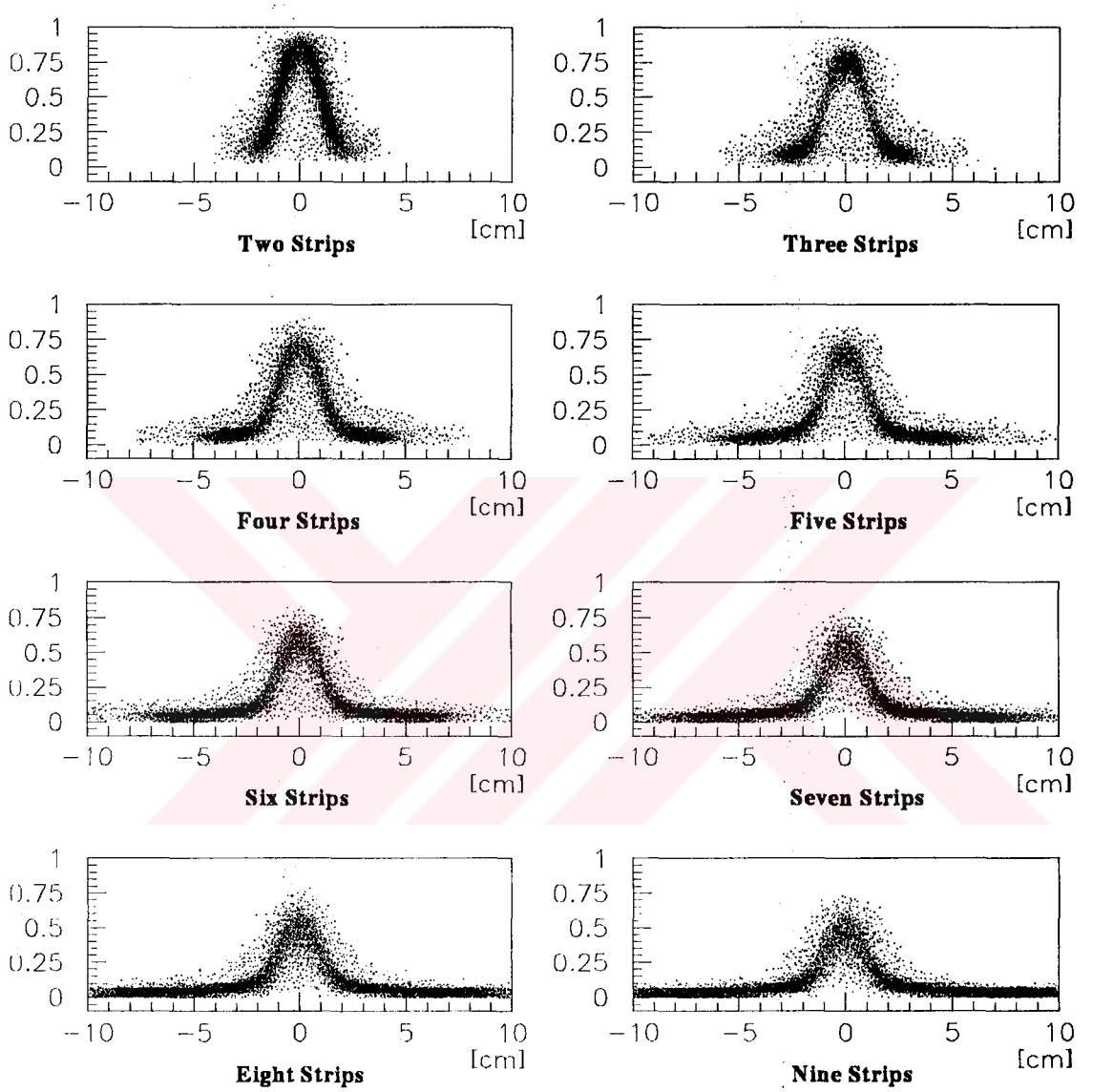

Figure 15: The Induced Charge Distribution

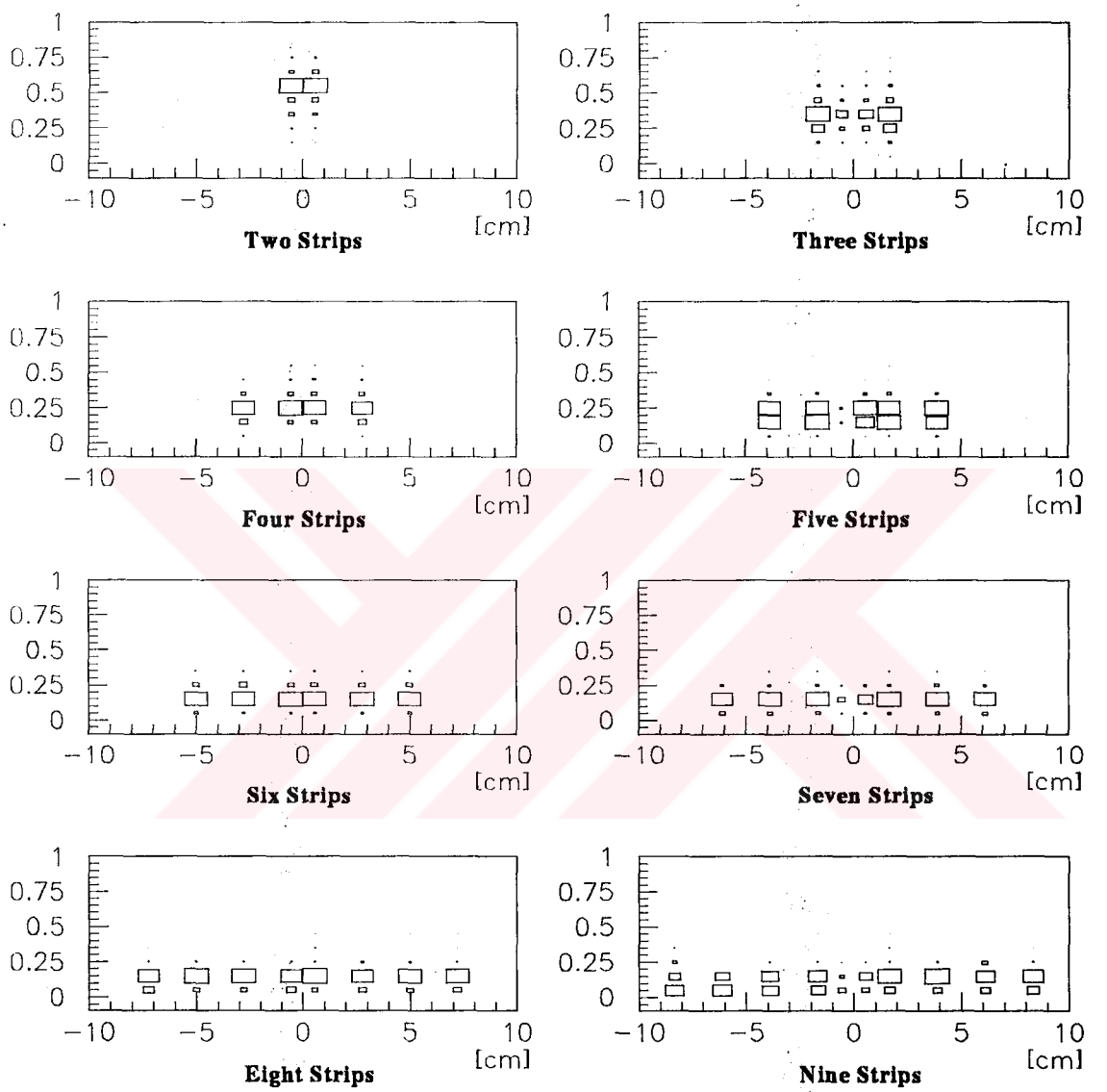

Figure 16: Noise Distribution on Strips

4.4 Common Bias Level Correction

The cluster definition that we have given at the end of section 4.2 is still subject to a final correction due to the large r.m.s. values of noise (section 4.1.2), namely the subtraction of a common bias level in the COG calculation. This was originally suggested by Charpak et al. in ref. [31] and [32]. A proper choice of the bias level allows reduction of the influence of pick-up and electronics noise in the COG determination.

The modified centroid calculation formula becomes

$$x_i = \frac{\sum_j (PH_{ij} - b) x_{ij}}{\sum_j (PH_{ij} - b)} , \qquad (4..10)$$

where $PH_{ij} - b$ is simply set to zero if it is negative. Here b is a small bias, which is subtracted from all the PH_{ij} contributing to a cluster. The effect of the subtraction is to decrease the contribution of the dispersion of the r.m.s. noise $\sigma_{PH_{ij}}$ on the strips. On the other hand one must be very careful, because choosing too high a bias level would result in a loss of information.

Charpak et al. stated that they had found the best results by choosing a bias level proportional to the total charge of a cluster:

$$b = k \sum_j PH_{ij} , \qquad (4..11)$$

where k is a constant factor which is determined experimentally as follows. We processed the same raw data file, i.e. 100 GeV beam muons, for increasing values of the constant k. The determination of the common bias level mostly depends on the goodness of the track residuals. We calculated track residuals as the distances between COG points used in successful fits and the reconstructed tracks and found that the spatial resolution has got a minimum value of 3.139 mm r.m.s. for the common bias level of 8%. This value is independent of a given projection. Figure 17 shows the track residuals of both projections, y and z, separately and the overall residuals for this choice.

Common bias level correction not only improves the spatial resolution but also the muon reconstruction efficiency. We observe that total number of reconstracted muons

is almost constant for values of k between five and ten per cent, but a glimpse at the number for the no bias case shows that with a little bias level the reconstruction efficiency improves considerably, namely from 9700 to 10700. Table 3 shows our results.

We can conclude this section by saying that the best estimate about the trajectories of muons we can obtain from the single hit clusters on muon tracks is given by the centers of clusters together with their two nearest neighbours with a common bias level of eight per cent if the COG method is used.

cb level [%]	Muons reconstructed	Resolution [mm]
no bias	9734	3.171
5	10684	3.194
6	10703	3.170
7	10713	3.150
8	10721	3.139
9	10721	3.140
10	10723	3.151

Table 3: Determination of the common bias level, 100 GeV muons

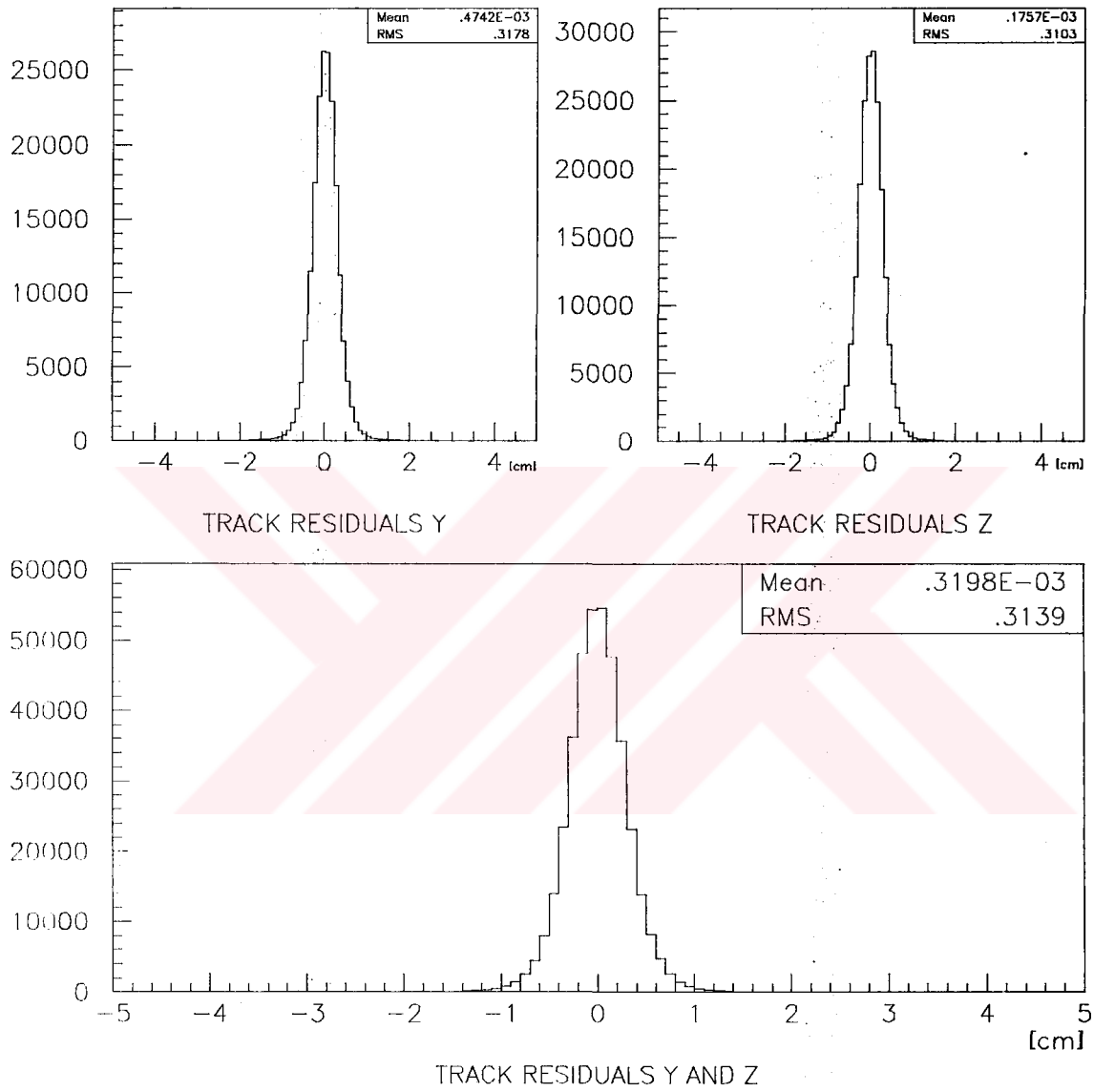

Figure 17: Track residuals for common bias level of 8%

4.5 Use of an Empirical Function

The basic idea in this approach is to match the observed charge distribution (Figure 15) to an analytic function with a small number of parameters. The minimum number of parameters is three; a_1 for overall normalization; a_2 for the position of the centre; a_3 for the width of the charge distribution. We reviewed different approaches and found out that the empirical function originally suggested by D.E.Lee et al. [33] for MWPC's fits well to the charge distribution we observe. This simple three-parameter function is given by:

$$PH_{ij} = \frac{a_1}{\cosh^2 \frac{\pi(x_j - a_2)}{a_3}}, \qquad (j = 1,2,3) \tag{4..12}$$

where x_j are the positions of the strips: j=1 central strip, j=2 left, and j=3 is right strip. The position of the centre can be calculated to be,

$$a_2 = \frac{a_3}{\pi} \tanh^{-1} \frac{\sqrt{PH_{i1}/PH_{i3}} - \sqrt{PH_{i1}/PH_{i2}}}{2\sinh(\pi w/a_3)}, \tag{4..13}$$

with

$$a_3 = \frac{\pi w}{\cosh^{-1} \frac{1}{2}(\sqrt{PH_{i1}/PH_{i3}} + \sqrt{PH_{i1}/PH_{i2}})} \tag{4..14}$$

for all PH_{i1}, PH_{i2}, and PH_{i3} different from zero, which implies that this formula can be applied to all clusters except for clusters of one and two strips. Here w is the pitch of the strips. Figure 18 shows the charge profile predicted by the inverse cosecant square model.

