

KfK 4016
August 1986

Monte-Carlo-Simulation von 4π -Detektorsystemen

M. Kunze
Institut für Kernphysik

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARLSRUHE

Institut für Kernphysik

KfK 4016

MONTE-CARLO-SIMULATION VON
4 π - DETEKTORSYSTEMEN *)

von
Marcel Kunze

*) Von der Fakultät Physik der Universität Karlsruhe
genehmigte Diplomarbeit

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Als Manuskript vervielfältigt
Für diesen Bericht behalten wir uns alle Rechte vor

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

ZUSAMMENFASSUNG

Für Experimente am LEAR (Low Energy Antiproton Ring) beim CERN wird ein Detektorsystem vorgeschlagen, das gleichzeitig geladene und neutrale Annihilationsprodukte mit guter Energieauflösung und nahezu 4π -Geometrie nachweisen kann. Hauptziele der Messungen sind das Verständnis der Proton-Antiproton Annihilation auf dem Quark-Gluon-Niveau und neue Beiträge zur Mesonenspektroskopie sowie eine hochempfindliche Suche nach exotischen Zuständen (Glue-Balls, Hybrids, Baryonium, etc.). Die neutralen Komponenten sollen in einem kugelförmigen modularen CsI-Detektor (ca. 1500 Module), die geladenen Teilchen in einer Vertexkammer hoher Ortsauflösung in einem Magnetfeld gemessen werden.

MONTE-CARLO-SIMULATION OF 4π - DETECTOR SYSTEMS

For experiments at LEAR (Low Energy Antiproton Ring) at CERN a detection system is proposed, which allows to measure charged and neutral annihilation products with nearly full solid angle. The main physics items are the understanding of proton-antiproton annihilation on the quark-gluon level and new contributions to meson spectroscopy; furthermore, sensitive search for exotic states (glueballs, hybrids, baryonia, etc.) is possible. The neutral components are measured in a spherical modular CsI-detector (1500 modules), the charged particles are measured in a vertex chamber with high spatial resolution in a magnetic field.

INHALTSVERZEICHNIS

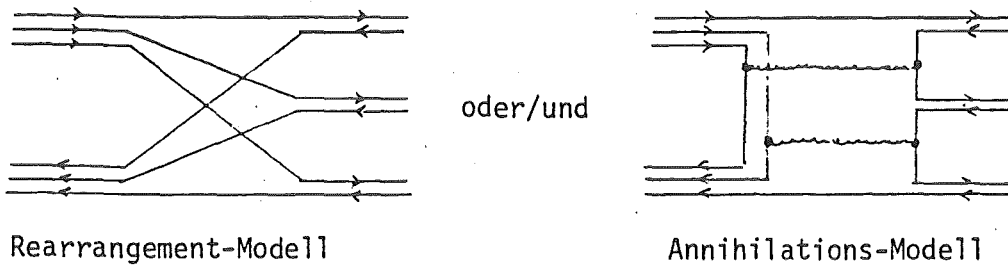
KAPITEL 1. DIE ERZEUGUNG DER ANNIHILATIONSKINEMATIK	1
1.1. Die Proton - Antiproton Annihilation	1
1.2. Phasenraumbetrachtungen	4
1.2.1. Der Phasenraum	4
1.2.2. Das Phasenraumgewicht	5
1.2.3. Zerfallsspektren	6
1.2.3. Berechnung des Phasenraumintegrals	7
1.2.4. Fehlerbetrachtung	11
1.3. Monte - Carlo Computersimulation	12
1.4. Energiespektren des Summenkanals	13
1.4.1. Energiespektrum der geladenen Pionen	13
1.4.2. Energiespektrum der neutralen Pionen	14
1.4.3. Energiespektrum der Gammas	15
KAPITEL 2. DIE SIMULATION	16
2.1. Das 'STANDARD-DETEKTOR-SYSTEM'	16
2.1.1. Parametrisierung	16
2.1.2. Programmtechnische Realisierung	20
2.2. Rekonstruktion von Invarianten Massen	24
2.3. Inklusive Energiespektren	27
2.3.1. Reduktion des Untergrundes	29
2.4. Exklusive Energiespektren	31
2.4.1. Nachweis resonanter Zerfälle	31
2.4.1.1. Radiativ zerfallende Resonanzen	32
2.4.1.2. In geladene Teilchen zerfallende Resonanzen	34
2.4.2. Nachweis exotischer Zerfälle	36
2.4.2.1. Baryonium	36
2.4.2.2. Glue - Balls	38
2.5. Variation der Parameter	42
2.5.1. Gammadetektor	42
2.5.1.1. Variation der Modularität	42
2.5.1.2. Variation der Energieauflösung	45
2.5.1.3. Variation des Raumwinkels	46
2.5.2. Driftkammer	48
2.5.2.1. Variation der Impulsauflösung	48
2.6. Alternative Lösungen	49
2.6.1. Detektorsystem ohne Magnetfeld	49
2.7. Schlusswort und Ausblick	52
ANHANG A. PROGRAMMPAKET BIGBANG	54
Programmablauf	54
Eingabedaten	55
Namelist 'INPUT'	55
Einfügen und Ändern von Annihilationskanälen	57
Eventdump	58
Ablaufdiagramm	59
Quellprogramm	60
ANHANG B. PROGRAMMPAKET DETEKTOR	91
Programmablauf	91
Eingabedaten	92
Namelist 'INPUT'	92
Eventdump	95
Ablaufdiagramm	96
Quellprogramm	97

ABBILDUNGSVERZEICHNIS	159
LITERATURVERZEICHNIS	160

EINLEITUNG

Die Untersuchung der Annihilation des Nukleon-Antinukleon-Systems ist für die Überprüfung von Aussagen der Quarktheorie (Das Nukleon besteht aus 3 Quarks, das Antinukleon aus 3 Antiquarks) von großer Bedeutung.

So ist zum Beispiel bislang ungeklärt, ob die Nukleon - Antinukleon Annihilation in drei Mesonen als Umordnung der beteiligten Quarks oder als Vernichtung und Wiedererzeugung von Quark-Antiquark-Paaren erklärt werden kann:



Rearrangement-Modell

Annihilations-Modell

Abbildung 1. Bild der Annihilation im Quarkmodell

Damit man Aussagen verschiedener Quarkmodelle miteinander vergleichen kann, ist es nötig, die Verzweungsverhältnisse bzw. die Wirkungsquerschnitte für möglichst viele Annihilationskanäle experimentell zu bestimmen. Hier sind quantitative Vergleiche bisher nicht möglich, da nur ein Bruchteil der auftretenden Annihilationskanäle untersucht werden konnte.

Um zu einem tieferen Verständnis der Nukleon - Antinukleon Annihilation zu gelangen, ist vor allen Dingen der Nachweis der Annihilationskanäle interessant, in denen direkt erzeugte mesonische Zwischenzustände auftreten ($\rho, \omega, f, \eta, \dots$).

Daraus ergibt sich die Forderung, ein Detektorsystem zu entwickeln, das den Nachweis neutraler wie geladener Teilchen in einem großen Raumwinkelbereich mit guter Energie- und guter Ortsauflösung erlaubt.

Die vorliegende Arbeit beschäftigt sich mit dem Entwurf und der Optimierung eines solchen Detektorsystems zur Durchführung eines kinematisch vollständigen Experimentes.

Durch Monte - Carlo Computersimulation sollten vor allen Dingen folgende Fragen geklärt werden:

1. Untersuchung des zu erwartenden Untergrunds in inklusiven und exklusiven Spektren
2. Entwicklung von Methoden zur Herabsetzung des Untergrunds
3. Untersuchung des Auftretens künstlicher, durch geometrische Effekte erzeugter Strukturen im Annihilationsspektrum
4. Untersuchung der zu erwartenden Sensitivität beim Nachweis komplizierter Zerfallsprozesse

KAPITEL 1. DIE ERZEUGUNG DER ANNIHILATIONSKINEMATIK

Die Erzeugung der Annihilationskinematik mit dem Computer wird ermöglicht durch das Programmpaket 'BIGBANG' (siehe Anhang A).

1.1.DIE PROTON - ANTIPROTON ANNIHILATION

In diesem Kapitel soll ein Abriss über die bisher bestehenden Vorstellungen von der Proton - Antiproton Annihilation gegeben werden.

Die Annihilation des Proton - Antiproton Systems wurde experimentell bisher vornehmlich in Blaskammern untersucht (ca. 100 000 Annihilationen) /BUR70/.

Ein mit niedriger Geschwindigkeit in einen Behälter mit Flüssigwasserstoff einlaufendes Antiproton wird von einem Wasserstoffkern eingefangen, wobei es den Platz des den Wasserstoffkern umkreisenden Elektrons einnimmt. Das System regt sich durch Aussenden von Roentgenstrahlen und durch Augereffekt ab, wobei das Antiproton auf niedrigere atomare Bahnen herabfällt. Von allen Zuständen ist die Annihilation möglich, in flüssigem Wasserstoff jedoch bevorzugt aus den S-Niveaux.

Einen Überblick über die bisher bekannten Annihilationskanäle gibt Abbildung 2. Wie daraus hervorgeht, ist eine Aufschlüsselung über resonante Zwischenzustände nur für den Fall möglich, in dem im Endzustand ein einzelnes oder gar kein neutrales Pion auftritt. In der Mehrzahl der Fälle (ca. 60%) sind die Zwischenzustände noch unbekannt.

Weiterhin sind in dieser Tabelle auch solche Annihilationskanäle aufgeführt, deren Evidenz experimentell nur schwach gesichert ist /BKS78/. Es sind dies tiefgebundene Zustände unterhalb der Schwelle des Proton - Antiproton Systems, deren Bildung in Ruhe und unter Aussendung eines monoenergetischen Teilchens geschieht (ID=54-64).

Es folgen rein radiative Kanäle, in denen kein geladenes Teilchen im Endzustand auftritt (ID=65,66,67).

Den Abschluß der Liste bilden rein hypothetische Annihilationskanäle, deren Existenz von der Theorie vorausgesagt wird, für die aber bislang noch keine experimentelle Evidenz besteht. Es sind dies die Annihilation über einen glyonischen Zwischenzustand (ID=68,69) und die Annihilation über das E^0 , hinter dem ein Glueball vermutet wird.

PROTON-ANTI-PROTON-ANNIHILATION

ID	ENDZUSTAND	ZWISCHENZUSTAND	VERZWEIGUNGSVERHÄLTNIS
01	$\pi^+ \pi^- 2\pi^0$		9.3%
02	$\pi^+ \pi^- 3\pi^0$		23.3%
03	$\pi^+ \pi^- 4\pi^0$		2.8%
04	$2\pi^+ 2\pi^- 2\pi^0$		16.6%
05	$2\pi^+ 2\pi^- 3\pi^0$		4.2%
06	$2\pi^0$		~0.04%
07	$n\pi^0, n>2$		3.2%
08	$\pi^+ \pi^- \pi^0$		0.86%
09		$\rho^0 \pi^0$	5.8%
10		$f^0 \pi^0$	0.24%
11	$2\pi^+ 2\pi^-$		2.38%
12		$A_2^{+/-} \pi^- / +$	2.0%
13		$\rho^0 f^0$	0.9%
14		$\rho^0 \pi^+ \pi^-$	1.5%
15		$\rho^0 \rho^0$	0.12%
16	$2\pi^+ 2\pi^- \pi^0$	$\omega \pi^+ \pi^-$	3.0%
17		$\omega \rho^0$	2.1%
18		ωf^0	1.7%
19		$\rho^0 \pi^+ \pi^- \pi^0$	3.7%
20		$B^{+/-} \pi^- / +$	0.7%
21		$\eta \pi^+ \pi^-$	0.35%
22		$A_2^{+/-} \pi^- / +$	0.13%
23	$3\pi^+ 3\pi^-$		2.1%
24	$3\pi^+ 3\pi^- \pi^0$		0.39%
25		$\omega 2\pi^+ 2\pi^-$	1.3%
26		$\eta 2\pi^+ 2\pi^-$	0.17%
27		$\eta' 2\pi^+ 2\pi^-$	0.04%
28	$\pi^+ \pi^- \pi^0$	$\rho^{+/-} \pi^- / +$	2.7%
29	$2\pi^+ 2\pi^- \pi^0$	$\rho^{+/-} 2\pi^- / + \pi^{+/-}$	6.4%
30	$K^+ K^-$		0.096%
31	$K_1 K_1 + K_2 K_2$		0.0008%
32	$K_1 K_2$		0.071%
33	$K_1 K_1 \pi^0 + K_2 K_2 \pi^0$		0.146%
34	$K_1 K_2 \pi^0$	$K^0 K^{0*}$	0.035%
35	$K_1 K_2 n\pi^0, n>1$		0.064%
36		$K_1 K_1 \eta$	0.0155%
37		$K_1 K_1 \omega$	0.0089%

38	$K^0 K^{+/-} \pi^{-/+}$		0.425%
39		$K^0 K^{*0}$	0.085%
40		$K^{+/-} K^{*-/+}$	0.057%
41		$A_2^{+/-} \pi^{-/+}$	0.064%
42		$A^{+/-} \pi^{-/+}$	0.025%
43	$K_1 K_1 \pi^+ \pi^-$		0.201%
44	$K_1 K_2 \pi^+ \pi^-$		0.241%
45		$\phi \pi^+ \pi^-$	0.018%
46	$K_1 K^{+/-} \pi^{-/+} \pi^0$		0.447%
47	$K_1 K_1 \pi^+ \pi^- \pi^0$		0.149%
48		$K_1 K_1 \omega$	0.0836%
49		$K_1 K_1 \eta$	0.0059%
50		$E^0 \pi^+ \pi^-$	0.014%
51	$K_1 K^{+/-} 2\pi^{-/+} \pi^{+/-}$		0.059%
52		$E^0 \pi^+ \pi^-$	0.071%
53	$K_1 K^+ \pi^- 3\pi$		~0.0%
54	$\gamma X(1694)$		0.16%
55	$\gamma X(1638)$		0.3%
56	$\gamma X(1421)$		0.06%
57	$\gamma X(1383)$		0.1%
58	$\gamma X(1210)$		0.09%
59	$\pi^{+/-} X(1638)^{-/+}$		0.5%
60	$\pi^{+/-} X(1561)^{-/+}$		0.5%
61	$\pi^{+/-} X(1447)^{-/+}$		0.5%
62	$\pi^0 X(1638)$		0.5%
63	$\pi^0 X(1561)$		0.5%
64	$\pi^0 X(1447)$		0.5%
65	$2\pi^0 \gamma$	$\pi^0 \omega$	2.38%
66	$\pi^0 2\gamma$	$\pi^0 \eta$	0.82%
67	$\pi^0 \gamma$		0.015%
68	$2\pi^0 \gamma$	Gluonium γ	0.1%
69	$2\pi^0 \pi^-$	Gluonium π^-	0.1%
70	$3\pi^0 2\gamma$	$E^0 \pi^0$	0.1%
71	$4\pi^0 2\gamma$	$E^0 2\pi^0$	0.1%
72	$\pi^+ \pi^- 2\pi^0 2\gamma$	$E^0 \pi^+ \pi^-$	0.2%

Abbildung 2. Anihilationskanäle des Proton - Antiprotonsystems

1.2. PHASENRAUMBETRACHTUNGEN

Bevor die Durchführung der Monte - Carlo Computersimulation der Proton - Antiproton Annihilation beschrieben wird, soll zunächst der Übergang eines Vielteilchen-Systems von einem Zustand in einen anderen betrachtet werden.

1.2.1. Der Phasenraum

Die Wahrscheinlichkeit W für den Übergang eines n -Teilchensystems vom Anfangszustand i in den Endzustand f ist in lorentzinvarianter Form /PER72/ gegeben durch

$$(1) \quad W = \frac{2\pi}{\hbar} \frac{|M.E.|^2}{\prod_i 2E_i} R_n$$

Hier ist $M.E.$ ein Matrixelement, welches die Dynamik des Überganges beschreibt. Der Faktor R_n beschreibt die Zahl der möglichen Zustände der n Teilchen der Anfangsenergie E_i im Impulsraum der Endzustände pro Einheitsintervall der Gesamtenergie (Phasenraumfaktor).

Dieser Phasenraumfaktor ist für ein Teilchen gegeben durch $d^3\vec{p}=p^2 dp d\vec{\Omega}$ und gibt die Wahrscheinlichkeit an, das Teilchen im Raumwinkelbereich Ω mit einem Impuls p vorzufinden.

Unter Vernachlässigung eines Normierungsvolumens ist dieser Phasenraumfaktor für ein Ensemble aus n Teilchen¹

$$R_n = \frac{dN}{dE} = \frac{\int d^3\vec{p} \dots d^3\vec{p} \delta^4(P - \sum p_f)}{\prod_f 2E_f}$$

Hier ist E_f die Energie eines Teilchens im Endzustand, P der Gesamtvierervektor des Anfangszustandes, p_f sind die Vierervektoren der Einzelteilchen im Endzustand. Die δ -Funktion gewährleistet die Viererimpulserhaltung.

¹ Die verwendete Metrik ist $P=(\vec{p}, iE)$ mit $c=1$.

Mit

$$\begin{aligned} \int \frac{d^3 \vec{p}}{2E} &= \int d^3 \vec{p} \int \frac{1}{2E} \delta(\vec{p}^2 + m^2 - E^2) d(E^2) \\ &= \int d^3 \vec{p} \int \delta(\vec{p}^2 + m^2 - E^2) dE \\ &= \int d^4 p \delta(p^2 + m^2) \end{aligned}$$

ergibt sich die folgende Formulierung des lorentzinvarianten Phasenraumintegrals:

$$(2) R_n(P, m_i) = \frac{dN}{dE} = \int \int \delta^4(P - \sum_{j=1}^n p_j) \prod_{i=1}^n \delta(p_i^2 + m_i^2) d^4 p_i$$

wobei P der Gesamtvierervektor des n - Teilchensystems, p_i der Vierervektor eines Einzelteilchens und m_i dessen Masse ist.

Die physikalischen Erhaltungssätze der Energie und des Impulses finden in dieser Formulierung des Phasenraumes ihre Entsprechung in δ -Funktionen, die physikalisch nicht zugängliche Gebiete des Phasenraumes ausschließen.²

1.2.2. Das Phasenraumgewicht

Im Phasenraum sind alle physikalisch möglichen Zustände des Systems enthalten. Die Zustandsdichte des Phasenraums ist aber i.a. nicht isotrop. Dies bedeutet für das Beispiel des n-Körperzerfalls, daß kinematisch verschiedene Zerfälle eines Systems, die zur selben Teilchenzusammensetzung des Endzustandes führen, mit unterschiedlichen Wahrscheinlichkeiten auftreten:

Jeder Zerfall ist mit einem Phasenraumgewicht versehen, das der Wahrscheinlichkeit für das Auftreten des jeweiligen Prozesses entspricht.

² Weitere Erhaltungssätze, wie z.B die Erhaltung der Leptonenzahl, Erhaltung der Baryonenzahl, Erhaltung des Isospins usw. werden durch die Beschränkung auf die Simulation experimentell beobachtbarer Annihilationsreaktionen des Proton - Antiproton Systems implizit erfüllt.

1.2.3. Zerfallsspektren

Durch den Phasenraum wird nur die Kinematik des Übergangs des Systems von einem Zustand in einen anderen beschrieben.

Eine vollständige Beschreibung der Spektren einer Größe α (Energie, Impuls, Winkel, etc.) des zerfallenden Systems ist gegeben durch

$$(3) \quad f(\alpha) = \frac{d}{d\alpha} (|M.E.|^2 R_n)$$

wobei M.E. ein Matrixelement ist, das die Dynamik des Zerfallsprozesses beinhaltet. Besteht keine Wechselwirkung zwischen den ausgehenden Teilchen, so ist M.E.=1, und alle Spektren sind nur durch den Phasenraum bestimmt.

Ein Beispiel für ein nur aus dem Phasenraum abgeleitetes Spektrum ist in Abbildung 3 gegeben für den Fall der Annihilation des Proton - Antiproton Systems in drei Pionen (ID=8,9,10). Es handelt sich hierbei um einen sog. 'Dalitz-Plot', bei dem die kinetische Energie des π^0 gegen die Differenz der kinetischen Energien der beiden geladenen Pionen aufgetragen ist. Die breite ρ -Resonanz (ID=9,10) führt zu den dicken Balken über einem Untergrund der isotropen Annihilation (ID=8).

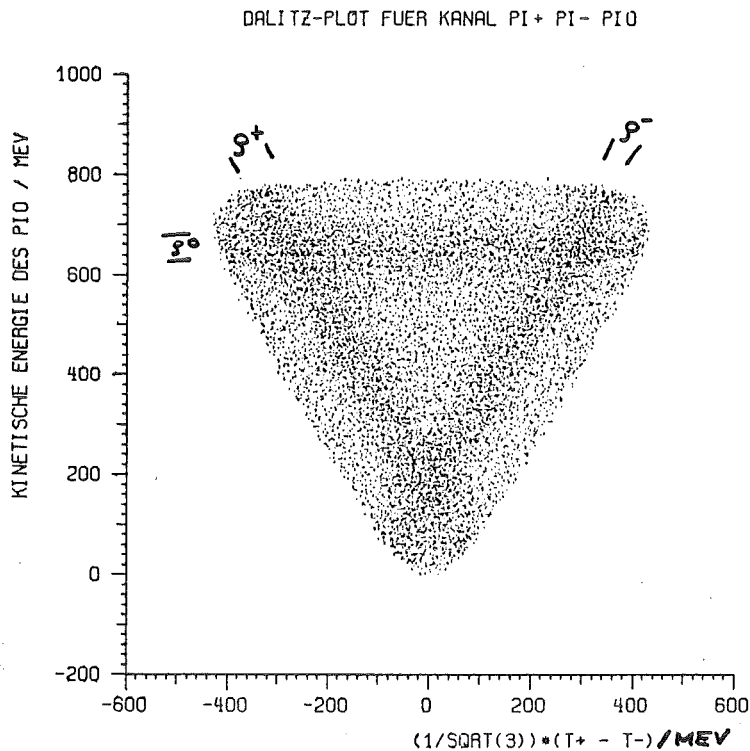


Abbildung 3. Dalitz-Plot eines Systems aus drei Pionen

1.2.3. Berechnung des Phasenraumintegrals

Zur Erzeugung der Zerfallsspektren des n-Körperzerfalls ist es erforderlich, den lorentzinvarianten Phasenraum (2) zu berechnen. Die analytische Integration des Phasenraumintegrals ist aber äußerst kompliziert und kann im besten Fall auf die Behandlung von Dreikörperproblemen angewendet werden. Für die Berechnung von n-Körperzerfällen (n>3) reicht nur eine Entwicklung des Integrals mit Monte-Carlo-Methoden in den Bereich des Möglichen.³

Im folgenden soll eine Monte-Carlo-Methode zur Berechnung des Phasenraumintegrals beschrieben werden, die mit invarianten Massen als kinematischen Parametern arbeitet /JAM68/.

Aus der Lorentz-Invarianz der Formulierung des Phasenraumintegrals (2) läßt sich eine Beschreibung des n-Körperzerfalls als Abfolge einzelner Zweikörperzerfälle herleiten:

$$(4) R_n = \frac{1}{M_1} \iint \prod_{i=1}^{n-1} \{ 2M_i R_2(M_{i+1}; M_i, m_{i+1}) \} dM_{n-1} \dots dM_2$$

$$R_2(M_{i+1}; M_i, m_{i+1}) = \frac{2\pi}{M_{i+1}} \sqrt{M_{i+1}^2 + \left(\frac{M_i^2 - m_{i+1}^2}{M_{i+1}}\right)^2 - 2(M_i^2 + m_{i+1}^2)}$$

wobei R_2 dem invarianten Zwei-Körper-Phasenraum beim Zerfall der Masse M_{i+1} in die beiden Tochtermassen M_i und m_{i+1} entspricht.

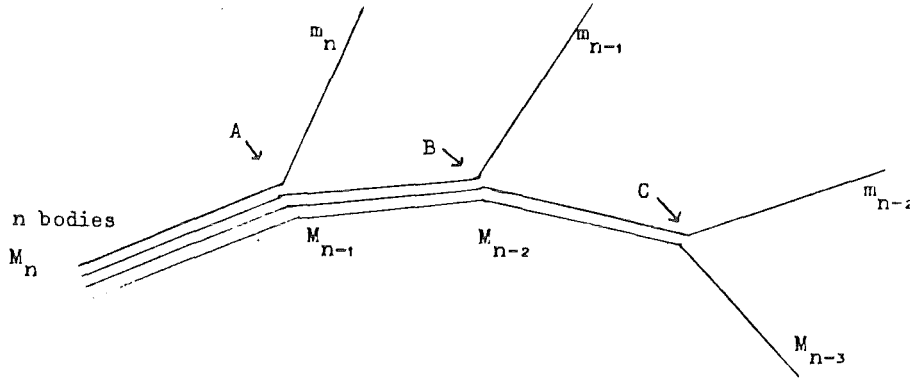


Abbildung 2. Annihilation als Abfolge von Zweikörperzerfällen

Zunächst wird der Zweikörperphasenraum nur im ersten Vertex (A) berechnet, danach wird über alle möglichen Zwischenmassen M_{n-1} integriert.

Da die 'Anfangsenergie' in B nun ebenfalls variabel ist, usw., wählt man dasselbe Vorgehen für den Punkt B, indem man über alle möglichen Zwischenmassen M_{n-2} und alle möglichen Zwischenmassen M_{n-1} integriert.

³ Für die Entwicklung eines 16-fach Integrals (n=4) nach einer anderen numerischen Methode (z.B. Simpson-Integration) mit jeweils 10 Stützstellen ergäbe sich eine Summe über 10^{16} Stützstellen insgesamt.

Eine Randbedingung hierbei ist, daß alle Teilreaktionen exotherm ablaufen müssen: An jedem Vertex muß mindestens soviel Energie bereitgestellt werden, wie zur Bildung der nachfolgenden invarianten Zwischenmasse und der Tochtermasse benötigt wird. Daher sind die Integrationsgrenzen für die Bildung der invarianten Zwischenmassen festgelegt durch die Bedingung

$$(5) \quad M_{j-1} + m_j < M_j < M_{j+1} - m_{j+1}$$

Unglücklicherweise stehen die so gewählten Integrationsgrenzen über den Integranden in direkter Beziehung zueinander. Eine solche Abhängigkeit muß aber bei Anwendung einer Monte-Carlo-Methode aus Gründen einer konstanten Integrationsdichte vermieden werden.

Die invarianten Zwischenmassen M müssen unabhängig voneinander gebildet werden. Dies gewährleistet die schwächere Bedingung

$$(6) \quad \sum_{i=1}^j m_i < M_j < M_n - \sum_{i=j+1}^n m_i$$

Hier ist M_n die Gesamtenergie im Ruhesystem der Muttermasse, so daß nur Konstanten in die Wahl der Zwischenmasse M_j eingehen. Diese ist dann mindestens so groß, daß noch alle folgenden Tochtermassen m_i gebildet werden können, aber kleiner als die in diesem Vertex zur Verfügung stehende gesamte freie Energie. Solange allerdings nicht auch die stärkere Bedingung (5) erfüllt ist, genügt der Zerfall nicht notwendig den Erhaltungssätzen der Physik.

Einen Weg, beide Bedingungen zu erfüllen, bietet

$$(7) \quad M_j = r_j \left(M_n - \sum_{i=1}^n m_i \right) + \sum_{i=1}^j m_i$$

wobei r_j eine Zufallszahl zwischen 0 und 1 ist, mit der die gesamte kinetische Energie auf die invarianten Zwischenmassen verwürfelt wird.

Die notwendige und hinreichende Bedingung, (5) und (6) zu erfüllen, ist, die Zufallszahlen r_j in aufsteigender Reihenfolge zu sortieren:

$$0 < r_1 < \dots < r_j < \dots < r_{n-1} < 1$$

Auf diese Weise ist es möglich, einen Zerfallsgenerator für n -Körperzerfälle zu konstruieren, bei dem jeder Zerfall unabhängig vom anderen generiert wird, und jeder Zerfall in ein Gebiet des Phasenraums fällt, das physikalisch zugänglich ist.

Da an jedem Zweikörper-Vertex nur eine Energievariation durchgeführt wird, stehen zur vollständigen Beschreibung des physikalischen Vorganges an jedem Vertex zwei weitere frei wählbare kinematische Parameter zur Verfügung. Es sind dies zwei Winkel, die im Ruhesystem jedes Vertex isotrop gewählt werden können, da jeder Zweikörperphasenraum isotrop ist.

Um allerdings zu einer Beschreibung des Gesamtsystems zu gelangen, müssen nacheinander alle Zweikörperzerfälle in das Ruhesystem der Teilchengruppe lorentztransformiert werden, die dem jeweiligen Zerfall vorangeht.

Eine Umsetzung des obigen Zerfallsalgorithmus in die höhere Programmiersprache FORTRAN steht als Unterprogramm GENBOD (W505) in der CERN Programmbibliothek zur Verfügung.

Bei der Verwendung dieses Unterprogramms zur Simulation der Proton - Antiproton Annihilation besteht ein großes Problem darin, daß die über die Entwicklung des Phasenraumintegrals erhaltenen Zerfälle bei der Simulation eines bestimmten Zerfallskanals bezüglich der Verzweigungsverhältnisse der verschiedenen anderen Annihilationsreaktionen unnormiert sind:

Denn jeder Zerfall ist durch ein Phasenraumgewicht gewichtet, das der Wahrscheinlichkeit für das Auffinden eines bestimmten Ensembles von invarianten Zwischenmassen entspricht. Dieses Phasenraumgewicht erhält man durch Differenzieren des Phasenraumes (4) nach allen auftretenden Zwischenmassen

$$(8) \quad w(M_i, m_i) = \frac{d^{n-2}}{dM_{n-1} \dots dM_2} \{ R_n \}$$

$$\sim \frac{1}{M_1} \prod_{i=1}^{n-1} \{ M_i R_2(M_{i+1}; M_i, m_{i+1}) \}$$

Nur für einen Zweikörperzerfall besitzt dieses Gewicht immer den Wert 1, bei einem Zerfall in mehr als zwei Teilchen gehorchen die Phasenraumgewichte einer endlichen statistischen Verteilung.

Da in diesem Falle dann ein Ereignis mit Maximalgewicht existiert, können unter Kenntnis dieses Maximalgewichtes durch die folgende Vorgehensweise ungewichtete Zerfälle aus gewichteten Zerfällen erzeugt werden:

Für jeden Zerfall wird eine gleichverteilte Zufallszahl zwischen Null und dem Maximalgewicht gewürfelt. Ist das Gewicht des Zerfalles kleiner als diese Zufallszahl, so wird das Ereignis verworfen, anderenfalls wird es akzeptiert. Dies bedeutet, daß ein Ereignis mit der Wahrscheinlichkeit seines Phasenraumgewichtes relativ zum Maximalgewicht übernommen wird (Hit-or-Miss-Monte-Carlo).⁴

Auf diese Weise wird die Erzeugung von Ereignissen möglich, die alle dasselbe Phasenraumgewicht (exakt 1) besitzen und damit im Bezug auf die verschiedenen Annihilationskanäle normiert sind.

⁴ Es hat sich gezeigt, daß im Falle der Simulation der Proton - Antiproton Annihilation ca. 50% aller erzeugten Zufallereignisse verworfen werden.

Eine Schwierigkeit bei der Anwendung dieses Verfahrens besteht darin, daß zu Beginn der Simulation der Annihilationskinematik des Proton - Antiproton Systems die Maximalgewichte für die einzelnen Annihilationskanäle nicht bekannt sind. Aus diesem Grunde wurde das folgende approximative Verfahren entwickelt:

Zu Beginn werden die Maximalgewichte aller zu simulierenden Annihilationskanäle auf Null gesetzt. Daraufhin werden im Verlauf der Simulation diese Maximalgewichte dem jeweils größten auftretenden Phasenraumgewicht angepaßt, wodurch man sich Schritt für Schritt an das globale Maximalgewicht jeden Annihilationskanales annähert.

Die Verwirklichung dieses Verfahrens bringt zu Beginn der Simulation Fehler in der Normierung mit sich, es hat sich aber gezeigt, daß diese Methode der 'sukzessiven Annäherung' relativ schnell konvergiert: Nach ca. 100 erzeugten Zufallsereignissen beträgt die Abweichung vom globalen Maximalgewicht pro Annihilationskanal nur ungefähr 1% .

1.2.4. Fehlerbetrachtung

Im folgenden soll untersucht werden, wie groß der zu erwartende statistische Fehler bei Anwendung der Monte-Carlo-Methode zur Berechnung des Phasenraumintegrals ist.

Es soll zunächst ein kleiner Bereich des Phasenraumes betrachtet werden:

In einer Folge von N Ereignissen besitzt das i -te Ereignis das Phasenraumgewicht w_i . Dieses Phasenraumgewicht wird Null, wenn das Ereignis nicht in das gewünschte Gebiet des Phasenraumes fällt.

Repräsentiert dieses Phasenraumgebiet einen Kanal im Histogramm eines Zerfallsspektrums, so ist der Mittelwert für die Einträge in diesem Kanal

$$(9) \quad W = \frac{1}{N} \sum_{i=1}^N w_i$$

Man erhält eine Abschätzung für den Fehler (σ ist die Standardabweichung) durch

$$(10) \quad \Delta W = \sqrt{\frac{\sigma^2}{N}} = \sqrt{\frac{\sum (w_i - W)^2}{N(N-1)}} \\ = \sqrt{\frac{\sum (w_i)^2 - NW^2}{N(N-1)}} \\ = \sqrt{\frac{1}{N(N-1)} \left[\sum (w_i)^2 - \frac{1}{N} (\sum w_i)^2 \right]}$$

Für $N \gg 1$ verschwindet der Term $1/N(\sum w_i)^2$ und der Vorfaktor $\sqrt{1/N(N-1)}$ wird ungefähr $1/N$ (Dieser Vorfaktor stellt lediglich einen Skalierungsfaktor dar). Somit erhält man aus (9) und (10) für den Inhalt eines Kanals im Histogramm eines Spektrums

$$(11) \quad W \sim \sum_{i=1}^N w_i \pm \sqrt{\sum w_i^2}$$

Der Fehler erreicht einen Minimalwert $\Delta W = \sqrt{N}$ für den Fall, in dem alle Phasenraumgewichte gleich sind und entspricht dann dem Fehler einer Poissonverteilten Größe.

Dies ist bei Anwendung der Hit-or-miss Monte-Carlo Methode der Fall, da Ereignisse erzeugt werden, die alle dasselbe Phasenraumgewicht besitzen.

1.3.MONTE - CARLO COMPUTERSIMULATION

Zur Erzeugung der Annihilationskinematiken mit dem Computer wurde das Unterprogramm GENBOD (CERN Programmbibliothek W505) von F.James verwendet, das den Zerfall von Teilchen in ihrem Ruhesystem in andere Teilchen berechnet. Bei Vorgabe der Ruheenergie des zerfallenden Teilchens und der Massen der entstehenden Teilchen würfelt ein Aufruf von GENBOD

- o Die drei Impulskomponenten
- o Die Gesamtenergie
- o Den Gesamtimpuls

für jedes entstehende Teilchen sowie für das gesamte Ereignis ein Phasenraumgewicht, das der Wahrscheinlichkeit für das Zustandekommen des jeweiligen Ereignisses entspricht.

Mit dem Programmpaket BIGBANG (siehe Anhang A) ist es möglich, die einzelnen Annihilationskanäle unter Zuhilfenahme der Hit-or-Miss Monte-Carlo Methode in quasi zufälliger Reihenfolge zu erzeugen.

Die neutralen Pionen besitzen eine Lebensdauer von ca. 10^{-16} Sekunden. Da dies so kurz ist, daß die neutralen Pionen praktisch wieder am Ort ihres Entstehens in zwei Gammas zerfallen, wurde deren Kinematik durch die Kinematik ihrer Zerfallsgammas ersetzt.

Bei den Resonanzen wurde so verfahren, daß zunächst die Kinematik der Proton - Antiproton Annihilation in die Resonanz und die restlichen Teilchen berechnet wurde, im Anschluss daran der Zerfall der Resonanz in deren Ruhesystem. Nach Lorentztransformation der Kinematik der aus der Resonanz hervorgehenden Teilchen in das Proton - Antiproton Ruhesystem standen alle Teilchen des Endzustandes zur Verfügung. Das gesamte Phasenraumgewicht dieses Ereignisses ergab sich als Produkt aus dem Gewicht der Proton - Antiproton-Annihilation und dem Gewicht des resonanten Zerfalls.

Auf diese Art und Weise wurden jeweils 800.000 Annihilationen erzeugt und auf je eine Magnetbanddatei geschrieben.

Diese Magnetbanddatei war in Sätze zu 25 Bytes Länge eingeteilt. Jedem Ereignis wurde ein Kennsatz vorangestellt, der die Ereignisidentifikation (ID), die Anzahl der Teilchen, die Gesamtenergie der Annihilation in ihrem Schwerpunktsystem und die Nummer des laufenden Ereignisses enthielt. Für jedes Teilchen des Ereignisses gab es einen 'Teilchen-Kinematik-Satz', der die drei Impulskomponenten, die Gesamtenergie, den Gesamtimpuls und den Teilchennamen enthielt.

1.4. ENERGIESPEKTREN DES SUMMENKANALS

In die Energiespektren des Summenkanals sind die Energien sämtlicher im Endzustand der Annihilation auftretenden Teilchen einsortiert, wobei die Statistik jeweils 800.000 Ereignissen entspricht.

1.4.1. Energiespektrum der geladenen Pionen

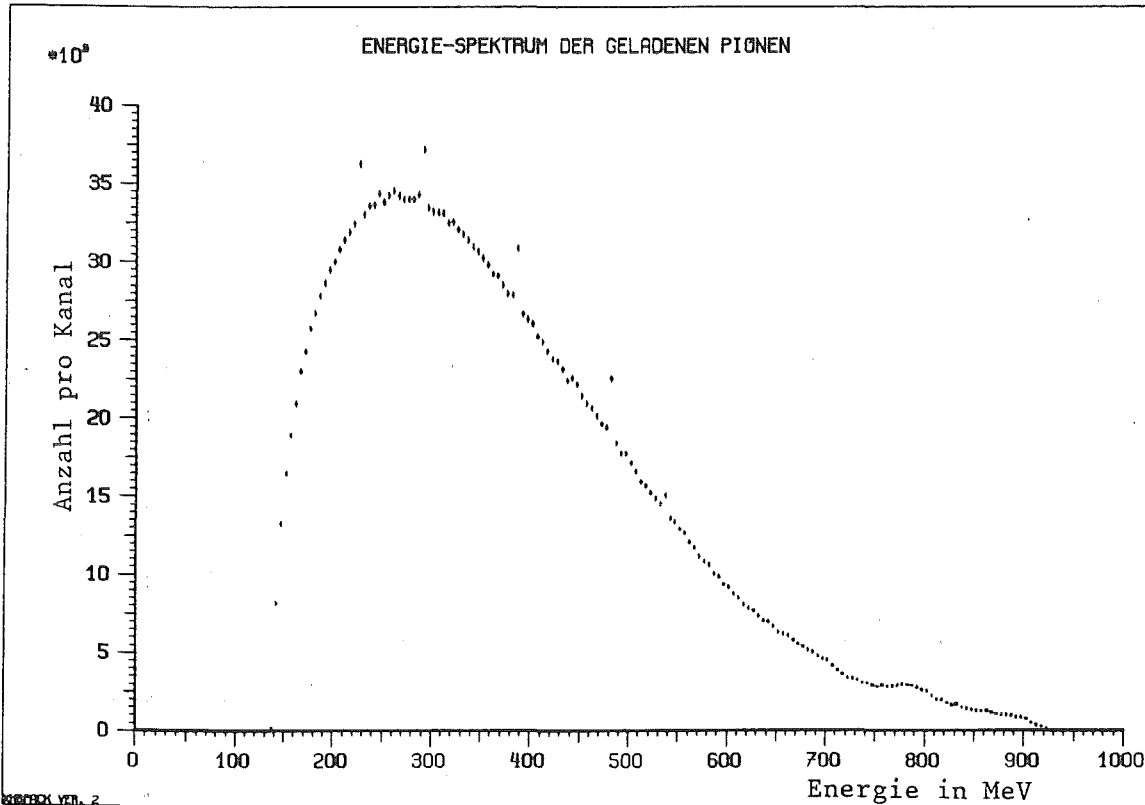


Abbildung 5. Energiespektrum der geladenen Pionen

Interessant sind bei diesem Energiespektrum der geladenen Pionen die beiden Stufen bei 800 MeV und 900 MeV. Die beiden Stufen werden durch die Pionen (ID=28) bewirkt, die vom Zweikörperzerfall des geladenen ρ -Mesons stammen (Kastenverteilung von 200-900 MeV) und den monoenergetischen Pionen von der Annihilation (800 MeV).

Die monoenergetischen Pionen bei 229 MeV, 294 MeV und 386 MeV rühren von der Zweikörperannihilation des Proton - Antiprotonsystems in einen tiefgebundenen Zustand her, dessen Bildung unter Aussendung eines geladenen Pions geschieht (ID=59,60,61).

Die monoenergetischen geladenen Pionen bei 475 MeV und 533 MeV stammen aus der Zweikörperannihilation in das A_2 (ID=12) bzw. das B (ID=20) und ein Pion.

1.4.2. Energiespektrum der neutralen Pionen

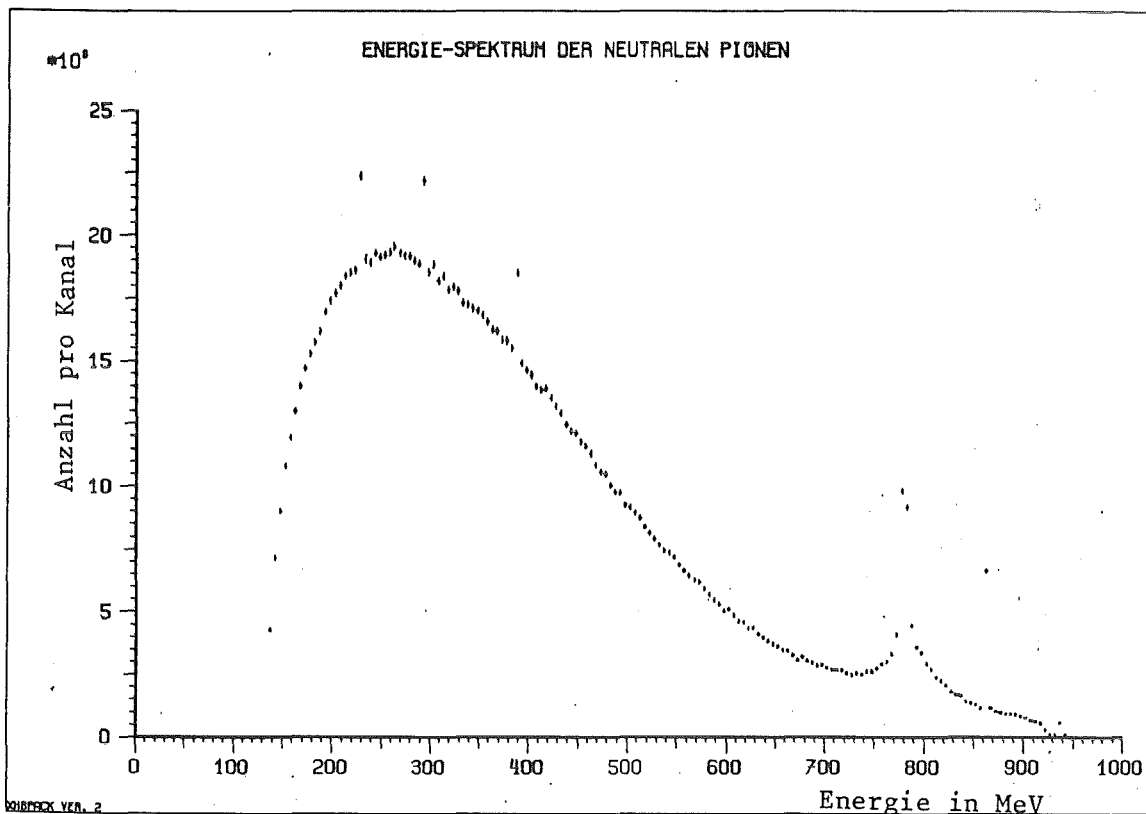


Abbildung 6. Energiespektrum der neutralen Pionen

Ins Auge springen in diesem Energiespektrum die monoenergetischen Pionen bei 228 MeV, 294 MeV und 385 MeV, die bei der Bildung eines tiefgebundenen Zustands ausgesendet werden (ID=62,63,64).

Auffallend sind die Höcker bei 780 MeV (ID=9,65), bei 862 MeV (ID=66) und 943 MeV (ID=67).

Diese entstehen bei den Annihilationskanälen, in denen die Annihilation unter Bildung eines breiten resonanten Zwischenzustands und eines neutralen Pions vor sich geht.

1.4.3. Energiespektrum der Gammas

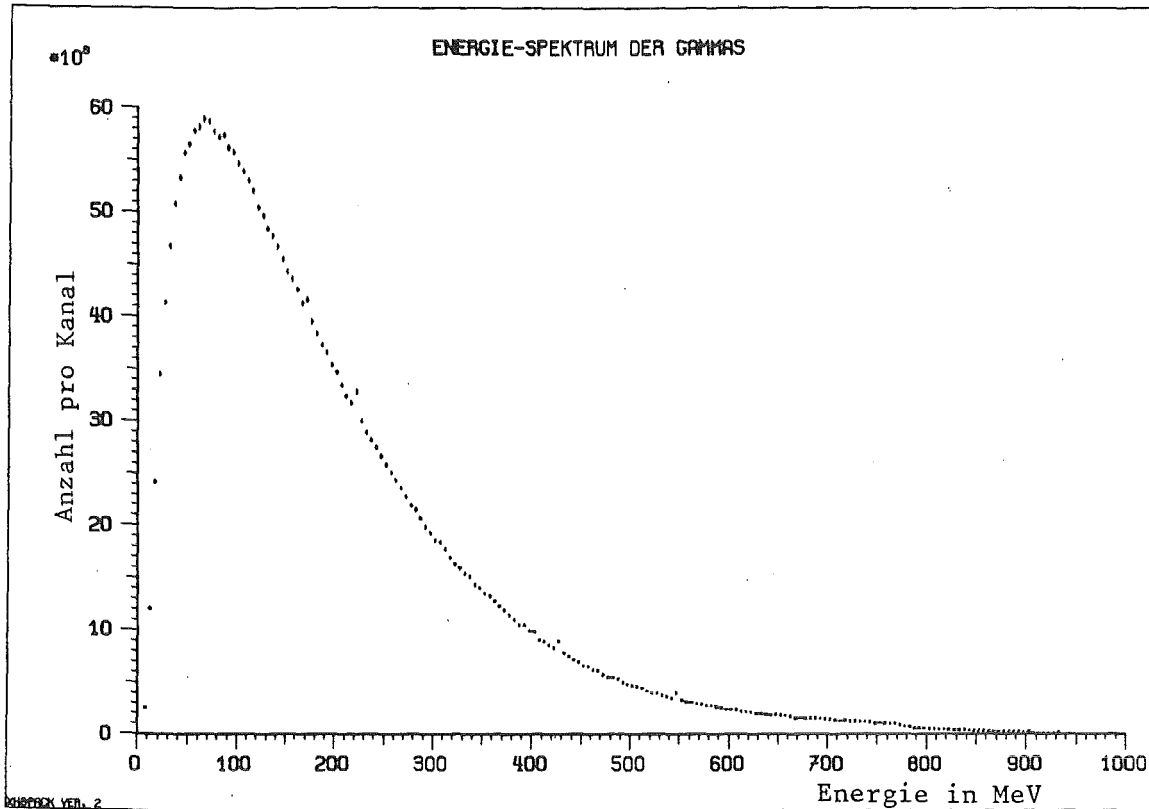


Abbildung 7. Energiespektrum der Gammas

Die Gestalt des Energiespektrums der Gammas wird hauptsächlich durch Kanäle mit vielen zerfallenden neutralen Pionen bestimmt.