Recently K.Lau and J.Pyrlik [34] reported that they get promising results in an extensive study about the spatial resolution of cathode strip chambers by means of this empirical function.

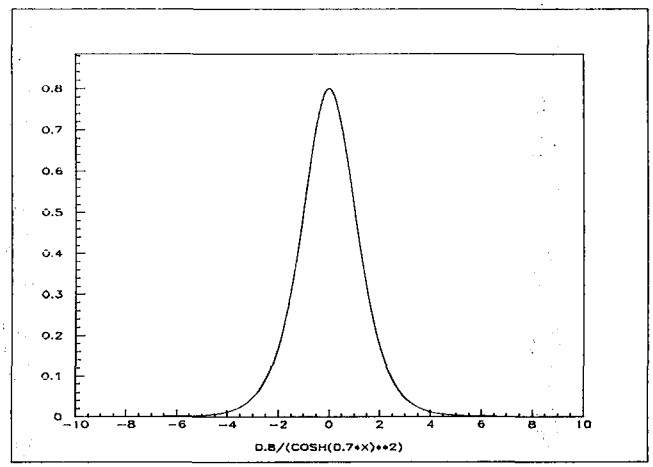

Figure 18: Induced charge cistribution as predicted by an empirical function

4.5.1 Common Bias Correction Incorporated

Considering that the intrinsic noise of the detector is independent of any method used for centroid determination, we also tried to obtain the common bias level for the empirical function of the induced charge distribution. Our approach has been exactly the same as before: PH_{ij}'s are modified by

$$PH_{ij} \longrightarrow PH_{ij} - b\,, \tag{4..15}$$

where b is

$$b = k \sum_{j} PH_{ij}\,, \quad j = 1,2,3. \tag{4..16}$$

Then we took runs for increasing values of the common bias level k and compared the track residuals. Our results are summarized in table 4. The best results are obtained for a common bias level subtraction of five per cent (Figure 19). One thing we ought to stress here is that the empirial formula which we are using for the induced charge distribution is applicable only to the clusters of cluster size greter than or equal to three

strips after the common bias level has been subtracted. For clusters of two strips at the raw data level or for the ones which are reduced to two strips after the common bias level subtraction, we calculate the centroid position by the COG method. Since we obtain the best spatial resolution in the case of the empirical fit together with a common bias level of five per cent this will be the method when we attempt to make an independent fit in which only the analog information is used.

The reconstruction efficieny is nearly the same as in the case of COG method with a common bias level of 8%, but one should notice that it is far better than the COG case even when there is no bias.

cb level [%]	Muons reconstructed	Resolution [mm]
no bias	10667	2.991
3	10682	2.952
4	10693	2.943
5	10704	2.945
6	10703	2.959
7	10705	2.987
8	10712	3.025

Table 4: Determination of the common bias level

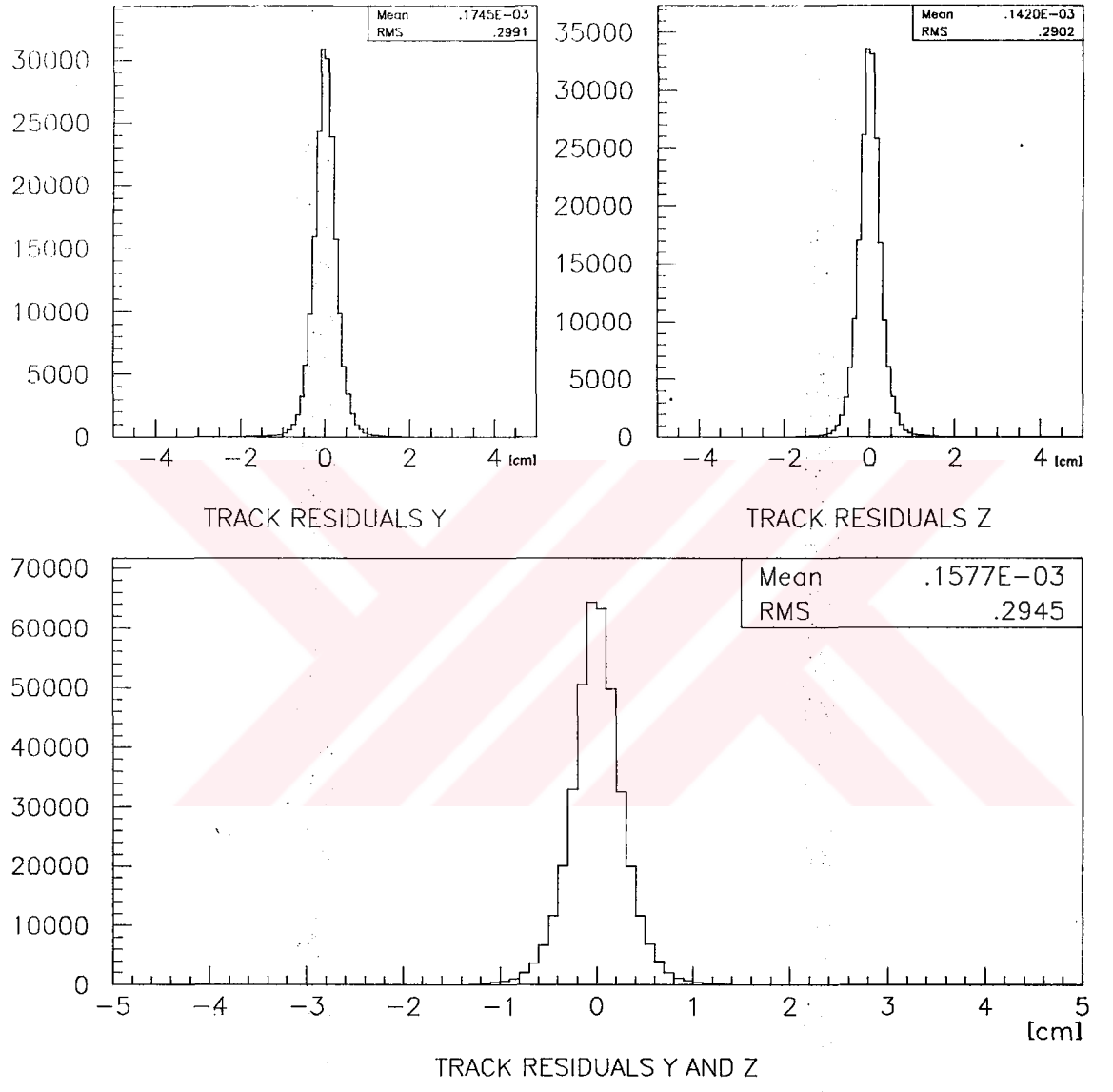

Figure 19: Track residuals for common bias level of 5%

4.6 Results from an Independent Fit

Since we already have a decent definition of what an analog cluster is and have already determined its essential corrections, it is tempting to try an independent fit, i.e. the use of analog information alone in reconstructing the muon track. Here basically the same algorithm as before is used. First, clusters on all planes are located by just the strips which have the maximum pulse height. Then a least squares fit is performed in the same way as before together with the same rejection algorithm for both projections. Here we will summarize some of the most important results (Table 5). For the muon beam we have chosen the same sample run of 11702 analog triggers that we have used in our analysis up to this point. Our algorithm was able to reconstruct 11255 muons in both projections with an r.m.s. value of 3.381 mm for the track residuals (Figure 20) As a comparison, we also give the results of an independent fit using only the digital hit information (i.e. no drift time information is used) from the streamer tubes. For the same run this was 10998 muons, with an r.m.s. value of 2.952 mm (Figure 21).

	Muons reconstructed	Resolution [mm]
beam muons analog	11255	3.381
bean muons digital	10998	2.952
cosmics analog	15334	4.138
cosmics digital	10919	3.131

Table 5: Results from the independent fit

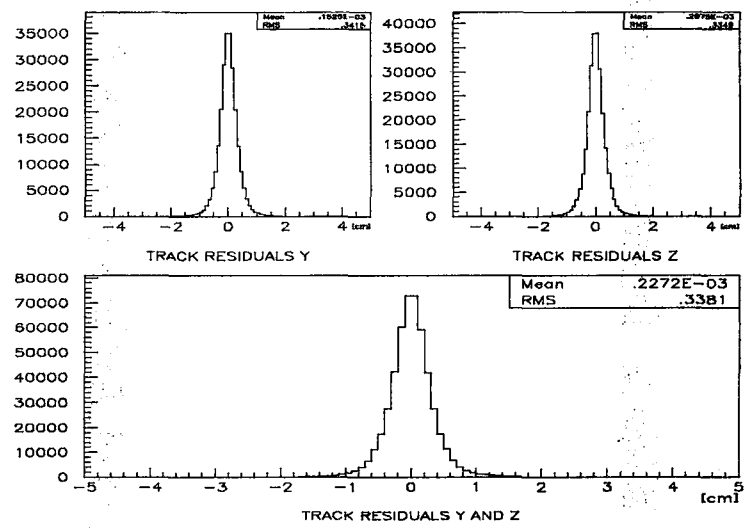

Figure 20: Track residuals for independent analog fit, 100 GeV beam muons

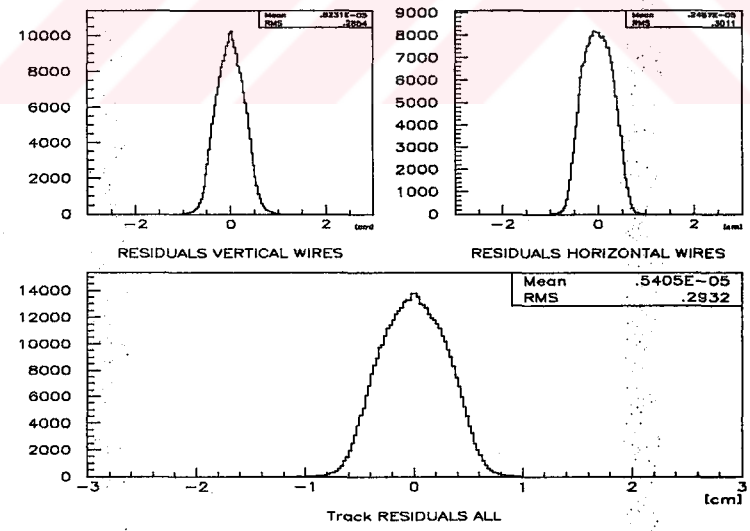

Figure 21: Track residuals for independent digital fit, 100 GeV beam muons

The difference between the two r.m.s. values in favour of digital can be easily explained by the fact that digital track reconstruction included only the single hit planes whereas analog fit included all the analog planes. In other words, the use of analog

information alone makes the inclusion of the planes through which several particles (e.g. a muon accompanied by a delta electron) have traversed, inevitable. We have observed (by eye scaning the raw data) that if this is the case, clusters of different particles overlap, which worsens ones ability to locate the streamer caused by the traversing muon.

The same algorithm applied to cosmic muons gave quite satisfactory results as well. For our sample run of 19212 cosmic triggers the number of muons reconstructed in both projections was 15334 for the analog strips and 10919 for the digital (see Table 5), with rms values of about 4.1 mm (Figure 22) for the analog and 3.1 mm for the digital (Figure 23). It should be stressed that the analysis with cosmic muons were done before the development of the empirical function and we actually expect a better spatial resolution than 4.1 mm for cosmics.

We also wanted to be sure that both the analog strips and the stramer tubes reconstruct the same track. For this purpose we have calculated the ratio of the slopes and intersection points given by the least squares fit for the analog tracks and the digital tracks (Figure 24). It is clearly seen that they are in complete agreement.