Der Höcker bei 770 MeV und die in diesem Maßstab kaum sichtbaren Stufen bei 850 MeV und 933 MeV kommen durch die radiativen Zerfälle des ω (ID=65), des η (ID=66) und der Proton - Antiproton Annihilation in π^0 und γ (ID=67) zustande.

Auch hier sieht man Hinweise auf die Existenz tiefgebundener Zustände in Form monoenergetischer Gammas, die im Spektrum bei 173 MeV, 223 MeV, 400 MeV, 429 MeV und 548 MeV deutlich sichtbar sind (ID=54,55,56,57,58).

KAPITEL 2. DIE SIMULATION

Zur Monte - Carlo Computersimulation des Detektorsystems wurde das Programmpaket 'DETEKTOR' entwickelt (siehe Anhang B) .

2.1. DAS 'STANDARD-DETEKTOR-SYSTEM'

2.1.1. Parametrisierung

Das Programmpaket 'DETEKTOR' simuliert ein Detektorsystem, dessen Geometrie in Abbildung 8 dargestellt ist:

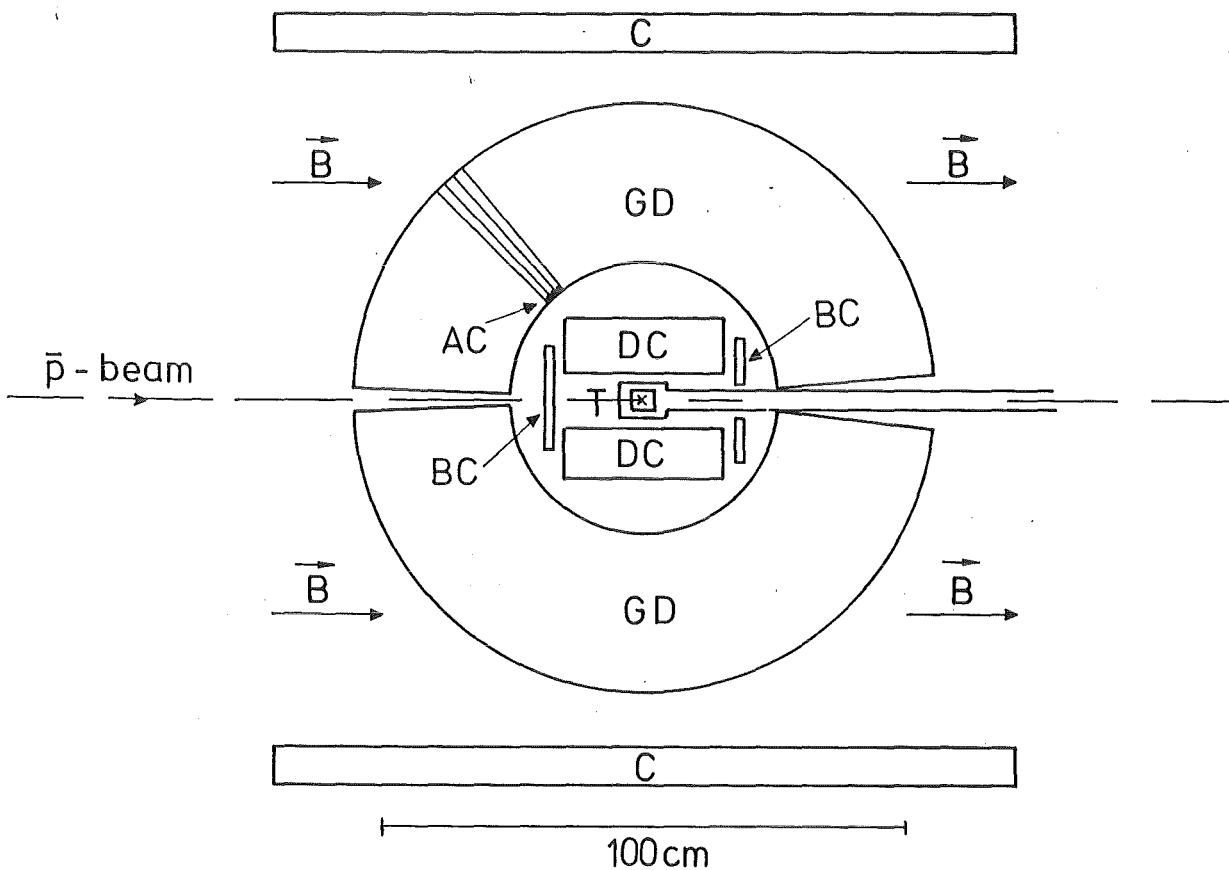


Abbildung 8. Standard Detektor System

Durch eine schmale Öffnung treten Antiprotonen mit niedrigem Impuls (100 MeV/c) in das Detektorsystem ein und treffen in der Mitte auf das Target (T). Die Antiprotonen werden im Target gestoppt und annihilieren mit den Protonen des Targetmaterials, wobei die Stopdichten der Antiprotonen um den Targetmittelpunkt in jede Raumrichtung einer Gaussverteilung gehorchen.

Ausgehend von dem Wechselwirkungspunkt (Vertex) der Antiprotonen mit den Protonen des Targetmaterials breiten sich die Annihilationsprodukte isotrop aus :

Die Richtung der geladenen Teilchen (π, K) wird in einer Driftkammer (DC) gemessen, die das Target in Strahlrichtung zylindrisch umgibt. Um dabei auch deren Impuls bestimmen zu können, ist die gesamte Anordnung von einem homogenen Magnetfeld durchsetzt, das durch die Magnetspulen (C) erzeugt wird.

Strahlkammern (BC) vor und hinter der Driftkammer dienen zur Identifikation der einlaufenden Antiprotonen bzw. als Antizähler für geladene Annihilationsprodukte, die die Driftkammer verfehlen.

Die neutralen Teilchen (γ) werden in einem modular aufgebauten Kristalldetektor (GD) nachgewiesen, dessen Einzelkristalle kugelförmig um die Driftkammer angeordnet sind.

Trifft nun ein neutrales Teilchen ein Kristallmodul, so konvertiert dessen Energie in Elektronen und Positronen, die sich längs der Kristallachse ausbreiten (Schauerbildung). Die Richtung eines neutralen Teilchens lässt sich hierbei aus der Lage des getroffenen Kristallmoduls bestimmen, dessen Energie aus der Stärke des elektromagnetischen Schauers.

Wahlweise können vor jedem Einzelmodul Plastiksintillatoren als Antizähler (AC) angebracht werden, die ein einlaufendes geladenes Teilchen von einem neutralen Teilchen unterscheiden.

Zur Durchführung der Computersimulation wurden für das betrachtete Detektorsystem folgende Annahmen getroffen:⁵

Koordinatensystem

- o Nullpunkt in Targetmitte
- o z - Achse in Strahlrichtung
- o x,y - Achse senkrecht zur Strahlrichtung

Target

- o Flüssigwasserstoff, Dichte: 0.0708 g/cm^3
- o Länge: 3 cm, Durchmesser: 3 cm
- o Breite der Antiprotonen-Stoppverteilung: 1 cm

Driftkammer

- o Länge: 40 cm, Durchmesser: 40 cm
- o Räumliches Auflösungsvermögen in Strahlrichtung: $5000 \mu\text{m}$
- o Räumliches Auflösungsvermögen senkrecht zum Strahl: $250 \mu\text{m}$
- o Stärke des Magnetfeldes: 1 T
- o Impulsauflösung: 5% (FWHM) bei 300 MeV/c, unterhalb const., oberhalb prop. zu p_{\perp}
- o Nachweisuntergrenze: 50 MeV/c

Gammadetektor

- o Innenradius: 34.4 cm
- o Anzahl der Module: 1647
- o Bedeckter Raumwinkel: 99.7%
- o Winkelauflösung: 5°
- o Energieauflösung: 7% (FWHM) bei 100 MeV, sonst prop. zu $1/\sqrt{E}$
- o Nachweisuntergrenze: 10 MeV

Dieser Satz von Parametern charakterisiert das im Programmpaket 'DETEKTOR' realisierte 'STANDARD-DETEKTOR-SYSTEM', von dem im folgenden die Rede sein soll.

⁵ Alle Auflösungen sind FWHM.

Bei der technischen Realisierung eines solchen Detektorsystems wird man aus Gründen der besseren Raumnutzung allerdings von einer Kugelgeometrie des Gammadetektors zu einer Zylindergeometrie übergehen.

Wie ein solches Detektorsystem aussehen könnte, zeigt Abbildung 9.

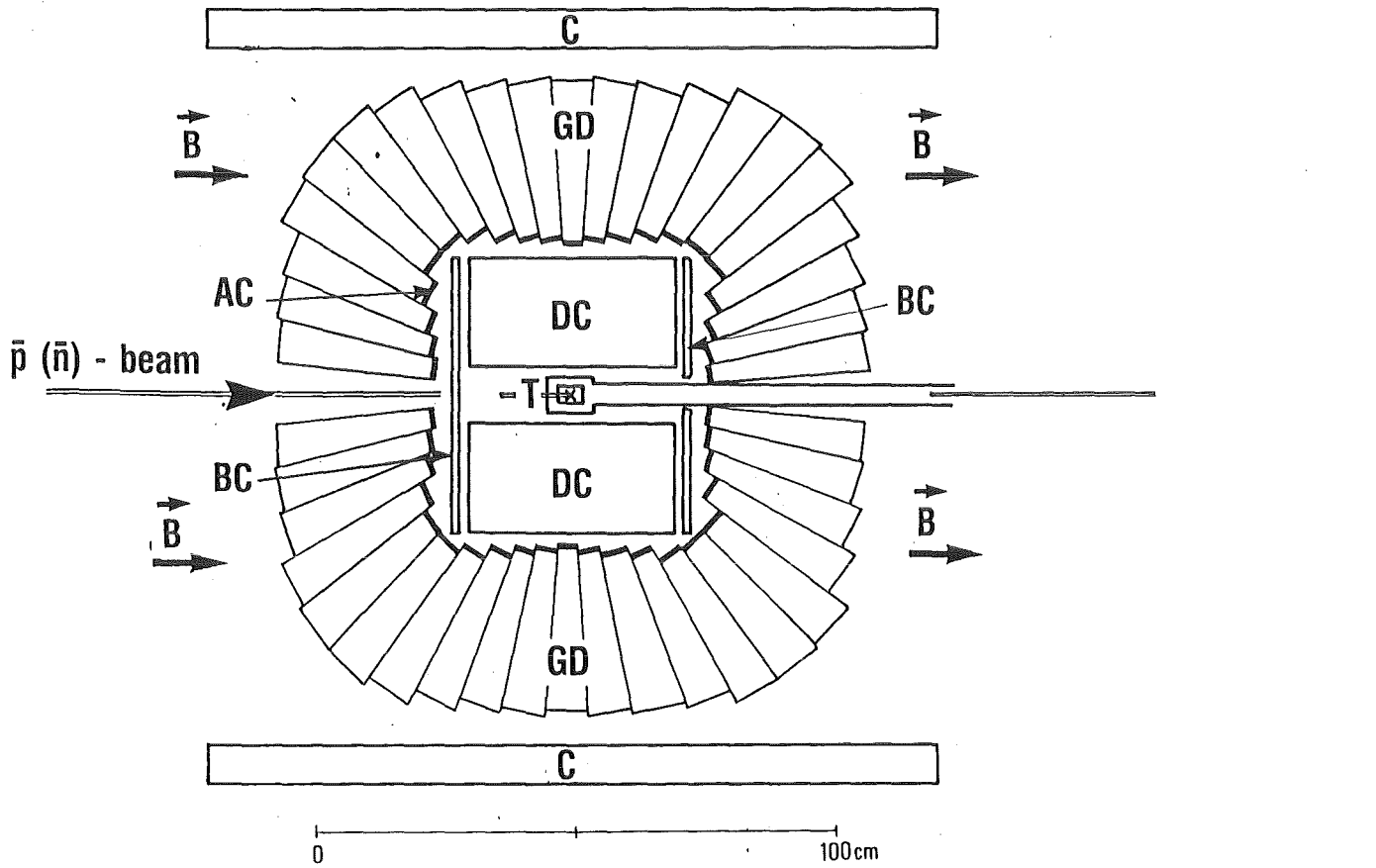


Abbildung 9. Detektor-System (praxisnahe Geometrie)

2.1.2. Programmtechnische Realisierung

Mit den erzeugten Annihilationskinematiken wurde nun eine Monte - Carlo Computersimulation des 'STANDARD-DETEKTOR-SYSTEMS' durchgeführt: Dazu wurden für jeden Durchlauf 3.200.000 Annihilationen von der Magnetbanddatei wieder eingelesen und wie folgt verarbeitet.

In dem zylinderförmigen Target wurden zunächst Wechselwirkungspunkte für die Proton - Antiproton Annihilation gewürfelt, deren Verteilung in Abbildung 10 dargestellt ist. Die Breite der Gaussverteilung wurde mit 1 cm so gewählt, daß sie der Stopdichtenverteilung von niederenergetischen Antiprotonen in flüssigem Wasserstoff entsprach.

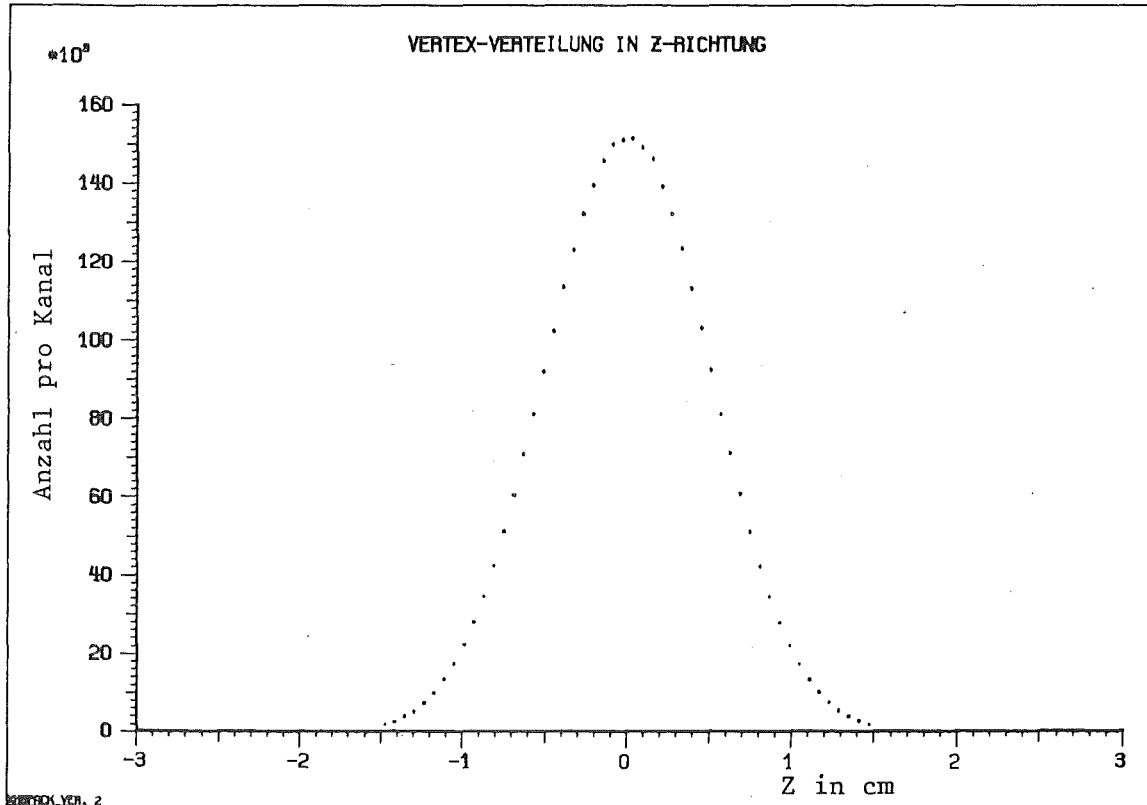
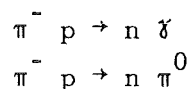


Abbildung 10. Stopdichtenverteilung der Antiprotonen im Target

Ausgehend vom jeweiligen Vertex wurden die aus der Annihilation hervorgehenden Teilchen auf ihrer Bahn durch das Detektorsystem weiterverfolgt.

Nun gibt es aber Reaktionen der Sekundärteilchen mit dem Targetmaterial, die ihrerseits wiederum neue Teilchen produzieren. Es sind dies vor allen Dingen die Reaktionen der negativ geladenen Pionen mit dem Target-Wasserstoff, wenn diese im Target zur Ruhe kommen /SPU77/:



Das Häufigkeitsverhältnis der beiden Reaktionen zueinander ist ungefähr 1/1.55.

Im ersten Fall entsteht ein monoenergetisches Gamma, dessen Energie 129.4 MeV beträgt, im zweiten Fall ein monoenergetisches neutrales Pion, das beim Zerfall ein Gamma-Kastenspektrum von 56 MeV bis 86 MeV erzeugt.

Kam nun im Verlauf der Simulation ein geladenes Pion im Wasserstoff des Targets zur Ruhe, so wurde es aus der Liste der Teilchen gestrichen und im Falle des negativen Pions durch die entstehenden neuen Teilchen ersetzt (Unterprogramm 'PISTOP').

Abbildung 11 zeigt die Stopverteilung der geladenen Pionen im Target ,
Abbildung 12 deren Energiespektrum .

Wie daraus hervorgeht, kommt aufgrund der geringen Targetabmessungen nur ein geringer Anteil der geladenen Pionen im Gesamtenergiebereich von 139 MeV bis 147 MeV im Target zur Ruhe.

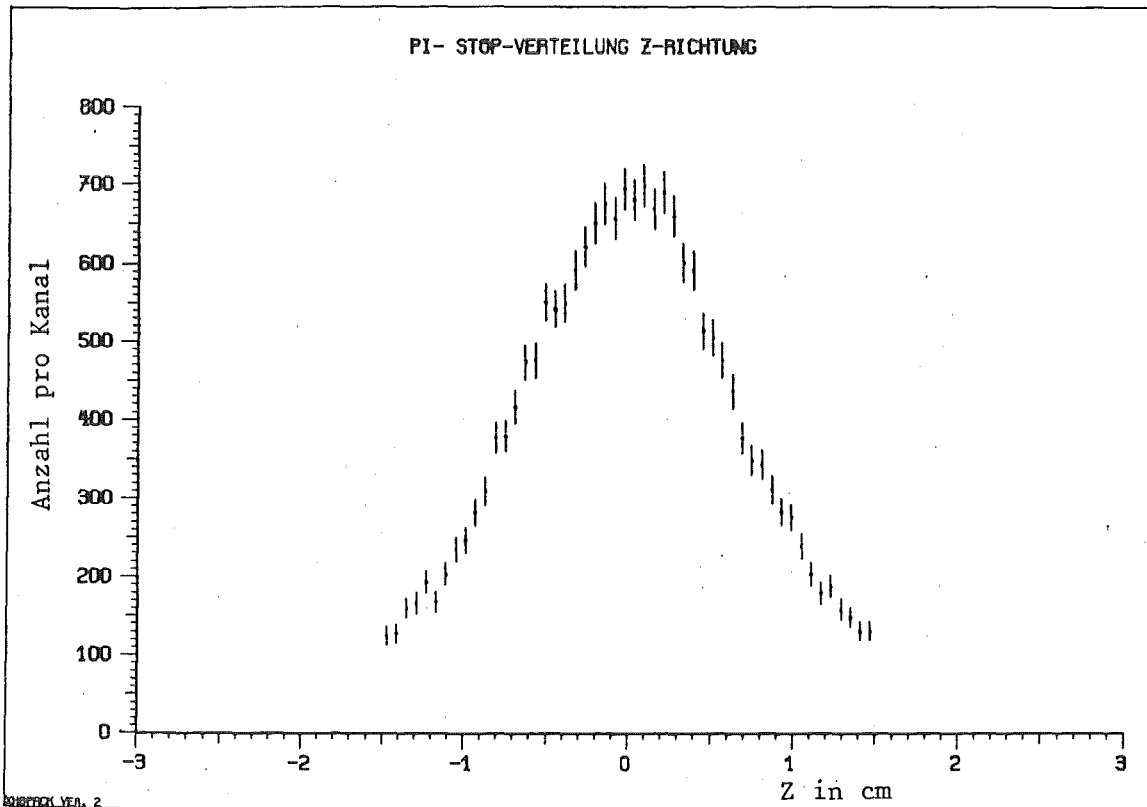


Abbildung 11. Stopverteilung der geladenen Pionen im Target

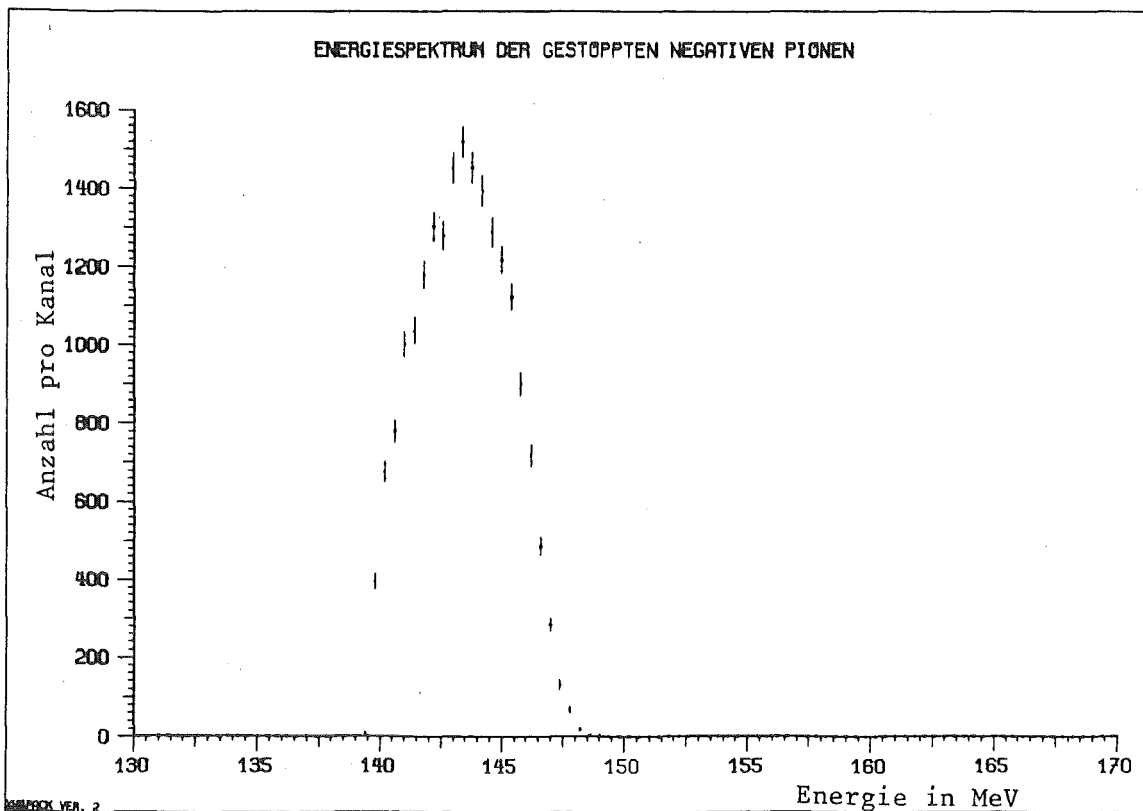


Abbildung 12. Energiespektrum der stoppenden geladenen Pionen

Die Bahnen der geladenen Teilchen wurden außerhalb des Targets weiterverfolgt und im Falle des Auftreffens auf die Targethalterung (Unterprogramm 'PARDET') oder des Verfehlens der Driftkammer in Strahlrichtung getilgt (Unterprogramm 'CHAMBR').

Die übrig gebliebenen geladenen Teilchen durchquerten die Driftkammer, in der ihr Impuls gemessen wurde (Unterprogramm 'MOMMES').⁶

Als Abhängigkeit der Impulsauflösung vom Impulsbetrag wurde dabei folgender Zusammenhang zugrunde gelegt /WAG80/:

$$\frac{\Delta P}{P} \sim \text{const. } 5\% \text{ für } 50 \text{ MeV/c} < P < 300 \text{ MeV/c} \text{ (FWHM)}$$

$$\frac{\Delta P}{P} \sim \frac{P_0}{P} \text{ sonst, } \frac{\Delta P}{P} = 5\% \text{ bei } P_0 = 300 \text{ MeV/c} \text{ (FWHM)}$$

Auch wurde die Raumwinkelabhängigkeit der Impulsauflösung durch folgenden Zusammenhang berücksichtigt:

$$\Gamma(\theta) \sim \sin(\theta)$$

$$\Gamma(\phi) \sim 1 / \sin(\theta)$$

Hier ist θ der Winkel der Teilchenbahn zur Strahlachse und ϕ der Meridianwinkel. Γ stellt die Breite der Gaussverteilung dar, nach der die Impulsauflösung für die geladenen Teilchen gewürfelt wurde.

⁶ Die kaonischen Annihilationskanäle wurden zwar erzeugt, der Nachweis der Kaonen wurde aber bislang noch nicht in die Detektorsimulation mit einbezogen.

Zum Nachweis der Gammaquanten wurde ein ähnliches Verfahren herangezogen:

Aufgrund ihres hohen Durchdringungsvermögens für Materie blieb eine evtl. Konversion von Gammaquanten im Target und im Material der Driftkammer unberücksichtigt. Auf ihrem weiteren Weg wurde untersucht, ob sie durch eine Öffnung unentdeckt aus der Kristallkugel entwichen.

Notwendige, in der Simulation berücksichtigte Öffnungen sind z.B. Targethalterung (5°), Strahlzuführung (5°) und Driftkammerauslese (10°).

Drang ein Gammaquant in ein Kristallmodul ein, wurde die deponierte Energie gemessen (Unterprogramm 'ENEMES')⁷, wobei die Energieauflösung des Gammadetektors nach folgender Abhängigkeit gebildet wurde:

$$\frac{\Delta E}{E} \sim \frac{\sqrt[4]{E_0}}{\sqrt[4]{E}}, \quad \frac{\Delta E}{E} = 7\% \text{ bei } E_0 = 100 \text{ MeV (FWHM)}$$

Hier ist ΔE die Gaussbreite der Verteilung zum Würfeln der Energieauflösung eines Kristallmoduls beim Nachweis eines Gammaquants.

Im Anschluß daran wurden die nachgewiesenen Teilchen ihren Namen nach sortiert, in einer Liste zusammen mit den zugehörigen Meßwerten festgehalten (Unterprogramm 'PSORT') und das so erhaltene Ereignis auf die Gültigkeit der physikalischen Erhaltungssätze (Ladung, Impuls, Energie) überprüft (Unterprogramm 'CONLAW').

Handelte es sich um ein kinematisch vollständig erfaßtes Ereignis, wurden auf die in der Liste abgelegten Teilchen Masken der Annihilationskanäle aus Abbildung 2 aufgelegt und getestet, ob ein bekanntes Ereignis nachgewiesen werden konnte (Unterprogramm 'TESTID'). In diesem Falle wurden für spezielle Annihilationskanäle invariante Massen berechnet und in Histogramme eingetragen.

Waren alle Annihilationsereignisse der Magnetbanddatei abgearbeitet, oder stand zur Bearbeitung weiterer Ereignisse keine Rechenzeit mehr zur Verfügung, wurde die Programmausführung durch Ausdrucken eines Abschlussberichts beendet (siehe Anhang B).

⁷ Anm.: Auf eine Simulation der Entwicklung des elektromagnetischen Schauers wurde aus Rechenzeitgründen verzichtet.

2.2. REKONSTRUKTION VON INVARIANTEN MASSEN

Unter der Annahme, daß ein Teilchen in n weitere Teilchen zerfällt, deren Gesamtenergien und Impulscomponenten gemessen werden können, läßt sich die invariante Ruhemasse des ursprünglichen Teilchens über die Umkehrung des relativistischen Impuls-Energie-Satzes gewinnen:

$$(12) \quad M = \frac{1}{c^2} \sqrt{(\sum_{i=1}^n E_i)^2 - (\sum_{i=1}^n \vec{P}_i)^2 c^2}$$

Speziell für die Bildung der invarianten Masse von Gammapaaren gilt wegen der verschwindenden Ruhemasse der Gammas

$$(13) \quad M = \frac{1}{c^2} \sqrt{2E_1 E_2 \cos(1 - \chi(\gamma_1, \gamma_2))}$$

wobei E_i die gemessenen Gamma-Energien sind und $\chi(\gamma_1, \gamma_2)$ der gemessene Zwischenwinkel zwischen zwei Gammas.

Werden nun je zwei nachgewiesene Gammas zur invarianten Masse rekombiniert, so ergibt sich unter Berücksichtigung aller Annihilationsereignisse das folgende Bild:

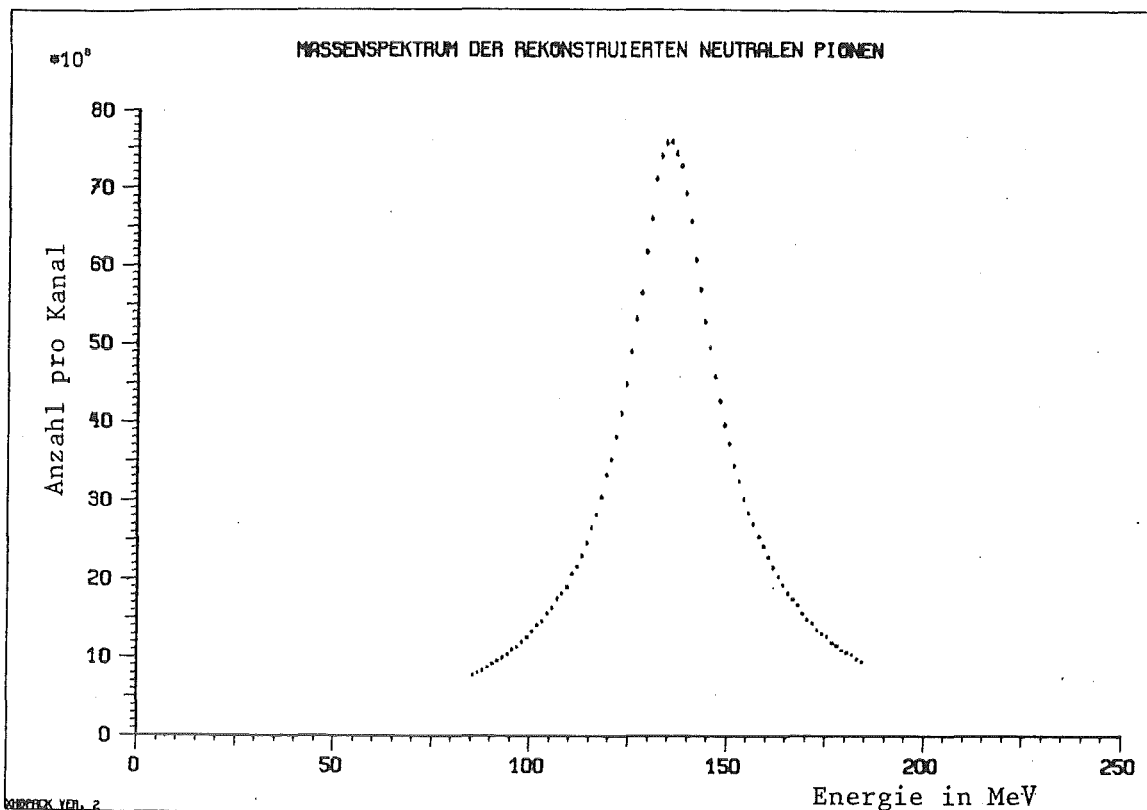


Abbildung 13. Spektrum der invarianten Masse der rekonstruierten neutralen Pionen

Man erkennt deutlich, daß die meisten Gammapaare um eine invariante Masse von 135 MeV gaussverteilt die Masse des kurzlebigen neutralen Pions ergeben. Die Breite der Verteilung ist dabei bestimmt durch die Energie- und die Ortsauflösung des Gammadetektors.

Ein neutrales Pion gilt dabei als mit großer Wahrscheinlichkeit erkannt, wenn die rekonstruierte Masse in einen Bereich von ± 50 MeV um die wirkliche Pionenmasse fällt.

Die Abhängigkeit der Rekonstruktionsrate für neutrale Pionen von den Detektorgrößen Energieauflösung und Ortsauflösung ist für den Fall einer Raumwinkelbedeckung von $\Omega=99\%$ in der folgenden Tabelle wiedergegeben:

$\Delta E/E$ (FWHM) $\Delta\phi$ (FWHM)	0%	4%	8%	12%	25%	40%
0°	99%	97%	90%	85%	77%	75%
2°	98%	96%	89%	84%	76%	74%
4°	96%	93%	86%	82%	75%	73%
6°	91%	88%	82%	78%	72%	70%
8°	85%	83%	78%	74%	68%	66%
10°	80%	78%	74%	70%	65%	63%
12°	75%	73%	69%	67%	62%	60%

Rekonstruktionsvermögen des Detektorsystems für neutrale Pionen

Bei der Rekonstruktion der neutralen Pionen treten aufgrund von zufällig zueinander passenden Gammas allerdings auch Fehlrekonstruktionen auf: Dies trifft im besten Fall ($\Delta E/E=0\%$, $\Delta\phi=0^\circ$) für ca. 0.5% aller nachgewiesenen Gammas zu, im schlechtesten Fall ($\Delta E/E=40\%$, $\Delta\phi=12^\circ$) für ca. 12% aller Gammas.

Ein weiterhin bemerkenswertes Ergebnis ist die Tatsache, daß aufgrund von Fehlrekonstruktionen rund 10% der auftretenden monoenergetischen Gammas, die ja nachgewiesen werden sollen, zu 'falschen' neutralen Pionen wegrekonstruiert werden.

Ein Gesamtenergiespektrum der aus den mit dem 'STANDARD-DETEKTOR-SYSTEM' gemessenen Gammas rekonstruierten neutralen Pionen ist in Abbildung 14 wiedergegeben:

Es entspricht dem Gesamtenergiespektrum der neutralen Pionen in Abbildung 6, allerdings sind hier aufgrund des endlichen Auflösungsvermögens des 'STANDARD-DETEKTOR-SYSTEMS' die Linien der monoenergetischen Pionen stark verbreitert.

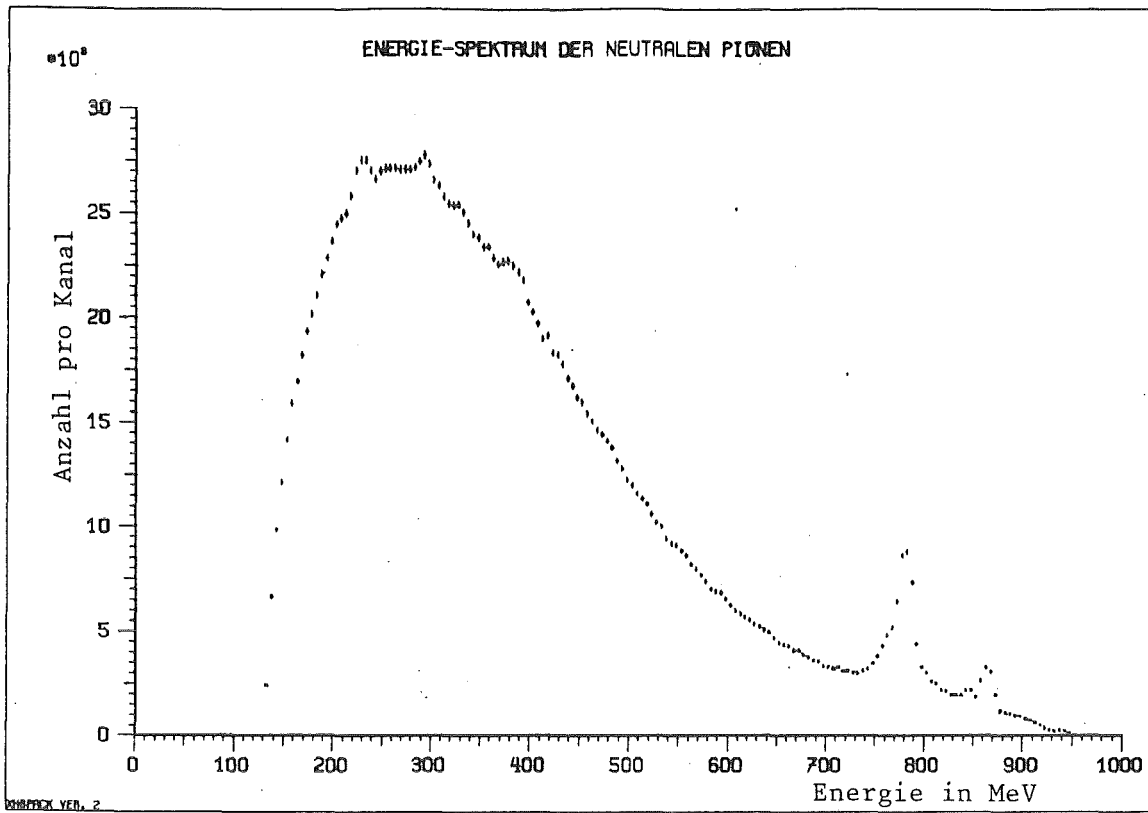
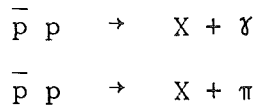


Abbildung 14. Inklusives Gesamtenergiespektrum der rekonstruierten neutralen Pionen

2.3. INKLUSIVE ENERGIESPEKTREN

Im folgenden soll eine Diskussion der Ergebnisse der Monte - Carlo Computersimulation des 'STANDARD-DETEKTOR-SYSTEMS' an willkürlich herausgegriffenen Beispielen erfolgen.

So wurde z.B in den inklusiven Gesamtenergiespektren (D.h. alle auftretenden Teilchen einer Sorte wurden in ein Histogramm einsortiert) der Gammaquanten und der geladenen Pionen nach Reaktionen der folgenden Art gesucht:



Das Auftreten eines monoenergetischen Gammaquants oder eines monoenergetischen Pions der Gesamtenergie $E_{\gamma, \pi}$ weist auf die Existenz langlebiger Zustände mit der Masse

$$(14) M_X = \frac{1}{c^2} \sqrt{E_T^2 - 2 E_T E_{\gamma, \pi}} \quad , \quad E_T = 1876 \text{ MeV}$$

hin (E_T ist die Schwellenenergie des Proton-Antiproton-Systems).

Die Energien der bei der Annihilation entstehenden geladenen Pionen und der Gammaquanten wurden wie beschrieben mit dem 'STANDARD-DETEKTOR-SYSTEM' gemessen.

Das so erhaltene Gesamtenergiespektrum der geladenen Pionen ist in Abbildung 15 wiedergegeben, das Energiespektrum der Gammaquanten in Abbildung 16. Beim Vergleich mit den direkt erzeugten Energiespektren der geladenen und der neutralen Teilchen (Abbildung 5 und Abbildung 7, Detektor unendlich guter Auflösung) fällt insbesondere eine Verschmierung der dort gut sichtbaren Linien monoenergetischer Gammas auf.

Diese Verschmierung wird durch die endliche Energieauflösung des Detektorsystems bewirkt.

Um nun die so erhaltenen inklusiven Gesamtenergiespektren von dem hohen Untergrund zu reinigen, muß man einiges an physikalischem Wissen in die Aufbereitung der inklusiven Spektren investieren.

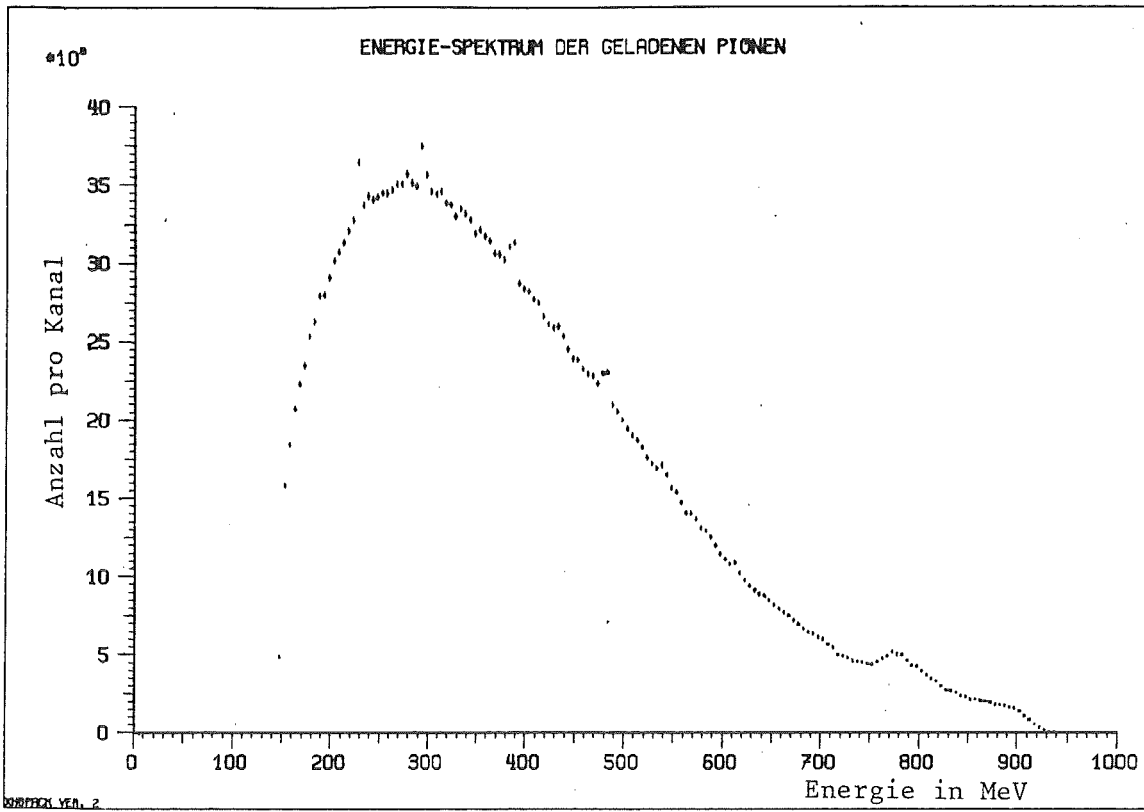


Abbildung 15. Gemessenes Energiespektrum der geladenen Pionen

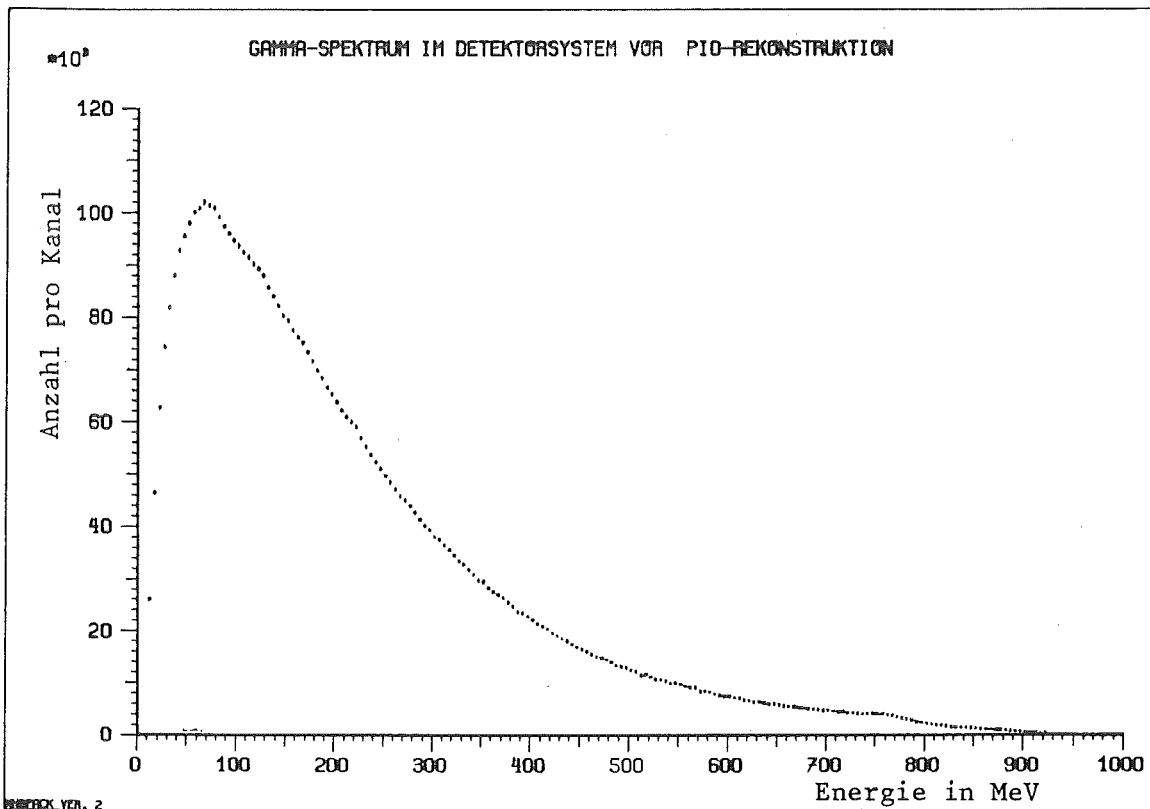


Abbildung 16. Gemessenes Energiespektrum der Gammas

2.3.1.Reduktion des Untergrundes

Es bietet sich an, in den inklusiven Gesamtenergiespektren nach Teilchenkombinationen zu suchen, die die Rekonstruktion der invariante Masse eines bekannten resonanten Zwischenzustandes erlauben. Diese Teilchen können dann aus dem inklusiven Gesamtenergiespektrum aussortiert werden, denn ihr Ursprung ist bekannt und sie tragen zur Untergrundbildung bei.