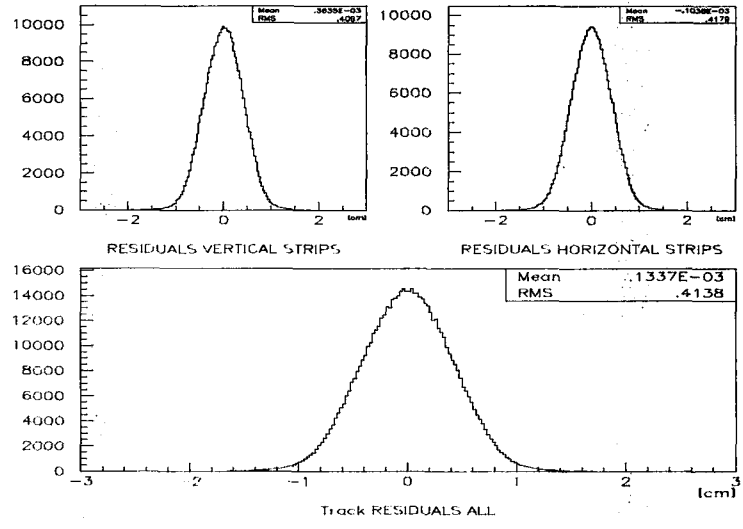

Figure 22: Track residuals for independent analog fit, cosmic muons

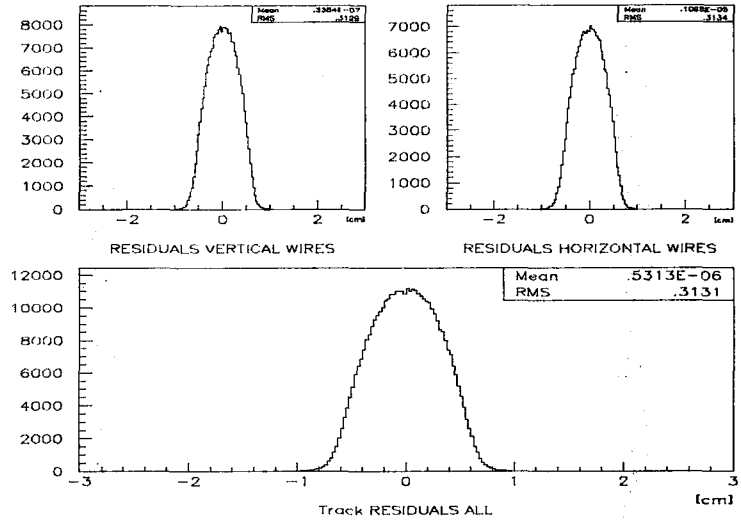

Figure 23: Track residuals for independent digital fit, cosmic muons

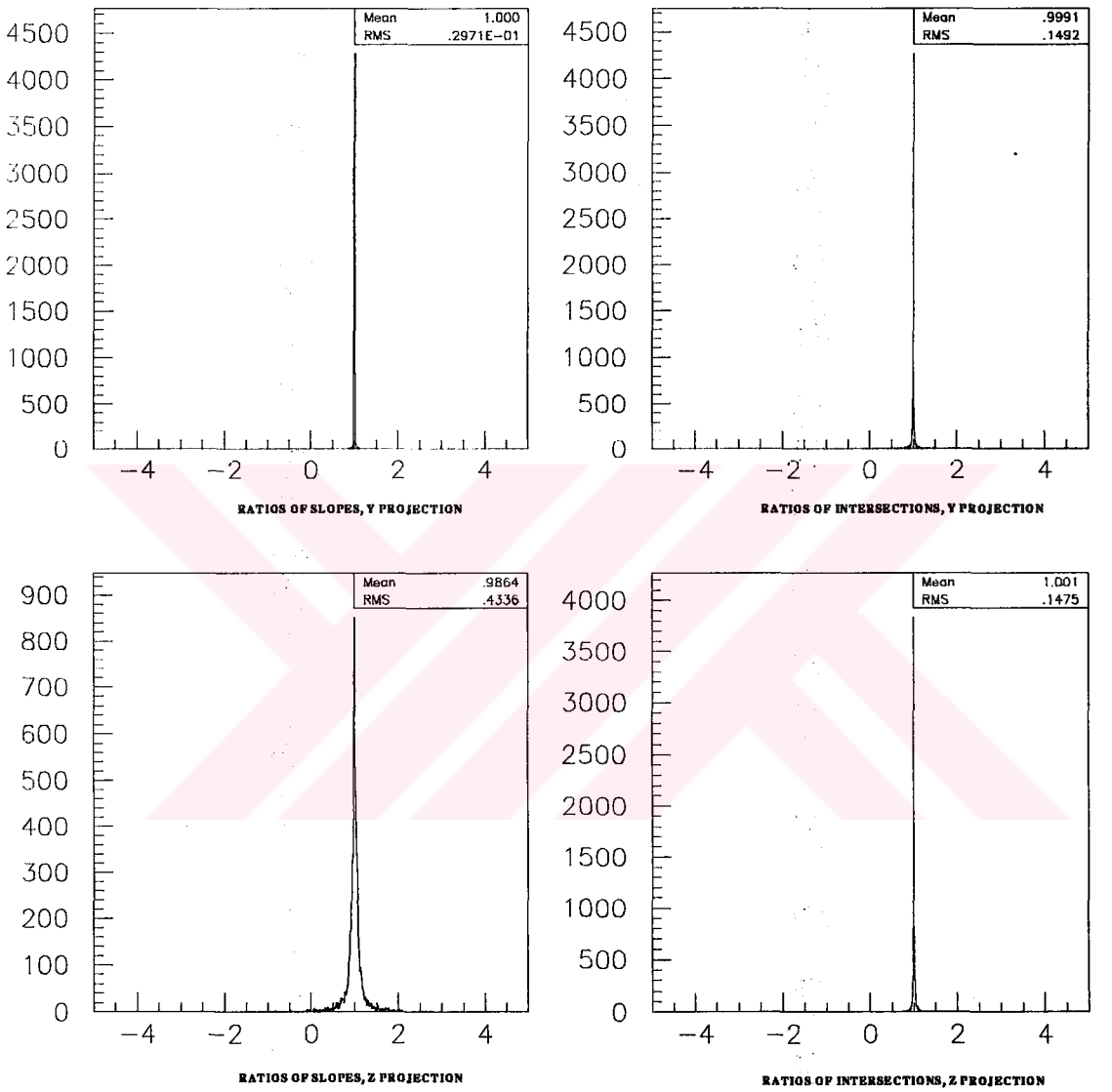

Figure 24: Ratios of the slopes and intersections

4.7 Pulse Height Calibration of the Analog System

Since every single strip has its own readout electronics chain as far as the charge amplifiers on the analog cards and the ADC's on the digital cards are concerned, in principle it may be neccessary to make strip by strip pulse height calibration. A very effective way of testing the cathode strips and the readout electronics is to use a special pulser system, which is a voltage of adjustable height that stimulates an effective charge [11]. This can be addressed to every strip by an external trigger signal. In this way, the noise level, the linearity of the gain of the charge amplifieers, and the overall response of the electronics channels can be checked. This was the method used in CHARM II [10] and in most of the other experiments such as [35]. In CHORUS we do not have such a test pulser system. Therefore it becomes very hard to analyse the individual characteristics of electronics channels if not impossible. One might attempt to take huge amounts of cosmics data and choose very clean events to analyse the streamer response of strips as in CHARM II [10]. But during our analysis we had to make frequent hardware changes, which have been neccessary due to malfunctioning electronic components. Every new change would mean a new amplification factor for the related strips, which would immediately necessitate a new set of calibration factors for the dedector. Therefore it did not seem to be feasible to make any strip by strip pulse height calibration by using cosmics data as long as the hardware changes continue.

5. CONCLUSIONS

In our analysis, we basically tried to determine the characteristics of particle and noise clusters, and tried to develop the best algorithm to calculate the centroid of the induced charge distribution. Our data analysis has shown us that for the case of single hit planes the analog system can contribute seventeen space points per projection on the average to the track information of the Muon Spectrometer with a spatial resolution of around three milimeter. In the case of single particle clusters the contribution is straightforward, because the shape of the particle and noise clusters can be distinguished very well by the empirical formula given in section 4.1.2. In the case of several particles traversing a streamer plane at the same time (e.g. a muon accompanied by a delta electron), however, clusters of different particles might overlap. In this case the localization of the avalanche position due to the traversing muon becomes tricky and it would be better to incorporate the information from the other tracking sections of the Muon Spectrometer.

Appendix

```
C This program reads the spectrometer raw data and calculates the
C spatial resolution for various cases. The program is based on the operating
C system IBM AIX Version 3 for RISC 6000, and CERN Program Library.
C*************************
      PROGRAM analysis
C*************************

      COMMON /ACCNT / MAX_EV
      INTEGER         MAX_EV

      COMMON /PEDESTAL/ PDT(176,56),PDTCTR(176,56)
      REAL PDT,PDTCTR

      COMMON /PROBABILITY/ N_STRIPS, N_TIMES
      REAL                 N_STRIPS, N_TIMES

      COMMON  /SAMEMUON/ SAME
      INTEGER            SAME

      INTEGER EVBUF(15000)   ! "event buffer"
      INTEGER EVLEN          ! total event length

      DATA    LUN/5/
      INTEGER LUN            ! logical unit # for the raw data file

C ----------------------------------------------------------------------
      SAME=0
      N_STRIPS=0.0
      N_TIMES=0.0
C ----------------------------------------------------------------------
      CALL VZERO(PDT,9856)     ! initialise pedestal arrays
      CALL VZERO(PDTCTR,9856)
C ----------------------------------------------------------------------
      CALL VZERO(EVBUF,15000)
C
C SUBROUTINE INITF : OPEN THE RAW DATA FILE USING IOOPEN
C
      CALL INITF(IRC,LUN)

      IF (IRC.NE.0) THEN
         STOP
      END IF
C
C SUBROUTINE INIT : LOADING THE GEOMETRY FILES AND HISTOGRAMS INITIALIZATION
C
      CALL INIT
C
C SUBROUTINE NOISE : READING MEAN NOISE VALUES FOR THE PARTICULAR RUN
C
      CALL MEAN_NOISE

      NEV    = 0     ! counter for the # of accepted events

C*********************************************************************
C READ EVENT BUFFER and put it into EVBUF()
C EVLEN --> EVENT LENGTH = event header + raw data banks in the event
C*********************************************************************

C ioread READS ONE EVENT AT A TIME

 1    IRC = IOREAD(LUN,EVBUF,1)  ! returns one if the event is there

      EVLEN  = EVBUF(1)
```

```
      NB     = 4 * IOREAD(LUN,EVBUF(2),EVLEN - 1)
      NEV    = NEV + 1

C*************************************************************
C*------ SIGNAL AT EVERY 1000-th EVENT and
C*       check END-OF-FILE mark
C*************************************************************

      IF(MOD(NEV,1000).EQ.0)THEN
        WRITE(*,*)NEV,'events '
      END IF
C
C Check whether the EOF is reached or not.
C
      IF (NB.EQ.0.OR.IRC.NE.1.OR.EVLEN.LE.0) THEN
         IRC = IOCLOSE(LUN)          ! Close the raw data file
         GOTO 99
      END IF

      HLEN   =  EVBUF(2)     ! EVENT HEADER LENGTH (HLEN = 14)
      NEVONL =  EVBUF(6)     ! ONLINE EVENT # (STARTS FROM ZERO)
      LABEL  =  EVBUF(3)     ! LABEL = 111 "PHYSICS" EVENT
      DATE   =  EVBUF(7)     ! DATE OF WRITING THE EVENT
      TIME   =  EVBUF(8)     ! TIME OF WRITING THE EVENT

      IF (NEV.GT.MAX_EV) THEN
         GO TO 99
      END IF

C*****************************************************************
C*    NOW THE EVENT IS COMPLETELY STORED in EVBUF array !!!
C*    START OF THE LOOP OVER BANKS ( LABEL 3 )
C*****************************************************************

C*****************************************************************
C*---- BKLEN --> RAW DATA BANK LENGTHS IN THE NEVth EVENT
C*****************************************************************

      IBK    = HLEN              ! Event Header Length
 3    BKLEN = EVBUF(IBK + 1)

      IF (BKLEN.LE.0) THEN
         GOTO 1
      END IF                     ! happens at the last record
C
C SUBROUTINE BANK : UNPACKING OF THE RAW DATA BANK
C
      CALL BANK(EVBUF(IBK + 1))
      IBK = IBK + BKLEN

      IF (IBK.LT.EVLEN) THEN
         GO TO 3                 ! repeat till you reach the end of the event
      END IF

      GOTO 1

 99   CONTINUE
      WRITE (*,*) 'END OF RUN : '
      WRITE (*,*) 'accepted events =',NEV - 1,'out of',
     +            NEVONL + 1,'events '
C
C SUBROUTINE TERMIN : WRITING THE OUTPUT FILE IN RZ format
C
      CALL TERMIN
```

```
      END   ! End of the Program

C ****************************
      SUBROUTINE MEAN_NOISE
C ****************************

      PARAMETER(NSTRIPS = 176,NPLANES = 56)
      COMMON /NOISE/ M_NOISE(NSTRIPS,NPLANES)
      REAL           M_NOISE

C ---------------------------------------------------------------------

      DO II=1,NPLANES
         DO JJ=1,NSTRIPS
            READ(42)M_NOISE(JJ,II)
         END DO
      END DO

      END                    ! End of PEDESTAL

**************************************************************************
      SUBROUTINE BANK(DATA)
**************************************************************************

C************************************************************************
C*
C*--------------------        DATA HEADER        -------------------------
C*
C*---------- BKLEN   DATA(1) --> DATA BANK LEGTH
C*---------- HLEN    DATA(2) --> HEADER LENGTH
C*---------- NEVONL DATA(3) --> A POSITIVE INTEGER BANK KEY
C*----------         DATA(4) --> A POSITIVE INTEGER BANK FORMAT VERSION
C*---------- LABEL   DATA(5) --> ON-LINE EVENT NUMBER
C*----------         DATA(6) --> BANK STATUS WORD ("ERROR WORD")
C*---------- DATE    DATA(7) --> DATE OF THE WRITING OF THE BANK, yy-mm-dd
C*---------- TIME    DATA(8) --> TIME OF THE WRITING OF THE BANK, hh-mm-ss
C*
C************************************************************************

      INTEGER BKEY, BLEN, DATA(*)
      LOGICAL DIGITAL

C ---------------------------------------------------------------------
C ----------- BANK KEY = 1012  ----->  RETU = CHORUS STREAMER TUBES
C ----------- BANK KEY = 1014  ----->  REAN = CHORUS ANALOG STRIPS
C ---------------------------------------------------------------------

      BKEY = DATA(3)
C
C unpack digital data
C
C       IF (BKEY.EQ.1012) THEN
C          CALL RETU(DATA(1))
C          BLEN=DATA(1)
C         WRITE(*,*)'Digital unpacked'
C          DIGITAL=.TRUE.
C       END IF
C
C unpack analog data
C
       IF (BKEY.EQ.1014) THEN
C          IF (DIGITAL) THEN
             CALL REAN(DATA(1))
C           WRITE(*,*)'Analog unpacked'
C          END IF
```

```
         DIGITAL=.FALSE.
       END IF

      END  ! End of BANK

*************************************************************************
      SUBROUTINE INIT  ! Initialise ZEBRA system and various local flags
*************************************************************************

      CHARACTER*80 FILENAME

      COMMON /LKBITS/ LKBITS(0:255)

      COMMON /PAWC/ H(1000000)! The working space of HBOOK is an array
                              ! allocated to the labelled COMMON /PAWC/
                              ! This is a ZEBRA store.

      CALL HLIMIT (1000000)! Defines the max. size of common /PAWC/ .
                           ! Must be called before any other HBOOK routine

      CALL HBOOK1(100,'P-H Distribution of pedestals'
     +              ,250,0.,250.,0.)
      CALL HBOOK1(200,'TOTAL CHARGE DISTRIBUTION '
     +              ,200,0.,200.,0.)

      DO ii = 1,56
        CALL HBOOK1(4000+ii,'AVERAGE PEDESTALS',176,0.5,176.5,0.)
        CALL HBOOK1(200+ii,'TOTAL CLUSTER CHARGE OF NOISE PER PLANE'
     +       ,200,0.,200.,0.)
        CALL HBPROF(2500+ii,'MEAN NOISE DISTRIBUTION',176,0.5,176.5,
     +       -1.0,100.,'S')
      END DO
C
C raw data histograms ------------------------------------------------
C
      CALL HBOOK1(1001,'raw data maximum pulse height distribution'
     +               ,500,0.,500.,0.)
      CALL HBOOK1(1002,'mean raw charge of planes',400,0.,400.,0.)
      CALL HBOOK1(1003,'total raw charge of planes',500,0.,500.,0.)
      CALL HBOOK1(1004,'raw data pulse height distribution of clusters'
     +               ,401,0.,401.,0.)
      CALL HBOOK1(1005,'mean raw cluster charge',301,0.,301.,0.)
      CALL HBOOK1(1006,'raw cluster lengths',44,0.,44.,0.)
      CALL HBOOK1(1007,'RAW DATA TOTAL CLUSTER CHARGE',300,0.,300.,0.)

C
C NOISE HISTOGRAMS ---------------------------------------------------
C
      CALL HBOOK1(3001,'CLUSTER PULSE HEIGHT DIST OF NOISE',
     +          200,0.,200.,0.)
      CALL HBOOK1(3002,'CLUSTER TOTAL PULSE HEIGHT DIST OF NOISE',
     +          300,0.,300,0.)
      CALL HBOOK1(3003,'CLUSTER MEAN PULSE HEIGHT DIST OF NOISE',
     +          150,0.,150.,0.)
      CALL HBOOK1(3004,'CLUSTER LENGTH OF NOISE',26,0.,26.,0.)

      CALL HBOOK2(6002,'NOISE Charge Distribution 2 STRIPS'
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)
      CALL HBOOK2(6003,'NOISE Charge Distribution 3 STRIPS'
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)
      CALL HBOOK2(6004,'NOISE Charge Distribution 4 STRIPS'
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)
      CALL HBOOK2(6005,'NOISE Charge Distribution 5 STRIPS'
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)
      CALL HBOOK2(6006,'NOISE Charge Distribution 6 STRIPS'
```

```
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)
      CALL HBOOK2(6007,'NOISE Charge Distribution 7 STRIPS'
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)
      CALL HBOOK2(6008,'NOISE Charge Distribution 8 STRIPS'
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)
      CALL HBOOK2(6009,'NOISE Charge Distribution 9 STRIPS'
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)
      CALL HBOOK2(6010,'NOISE Charge Distribution 10 STRIPS'
     +                              ,18,-10.,10.,11.,-0.1,1.0,0.)

      CALL HBOOK1(7002,'NOISE CLUSTER 2 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(7003,'NOISE CLUSTER 3 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(7004,'NOISE CLUSTER 4 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(7005,'NOISE CLUSTER 5 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(7006,'NOISE CLUSTER 6 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(7007,'NOISE CLUSTER 7 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(7008,'NOISE CLUSTER 8 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(7009,'NOISE CLUSTER 9 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(7010,'NOISE CLUSTER 10 STRIPS',300,0.,300.,0.)
C
C INDUCED CHARGE DISTRIBUTION HISTOGRAMS
C
      CALL HBOOK2(5002,'Induced Charge Distribution 2 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
      CALL HBOOK2(5003,'Induced Charge Distribution 3 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
      CALL HBOOK2(5004,'Induced Charge Distribution 4 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
      CALL HBOOK2(5005,'Induced Charge Distribution 5 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
      CALL HBOOK2(5006,'Induced Charge Distribution 6 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
      CALL HBOOK2(5007,'Induced Charge Distribution 7 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
      CALL HBOOK2(5008,'Induced Charge Distribution 8 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
      CALL HBOOK2(5009,'Induced Charge Distribution 9 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
      CALL HBOOK2(5010,'Induced Charge Distribution 10 STRIPS'
     +                              ,125,-10.,10.,33.,-0.1,1.0,0.)
C
C CLUSTER HISTOGRAMS
C
      CALL HBOOK1(8002,'CLUSTER ON TRACK 2 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(8003,'CLUSTER ON TRACK 3 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(8004,'CLUSTER ON TRACK 4 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(8005,'CLUSTER ON TRACK 5 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(8006,'CLUSTER ON TRACK 6 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(8007,'CLUSTER ON TRACK 7 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(8008,'CLUSTER ON TRACK 8 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(8009,'CLUSTER ON TRACK 9 STRIPS',300,0.,300.,0. )
      CALL HBOOK1(8010,'CLUSTER ON TRACK 10 STRIPS',300,0.,300.,0.)

      CALL HBOOK2(8020,'TOTAL CLUSTER PULSE HEIGHT VS CLUSTER LENGTH'
     +               ,30,0.,30.,300,0.,300.,0.)

C ---------------------------------------------------------------------
C
C TRACK FITTING HISTOGRAMS
C
      CALL HBOOK1(9001,'NO OF PLANES IN FITTED TRACK Y',36,0.,36.,0.)
      CALL HBOOK1(9002,'NO OF PLANES IN FITTED TRACK Z',36,0.,36.,0.)

      CALL HBOOK1(9003,'CHI2 PROBABILITY Y',100,0.,1.,0.)
      CALL HBOOK1(9004,'CHI2 PROBABILITY Z',100,0.,1.,0.)
```

```
      CALL HBOOK1(9005,'CHI2 PER DEGREE OF FREEDOM Y',100,0.,2.,0.)
      CALL HBOOK1(9006,'CHI2 PER DEGREE OF FREEDOM Z',100,0.,2.,0.)

      CALL HBOOK1(9007,'TRACK RESIDUALS Y',100,-5.,5.,0.)
      CALL HBOOK1(9008,'TRACK RESIDUALS Z',100,-5.,5.,0.)

      CALL HBOOK1(9009,'TRACK RESIDUALS Y AND Z',100,-5.,5.,0.)

      CALL HBOOK1(783,'CLUSTER PULSE HEIGHT DISTRIBUTION',
     +                                    400,0.,400.,0.)
      CALL HBOOK1(784,'Max PH Distribution',451,0.,451.,0.)
      CALL HBOOK1(785,'CLUSTER LENGTH',31,0.,31.,0.)
      CALL HBOOK1(786,'TOTAL CLUSTER CHARGE',500,0.,500.,0.)
      CALL HBOOK1(787,'MEAN CLUSTER CHARGE', 200,0.,200.,0.)

      CALL HBPROF(1500,'MEAN CHARGE DISTRIBUTION',50,-25.,25.,
     +                   -1.0,100.,'S')

      CALL HBPROF(1600,'PERCENTAGE OF MAX PH',50,-25.,25.,
     +                   -1.0,100.,'S')

      CALL HBPROF(45002,'Mean Cluster Charge',56,0.5,56.5,
     +                                        -1.0,500.,'S')
      CALL HBPROF(45001,'Mean Noise Charge',56,0.5,56.5,
     +                                        -1.0,500.,'S')

      CALL HBOOK1(20001,'avrg pulse height dist. (H)',800,0.,10.,0.)
      CALL HBOOK1(20002,'avrg pulse height dist. (V)',800,0.,10.,0.)

      CALL HIDOPT(0,'STAT')

      WRITE(6,*)'histograms ........: booked'

      CALL SPEC_GEOM  ! load geometric data file

      READ(40,255,END = 277) FILENAME
 255  FORMAT(A80)
 277  WRITE(6,*) FILENAME

      OPEN(95,FILE = FILENAME,FORM = 'UNFORMATTED')
      WRITE(6,*)'pedestal file .....: opened'

      DO i = 0,255
        LKBITS(i) = NUMBIT(i)  ! counts the one-bits in a word
      END DO                   ! From CERN Program Library
C      WRITE(*,*) 'end of init'


      END ! End of INIT

*****************************************************************
      SUBROUTINE RSOF(IDATA)  ! MAIN UNPACKING ROUTINE
*****************************************************************

      PARAMETER(NGAPS   =   7) ! # of streamer tube packs
      PARAMETER(NPLANES =  56) ! total # of planes (8 planes per pack)
      PARAMETER(NSTRIPS = 176) ! number of pickup strips
      PARAMETER(NWIRES  = 352) ! number of digital wires

      COMMON /SPEC_GEOM_DATA/
     + X_PL(NPLANES),             ! X-coordinates of streamer tube planes
     + ANA_PL_SHIFT(NPLANES),     ! analog plane shift array
     + ADJ_ANA_PL_SHIFT(NPLANES), ! adjustment to the analog plane shift
     + DIG_PL_SHIFT(NPLANES),     ! digital plane shift array
```

```
+ ADJ_DIG_PL_SHIFT(NPLANES), ! adjustment to the digital plane shift
+ STRIP_YZ(NSTRIPS),         ! coordinates of 176 analog strips per plane