Auf diese Weise gelingt es z.B., mit dem 'STANDARD-DETEKTOR-SYSTEM' ungefähr 85% der nachgewiesenen Gammas als zu neutralen Pionen gehörig aus dem inklusiven Energiespektrum der Gammas zu entfernen. Abbildung 17 zeigt, daß sich der Untergrund im inklusiven Energiespektrum der Gammas um fast eine Größenordnung verringert und die Linien monoenergetischer Gammas vor allem im niederenergetischen Bereich des Spektrums klarer hervortreten.

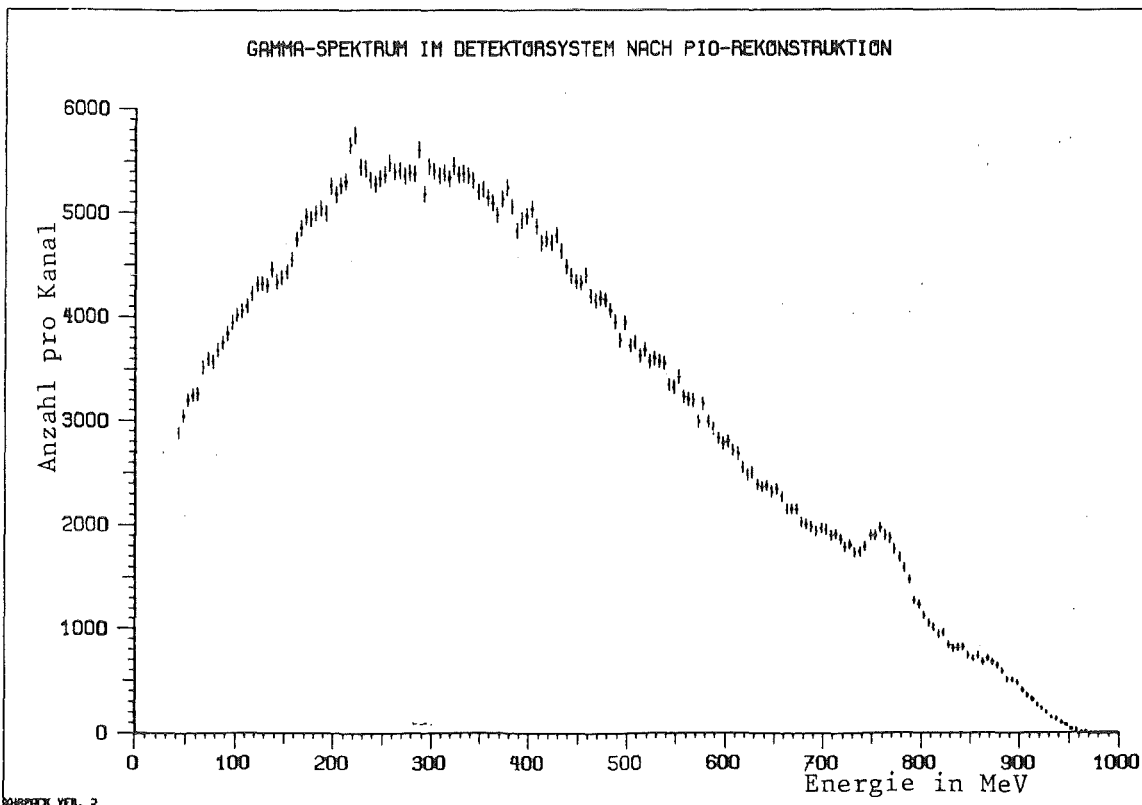


Abbildung 17. Inklusives Energiespektrum der Gammas nach Rekonstruktion der neutralen Pionen

Deutlich sichtbar ist ein Hinweis auf die Existenz eines hypothetischen, breiten 'Gluonium'-Zustands (800 MeV, $\Gamma=20\text{MeV}$, ID=68). Dieser Zustand war bislang im Untergrund verborgen und wird unter Aussendung eines Gammaquants einer Energie bei 768 MeV gebildet.

Eine Verbesserung der Unterdrückung des Untergrundes in noch höherem Maße wird erreicht durch die Beschränkung auf Annihilationskanäle, in denen eine nur ungerade Zahl von Gammas auftritt.

Zusätzlich wird durch Anwendung der Energieerhaltung ($\Sigma E = 1877 \pm 50 \text{ MeV}$) und der Impulserhaltung ($\Sigma P = \pm 50 \text{ MeV}/c$) mit hoher Wahrscheinlichkeit sichergestellt, daß alle zur Annihilation gehörigen Teilchen nachgewiesen wurden.

In diesem Falle wurde dann eines der Gammas mit großer Gewißheit bei der Bildung eines tiefgebundenen langlebigen Zustandes freigesetzt.

Durch die Anwendung der Methode der 'ungeraden Gammazahlen' gelingt es, den Untergrund unter den Linien der monoenergetischen Gammaquanten entscheidend zu reduzieren. Der verbleibende Untergrund ist dadurch zu erklären, daß durch Öffnungen im Gammadetektor ein niederenergetisches Gamma ($E < 50$ MeV) entweicht, das aus dem Zerfall eines neutralen Pions stammt, und somit eine ungerade Zahl an Gammas übrigbleibt.

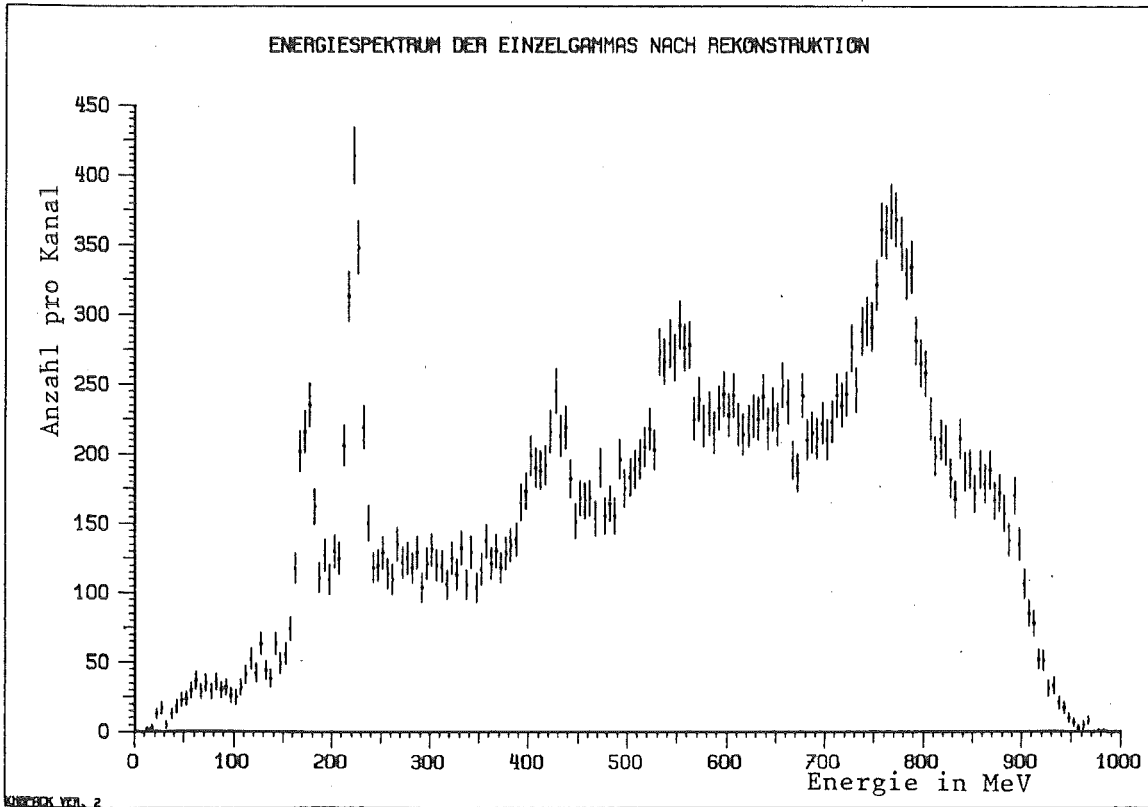


Abbildung 18. Inklusives Energiespektrum der Einzelgammas nach Rekonstruktion der neutralen Pionen

Wie das gegebene Beispiel zeigt, erlaubt das 'STANDARD-DETEKTOR-SYSTEM' z.B. in inklusiven Gammaskpektren den Nachweis von Zuständen mit Wahrscheinlichkeiten bis zu 10^{-4} bei einer Statistik von nur 3.2 Millionen Annihilationsereignissen.

2.4.EXKLUSIVE ENERGIESPEKTREN

Durch die fast vollständige Überdeckung des Raumwinkels wird durch das vorgestellte Detektorsystem ein kinematisch vollständiges Experiment ermöglicht:

Indem alle Reaktionsprodukte nachgewiesen werden, die aus einer Proton - Antiproton Annihilation hervorgehen, gelingt die Bestimmung der Quantenzahlen eines evtl. gebildeten Zwischenzustandes. Die Überprüfung der Vollständigkeit der Messung wird über die Überprüfung der physikalischen Erhaltungssätze erreicht (Impulssatz, Energiesatz).

2.4.1.Nachweis resonanter Zerfälle

Unter resonanten Zerfällen sind solche Zerfälle zu verstehen, bei denen die Proton - Antiproton Annihilation unter Bildung eines resonanten Zwischenzustandes abläuft.

Im folgenden wird mit dem 'STANDARD-DETEKTOR-SYSTEM' in einigen ausgewählten Annihilationskanälen nach solchen Resonanzen gesucht.

2.4.1.1. RADIATIV ZERFALLENDE RESONANZEN

Als Beispiel für die wichtigen radiativen Zerfälle von Mesonen wurden die Prozesse $\omega \rightarrow \pi^0 \gamma$ und $\eta \rightarrow \gamma \gamma$ betrachtet. Sie tauchen in den Annihilationskanälen im Endzustand $2\pi^0 \gamma$ bzw. $\pi^0 2\gamma$ mit 2.4% bzw. 0.9% Verzweigungsverhältnis auf.

Ihre Invarianten-Massen-Spektren (Abbildung 19, Abbildung 20) demonstrieren in anschaulicher Weise die gute Separation evtl. störender Kanäle und die hohe Sensitivität der Meßapparatur für alle radiativen Prozesse.

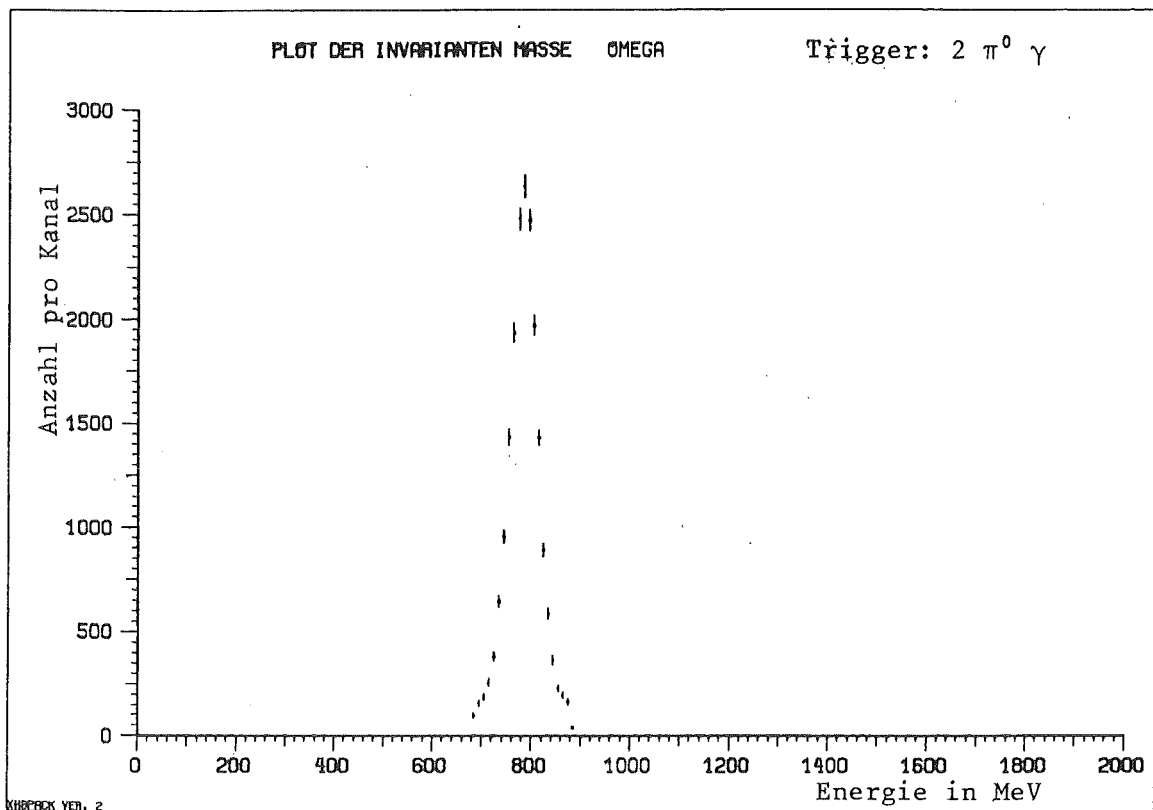


Abbildung 19. Invariante Masse des radiativ zerfallenden Omega

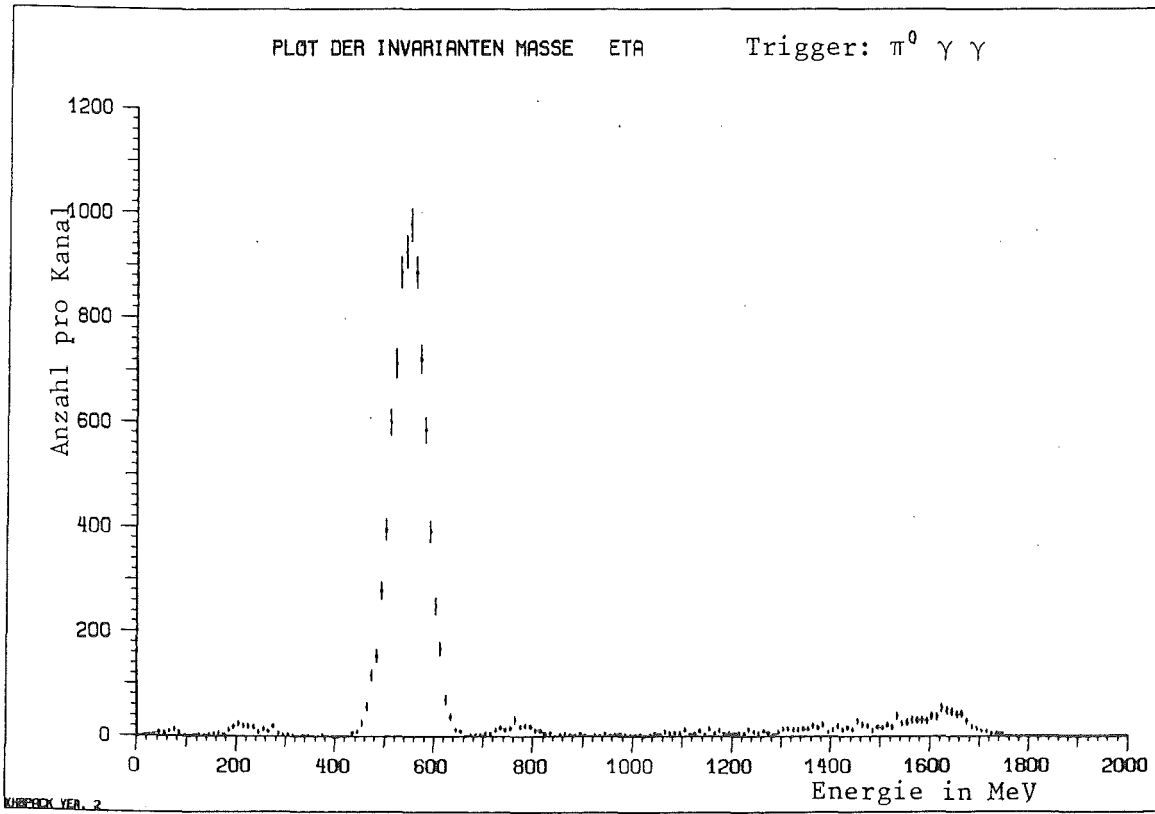


Abbildung 20. Invariante Masse des radiativ zerfallenden Eta

2.4.1.2. IN GELADENE TEILCHEN ZERFALLENDE RESONANZEN

In den Kanälen mit dem Endzustand $\pi^+ \pi^- \pi^0$ (ID=8-10) und $2\pi^+ 2\pi^- \pi^0$ (ID=16-22) wurde nach invarianten Massen gesucht. Die Ergebnisse sind in Abbildung 21 und Abbildung 22 dargestellt. Im Kanal (ID=9) dominiert wie erwartet das 154 MeV breite ρ^0 (769 MeV) neben dem im Verhältnis dazu seltenen f^0 (1273 MeV) (ID=10). Im Kanal (ID=16) erscheint das nur 9.9 MeV breite ω (782 MeV) und das η (548 MeV) (ID=21), die deutlich über dem nicht in ω 's und η 's zerfallenden Kontinuum auftauchen.

Aus der Breite dieser Resonanzen läßt sich die Massenauflösung des betrachteten Detektorsystems für den Nachweis in geladener Teilchen zerfallender Resonanzen mit ca. 20 MeV abschätzen.

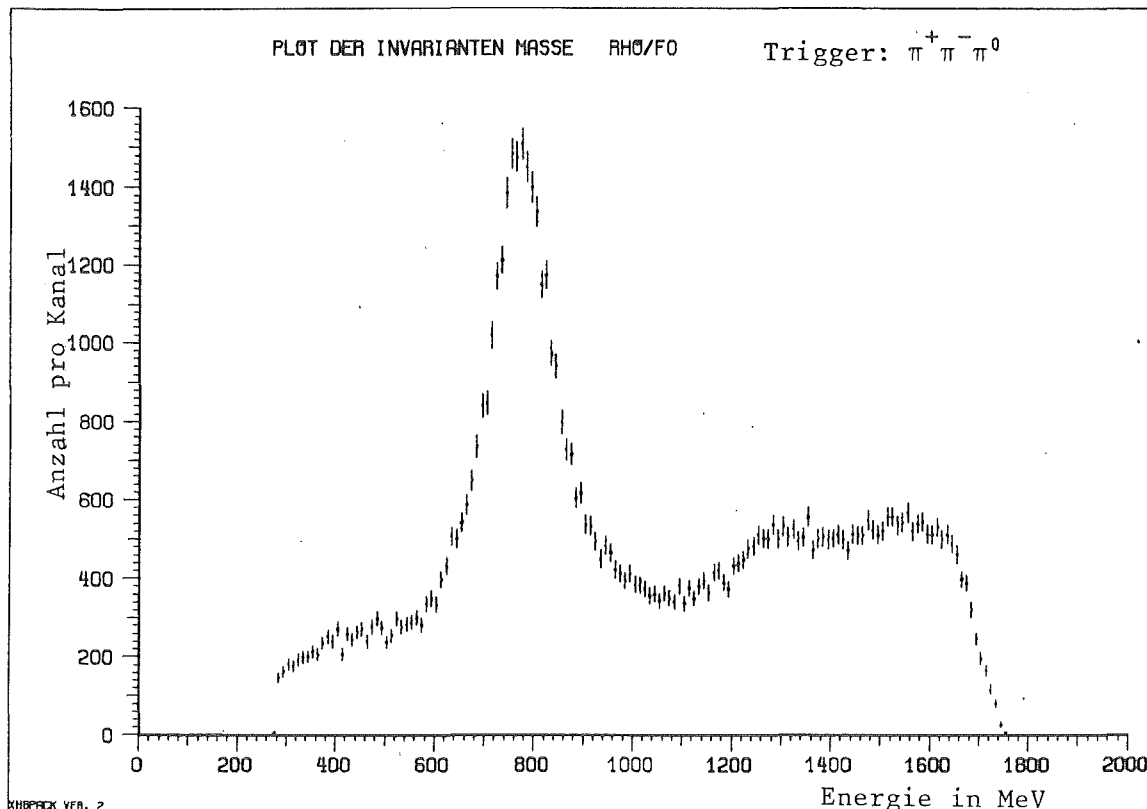


Abbildung 21. Invariante Masse des neutralen Rho

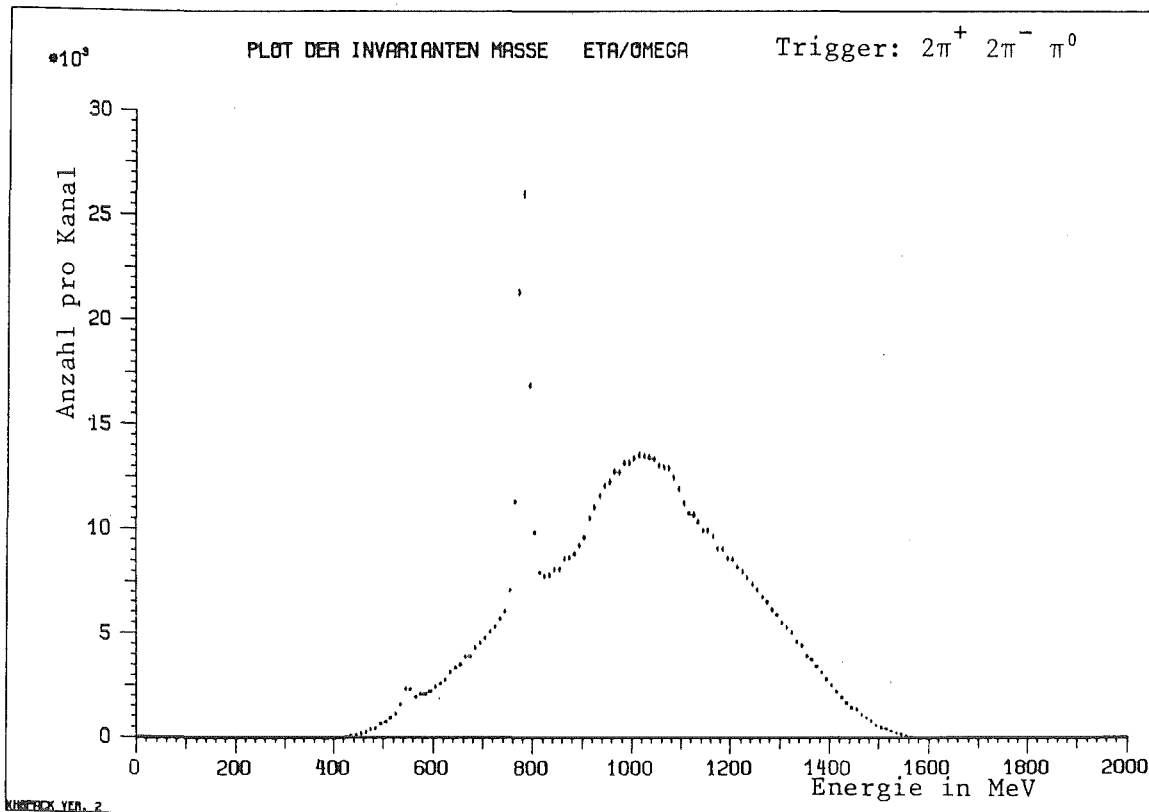


Abbildung 22. Invariante Masse des Omega

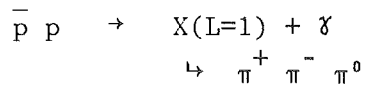
2.4.2. Nachweis exotischer Zerfälle

2.4.2.1. BARYONIUM

Hinweise auf exotische Zustände im Proton - Antiproton System wurden in den inklusiven χ -Energiespektren gefunden (ID=54 ff.). Die schmale Breite der χ -Linien weist auf langlebige Zustände hin, die in einem einfachen $(q\bar{q})$ -Bild nicht zu erklären sind, wohl aber als $(qq)(\bar{q}\bar{q})$ -Quark-Moleküle mit einem von Null verschiedenen Drehimpuls zwischen den Quarkpaaren.