+ WIRE_YZ(NWIRES)            ! coordinates of 352 digital wires per plane

 COMMON /PEDESTAL/ PDT(NSTRIPS,NPLANES),PDTCTR(NSTRIPS,NPLANES)
 REAL PDT,PDTCTR

 COMMON /NOISE/ M_NOISE(NSTRIPS,NPLANES)
 REAL           M_NOISE

 COMMON /LKBITS/ LKBITS(0:255)

 INTEGER IDATA(*)          ! dynamic array of raw data

 COMMON  /MUONBOOL/ BOTH
 LOGICAL            BOTH

 COMMON /CLCONADC/ CLCON(NSTRIPS,NPLANES)
 REAL              CLCON

 COMMON /CLSTRSEARCH/ PL_MAX(NPLANES),REJECT(NPLANES),KEEP(NPLANES)
+               ,KEPT_DIG(NPLANES),NO_INFO(NPLANES),SHOWER(NPLANES)
 INTEGER PL_MAX       ! strip having the max ADC counts
 INTEGER REJECT
 INTEGER KEEP
 INTEGER KEPT_DIG
 INTEGER NO_INFO
 INTEGER SHOWER

 REAL PH(NSTRIPS)     ! ADC content of strips of each plane

 INTEGER BEGIN
 INTEGER END
 INTEGER LENGTH

 INTEGER ISTART,  PH_CTR, CLEN  ! raw data cluster
 REAL    PH_AVRG, TOTALQ        ! variables

 COMMON /SINGLE_HITS/ LISTPL(NPLANES)
 INTEGER              LISTPL

 COMMON /DIG_HITS/ LISTPL_1(NPLANES)
 INTEGER           LISTPL_1

 DIMENSION LISTW(70) ! array of positions of one-bits
 INTEGER IHITS1      ! # of hits in first half of a sleeve
 INTEGER IHITS2      ! # of hits in second half of a sleeve
 INTEGER ISLEEVE     ! sleeve # in a streamer plane
 INTEGER NZ          ! total # of digital hits in a plane

 INTEGER IPL_AN       ! analog strip plane #
 INTEGER IPR          ! projection # : 1 = vertical, 2 = horizontal
 INTEGER ISTRIP       ! analog strip # having a hit
 INTEGER IPH_HIGH     ! ADC counts with a reference of 1 Volts
 INTEGER IPH_LOW      ! ADC counts with a reference of 5 Volt
 INTEGER NUM_GAP      ! gap # (1...7)

 INTEGER IPL1         ! The first odd or even plane in each gap
 INTEGER IPL2         ! The last odd or even plane in each gap
 INTEGER NPL          ! coordinate used in the fit

 REAL ST_CHARGE       ! charge in ADC counts

 INTEGER ABSENT(14)   ! first and last 22 strips of the first and second
 DATA    ABSENT   /1,2,9,10,17,18,25,26,33,34,41,42,49,50/
```

```
      INTEGER CORRECT(11)
      DATA    CORRECT   /132,131,130,129,128,127,126,125,124,123,122/
      INTEGER CORRECT1(11)
      DATA    CORRECT1   /121,120,119,118,117,116,115,114,113,112,111/

      INTEGER DUMMY

      INTEGER END_OF_PACK(7) ! last plane of each pack
      INTEGER BEG_OF_PACK(7) ! first plane of each pack
      INTEGER REF            ! digital plane to refer

C-------------------------------------------------------------------------------

      RETURN

C-------------------------------------------------------------------------------

      ENTRY RETU(IDATA) ! UPACKING OF STREAMER TUBE DIGITAL DATA

C*****************************************************************
C*  8 planes of streamer tubes per gap
C*  Even Numbered Planes have horizontal wires ( 2, 4, 6,..., 52 )
C*  Odd  Numbered Planes have vertical   wires ( 1, 3, 5,..., 51 )
C*****************************************************************

C --------------------------------
      CALL VZERO(LISTPL,NPLANES  )
      CALL VZERO(LISTPL_1,NPLANES)
      CALL VZERO(KEPT_DIG,NPLANES)
      CALL VZERO(NO_INFO,NPLANES )
      CALL VZERO(SHOWER,NPLANES  )
C --------------------------------

      BLEN = IDATA(1)
      HLEN = IDATA(2)

      DO I=(HLEN+1),BLEN  ! Loop over streamer tube data
         IF (JBYT(IDATA(I),31,2).EQ.0) THEN ! check error bit data or error?
            IPL         = JBYT(IDATA(I),22,8)
            IHITS1      = JBYT(IDATA(I),1,8)
            IHITS2      = JBYT(IDATA(I),9,8)
            NZ          = LKBITS(IHITS1) + LKBITS(IHITS2)
            LISTPL(IPL) = LISTPL(IPL) + NZ
            LISTPL_1(IPL) = LISTPL(IPL)
         END IF
      END DO

      DO ii=1,NPLANES

          IF (LISTPL_1(ii).EQ.0) THEN
             NO_INFO(ii) = 1
          END IF

          IF (LISTPL_1(ii).GT.1) THEN
             SHOWER(ii) = 1
          END IF

      END DO

      DO I=(HLEN+1),BLEN
         CALL VZERO(LISTW,70)
         IF (JBYT(IDATA(I),31,2).EQ.0) THEN  ! check error bit data or error?
            IPL = JBYT(IDATA(I),22,8)
C
```

```
C skip planes having no hit and planes having more than one hit
C
            IF (NO_INFO(IPL).EQ.1) THEN
               GO TO 199
            END IF

            IF (SHOWER(IPL).EQ.1) THEN
               GO TO 199
            END IF
C
C take planes with a single hit
C
            IF (LISTPL(IPL).EQ.1) THEN
               CALL BITPOS(IDATA(I),16,LISTW,NX)
               ISLEEVE     = JBYT(IDATA(I),17,5)
               LISTPL(IPL) = (ISLEEVE - 1) * 16 + LISTW(1) + 1

               KEPT_DIG(IPL) = 1
               GO TO 199
            END IF

 199     END IF

         IF (KEPT_DIG(IPL).EQ.0) THEN
            LISTPL(IPL) = 0
         END IF

      END DO

      RETURN  ! Back to BANK

C ---------------------------------------------------------------------

      ENTRY REAN(IDATA)  !  UNPACKING OF THE ANALOG STRIPS DATA

C*************************************************************************
C*  Even Numbered Planes have vertical   strips ( 2, 4, 6,..., 52 )
C*  Odd  Numbered Planes have horizontal strips ( 1, 3, 5,..., 51 )
C*************************************************************************

C ---------------------------------------------------------------------
      CALL VZERO(CLCON,9856    )
      CALL VZERO(PH,NSTRIPS    )
      CALL VZERO(REJECT,NPLANES)
      CALL VZERO(KEEP,NPLANES  )
C ---------------------------------------------------------------------
      BLEN = IDATA(1)
      HLEN = IDATA(2)
C -------------------

      DO i = HLEN + 1, BLEN  ! Loop over analog strip data
         IF (JBYT(IDATA(I),31,2).EQ.0) THEN  ! check error bit
            IPL_AN   = JBYT(IDATA(i),25,6)
            IPR      = MOD(IPL_AN,2) + 1
            ISTRIP   = JBYT(IDATA(i),17,8)
            IPH_HIGH = JBYT(IDATA(i),1,8)
            IPH_LOW  = JBYT(IDATA(i),9,8)
            NUM_GAP  = (IPL_AN - 1) / 8 + 1

            IF (IPH_HIGH.LE.240) THEN
               ST_CHARGE = FLOAT(IPH_HIGH)
            END IF

            IF (IPH_HIGH.GT.240) THEN
               ST_CHARGE = FLOAT(IPH_LOW) * 5.
```

```
            END IF
C
C CORRECTION FOR PLANE 28
C
            IF (IPL_AN.EQ.28) THEN

               IF (ISTRIP.GE.122.AND.ISTRIP.LE.132) THEN
                  IPH_LOW  = 0
                  IPH_HIGH = 0
               END IF

               DO kkk=1,11
                  DUMMY = 122 - kkk
                  IF (ISTRIP.EQ.DUMMY) THEN
                     ISTRIP = CORRECT(kkk)
                  END IF
               END DO

               IF (ISTRIP.GE.111.AND.ISTRIP.LE.121) THEN
                  IPH_LOW  = 0
                  IPH_HIGH = 0
               END IF

               DO kkk = 1,11
                  DUMMY = kkk + 121
                  IF (ISTRIP.EQ.DUMMY) THEN
                     ISTRIP = CORRECT1(kkk)
                  END IF
               END DO

            END IF
C
C END OF CORRECTION
C
            CLCON(ISTRIP,IPL_AN) = ST_CHARGE

         END IF

      END DO

      DO mmm = 1,7
         BEG_OF_PACK(mmm) = 1 + 8 * (mmm - 1)
         END_OF_PACK(mmm) = 8 + 8 * (mmm - 1)
      END DO

C ----------------------------------------------------------------

      DO ii=1,NPLANES ! Loop over pickup strip planes

C ---------------------- ----------------------------------------
C
C plane of reference to locate clusters
C
         REF = ii + 1
C ----------------------------------------------------------------
C CHECK THE END OF THE PACK
         DO JJ=1,7
            IF (ii.EQ.END_OF_PACK(JJ)) THEN
               REF = ii - 1
            END IF
         END DO
C ----------------------------------------------------------------
C
C     where it begins, where it ends..
C
```

```
        BEGIN = 1
        END   = NSTRIPS
C
C     in each gap in the first and second planes first 22 & last 22 strips are
C     not read out
C
        DO JJ=1,14
           IF (ii.EQ.ABSENT(JJ)) THEN
              BEGIN = 23
              END   = 154
           END IF
        END DO
C
C     put ADC values of strips into PH array
C
        DO JJ=BEGIN,END
           PH(JJ) = CLCON(JJ,ii)
        END DO
C
C     get the strip number having the maximum pulse height in ADC counts
C
        LENGTH     = 1 + END - BEGIN
        PL_MAX(ii) = MAXFZE(PH(BEGIN),LENGTH)
        PL_MAX(ii) = PL_MAX(ii) + BEGIN - 1
C
C     reject data if the max. PH of the plane is zero!
C
        IF (PH(PL_MAX(ii)).LT.0.5) THEN
           REJECT(ii) = 1
           GO TO 99
        END IF
C
C     reject data if the max. pulse height happens to be "isolated"
C
IF (PL_MAX(ii).LT.END-1.AND.PL_MAX(ii).GT.BEGIN+1) THEN
   IF (CLCON(PL_MAX(ii)-1,ii).EQ.0. .AND.
     +        CLCON(PL_MAX(ii)+1,ii).EQ.0.) THEN
      REJECT(ii) = 1
      GO TO 77
   END IF
END IF
C
C reject data if the max pulse height occurs near the edge of a plane
C
        IF (PL_MAX(ii).LE.BEGIN+1.OR.PL_MAX(ii).GE.END-1) THEN
           REJECT(ii) = 1
           GO TO 77
        END IF
C
C ------------------------------------------------------------------

 77     CONTINUE

        PH_CTR  = 0
        TOTALQ  = 0.
C
C maximum pulse heights even if they are isolated
C
        CALL HFILL(1001,PH(PL_MAX(ii)),0.,1.)
C
C raw pulse height of pickup strips (in ADC counts)
C
        DO JJ =BEGIN,END
           IF (PH(JJ).GT.0.) THEN
              TOTALQ = TOTALQ + PH(11)
```

```
            PH_CTR = PH_CTR + 1
          END IF
        END DO
C
C raw mean charge (in ADC counts) of planes
C
        IF (PH_CTR.GT.0) THEN
          PH_AVRG = TOTALQ / PH_CTR
          CALL HFILL(1002,PH_AVRG,0.,1.)
        END IF
C
C     raw total charge (in ADC counts) of planes
C
        CALL HFILL(1003,TOTALQ,0.,1.)

C     ------------------------------------------------------------------
C
C     USE OF DIGITAL SINGLE HIT PLANES AS REFERENCE
C
C     reject data if there is more than one digital hit in plane ii
C
C     IF (SHOWER(ii).EQ.1) THEN
C       REJECT(ii) = 1
C       GO TO 88
C     END IF
C
C reject data if there is no digital info available
C
C       IF (NO_INFO(ii).EQ.1) THEN
C           REJECT(ii) = 1
C           GO TO 88
C       END IF
C
C check whether there is a single hit near the maximum pulse height
C
C         HIT_POS = (LISTPL(REF)+1) / 2
C
C         IF (HIT_POS.NE.0) THEN
C           IF (KEPT_DIG(REF).EQ.1) THEN
C             IF (ABS(PL_MAX(ii)-HIT_POS).LT.2) THEN
C               KEEP(ii) = 1
C              WRITE(*,*)'kept at the first trial!!!','plane =',ii
C               GO TO 88
C             END IF
C           END IF
C         END IF
C
C SECOND TRIAL TO FIND A REFERENCE
C
C         DO JJ = 1,7
C           IF (ii.EQ.BEG_OF_PACK(JJ).OR.
C     +         ii.EQ.END_OF_PACK(JJ)) THEN
C             GO TO 88
C           END IF
C         END DO

C         REF=ii-1
C         HIT_POS = (LISTPL(REF) + 1) / 2

C         IF (KEPT_DIG(REF).EQ.1) THEN
C           IF (ABS(PL_MAX(ii)-HIT_POS).LT.2) THEN
C             KEEP(ii) = 1
C            WRITE(*,*)'kept at the second trial!!!','plane =',ii
C             GO TO 88
C           END IF
```

```
C          END IF

C ------------------------------------------------------------------------

 88     CONTINUE

C
C raw data clusters (treat everything as a cluster)
C
        ISTART  = 0
        PH_CTR  = 0
        TOTALQ  = 0.

        DO JJ = BEGIN,END

           IF (PH(JJ).GT.0.0.AND.JJ.LT.END) THEN

              IF (ISTART.EQ.0) THEN
                 ISTART = jj
              END IF

              PH_CTR  = PH_CTR + 1
              TOTALQ  = TOTALQ + PH(jj)
C
C raw data cluster pulse heights (in ADC counts)
C
              CALL HFILL(1004,PH(jj),0.,1.)

           ELSE IF (ISTART.NE.0) THEN

              PH_AVRG = TOTALQ / PH_CTR
C
C raw mean cluster charge (in ADC counts) of planes
C
              CALL HFILL(1005,PH_AVRG,0.,1.)
C
C raw total cluster charge (in ADC counts) of planes
C
              CALL HFILL(1007,TOTALQ,0.,1 )
C
C raw data cluster lengths
C
              CLEN=JJ-ISTART
              CALL HFILL(1006,FLOAT(CLEN),0.,1.)

              ISTART  = 0
              PH_CTR  = 0
              TOTALQ  = 0

           END IF

        END DO                  ! end of jj

C ------------------------------------------------------------------------

 99     CONTINUE

      END DO                    ! ii

C ------------------------------------------------------------------------
C
C calculate COG points
C
      CALL CALCULATE_COG
C
```

```
C least squares fit using COG points (REJECTION OF BAD POINTS BY PIPE)
C
      CALL ANALOGFIT
C
C FIND NOISE IF THERE IS A MUON RECONSTRUCTED IN BOTH PROJECTIONS
C
      IF (BOTH) THEN
         CALL FIND_NOISE
      END IF

      END  ! End of REAN

*********************************************************************
      SUBROUTINE CALCULATE_COG
*********************************************************************

      PARAMETER (SIGMA = 0.51961524) ! d / SQRT(12), d=1.8 cm (strip width)
      PARAMETER (NGAPS   = 7       )
      PARAMETER (NPLANES = 56      )
      PARAMETER (NSTRIPS = 176     )

      COMMON /SPEC_GEOM_DATA/
     +     X_PL(NPLANES),
     +     ANA_PL_SHIFT(NPLANES),
     +     ADJ_ANA_PL_SHIFT(NPLANES),
     +     DIG_PL_SHIFT(NPLANES),
     +     ADJ_DIG_PL_SHIFT(NPLANES),
     +     STRIP_YZ(NSTRIPS),
     +     WIRE_YZ(352)

      COMMON /FITDATA/ CNTR_OF_G(NPLANES),PLANE(NPLANES)
      REAL             CNTR_OF_G
      INTEGER                          PLANE

      COMMON /CLCONADC/ CLCON(NSTRIPS,NPLANES)
      REAL              CLCON

      COMMON /NOISE/ M_NOISE(NSTRIPS,NPLANES)
      REAL           M_NOISE

      COMMON /CLSTRSEARCH/ PL_MAX(NPLANES),REJECT(NPLANES),KEEP(NPLANES)
     +     ,KEPT_DIG(NPLANES),NO_INFO(NPLANES),SHOWER(NPLANES)
      INTEGER PL_MAX
      INTEGER REJECT
      INTEGER KEEP
      INTEGER KEPT_DIG
      INTEGER NO_INFO
      INTEGER SHOWER

      REAL STRIP_PH(176)

      REAL STRIPQ               ! total charge of strips in a cluster
      REAL AVRG_CH              ! average charge of clusters
      REAL NUM, DEN             ! NUMERATOR AND DENOMINATOR in COG calculation
      REAL COG                  ! center of gravity
      REAL STRIP_Q

      COMMON /ERRORS/ ERR(NPLANES)
      REAL            ERR

      REAL ERR_SUM

      INTEGER DUMMY
      INTEGER IPR               ! 1 = vertical strips, 2 = horizontal strips
```

```
      INTEGER CL, CL1, CL2

      INTEGER N
      INTEGER LSTART,LEND

      INTEGER STR_MAX         ! strip # having the max. ADC counts in a cluster
      REAL    MAX_PH            ! max. ADC counts in a cluster


      REAL    F_R, F_L, F_T, TQ, LL, RR

      COMMON /OFFTRACK/ NOISE_REGION(56,2)
      INTEGER           NOISE_REGION

      COMMON /INDUCEDQ/ CLUSTER(56,4)
      REAL              CLUSTER

      REAL P1, P2, P3, C1, C2, C3, TOT_PH

C     ---------------------------------
      CALL VZERO(PLANE,NPLANES    )
      CALL VZERO(CNTR_OF_G,NPLANES)
      CALL VZERO(ERR,NPLANES      )
C     ---------------------------------

      DO i = 1,2                ! Loop over two projections; 1 = Z, 2 = Y

C     ------------------------------------------------------------------------

         IF (i.EQ.1) THEN
            LSTART = 1
            LEND   = 55
         END IF

         IF (i.EQ.2) THEN
            LSTART = 2
            LEND   = 56
         END IF

         DO ipl = LSTART,LEND,2
C
C     Skip planes which have failed to have a cluster
C
            IF (REJECT(ipl).EQ.1) THEN
               GO TO 200
            END IF
C
C     SKIP PLANES WHICH DO NOT HAVE A DIGITAL REFERENCE PLANE
C
C     IF (KEEP(ipl).EQ.0) THEN
C     GO TO 200
C     END IF
C
C  INITIALIZATION OF THE CLUSTER PARAMETERS
C
            N        = 0
            STRIPQ   = 0.
            TQ       = 0.
            AVRG_CH  = 0.
            COG      = 0.
            NUM      = 0.
            STR_PT   = PL_MAX(ipl)
C
C     LOOP OVER STRIPS OF PLANES
C
```

```
            DO istrip=1,NSTRIPS
               STRIP_PH(istrip) = CLCON(istrip,ipl)
            END DO              ! istrip

            MAX_PH =  CLCON(PL_MAX(ipl),ipl)
            CALL HFILL(783,MAX_PH,0.,1.)
C
C     CALCULATE TOTAL CLUSTER CHARGE
C
            KK=PL_MAX(ipl)+1
            TQ=MAX_PH
 1          CONTINUE
            IF (STRIP_PH(KK).GT.0.5) THEN
               CALL HFILL(783,STRIP_PH(KK),0.,1.)
               TQ = TQ+STRIP_PH(KK)
               KK = KK+1
               GOTO 1
            END IF

            KK=PL_MAX(ipl)-1
 2          CONTINUE
            IF (STRIP_PH(KK).GT.0.5) THEN
               CALL HFILL(783,STRIP_PH(KK),0.,1.)
               TQ = TQ+STRIP_PH(KK)
               KK = KK-1
               GOTO 2
            END IF

            CALL HFILL(786,TQ,0.,1.)

            CL=0
            ERR_SUM = 0
C
C Start fom the pick up strip having the maximum pulse height
C
            DUMMY     = PL_MAX(ipl)
            YZ_STRIP  = STRIP_YZ(DUMMY)
     +           + ANA_PL_SHIFT(ipl)
     +           + ADJ_ANA_PL_SHIFT(ipl)
            NUM       = NUM + YZ_STRIP * STRIP_PH(DUMMY)
            CALL HFILL(784,STRIP_PH(DUMMY),0.,1.)
            ERR_SUM   = ERR_SUM + STRIP_PH(DUMMY) * STRIP_PH(DUMMY)
C
C     Go through strips on the left side of the maximum
C
            CL1=0
            LL=0.0

 784        CONTINUE

            IF (DUMMY.GE.2) THEN
               DUMMY = DUMMY - 1
               LL=LL-1.0
            END IF

            IF (DUMMY.EQ.1) THEN
               GO TO 801
            END IF

            IF (STRIP_PH(DUMMY).GT.0.) THEN
               IF (STRIP_PH(DUMMY+1).GE.STRIP_PH(DUMMY)) THEN
                  YZ_STRIP   = STRIP_YZ(DUMMY)
     +                 + ANA_PL_SHIFT(ipl)
     +                 + ADJ_ANA_PL_SHIFT(ipl)
                  NUM        = NUM + YZ_STRIP * STRIP_PH(DUMMY)
```

```
                CL1=CL1+1
                ERR_SUM    = ERR_SUM
     +                + STRIP_PH(DUMMY) * STRIP_PH(DUMMY)

                F_L = 100 * STRIP_PH(DUMMY) / TQ
                F_T = 100 * STRIP_PH(DUMMY) / MAX_PH

                CALL HFILL(1500,LL,F_L,1.)
                CALL HFILL(1600,LL,F_T,1.)

                GO TO 784
             END IF
          END IF
C
C     Go through strips on the right side of the maximum
C
          CL2=0
          RR=0.0
          F_R = 100 * MAX_PH / TQ
          CALL HFILL(1500,RR,F_R,1.)

          DUMMY = PL_MAX(ipl)

 801      CONTINUE
          IF (DUMMY.LE.175) THEN
             DUMMY = DUMMY + 1
             RR=RR+1.0
          END IF

          IF (DUMMY.EQ.176) THEN
             GO TO 2251
          END IF

          IF (STRIP_PH(DUMMY).GT.0.) THEN

             IF (STRIP_PH(DUMMY-1).GE.STRIP_PH(DUMMY)) THEN
                YZ_STRIP    = STRIP_YZ(DUMMY)
     +                + ANA_PL_SHIFT(ipl)
     +                + ADJ_ANA_PL SHIFT(ipl)
                NUM         = NUM + YZ_STRIP * STRIP_PH(DUMMY)
                CL2=CL2+1
                ERR_SUM = ERR_SUM
     +                + STRIP_PH(DUMMY) * STRIP_PH(DUMMY)

                F_R = 100 * STRIP_PH(DUMMY) / TQ
                F_T = 100 * STRIP_PH(DUMMY) / MAX_PH

                CALL HFILL(1500,RR,F_R,1.)
                CALL HFILL(1600,RR,F_T,1.)

                GO TO 801
             END IF
          END IF

 2251     IF (TQ.GT.0.5) THEN
             PLANE(ipl)=1

             CNTR_OF_G(ipl) = NUM / TQ

             CL = CL1 + CL2 + 1
             CALL HFILL(785,FLOAT(CL),0.,1.)

             AVRG_CH  = TQ / CL
             CALL HFILL(787,AVRG_CH,0.,1.)
```

```
                ERR(ip1) = SIGMA * (SQRT(ERR_SUM) / TQ)

                NOISE_REGION(ip1,1)  = STR_PT - CL1 - 11

                IF (NOISE_REGION(ip1,1).LT.0) THEN
                   NOISE_REGION(ip1,1) = 1
                END IF

                NOISE_REGION(ip1,2)  = STR_PT + CL2 + 11

                IF (NOISE_REGION(ip1,2).GT.176) THEN
                   NOISE_REGION(ip1,2)  = 176
                END IF

             END IF

             CLUSTER(ip1,1)  = PL_MAX(ip1) - CL1
             CLUSTER(ip1,2)  = PL_MAX(ip1) + CL2
             CLUSTER(ip1,3)  = FLOAT(CL)
             CLUSTER(ip1,4)  = TQ

C              IF (CL.GT.3) THEN
C                 NUM=0.0
C                 DEN=0.0
C                 YZ_STRIP  = STRIP_YZ(PL_MAX(ip1))
C     +                     + ANA_PL_SHIFT(ip1)
C     +                     + ADJ_ANA_PL_SHIFT(ip1)
C                 NUM = NUM + STRIP_PH(PL_MAX(ip1)) * YZ_STRIP
C                 DEN = DEN + STRIP_PH(PL_MAX(ip1))
C                 LL=1
C                 N=PL_MAX(ip1)+1
C                 M=PL_MAX(ip1)-1
C 999             IF (STRIP_PH(N).GE.STRIP_PH(M)) THEN
C                    KK=N
C                    LL=LL+1
C                    N=N+1
C                 ELSE
C                    KK=M
C                    LL=LL+1
C                    M=M-1
C                 END IF
C                 IF (LL.GT.3) GO TO 99
C                 YZ_STRIP  = STRIP_YZ(KK)
C     +                     + ANA_PL_SHIFT(ip1)
C     +                     + ADJ_ANA_PL_SHIFT(ip1)
C                 NUM = NUM + STRIP_PH(KK) * YZ_STRIP
C                 DEN = DEN + STRIP_PH(KK)
C                 GO TO 999
C 99              CNTR_OF_G(ip1) = NUM / DEN
C              END IF

             JJ=PL_MAX(ip1)
             TOT_PH = STRIP_PH(JJ-1) + STRIP_PH(JJ) + STRIP_PH(JJ+1)

             CL=0
             CB = 0.05 * TOT_PH
             P1 = STRIP_PH(JJ-1) - CB
             IF (P1.LT.0.0) P1 = 0.0
             IF (P1.GT.0.0) CL=CL+1
             P2 = STRIP_PH(JJ  ) - CB
             IF (P2.LT.0.0) P2 = 0.0
             IF (P2.GT.0.0) CL=CL+1
             P3 = STRIP_PH(JJ+1) - CB
             IF (P3.LT.0.0) P3 = 0.0
             IF (P3.GT.0.0) CL=CL+1
```

```
            TOT_PH = P1 + P2 + P3

            C1 = STRIP_YZ(JJ-1)
     +           + ANA_PL_SHIFT(ipl)
     +           + ADJ_ANA_PL_SHIFT(ipl)
            C2 = STRIP_YZ(JJ)
     +           + ANA_PL_SHIFT(ipl)
     +           + ADJ_ANA_PL_SHIFT(ipl)
            C3 = STRIP_YZ(JJ+1)
     +           + ANA_PL_SHIFT(ipl)
     +           + ADJ_ANA_PL_SHIFT(ipl)

            CNTR_OF_G(ipl) = (P1*C1 + P2*C2 + P3*C3) / TOT_PH
            ERR(ipl) = 0.51961524*SQRT(P1*P1 + P2*P2 + P3*P3)/TOT_PH
C
C Empirical fit for cluster sizes greater than or equla to three
C
            IF (CL.EQ.3) THEN
               PI  = 3.1415927
               PIW = 3.1415927 * 2.1
               F1  = SQRT(P2/P3)
               F2  = SQRT(P2/P1)
               A3  = PIW
     +            / LOG  (0.5  *  (F1 + F2)
     +            + SQRT((0.5  *  (F1 + F2))
     +            *      (0.5  *  (F1 + F2)) - 1))
               F3  = (EXP(PIW / A3) - EXP(-PIW / A3))
               A2  = (A3 / (2*PI))
     +            * LOG (ABS(((F1 - F2) / F3 + 1)
     +            /      ((F1 - F2) / F3 - 1)))
               CNTR_OF_G(ipl) = C2 - A2
C              X=A2*10000
C              XEST = X - ((-39.43 * SIN(2 * PI *  X   / 21000))
C     +               +      (  2.52 * SIN(4 * PI *  X   / 21000))
C     +               +      ( -0.72 * SIN(6 * PI *  X   / 21000))
C     +               +      (  0.30 * SIN(8 * PI *  X   / 21000)))
C              XCOR = X - ((-39.43 * SIN(2 * PI * XEST / 21000))
C     +               +      (  2.52 * SIN(4 * PI * XEST / 21000))
C     +               +      ( -0.72 * SIN(6 * PI * XEST / 21000))
C     +               +      (  0.30 * SIN(8 * PI * XEST / 21000)))
C             WRITE(*,*)CNTR_OF_G(ipl)
C               CNTR_OF_G(ipl) = C2 - XCOR * 0.0001
             END IF
C ---------------------------------------------------------------------

 200         CONTINUE
          END DO                ! ipl

C ---------------------------------------------------------------------

C     WRITE(*,*) '******************'

C ---------------------------------------------------------------------

       END DO                   ! i

C ---------------------------------------------------------------------

C      WRITE(*,*) '******************'

       RETURN

       END

*********************************
```

```
      SUBROUTINE ANALOGFIT
*********************************

      PARAMETER (NSTRIPS  = 176 )
      PARAMETER (NPLANES  = 56  )
      PARAMETER (CHI2MIN  = 0.05)
      PARAMETER (MIN_HITS = 9   )

      COMMON /SPEC_GEOM_DATA/
     +     X_PL(56),
     +     ANA_PL_SHIFT(56),
     +     ADJ_ANA_PL_SHIFT(56),
     +     DIG_PL_SHIFT(56),
     +     ADJ_DIG_PL_SHIFT(56),
     +     STRIP_YZ(176),
     +     WIRE_YZ(352)

      COMMON /INDUCEDQ/ CLUSTER(56,4)
      REAL           CLUSTER

      REAL PH, DD

      COMMON /CLCONADC/ CLCON(NSTRIPS,NPLANES)
      REAL              CLCON

      COMMON /FITDATA/ CNTR_OF_G(NPLANES),PLANE(NPLANES)
      REAL             CNTR_OF_G
      INTEGER                           PLANE

      COMMON /PROJECTIONS/ GAPBOOL(2)
      INTEGER GAPBOOL

      COMMON/ERRORS/ERR(NPLANES)
      REAL ERR

      COMMON /MUONBOOL/ BOTH
      LOGICAL           BOTH

      LOGICAL MUON

      COMMON  /SAMEMUON/ SAME
      INTEGER            SAME

      REAL    LF_X1(56)
      REAL    LF_X2(56)
      REAL    LF_SIG(56)
      INTEGER LF_PLN(56), N

      REAL MIN_CONF_LEV, CHI2_PROB, SLOPE, INTERSECTION, CHI2, TQ

C -------------------------
      CALL VZERO(GAPBOOL,2)
      CALL VZERO(LF_PLN,56)
C -------------------------

      BOTH=.FALSE.
      MUON=.FALSE.

      MIN_CONF_LEV = 0.05
      N=0
      DO I=2,56,2
         IF (PLANE(I).EQ.1) THEN
            N=N+1
            LF_X1(N)    = X_PL(I) - .757
            LF_X2(N)    = CNTR_OF_G(I)
```

```
      LF_SIG(N)  = ERR(I)
      LF_PLN(N)  = I
    ENDIF
  ENDDO

  IF (N.GT.MIN_HITS) THEN
    CALL PIPE(LF_X1,LF_X2,LF_SIG,LF_PLN,N,MIN_CONF_LEV,
+       CHI2,CHI2_PROB,SLOPE,INTERSECTION)
  END IF

  IF (N.GT.MIN_HITS) THEN
    CALL HFILL(9001,FLOAT(N),0.,1.)
    CALL HFILL(9003,CHI2_PROB,0.,1.)
    CALL HFILL(9005,CHI2/(FLOAT(N-2)),0.,1.)
    MUON=.TRUE.
    GAPBOOL(2) = 1
    DO I=1,N
      PLANE(LF_PLN(I)) = 100
      D = SLOPE * LF_X1(I) + INTERSECTION
      RESIDUAL = LF_X2(I) - D
      CALL HFILL(9007,RESIDUAL,0.,1.)
      CALL HFILL(9009,RESIDUAL,0.,1.)
    END DO
  END IF

  DO I=2,56,2
    IF (PLANE(I).EQ.100) THEN
      DO II=INT(CLUSTER(I,1)),INT(CLUSTER(I,2))
        PH  = CLCON(II,I) / CLUSTER(I,4)
        DD  = (SLOPE * (X_PL(I) - .757) + INTERSECTION)
+           - STRIP_YZ(II)
+           - ANA_PL_SHIFT(I)
+           - ADJ_ANA_PL_SHIFT(I)

        IF (CLUSTER(I,3).EQ.2.)  CALL HFILL(5002,DD,PH,1.)
        IF (CLUSTER(I,3).EQ.3.)  CALL HFILL(5003,DD,PH,1.)
        IF (CLUSTER(I,3).EQ.4.)  CALL HFILL(5004,DD,PH,1.)
        IF (CLUSTER(I,3).EQ.5.)  CALL HFILL(5005,DD,PH,1.)
        IF (CLUSTER(I,3).EQ.6.)  CALL HFILL(5006,DD,PH,1.)
        IF (CLUSTER(I,3).EQ.7.)  CALL HFILL(5007,DD,PH,1.)
        IF (CLUSTER(I,3).EQ.8.)  CALL HFILL(5008,DD,PH,1.)
        IF (CLUSTER(I,3).EQ.9.)  CALL HFILL(5009,DD,PH,1.)
        IF (CLUSTER(I,3).EQ.10.) CALL HFILL(5010,DD,PH,1.)

        TQ=CLUSTER(I,4)
        IF (CLUSTER(I,3).EQ.2.)  CALL HFILL(8002,TQ,0.,1.)
        IF (CLUSTER(I,3).EQ.3.)  CALL HFILL(8003,TQ,0.,1.)
        IF (CLUSTER(I,3).EQ.4.)  CALL HFILL(8004,TQ,0.,1.)
        IF (CLUSTER(I,3).EQ.5.)  CALL HFILL(8005,TQ,0.,1.)
        IF (CLUSTER(I,3).EQ.6.)  CALL HFILL(8006,TQ,0.,1.)
        IF (CLUSTER(I,3).EQ.7.)  CALL HFILL(8007,TQ,0.,1.)
        IF (CLUSTER(I,3).EQ.8.)  CALL HFILL(8008,TQ,0.,1.)
        IF (CLUSTER(I,3).EQ.9.)  CALL HFILL(8009,TQ,0.,1.)
        IF (CLUSTER(I,3).EQ.10.) CALL HFILL(8010,TQ,0.,1.)

      END DO
    END IF
  END DO

  N=0
  DO I=1,55,2
    IF (PLANE(I).EQ.1) THEN
      N=N+1
      LF_X1(N)   = X_PL(I) - .757
      LF_X2(N)   = CNTR_OF_G(I)
```

```
            LF_SIG(N)   = ERR(I)
            LF_PLN(N)   = I
         ENDIF
      ENDDO

      IF (N.GT.MIN_HITS) THEN
         CALL PIPE(LF_X1,LF_X2,LF_SIG,LF_PLN,N,MIN_CONF_LEV,
     +              CHI2,CHI2_PROB,SLOPE,INTERSECTION)
      END IF

      IF (N.GT.MIN_HITS) THEN
         CALL HFILL(9002,FLOAT(N),0.,1.)
         CALL HFILL(9004,CHI2_PROB,0.,1.)
         CALL HFILL(9006,CHI2/(FLOAT(N-2)),0.,1.)
         IF (MUON) THEN
            BOTH=.TRUE.
            SAME=SAME+1
         END IF
         GAPBOOL(1) = 1
         DO I=1,N
            PLANE(LF_PLN(I)) = 100
            D = SLOPE * LF_X1(I) + INTERSECTION
            RESIDUAL = LF_X2(I) - D
            CALL HFILL(9008,RESIDUAL,0.,1.)
            CALL HFILL(9009,RESIDUAL,0.,1.)
         END DO
      END IF

      DO I=1,55,2
         IF (PLANE(I).EQ.100) THEN
            DO II=INT(CLUSTER(I,1)),INT(CLUSTER(I,2))
               PH  = CLCON(II,I) / CLUSTER(I,4)
               DD  = (SLOPE * (X_PL(I) - .757) + INTERSECTION)
     +              - STRIP_YZ(II)
     +              - ANA_PL_SHIFT(I)
     +              - ADJ_ANA_PL_SHIFT(I)

               IF (CLUSTER(I,3).EQ.2.)  CALL HFILL(5002,DD,PH,1.)
               IF (CLUSTER(I,3).EQ.3.)  CALL HFILL(5003,DD,PH,1.)
               IF (CLUSTER(I,3).EQ.4.)  CALL HFILL(5004,DD,PH,1.)
               IF (CLUSTER(I,3).EQ.5.)  CALL HFILL(5005,DD,PH,1.)
               IF (CLUSTER(I,3).EQ.6.)  CALL HFILL(5006,DD,PH,1.)
               IF (CLUSTER(I,3).EQ.7.)  CALL HFILL(5007,DD,PH,1.)
               IF (CLUSTER(I,3).EQ.8.)  CALL HFILL(5008,DD,PH,1.)
               IF (CLUSTER(I,3).EQ.9.)  CALL HFILL(5009,DD,PH,1.)
               IF (CLUSTER(I,3).EQ.10.) CALL HFILL(5010,DD,PH,1.)

               TQ=CLUSTER(I,4)
               IF (CLUSTER(I,3).EQ.2.)  CALL HFILL(8002,TQ,0.,1.)
               IF (CLUSTER(I,3).EQ.3.)  CALL HFILL(8003,TQ,0.,1.)
               IF (CLUSTER(I,3).EQ.4.)  CALL HFILL(8004,TQ,0.,1.)
               IF (CLUSTER(I,3).EQ.5.)  CALL HFILL(8005,TQ,0.,1.)
               IF (CLUSTER(I,3).EQ.6.)  CALL HFILL(8006,TQ,0.,1.)
               IF (CLUSTER(I,3).EQ.7.)  CALL HFILL(8007,TQ,0.,1.)
               IF (CLUSTER(I,3).EQ.8.)  CALL HFILL(8008,TQ,0.,1.)
               IF (CLUSTER(I,3).EQ.9.)  CALL HFILL(8009,TQ,0.,1.)
               IF (CLUSTER(I,3).EQ.10.) CALL HFILL(8010,TQ,0.,1.)