Ein solcher Zustand wird als Baryoniumzustand bezeichnet und zerfällt mit großer Wahrscheinlichkeit in Mesonen mit $L_{q\bar{q}} > 0$.

Der Endzustand $\pi^+ \pi^- \pi^0 \chi$, wo der in $\pi^+ \pi^- \pi^0$ zerfallende Anteil aus einem hypothetischen Baryoniumzerfall hervorgeht, ist ein Beispiel für einen solchen seltenen exotischen Kanal:



Bei exklusiver Messung sollte dieser Anteil im invarianten Massenspektrum auftauchen. Wie Abbildung 23 zeigt, ist dies in beeindruckender Weise der Fall.

Selbst so seltene Prozesse wie $X(1421) \rightarrow \pi^+ \pi^- \pi^0$ mit einem Verzweigungsverhältnis von insgesamt nur 4×10^{-5} erscheinen auf sehr kleinem Untergrund.

Der geringe Untergrund zeigt das große Separationsvermögen der Apparatur für verschiedene Annihilationskanäle .

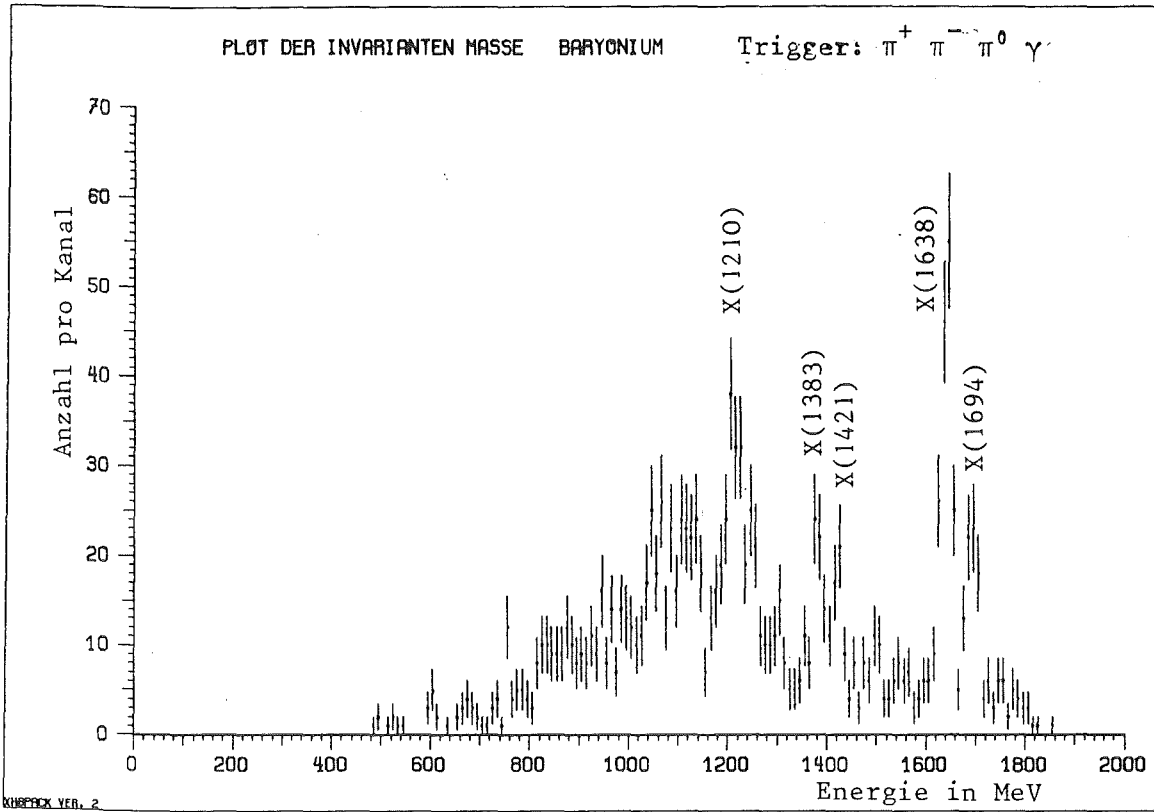


Abbildung 23. Invariante Masse hypothetischer Baryoniumzustände

2.4.2.2. GLUE - BALLS

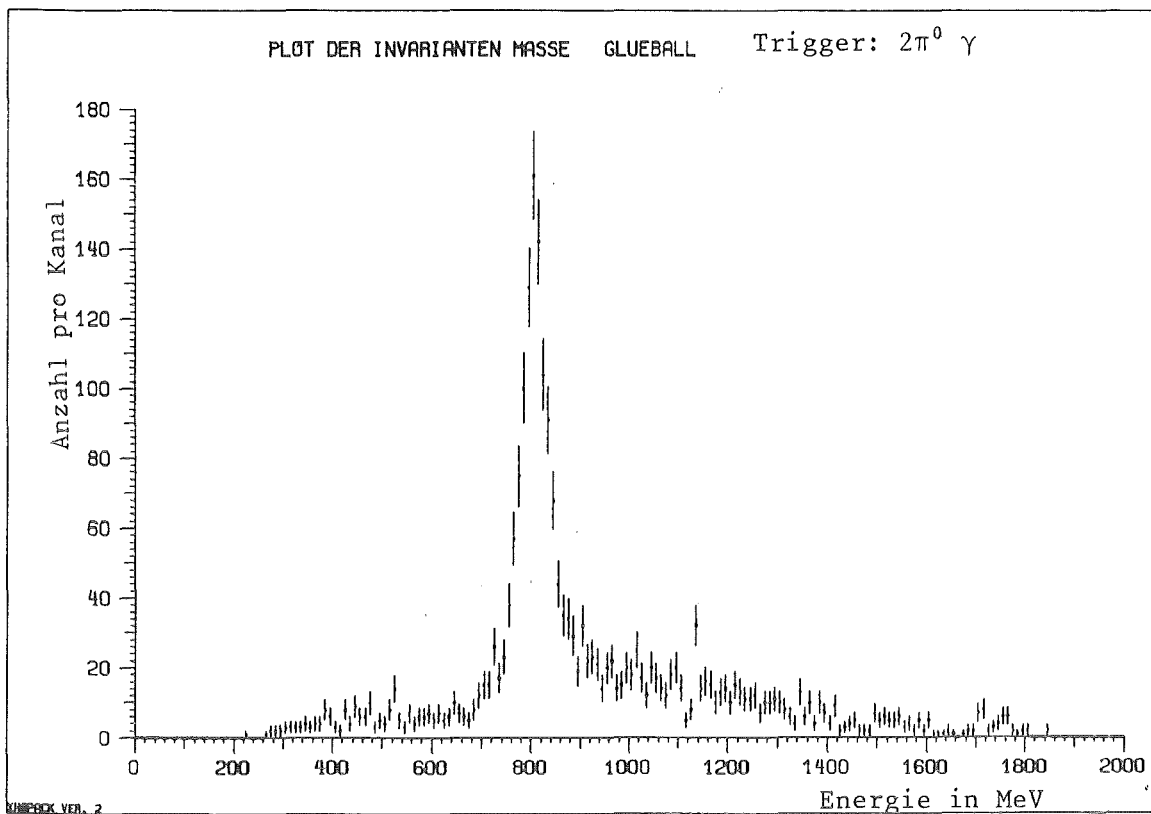
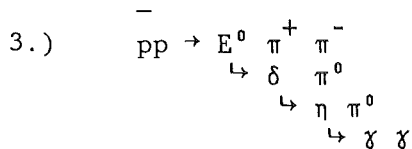
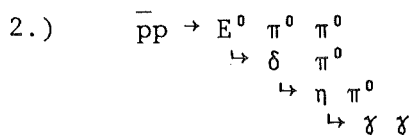
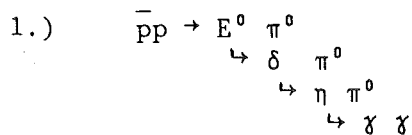


Abbildung 24. Invariante Masse des radiativ zerfallenden Glueballs

Wie aus dem invarianten Massenspektrum des hypothetischen Glueballzerfalls ($800\text{MeV}, \Gamma=20\text{MeV}$) in zwei neutrale Pionen (ID=68) ersichtlich ist, erlaubt ein Detektorsystem der hier vorgeschlagenen Art auch den Nachweis seltener breiter Resonanzen (Abbildung 24) .

Einige Kandidaten für solche Zustände sind im Massenbereich zwischen $1400\text{-}2200$ MeV vorwiegend in radiativen Ψ -Zerfällen gefunden worden, wie z.B. $i/E^0(1418, \Gamma=52\text{MeV})$ /FRAN83/.

Der komplizierteste simulierte Prozess ist der Zerfall des E^0 (ID=70,71,72), welches aus der Proton - Antiproton Annihilation unter Aussendung eines oder mehrerer geladener bzw. neutraler Pionen hervorgeht:



Im rein radiativen Fall (2) mit zehn Gammas im Endzustand zeigt Abbildung 26, daß selbst hier über einem kombinatorischen Untergrund das breite E^0 im Spektrum der invarianten $\delta \pi^0$ Masse sehr deutlich zu sehen ist.

Der kombinatorische Untergrund rührt daher, daß Kombinationen der neutralen Pionen, die ausschließlich vom zerfallenden resonanten Teilchen stammen, ununterscheidbar sind von Kombinationen mit neutralen Pionen, die die Annihilation direkt beiträgt.

Der rein radiative Zerfall (1) des E^0 mit acht Gammas im Endzustand ist um einiges schwieriger nachzuweisen, da unglücklicherweise das Maximum des kombinatorischen Untergrunds im Spektrum der invarianten $\delta \pi^0$ Masse mit dem gesuchten Teilchen zusammenfällt (Abbildung 25).

Zum Nachweis am besten geeignet erscheint der Zerfall des E^0 (3) mit geladenen Pionen im Endzustand. Hier ist im Spektrum der invarianten $\delta \pi^0$ Masse kein kombinatorischer Untergrund zu erwarten, was auf Abbildung 27 deutlich zu sehen ist.

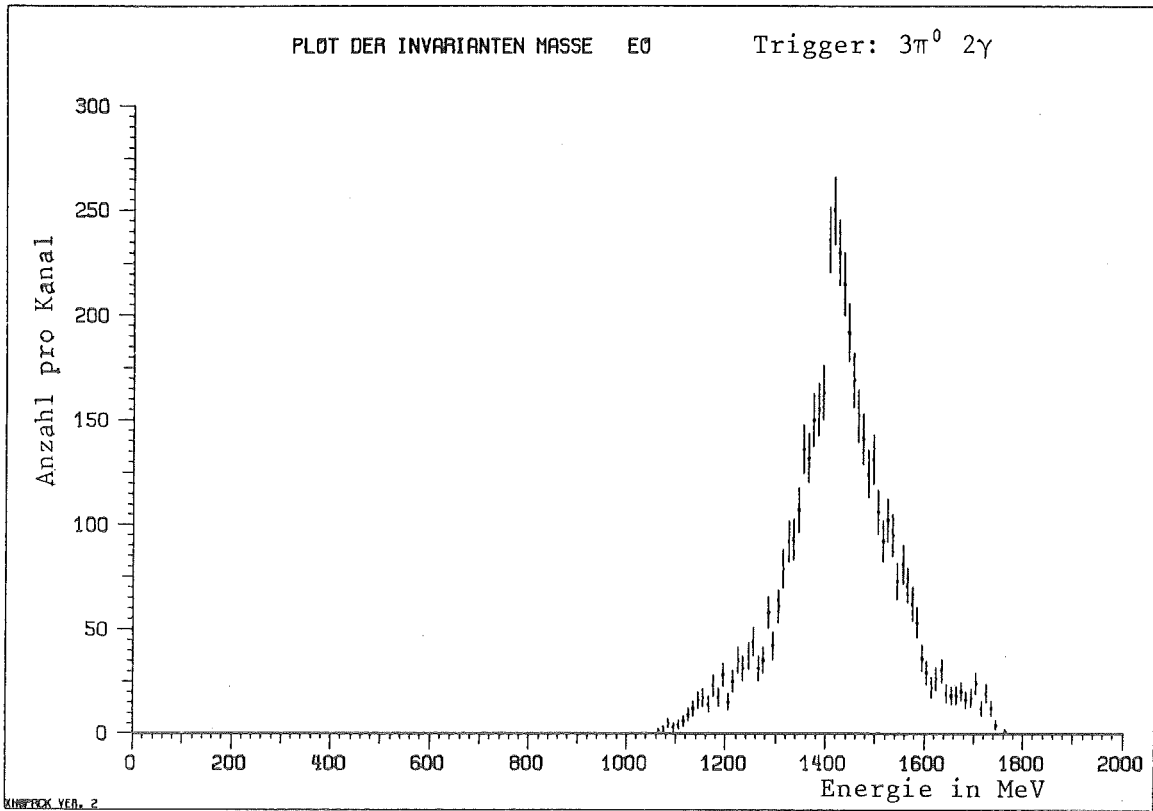


Abbildung 25. Invariante Masse des E(1418) im Kanal ID=70

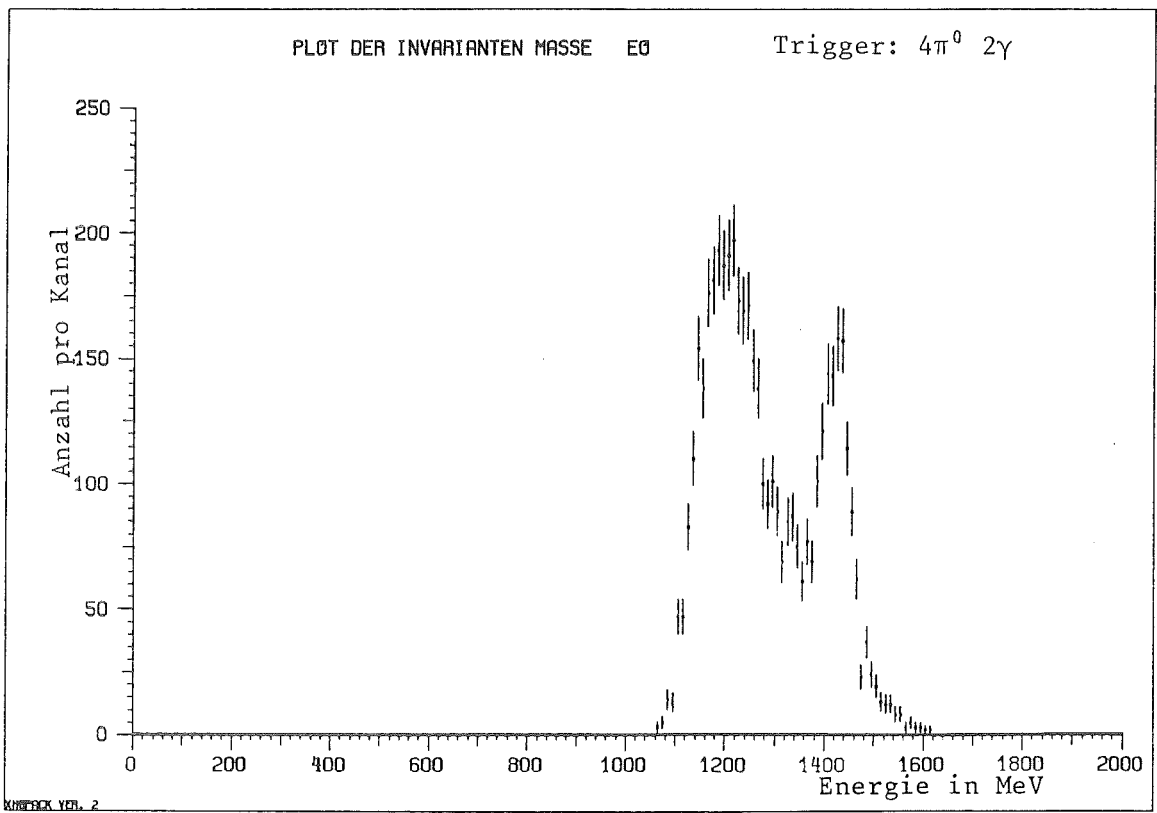


Abbildung 26. Invariante Masse des E(1418) im Kanal ID=71

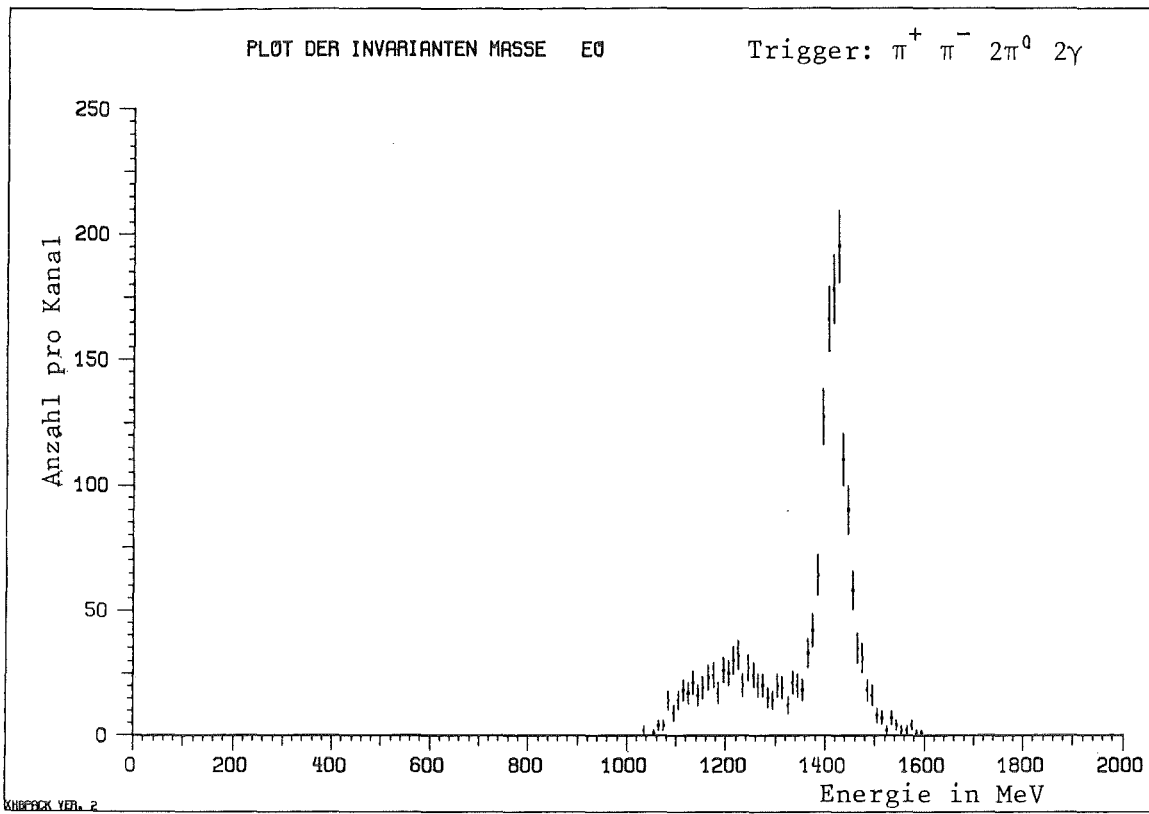


Abbildung 27. Invariante Masse des E(1418) im Kanal ID=72

2.5.VARIATION DER PARAMETER

Das hier vorgestellte Detektorsystem stellt das Ergebnis von mit dem Monte - Carlo Computersimulationsprogrammpaket 'DETEKTOR' durchgeführten Optimierungsrechnungen unter Berücksichtigung der technischen Realisierbarkeit dar.

Parallel hierzu wurden Rechnungen angestellt, in denen die Auswirkungen systematischer Veränderungen der optimalen Detektorparameter auf das Nachweisvermögen des Systems studiert wurden:

2.5.1.Gammadetektor

2.5.1.1.VARIATION DER MODULARITÄT

Die Modularität des 'STANDARD-DETEKTOR-SYSTEMS' war festgesetzt auf 1647 Kristallmodule, die zur Kugel zusammengefügt den Gammadetektor bildeten.

Das Zusammenfügen eines Detektorsystems aus Einzelmodulen entspricht einer Quantelung des Raumwinkels beim Nachweis von Gammas.

Da nun in die Bildung der invarianten Massen der neutralen Pionen die gemessenen Zwischenwinkel zwischen den Gammas eingehen, sollte eine Änderung der Modularität des Gammadetektors z.B. auf das Gesamtenergiespektrum der rekonstruierten neutralen Pionen Auswirkungen zeigen.

Bei einer sukzessiven Verringerung der Modularität tritt eine zunehmende Entstellung des Gesamtenergiespektrums der neutralen Pionen auf, von einer leichten Welligkeit bei 1000 Einzelmodulen bis zur völligen Deformation bei nur 45 Einzelmodulen, wie Abbildung 28 bis Abbildung 31 auf eindruckliche Weise belegen.

Verschlechterungen der Ergebnisse treten in gleicher Weise auch in den Spektren der invarianten Massen auf, bei deren Rekonstruktion eines oder mehrere Gammas beteiligt sind.

Um eine aus Gründen zu geringer Modularität evtl. auftretende Welligkeit der Spektren auszugleichen, kann man daran denken, die Quantelung des Raumwinkels durch die Anwendung einer 'stochastischen Verschmierung' der Winkel aufzuheben, wobei die diskreten Winkel nach einer stochastischen Verteilungsfunktion verwürfelt werden.