               CALL HFILL(8020,CLUSTER(I,3),TQ,1.)

            END DO
         END IF
      END DO

C ------------------------------------------------------------------
```

```
C      WRITE(*,*) 'end of analog_fit'
 100  RETURN

      END

*********************************************************************
      SUBROUTINE PIPE(A_X1,A_X2,A_SIG,A_PLN,NO,PRO_LIM,C2
     +                                      ,CHI2_PROB,MMM,NNN)
*********************************************************************

      REAL    A_X1(NO), A_X2(NO), A_SIG(NO)
      INTEGER A_PLN(NO)

      REAL MMM, NNN, CHI2_PROB
      REAL MAX,DIST,DIST2

      INTEGER MAX_I

      INTEGER KK
      INTEGER MAX_II(10)

C -----------------------------------------------------------------

 55   CALL LINEFIT(A_X1, A_X2, A_SIG, NO, MMM, NNN, C2)

      CHI2_PROB = PROB(C2,NO-2)
      IF (CHI2_PROB.GE.PRO_LIM) GOTO 70

      MAX=0.0
      DO I=1,NO
         DIST=1.0/SQRT(1.0+MMM*MMM)*ABS(MMM*A_X1(I)-A_X2(I)+NNN)
         IF (DIST.GT.MAX) THEN
            MAX_I=I
            MAX=DIST
         END IF
      END DO

      A_X1(MAX_I)  = A_X1(NO)   ! exclude the plane
      A_X2(MAX_I)  = A_X2(NO)   ! with the maximum distance
      A_SIG(MAX_I) = A_SIG(NO)  ! to the reconstructed
      A_PLN(MAX_I) = A_PLN(NO)  ! track
      NO=NO-1

      IF (NO.LT.3) GOTO 70
      GOTO 55

 70   CONTINUE

      RETURN
      END

*************************************************
      SUBROUTINE LINEFIT(X1,X2,SIGM,NNN,AA,BB,CHI2)
*************************************************

      DIMENSION X1(NNN),X2(NNN),SIGM(NNN)
      INTEGER II

      REAL LF_A, LF_B, LF_C, LF_D, LF_E, LF_F,CHI2

C -------------------------------------------------

      LF_A=0.0
      LF_B=0.0
```

```
      LF_C=0.0
      LF_D=0.0
      LF_E=0.0
      LF_F=0.0

      DO II=1,NNN
         LF_A  =  LF_A + X1(II) / SIGM(II) / SIGM(II)
         LF_B  =  LF_B +   1    / SIGM(II) / SIGM(II)
         LF_C  =  LF_C + X2(II) / SIGM(II) / SIGM(II)
         LF_D  =  LF_D + X1(II) *  X1(II)  / SIGM(II) / SIGM(II)
         LF_E  =  LF_E + X2(II) *  X1(II)  / SIGM(II) / SIGM(II)
         LF_F  =  LF_F + X2(II) *  X2(II)  / SIGM(II) / SIGM(II)
      ENDDO

      AA = (LF_E * LF_B - LF_C * LF_A) / (LF_D * LF_B - LF_A *LF_A)
      BB = (LF_D * LF_C - LF_E * LF_A) / (LF_D * LF_B - LF_A *LF_A)
C
C calculation of chi2
C
      CHI2=0.0
      DO II=1,NNN
         CHI2 = CHI2 + ( X2(II) - AA * X1(II) - BB) *
     +                 ( X2(II) - AA * X1(II) - BB) /
     +                 SIGM(II)  /   SIGM(II)
      ENDDO

      RETURN

      END

*********************************************************************
      SUBROUTINE FIND_NOISE
*********************************************************************

      PARAMETER (NGAPS   = 7  )
      PARAMETER (NPLANES = 56 )
      PARAMETER (NSTRIPS = 176)

      COMMON /SPEC_GEOM_DATA/
     + X_PL(NPLANES),
     + ANA_PL_SHIFT(NPLANES),
     + ADJ_ANA_PL_SHIFT(NPLANES),
     + DIG_PL_SHIFT(NPLANES),
     + ADJ_DIG_PL_SHIFT(NPLANES),
     + STRIP_YZ(NSTRIPS),
     + WIRE_YZ(352)

      COMMON /FITDATA/ CNTR_OF_G(NPLANES),PLANE(NPLANES)
      REAL CNTR_OF_G
      INTEGER PLANE

      COMMON /CLCONADC/ CLCON(NSTRIPS,NPLANES)
      REAL              CLCON

      COMMON /PEDESTAL/ PDT(NSTRIPS,NPLANES),PDTCTR(NSTRIPS,NPLANES)
      REAL              PDT,                 PDTCTR

      COMMON /PROBABILITY/ N_STRIPS, N_TIMES
      REAL                 N_STRIPS, N_TIMES

      COMMON /CLSTRSEARCH/ PL_MAX(NPLANES),REJECT(NPLANES),KEEP(NPLANES)
     +              ,KEPT_DIG(NPLANES),NO_INFO(NPLANES),SHOWER(NPLANES)

      INTEGER PL_MAX, REJECT, KEEP, KEPT_DIG, NO_INFO, SHOWER
```

```
      REAL PH(NSTRIPS)      ! ADC content of strips of each plane
      REAL CL_Q
      LOGICAL FIRST
      INTEGER BEGIN
      INTEGER END
      INTEGER LENGTH
      INTEGER ISTART
      INTEGER NCL

      INTEGER PH_CTR, CLEN
      REAL    TOTALQ, P

      REAL STRIP_PH(NSTRIPS)
      REAL STRIPQ         ! total charge of pedestals
      REAL CLT_CHG        ! total charge of strips in a cluster
      REAL MAX_PH
      REAL CL, NC, DD

      INTEGER STR_PT      ! where the clstr starts
      INTEGER STP_PT      ! where the clstr ends

      INTEGER LSTART
      INTEGER LEND

      INTEGER ABSENT(14) ! first and last 22 strips of the first and second
      DATA    ABSENT       /1,2,9,10,17,18,25,26,33,34,41,42,49,50/

      COMMON /OFFTRACK/ NOISE_REGION(56,2)
      INTEGER           NOISE_REGION

* ------------------------------------------------------------------
*
*-- LOOP OVER PLANES IN EACH EVENT
*
      DO i=1,2 ! loop over the two projections 1=horizontal 2=vertical

           IF (i.EQ.1) THEN
              LSTART = 1
              LEND   = 55
           END IF

           IF (i.EQ.2) THEN
              LSTART = 2
              LEND   = 56
           END IF

           DO ipl=LSTART,LEND,2

             IPR  = MOD(ipl,2) + 1
C
C reject planes not having participated in least squares fit
C
             IF (PLANE(ipl).NE.100) THEN
                GO TO 200
             END IF

             CLT_CHG=0.
             CL=1.0

          DO ii = 1,NSTRIPS
             STRIP_PH(ii) = CLCON(ii,ipl)
          END DO

          STR_PT = PL_MAX(ipl)
          MAX_PH = CLCON(STR_PT,ipl)
```

```
          CLT_CHG = CLT_CHG + CLCON(STR_PT,ipl)
          CLCON(STR_PT,ipl) = 0.

 784      CONTINUE

          IF(STR_PT.GE.2)THEN
            STR_PT =  STR_PT - 1
          END IF

          IF(STR_PT.EQ.1)THEN
            GO TO 801
          END IF

          IF(STRIP_PH(STR_PT).GT.0.)THEN
            CLT_CHG = CLT_CHG + CLCON(STR_PT,ipl)
            CL=CL+1.0
            CLCON(STR_PT,ipl) = 0.
            GO TO 784
          END IF

          STR_PT = PL_MAX(ipl)

 801      CONTINUE
          IF(STR_PT.LE.175)THEN
            STR_PT = STR_PT + 1
          END IF

          IF(STR_PT.EQ.176)THEN
            GO TO 2251
          END IF

          IF (STRIP_PH(STR_PT).GT.0.) THEN
             CLT_CHG = CLT_CHG + CLCON(STR_PT,ipl)
             CL=CL+1.0
             CLCON(STR_PT,ipl) = 0.0
             GO TO 801
          END IF

 2251     CONTINUE

          DO JJ=1,14
             IF (ipl.EQ.ABSENT(JJ)) THEN
                N_STRIPS=N_STRIPS+154.0-CL
                GO TO 999
             END IF
          END DO
          N_STRIPS=N_STRIPS+176.0-CL
 999      CONTINUE
C
C accumulation of pedestal values of each strip
C

C
C put ADC values of noisy strips into PH array
C

        DO JJ=NOISE_REGION(ipl,1),NOISE_REGION(ipl,2)
           CLCON(JJ,ipl) = 0.0
        END DO

        DO JJ=1,176
          PH(JJ) = CLCON(JJ,ipl)
        END DO
C
```

```
C STUDY OF NOISE CLUSTERS
C
        ISTART  = 0
        PH_CTR  = 0
        TOTALQ  = 0.

        DO jj = 1,176

          IF (PH(jj).GT.0.5) THEN

             IF (ISTART.EQ.0) THEN
                 ISTART = jj
             END IF

             PH_CTR  = PH_CTR + 1
             TOTALQ  = TOTALQ + PH(jj)
C
C CLUSTER NOISE PULSE HEIGHT
C
             CALL HFILL(3001,PH(jj),0.,1.)

          ELSE IF (ISTART.NE.0) THEN
C
C CLUSTER TOTAL NOISE CHARGE
C
             CALL HFILL(3002,TOTALQ,0.,1.)

             PH_AVRG = TOTALQ / PH_CTR
C
C CLUSTER AVERAGE NOISE CHARGE
C
             CALL HFILL(3003,PH_AVRG,0.,1.)

             CLEN=jj-ISTART
C
C CLUSTER LENGTH OF NOISE
C
            CALL HFILL(3004,FLOAT(CLEN),0.,1.)

            IF (CLEN.EQ.2)  CALL HFILL(7002,TOTALQ,0.,1.)
            IF (CLEN.EQ.3)  CALL HFILL(7003,TOTALQ,0.,1.)
            IF (CLEN.EQ.4)  CALL HFILL(7004,TOTALQ,0.,1.)
            IF (CLEN.EQ.5)  CALL HFILL(7005,TOTALQ,0.,1.)
            IF (CLEN.EQ.6)  CALL HFILL(7006,TOTALQ,0.,1.)
            IF (CLEN.EQ.7)  CALL HFILL(7007,TOTALQ,0.,1.)
            IF (CLEN.EQ.8)  CALL HFILL(7008,TOTALQ,0.,1.)
            IF (CLEN.EQ.9)  CALL HFILL(7009,TOTALQ,0.,1.)
            IF (CLEN.EQ.10) CALL HFILL(7010,TOTALQ,0.,1.)

 2          DO II=ISTART,ISTART+PH_CTR-1
               P   = PH(II) / TOTALQ
               DD  = (STRIP_YZ(ISTART)
     +             + STRIP_YZ(ISTART+PH_CTR-1)) / 2
     +             - STRIP_YZ(II)
               IF (CLEN.EQ.2)  CALL HFILL(6002,DD,P,1.)
               IF (CLEN.EQ.3)  CALL HFILL(6003,DD,P,1.)
               IF (CLEN.EQ.4)  CALL HFILL(6004,DD,P,1.)
               IF (CLEN.EQ.5)  CALL HFILL(6005,DD,P,1.)
               IF (CLEN.EQ.6)  CALL HFILL(6006,DD,P,1.)
               IF (CLEN.EQ.7)  CALL HFILL(6007,DD,P,1.)
               IF (CLEN.EQ.8)  CALL HFILL(6008,DD,P,1.)
               IF (CLEN.EQ.9)  CALL HFILL(6009,DD,P,1.)
               IF (CLEN.EQ.10) CALL HFILL(6010,DD,P,1.)
            END DO
```

```
            ISTART  = 0
            PH_CTR  = 0
            TOTALQ  = 0.

          END IF

        END DO ! end of jj

        STRIPQ=0.0
        IF (NOISE_REGION(ipl,1).GT.1) THEN
             NC=0.0
             DO JJ=1,NOISE_REGION(ipl,1)

               PDTCTR(JJ,ipl) = PDTCTR(JJ,ipl) +      1
               PDT(JJ,ipl)    = PDT(JJ,ipl)    + STRIP_PH(JJ)

               STRIPQ  = STRIPQ + STRIP_PH(JJ)

               CALL HFILL(2500+ipl,FLOAT(JJ),CLCON(JJ,ipl),1.)

               IF (CLCON(JJ,ipl).GT.0.5) THEN
                  NC=NC+1.0
                  CALL HFILL(100,STRIP_PH(JJ),0.,1.)
                  CALL HFILL(100+ipl,STRIP_PH(JJ),0.,1.)
                  CALL HFILL(400+ipl,STRIP_PH(JJ),FLOAT(JJ),1.)
                  CALL HFILL(300+ipl,FLOAT(JJ),0.,1.)
               END IF

             END DO
             N_TIMES=N_TIMES+NC
        END IF

        IF (NOISE_REGION(ipl,1).LT.176) THEN
             NC=0.0
             DO JJ=NOISE_REGION(ipl,2),NSTRIPS

               PDTCTR(JJ,ipl) = PDTCTR(JJ,ipl) +      1
               PDT(JJ,ipl)    = PDT(JJ,ipl)    + STRIP_PH(JJ)

               STRIPQ  = STRIPQ + STRIP_PH(JJ)

               CALL HFILL(2500+ipl,FLOAT(JJ),CLCON(JJ,ipl),1.)

               IF (CLCON(JJ,ipl).GT.0.5) THEN
                  NC=NC+1.0
                  CALL HFILL(100,STRIP_PH(JJ),0.,1.)
                  CALL HFILL(100+ipl,STRIP_PH(JJ),0.,1.)
                  CALL HFILL(400+ipl,STRIP_PH(JJ),FLOAT(JJ),1.)
                  CALL HFILL(300+ipl,FLOAT(JJ),0.,1.)
               END IF

             END DO
             N_TIMES=N_TIMES+NC
        END IF
        CALL HFILL(45001,FLOAT(ipl),STRIPQ,1.)
        CALL HFILL(45002,FLOAT(ipl),CLT_CHG,1.)
C
C filling the relevant histograms
C
             IF (STRIPQ.NE.0.) THEN
                CALL HFILL(200,STRIPQ,0.,1.)
                CALL HFILL(200+ipl,STRIPQ,0.,1.)
             END IF

 200         CONTINUE
```

```
          END DO  ! ipl

 100    CONTINUE
      END DO  ! i (projection loop)

      RETURN

      END

*****************************
      SUBROUTINE SPEDESTAL
*****************************

      PARAMETER (NSTRIPS = 176,NPLANES = 56)
      COMMON /PEDESTAL/ PDT(NSTRIPS,NPLANES),PDTCTR(NSTRIPS,NPLANES)
      REAL              PDT,                 PDTCTR

      DIMENSION PEDESTAL(NSTRIPS+1)

      COMMON /PROBABILITY/ N_STRIPS, N_TIMES
      REAL                 N_STRIPS, N_TIMES

C ---------------------------------------------------------------------
      DO ii=1,NPLANES
         DO jj=1,NSTRIPS
C
C calculate the mean pedestal values of ecah strip
C
            IF (PDTCTR(jj,ii).GT.0.0) THEN
               PDT(jj,ii) = PDT(jj,ii) / PDTCTR(jj,ii)
            END IF
C
C fill histograms of mean pedestals for horizontal planes
C
            IF (PDT(jj,ii).GT.0.0) THEN
               IF (MOD(ii,2).EQ.1) THEN
                  CALL HFILL(20001,PDT(jj,ii),0.,1.)
               END IF
            END IF
C
C fill histograms of mean pedestals for vertical planes
C
            IF (PDT(jj,ii).GT.0.) THEN
               IF (MOD(ii,2).EQ.0) THEN
                  CALL HFILL(20002,PDT(jj,ii),0.,1.)
               END IF
            END IF

            PEDESTAL(jj) = PDT(jj,ii)
            CALL HFILL(4000+ii,FLOAT(jj),0.,1.)
            WRITE(95)PDT(jj,ii)
         END DO
         CALL HPAK(4000+ii,PEDESTAL)
      END DO

      CLOSE(95)


      END  ! End of PEDESTAL

*****************************
      SUBROUTINE TERMIN
*****************************

      CHARACTER*80 FILENAME
```

```
      COMMON /PROBABILITY/ N_STRIPS, N_TIMES
      REAL                 N_STRIPS, N_TIMES

      COMMON  /SAMEMUON/ SAME
      INTEGER            SAME

      REAL PROB
C ---------------------------------------------------------------------
      READ(40,255,END = 277) FILENAME
 255  FORMAT(A80)
 277  WRITE(6,*) FILENAME
C ---------------------------------------------------------------------

      PROB=N_TIMES/N_STRIPS
      WRITE(*,*)'# OF TIMES THAT STRIPS MIGHT BE NOISY',N_STRIPS
      WRITE(*,*)'# OF TIMES STRIPS HAD A FINITE NOISE',N_TIMES
      WRITE(*,*)PROB,'IS THE FINITE NOISE PROBABILITY'

      WRITE(*,*)SAME,'MUONS RECONSTRUCTED IN BOTH PROJECTIONS'

      WRITE (*,*) '>>>>>>>>>>>>>END OF THE PROGRAM<<<<<<<<<<<<<'
      CALL HRPUT(0,FILENAME,'N') !writes all histograms to .rz

      END  ! End of TERMIN

************************************
      SUBROUTINE SPEC_GEOM
************************************

      COMMON /SPEC_GEOM_DATA/
     +     X_PL(56),
     +     ANA_PL_SHIFT(56),
     +     ADJ_ANA_PL_SHIFT(56),
     +     DIG_PL_SHIFT(56),
     +     ADJ_DIG_PL_SHIFT(56),
     +     STRIP_YZ(176),
     +     WIRE_YZ(352)

C ---------------------------------------------------------------------

      READ (91,*)               ! is used to skip an empty line in the file
      DO i = 1, 176
         READ (91,*) STRIP_YZ(i)
      END DO

      READ (91,*)
      DO i = 1, 56
         READ (91,*) ANA_PL_SHIFT(i)
      END DO

      DO i = 1, 56
         READ (56,10) IPL,PS
 10      FORMAT (I3,F8.4)
         ADJ_ANA_PL_SHIFT(ipl) = PS
         ADJ_DIG_PL_SHIFT(ipl) = 0.
      END DO

      READ (91,*)
      DO i = 1, 56
         READ (91,*) DIG_PL_SHIFT(i)
      END DO

      READ (91,*)
      DO i = 1, 352
```

```
         READ (91,*) WIRE_YZ(i)
      END DO

      READ (91,*)
      DO i = 1, 56
         READ (91,*) X_PL(i)
      END DO

      WRITE(6,*)'geometry    ........: loaded'

      END                        ! End of SPEC_GEOM

*************************************
      SUBROUTINE INITF(IRC,LUN)
*************************************

      COMMON /ACCNT/ MAX_EV
      INTEGER        MAX_EV

      CHARACTER*80 FILENAME

      INTEGER LUN
      INTEGER IRC
      INTEGER LEN

C -----------------------------------------------------------------------

      IRC    = 1

      WRITE(*,*)'START'
      READ(40,*) MAX_EV
      WRITE(6,*) 'MAXIMUM EVENTS=',MAX_EV

      READ(40,155,END = 177)FILENAME
 155  FORMAT(A80)
 177  WRITE(6,*)FILENAME
      LEN = INDEX(FILENAME,' ') - 1
      IRC = IOOPEN(LUN,FILENAME(1:LEN),0) ! returns zero if the file is there
      IF (IRC.NE.0) THEN
         WRITE(*,*)'COULD NOT OPEN RAW DATA FILE ',IRC
      END IF
      END                        ! End of INITF
```