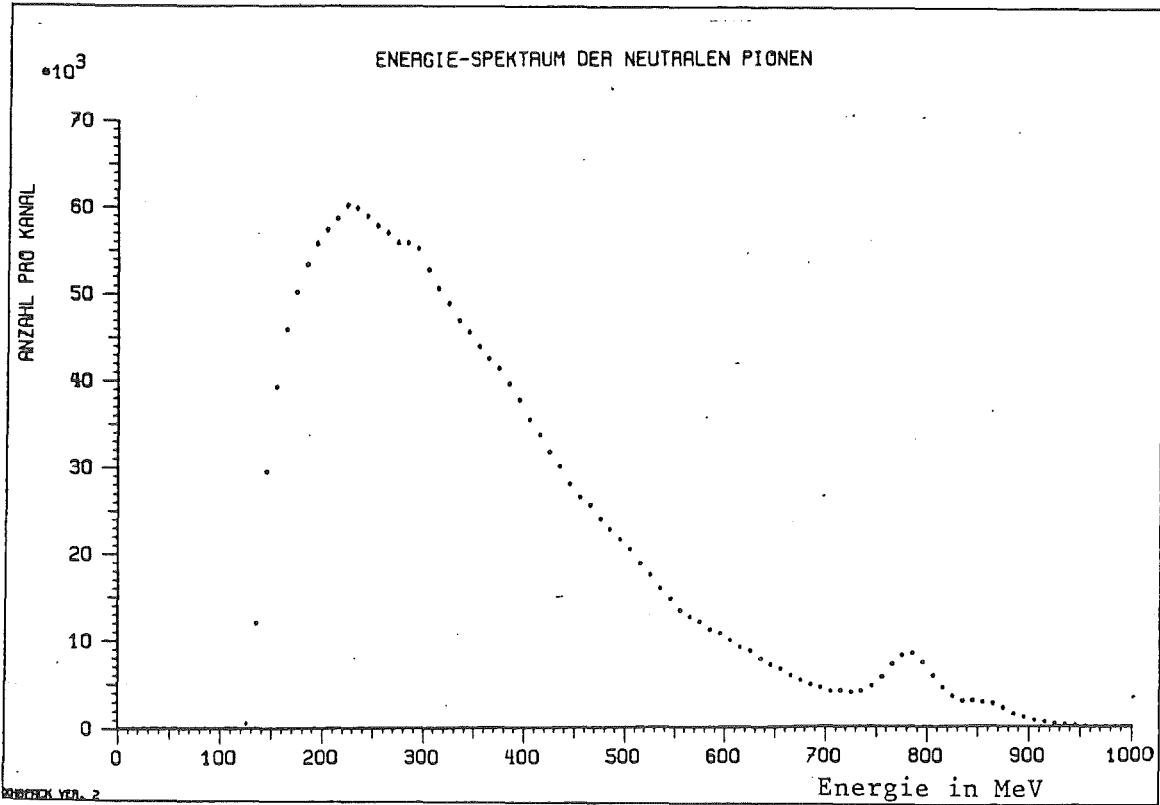


Abbildung 28. Energiespektrum der rekonstruierten neutralen Pionen (Modularität: 1000)

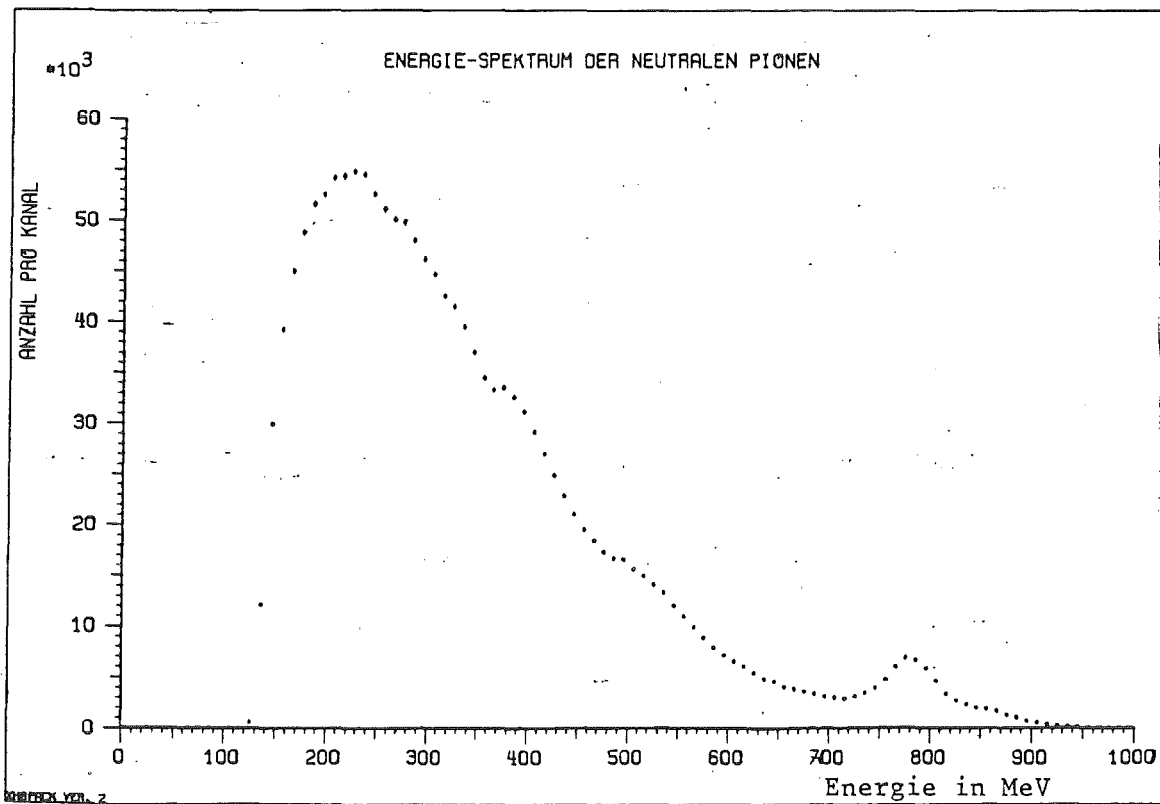


Abbildung 29. Energiespektrum der rekonstruierten neutralen Pionen (Modularität: 600)

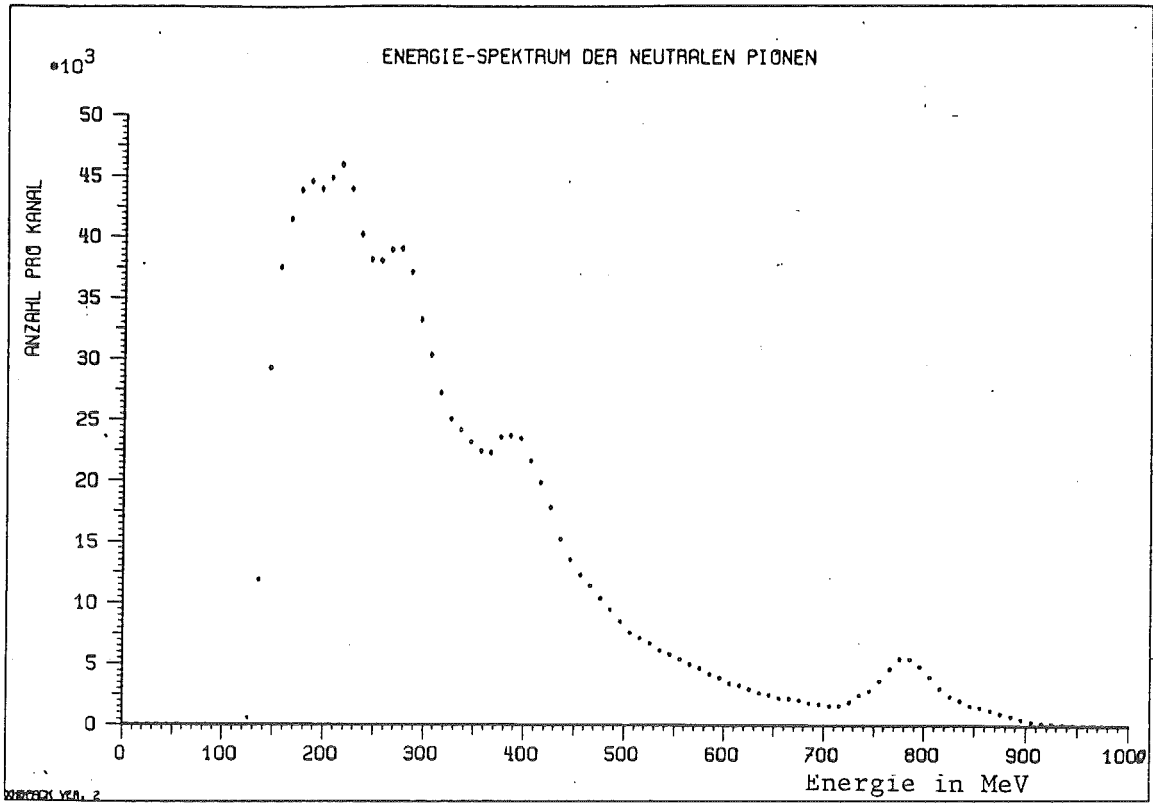


Abbildung 30. Energiespektrum der rekonstruierten neutralen Pionen (Modularität: 150)

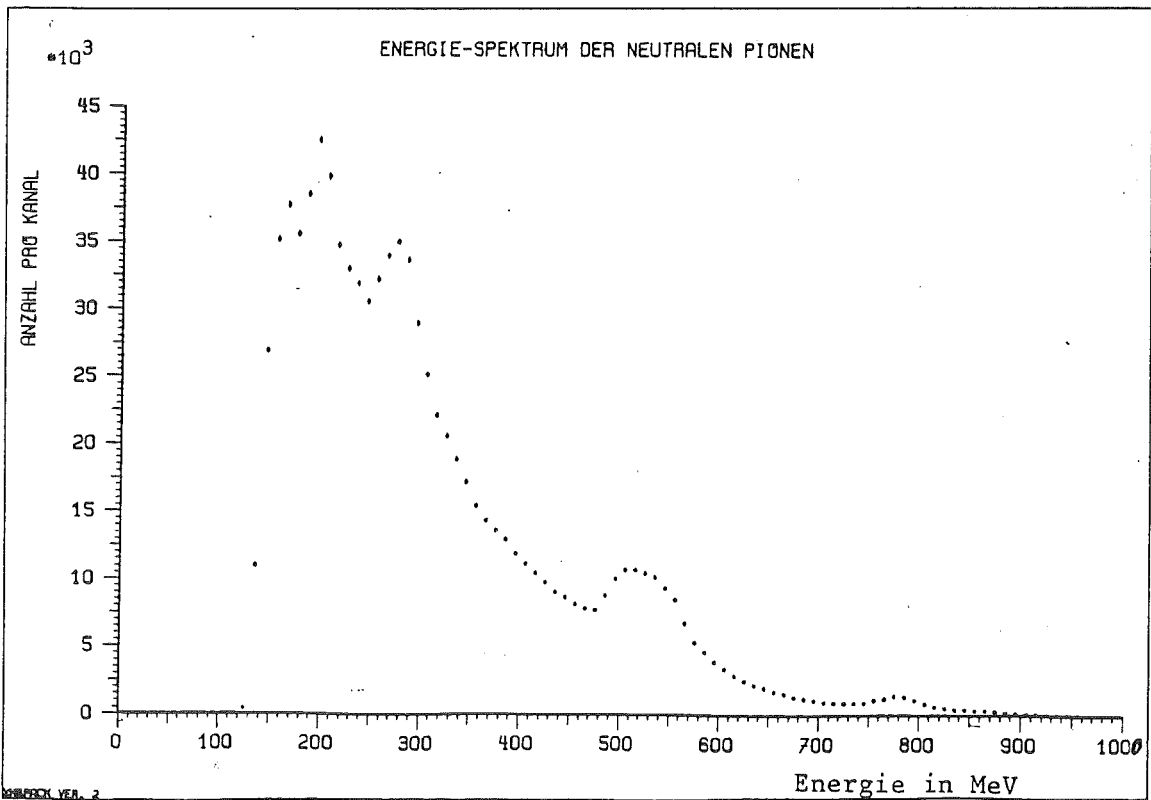
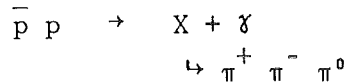


Abbildung 31. Energiespektrum der rekonstruierten neutralen Pionen (Modularität: 45)

2.5.1.2.VARIATION DER ENERGIEAUFLÖSUNG

Würde anstelle der Energieauflösung für Gammas von 7% bei 100 MeV ein Wert von 25% bei 100 MeV angesetzt, was in etwa dem Auflösungsvermögen von Bleiglas entspricht, so würde aufgrund der größeren Energieverschmierung der gemessenen Gammas eine erfolgreiche Energiesatzüberprüfung erschwert und damit das sichere Erkennen eines vollständigen Annihilationskanals unmöglich.

Um die Auswirkungen einer schlechteren Energieauflösung für Gammas auf das Nachweisvermögen des 'STANDARD-DETEKTOR-SYSTEMS' zu untersuchen, wurde das Beispiel eines hypothetischen Baryoniumzerfalls gewählt:



Für den Nachweis dieser seltenen Reaktion bedeutet eine Verschlechterung der Energieauflösung eine Verbreiterung der Linien.

Außerdem trägt der Annihilationskanal $\pi^+ \pi^- 2\pi^0$ (ID=1) in verstärktem Maße zur Untergrundbildung bei (Abbildung 32).

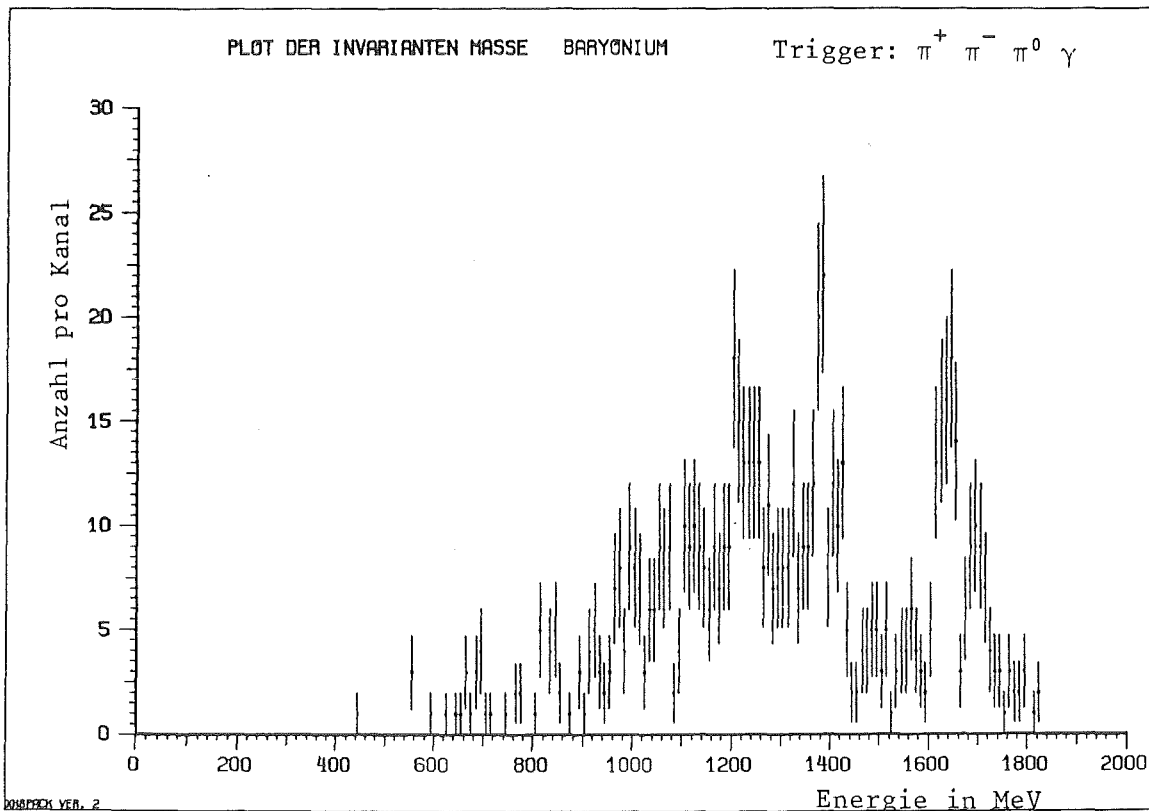
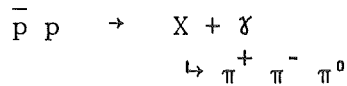


Abbildung 32. Verringerung der Energieauflösung auf 25%

2.5.1.3. VARIATION DES RAUMWINKELS

Eine Verringerung der Flächenbedeckung des Gammadetektors hat zur Folge, daß nur noch ein Bruchteil der aus der Proton - Antiproton Annihilation hervorgehenden Gammas nachgewiesen werden kann.

Um die Auswirkungen einer geringeren Flächenbedeckung auf das Nachvermögen des 'STANDARD-DETEKTOR-SYSTEMS' zu untersuchen, wurde wiederum das Beispiel des hypothetischen Baryoniumzerfalls gewählt:



Auch in diesem Fall trägt der Annihilationskanal $\pi^+ \pi^- 2\pi^0$ (ID=1) aufgrund der größeren Wahrscheinlichkeit des Entweichens eines niederenergetischen Gammaquants in verstärktem Maße zur Untergrundbildung bei (Abbildung 33, Abbildung 34).

Aus den betrachteten Beispielen kann man schließen, daß die Erfassung eines möglichst großen Raumwinkelbereichs ($\Omega > 90\%$) für die Durchführung eines kinematisch vollständigen Experimentes hoher Sensitivität grundlegend wichtig ist.

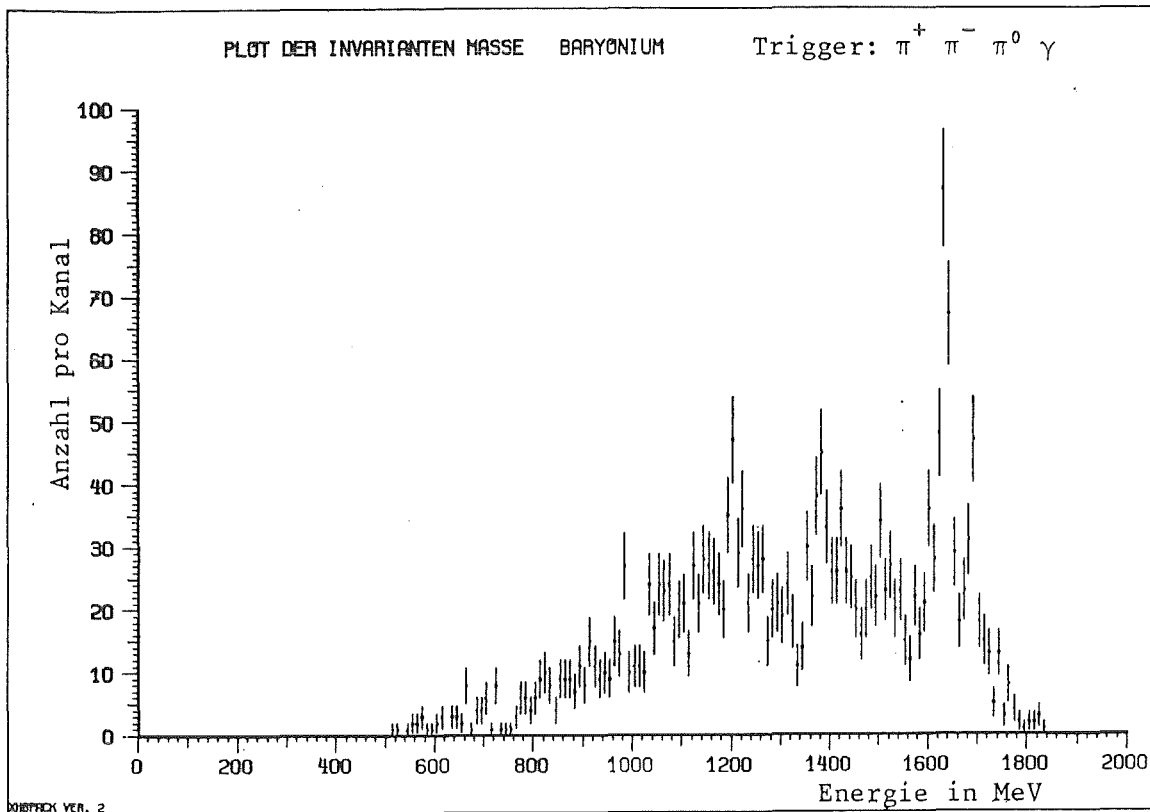


Abbildung 33. Verringerung der Flächenbedeckung auf 95%

2.5.2. Driftkammer

2.5.2.1. VARIATION DER IMPULSAUFLÖSUNG

Eine Verringerung der Impulsauflösung für geladene Teilchen führt prinzipiell dieselben Änderungen in den exklusiven Spektren der invarianten Masse mit sich, wie dies bei der Verringerung der Energieauflösung für neutrale Teilchen der Fall war. Für das Spektrum der invarianten Masse des hypothetischen Baryoniumzerfalls gilt auch hier das dort gesagte (Abbildung 35).

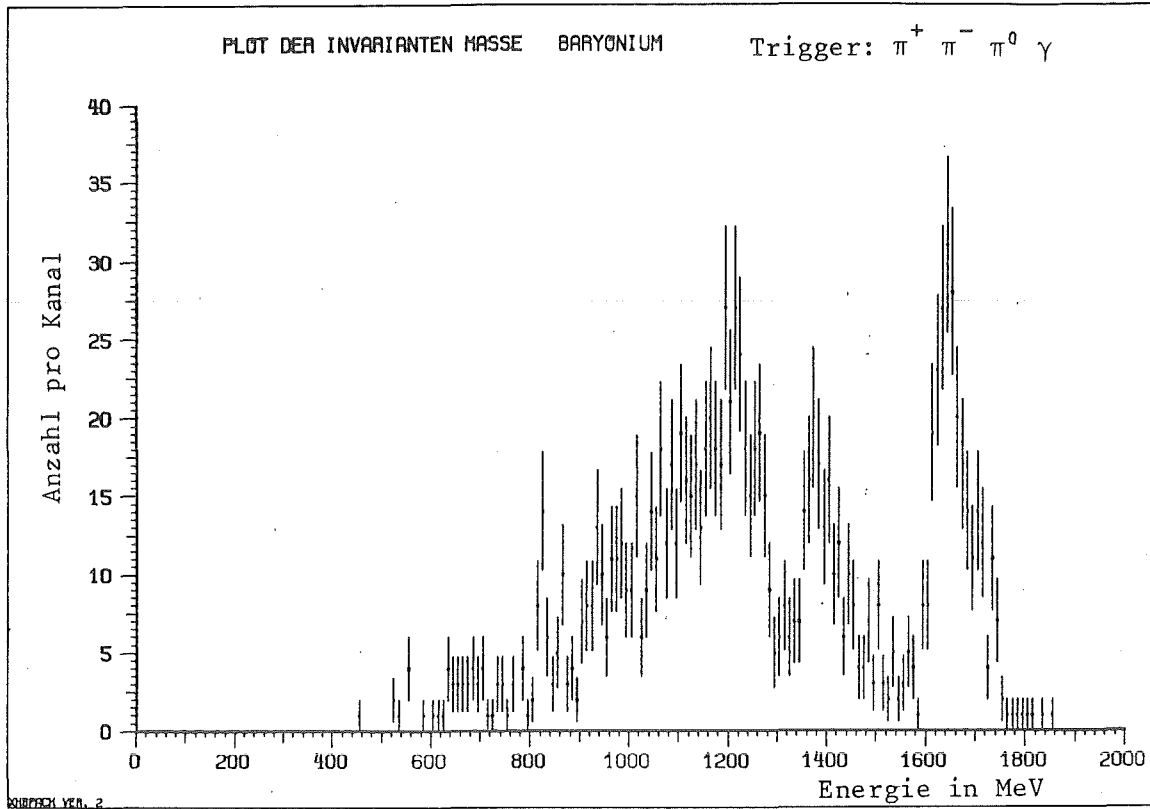


Abbildung 35. Verringerung der Impulsauflösung auf 15%

2.6. ALTERNATIVE LÖSUNGEN

2.6.1. Detektorsystem ohne Magnetfeld

Alternativ zu dem hier vorgestellten 'STANDARD-DETEKTOR-SYSTEM' wurde untersucht, welche Eigenschaften ein entsprechendes Detektorsystem ohne Magnetfeld besitzt. Als Parametersatz wurden die Größen des 'STANFORD-CRYSTAL-BALL' gewählt. Dieser Detektor besteht aus ca. 700 NaJ(Tl)-Modulen, bedeckt einen Raumwinkel von ca. 93% - 95% und hat zur Zeit eine Energieauflösung von 10.3% (FWHM) bei 130 MeV.