References

[1] Gomez-Cadenas, J.J., and M.C. Gonzalez-Garcia, "Future of the ν_τ oscillation experiments and present data", CERN preprint, HEP-PH-9504246, TH-95-080, 1995.

[2] GALLEX Collaboration (Anselmann, P. et al.), "GALLEX Results from the first 30 solar neutrino runs". Phys. Lett. Vol., B327, pp 377-385, 1994.

[3] SAGE collaboration, (Abdurashitov, Dzh.N. et al.) "Results from SAGE (The Russian-American solar neutrino experiment)", Phys. Lett., Vol. B328, pp 234-248, 1994.

[4] Homestake Collaboration (Cleveland, B.T. et al.), "Update on the measurement of the solar neutrino flux with the homestake chlorine detector", Nucl. Phys. B (Proc. Suppl.), Vol. 38, pp 47-53, 1995.

[5] Suzuki, Y., "Kamiokande solar neutrino results", Nucl. Phys. B (Proc. Suppl.), Vol. 38, pp 54-59, 1995.

[6] Kamiokande Collaboration (Hirata, K.S. et al.), "Observation of a small atmospheric ν_μ/ν_e ratio in Kamiokande", Phys. Lett., Vol. B280, pp 146-152, 1992.

[7] Kamiokande Collaboration (Fukuda, Y. et al.), "Atmospheric ν_μ/ν_e ratio in the multi-GeV energy range" Phys. Lett., Vol. B335, pp 237-245, 1994.

[8] Becker-Szendy, R. et al., "The ν_e and ν_μ content of the atmospheric flux", Phys Rev., Vol. D46, pp 3720-3724, 1992.

[9] Battistoni, G. et al., "Detection of induced pulses in proportional wire devices with resistive cathodes", Nucl. Instr. Meth. Vol. 152, pp 423-430, 1978.

[10] De Winter, K. et al., "A detector for the study of neutrino-electron scattering", Nucl. Instr. and Meth. Vol. A278, pp 670-686, 1989.

[11] Friend, B., CERN, Geneva, Private communication.

[12] CHARM II Collaboration (Geiregat, D. et al.), "A new measurement of the cross section of the inverse muon decay reaction $\nu_\mu + e^- \rightarrow \mu^- + \nu_e$", Phys. Lett. Vol B247, pp 131-136, 1990.

[13] Grunberg, C. et al., "Multiwire proportional and semiproportional counter with a variable sensitive volume", Nucl. Instr. and Meth. Vol. 78, pp 102-108, 1970.

[14] Brehin, S. et al. "Some observations concerning the construction of proportional chambers with thick sense wires", Nucl. Instr. and Meth. Vol. 123, pp 225-229, 1975.

[15] Charpak, G. et al., "High accuracy measurements in proportional chambers and clarification of the avalanche mechanisms around wires", IEEE Trans. Nucl. Sci. Vol. NS-25, pp 122-125, 1978.

[16] Fischer, J. et al., "Spatial distribution of the avalanche in proportional counters", Nucl. Instr. and Meth. Vol. 151, pp 451-460, 1978.

[17] Alekseev, G.D. et al., "Self-quenching streamer discharge in a wire chamber", Lett. Nuovo Cimento, Vol 25, pp 157-160, 1979.

[18] Atac, M., and A.V. Tollestrup "Self-quenching streamers", IEEE Trans. Nucl. Sci. Vol. NS-29, pp 388-395, 1982.

[19] Atac, M., and A.V. Tollestrup "Self-quenching streamers", Nucl. Instr. and Meth. Vol 200, pp 345-354, 1982.

[20] Jonker, M. et al., Nucl. Instr. and Meth. Vol. 215, "Test results of the streamer tube system of the CHARM neutrino detector", pp 361-367, 1983.

[21] DeWulf, J.P. et al., "Test results of the streamer-tube system of the CHARM II neutrino detector", Nucl. Instr. and Meth. Vol. A252, pp 443-449, 1986.

[22] Mellisinos, A.C. Mellisinos, "Experiments in Modern Physics", Academic Press, 1966.

[23] Taylor, F.E., "A model of the limited streamer mechanism", Nucl. Instr. and Meth. Vol. A289, pp 283-293, 1990.

[24] Alekseev, G.D. et al., "Investigation of self-quenching streamer discharge in a wire chamber", Nucl. Instr. and Meth. Vol. 177, pp 385-397, 1980.

[25] Kamyshkov, Y. et al., "The self-quenching streamer discharge in Ar-CO_2 mixtures", Nucl. Instr. and Meth. Vol. A257, pp 125-131, 1987.

[26] Benvenuti, A. et al., "The limited streamer tubes system for the SLD warm iron calorimeter", SLAC preprint, SLAC PUB 5713, 1992.

[27] Iarocci, E., "Plastic streamer tubes and their applications in high energy physics", Nucl. Instr. and Meth. Vol. 217, pp 30-42, 1983.

[28] Battistoni, G. et al., "Resistive cathode detectors with bidimensional strip readout:tubes and drift chambers", Nucl. Instr. and Meth. Vol. 176, pp 297-303, 1980.

[29] CHARM II Collaboration (DeWulf, J.P. et al.), "Test results and conditioning procedure of a limited streamer-tube calorimeter" Nucl. Instr. and Meth. Vol. A263, pp 109-113, 1988.

[30] Leo, W.R., "Techniques for Nuclear and Particle Physics Experiments", Springer-Verlag Press, 1987.

[31] Charpak, G. et al., "Progress in high-accuracy proportional chambers", Nucl. Instr. and Meth. Vol. 148, pp 471-482, 1978.

[32] Charpak, G. et al., "High-accuracy localization of minimum ionizing particles using the cathode-induced charge cetre-of-gravity read-out", Nucl. Instr. and Meth. Vol. 167, pp 455-464, 1979.

[33] Lee, D.M., and S.E. Sobottka, "A bifilar helical multiwire proportional chamber for position sensitive detection of minimum ionizing particles", Nucl. Instr. and Meth. Vol. 104, pp 179-188, 1972.

[34] Kwong,L, and J. Pyrlik, "Optimization of centroid-finding algorithms for cathode strip chambers", preprint, UHIBPD-HEP-94-002.

[35] Lau, K., and B. Mayes, and J. Pyrlik, "Test results of a high precision cathode strip chamber based on plastic streamer tubes", Nucl. Instr. and Meth. Vol. A354, pp 376-388, 1995.