Wegen der Lichtauslese mit Photomultipliern (Diese können aus Gründen der Lichtintensität nicht durch Photodioden ersetzt werden) läßt er sich nicht in einem Magnetfeld betreiben.

Es wurde dieselbe Driftkammer 'installiert', wie dies bei den Simulationen des 'STANDARD-DETEKTOR-SYSTEMS' der Fall war, doch sind von den geladenen Teilchen wegen des fehlenden Magnetfeldes nur die Richtungskosinusse meßbar, nicht aber Ladungsvorzeichen und der Impulsbetrag.

Die fehlenden Impulsbeträge können für maximal vier geladenen Teilchen aus der Forderung nach Erfüllung der Energieerhaltung und der Impulserhaltung gewonnen werden:

$$(15) \quad \sum_{i=1}^n \vec{P}_i = 0$$

$$(16) \quad \sum_{i=1}^n E_i = 1877 \text{ MeV}$$

Sind c_x , c_y und c_z die aus der Messung bekannten Richtungskosinusse des i -ten geladenen Teilchens im Bezug auf die x, y und z -Achse, so erhält man aus (15) und (16) ein Gleichungssystem zur Bestimmung der Impulsbeträge:

$$\text{I) } c_{x_1} |\vec{P}_1| + c_{x_2} |\vec{P}_2| + c_{x_3} |\vec{P}_3| + c_{x_4} |\vec{P}_4| = 0$$

$$\text{II) } c_{y_1} |\vec{P}_1| + c_{y_2} |\vec{P}_2| + c_{y_3} |\vec{P}_3| + c_{y_4} |\vec{P}_4| = 0$$

$$\text{III) } c_{z_1} |\vec{P}_1| + c_{z_2} |\vec{P}_2| + c_{z_3} |\vec{P}_3| + c_{z_4} |\vec{P}_4| = 0$$

$$\text{IV) } \left[m_\pi^2 c^2 + |\vec{P}_1|^2 \right]^{\frac{1}{2}} + \left[m_\pi^2 c^2 + |\vec{P}_2|^2 \right]^{\frac{1}{2}} + \left[m_\pi^2 c^2 + |\vec{P}_3|^2 \right]^{\frac{1}{2}} + \left[m_\pi^2 c^2 + |\vec{P}_4|^2 \right]^{\frac{1}{2}} = 1877 \text{ MeV}/c$$

Die Lösungen dieses nicht linearen Gleichungssystems werden durch Anwendung eines iterativen Verfahrens im Unterprogramm 'MOMENT' ermittelt /AMS84/.

Abbildung 36 zeigt die der Abbildung 21 entsprechende Situation:

Während im Kanal mit zwei geladenen Pionen ($\pi^+ \pi^- \pi^0$) die ρ^0 -Rekonstruktion weiterhin recht gut funktioniert, werden die Ergebnisse für Annihilationskanäle, in denen vier geladene Pionen auftreten sehr viel schlechter.

Die ω -Resonanz (Abbildung 37) wird wesentlich breiter und liegt auf einem höheren Untergrund. Das liegt daran, daß hier alle vier kinematischen Gleichungen zur Bestimmung der vier fehlenden Impulse ausgenutzt werden und auf diese Art und Weise keiner der Erhaltungssätze mehr zur Vollständigkeitsüberprüfung für das betrachtete Ereignis herangezogen werden kann.

Wie Abbildung 38 zeigt (entspricht Abbildung 23 auf Seite 37) wird selbst für den Fall der Rekonstruktion nur zweier geladener Pionen bei sehr seltenen Kanälen die Massenrekonstruktion gegenüber dem 'STANDARD-DETEKTOR-SYSTEM' mit Magnetfeld zur Impulsmessung der geladenen Teilchen wesentlich schlechter.

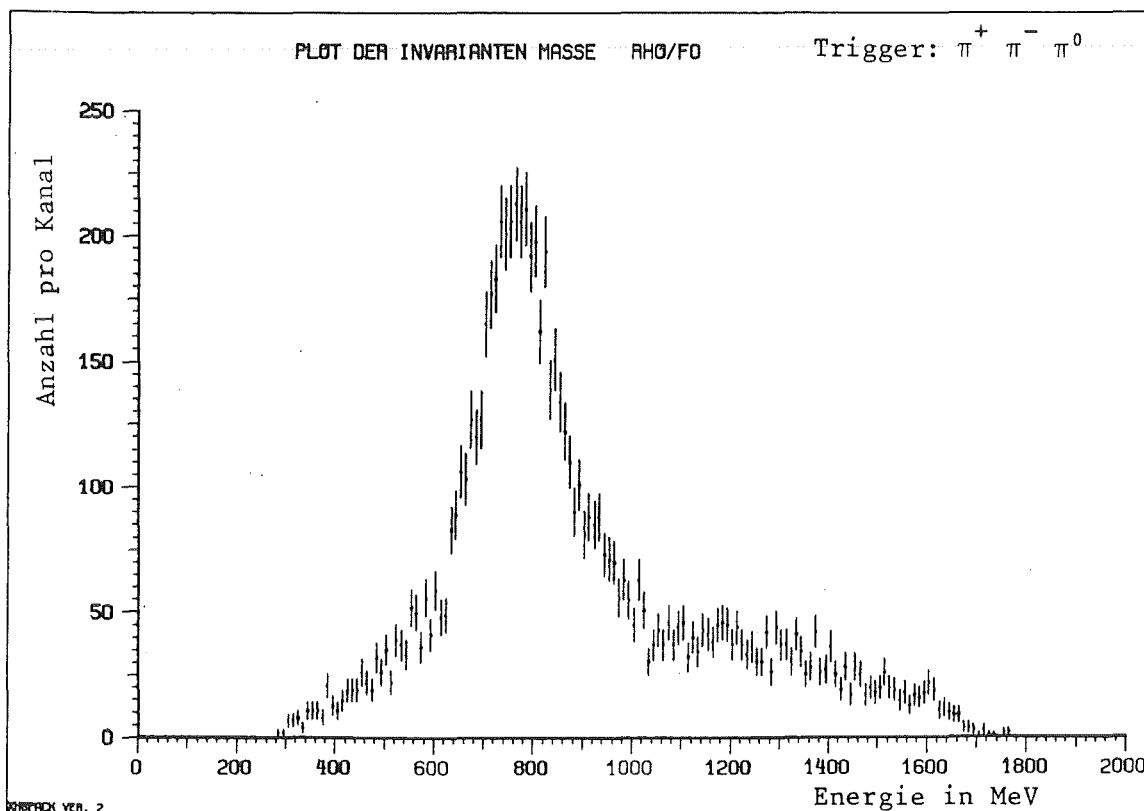


Abbildung 36. Invariante Masse des neutralen Rho (Detektor ohne Magnetfeld)

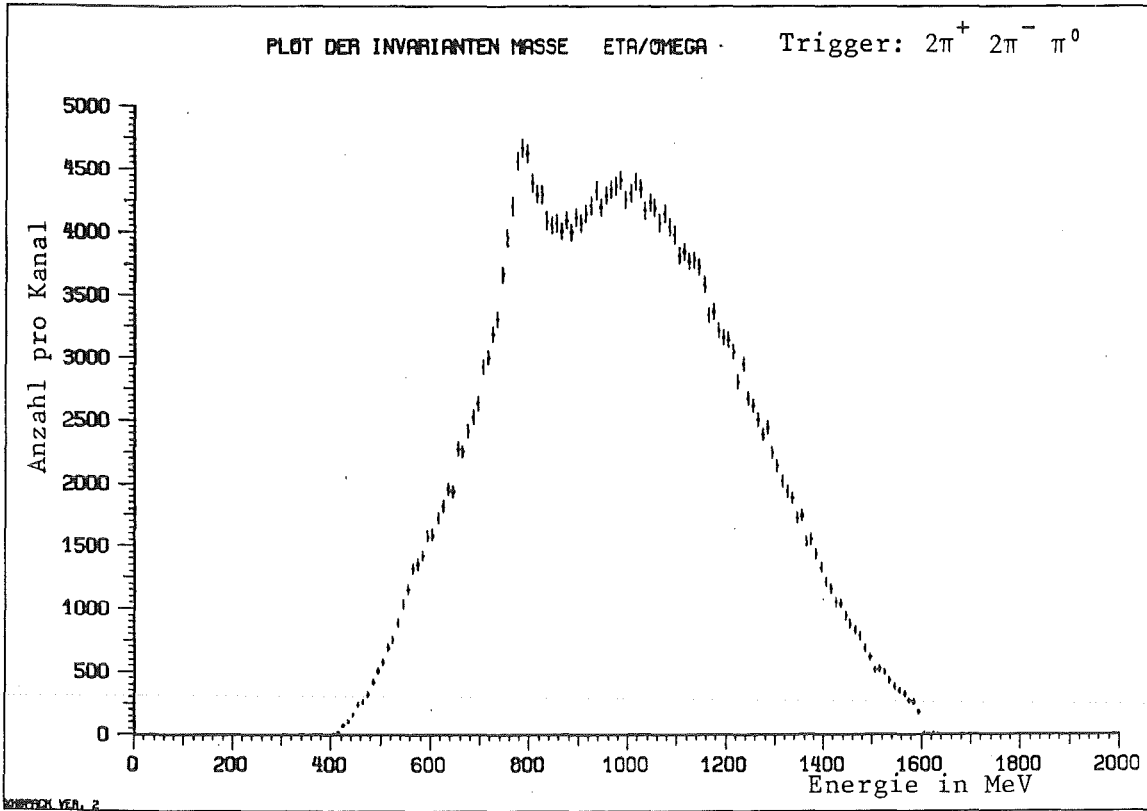


Abbildung 37. Invariante Masse des Omega (Detektor ohne Magnetfeld)

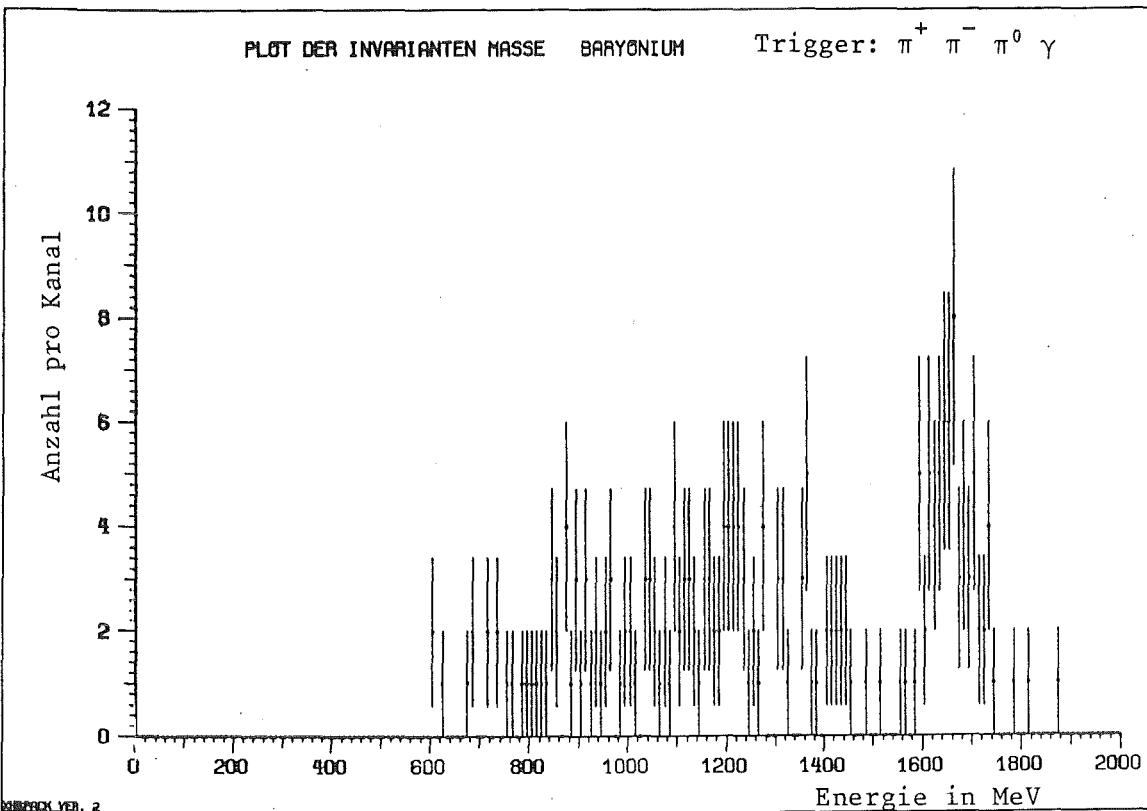


Abbildung 38. Invariante Masse des hypothetischen Baryoniums (Detektor ohne Magnetfeld)

2.7. SCHLUSSWORT UND AUSBLICK

In der vorliegenden Arbeit wird die Monte - Carlo Computersimulation eines Detektorsystems an einer Rechenanlage beschrieben und deren Ergebnisse diskutiert.

Durch Variation der Parameter der verschiedenen Detektorkomponenten (modularer Cs-J-Detektor, zylindrische Driftkammer) im Rahmen technisch realisierbarer Grenzen wurde die Geometrie des Detektorsystems dahingehend optimiert, daß ein kinematisch vollständiges Experiment zur Untersuchung der Proton - Antiproton Annihilation durchgeführt werden kann.

Inbesondere zeigte sich bei Durchführung der Computersimulationen, daß seltene hypothetische Zwischenzustände, wie z.B. Glueballs, Baryoniumzustände usw. mit sehr hoher Empfindlichkeit nachgewiesen werden können.

Die hier beschriebenen Rechnungen bildeten die Grundlage zur Erstellung eines 'Letter of Intent' für den Bau eines 4π -Detektorsystems zur Untersuchung der Proton - Antiproton Wechselwirkung am LEAR im CERN.

Selbstverständlich kann ein Monte - Carlo Computersimulationsprogramm die experimentelle Wirklichkeit nur näherungsweise nachbilden, da die programmtechnische Verwirklichung eines reellen Parameters Rechenzeit kostet und bei jedem Programmlauf mehrere Millionen Ereignisse verarbeitet werden müssen.

Das Programm-Modell 'DETEKTOR' benötigt in der vorliegenden Version bei der Simulation des 'STANDARD-DETEKTOR-SYSTEMS' (Statistik: 3.2 Millionen Annihilationsereignisse) ca. 1.5 Stunden Rechenzeit auf einer IBM 3081/Siemens 7890 Rechenanlage. Die Statistik ist hierbei zum einen begrenzt durch die Rechenzeit, zum andern durch die Bereitschaft des Bedienungspersonals, Magnetbänder zu montieren.⁸

⁸ Die Datenmenge für 3.200.000 Proton - Antiproton Annihilationen beträgt ca. 700 Megabyte und füllt 4 Magnetbänder (6250 BpI).

Aufgrund der konsequenten Einhaltung des Prinzips der 'Strukturierten Programmierung' sind die Programmpakete modular aufgebaut, so daß es keine Schwierigkeiten bereiten sollte, Ergänzungen und Erweiterungen durchzuführen:

- o So ist zum Beispiel daran gedacht, die Annihilationsmechanismen zu erweitern und die Proton - Antiproton Annihilation im Fluge zuzulassen, was die Vielzahl der Annihilationskanäle nicht unbeträchtlich bereichern würde.
- o Das Separationsvermögen des Detektorsystems für Kaonen könnte getestet werden (π/K -Separation)
- o Eine weitere Verbesserung des Programmpakets 'DETEKTOR' könnte darin bestehen, wie bei der Auswertung von experimentellen Daten üblich, einen kinematischen Fit der Meßwerte durchzuführen.
- o Wenn das Aufzeichnungsformat der im späteren Experiment anfallenden Daten bekannt wäre, könnten noch vor Beginn des Experimentes Computersimulationen durchgeführt werden, mit dem Ziel, Auswerteprogramme zu erstellen.

Abschließend sei bemerkt, daß trotz einiger Vereinfachungen der physikalischen Realität das vorliegende Modell es nicht nur erlaubt, die Leistungsfähigkeit eines modernen realen Detektorsystems abzuschätzen, sondern sogar bei der Konzeption und der Auswertung von Experimenten eingesetzt werden kann.

ANHANG A. PROGRAMMPAKET BIGBANG

Das Programmpaket 'BIGBANG' wurde in der höheren Programmiersprache FORTRAN77 geschrieben und an der IBM 3081/Siemens 7890 Grossrechenanlage des Rechenzentrums im Kernforschungszentrum Karlsruhe getestet. Es generiert die Annihilationskinematiken der Proton - Antiproton Annihilation im Proton - Antiproton Ruhesystem.

PROGRAMMABLAUF

Im Hauptprogramm 'CREATE' werden zunächst Variablen deklariert und mit Default- bzw. Startwerten versehen. Im Anschluß daran erfolgt die Initialisierung der Histogramme (Verwendung des Unterprogrammpakets HBOOK (CERN)).

In der Hauptschleife, die so oft durchlaufen wird, bis entweder die gewünschte Anzahl der Zerfälle erreicht ist oder keine Rechenzeit mehr zur Verfügung steht, wird das Unterprogramm 'EVENT' aufgerufen, welches in quasi zufälliger Reihenfolge die Kinematik der verschiedenen Annihilationskanäle produziert.

Daraufhin wird untersucht, ob es sich bei dem erzeugten Ereignis um einen sog. Baryoniumzustand handelt. Ist dies der Fall, so wird die Kinematik einer weiteren Annihilation bereitgestellt, eine Lorentztransformation (Unterprogramm 'LORTRN') der Daten vom Ruhesystem des Baryoniums ins Ruhesystem der Proton - Antiproton Annihilation durchgeführt, und das Baryonium durch diese Teilchen ersetzt. Da sich die Wahrscheinlichkeiten der Zerfälle bei Anwendung dieses Verfahrens multiplizieren, werden auf diese Art und Weise äußerst seltene hypothetische Zerfälle von tiefgebundenen Zuständen simuliert.

Im Anschluß daran werden die erzeugten Teilchen sortiert, die Ergebnisse ausgedruckt, die Histogramme gefüllt und die Daten in eine Magnetbanddatei ausgegeben.

Wird die Hauptschleife verlassen, erfolgt die Ausgabe und das Abspeichern der Histogramme, sowie das Ausdrucken eines Abschlußberichts.

EINGABEDATEN

Mit dem Programmpaket 'BIGBANG' ist es dem Benutzer möglich, die Kinematik von Zerfällen beliebiger Art zu erzeugen. So kann z.B. auch eine beliebige Kette ineinander zerfallender Resonanzen berechnet werden, vorausgesetzt, diese Kette terminiert (Oszillationen z.B. können nicht berechnet werden). Nicht berücksichtigt bleiben im Programmpaket auch spinabhängige Vorgänge.

Unter Berücksichtigung dieser Tatsachen erstreckt sich die Kontrolle des Benutzers auf folgende Eingabedaten:

Namelist 'INPUT'

Die Parameter zur Steuerung des Programmablaufs werden durch die Namelist 'INPUT' übergeben. Folgende Parameter sind definiert:

TECM	Gesamtenergie der Annihilation im Ruhesystem (MeV)
BARDEC	Baryoniumzerfall soll gerechnet werden (Logical)
MAXGEN	Anzahl der zu erzeugenden Annihilationen
LOOP	Startwert für den Zufallszahlengenerator
ISTORE	Dateinummer unter der die Histogramme abgelegt werden
LUNOUT	Dateinummer der Magnetbanddatei
NLIST	Zahl der zu listenden Ereignisse ('Event-Dump')

Es schließt sich die Eingabe der einzelnen Annihilationskanäle sowie deren Häufigkeiten an. In freiem Format werden die Kenn-Nummer des Kanals und dessen Verzweigungsverhältnis angegeben. Auf der nächsten 'Karte' folgen die Namen der beteiligten Teilchen im Format (16A5). Hierbei können die einzelnen Annihilationskanäle in beliebiger Reihenfolge aufeinanderfolgen. Tritt im Laufe der Programmausführung der Fall ein, daß ein Teilchen nicht erkannt wird, wird ein Hinweis darauf ausgedruckt.

Ein Beispiel für eine Benutzereingabe könnte folgendermaßen aussehen:

```
&INPUT
TECM=1876.,BARDEC=.TRUE.,MAXGEN=100000,
LUNOUT=1,NLIST=50
&END
1 9.3
PI+ PI- PIO PIO
2 23.3
PI+ PI- PIO PIO PIO
.
.
72 0.2
E0 PI+ PI-
```

Folgende Teilchennamen werden bislang vom Programm erkannt und im Rahmen dessen richtig weiterverarbeitet:

```
GAMMA,PIO ,PI+ ,PI- ,RHOO ,RHO- ,RHO+ ,FO ,OMEGA,ETA ,ETA" ,
KO ,K+ ,K- ,K1 ,K2 ,KO* ,K*0 ,K*+ ,K*- ,A1- ,A2+ ,
A2- ,B+ ,PHI ,E0 ,DELTA,GLUE ,B1694,B1638,B1561,B1447,B1421,
B1383,B1210
```

Einfügen und Ändern von Annihilationskanälen

Das Einfügen oder Ändern von Annihilationskanälen kann beim jetzigen Stand des Programmpakets nur über eine Programmänderung vorgenommen werden. Die Annihilationskanäle sind im Unterprogramm 'GETDAT' definiert und können auf einfache Art und Weise ergänzt werden:

Am Beispiel des Zerfalls des neutralen Pions soll dies gezeigt werden:

```
IF (TYP(I).EQ.'PIO ') THEN
  BARYON(I) = .FALSE.           ; Kein Baryoniumzustand
  AMASS(I)  = 134.9645         ; Masse des neutralen Pions in MeV
  GAMMA(I)  = 0.0              ; Breite des neutralen Pions in MeV
  NRES(I)   = 2                ; Zahl der resonanten Teilchen
  RTYP(1,I) = 'GAMMA'          ; Erstes Teilchen ist ein Gamma
  RESPAR(1,I) = 0.0            ; Dessen Breite ist Null
  RTYP (2,I) = 'GAMMA'         ; dto.
  RESPAR(1,I) = 0.0            ; dto.
ENDIF
```

Selbstverständlich muß dafür Sorge getragen werden, daß alle auftretenden Teilchen definiert sind.

EVENTDUMP

Beim Entwurf des Programmpakets 'BIGBANG' wurde darauf Wert gelegt, den Verlauf der Generierung der Annihilationskanäle für den Benutzer übersichtlich darzustellen. Durch Angabe des 'NLIST'-Parameters erhält man die folgende Druckerliste:

```

*****
# EVENT NO.      7      CHANNEL NO. 71      NUMBER OF PARTICLES: 3      TOTAL ENERGY (CM): 1876.56      #
*****
EO      1514.38918 | PX = 21.2239197 | PY = -107.8362818 | PZ = -120.0969014 | E = 1523.1142397 | P = 162.7955287
PIO     134.96449 | PX = 54.2838879 | PY = 12.4163745 | PZ = 83.1161891 | E = 168.0018513 | P = 100.0460231
PIO     134.96449 | PX = -75.5078088 | PY = 95.4199089 | PZ = 36.9807129 | E = 185.4437264 | P = 127.1768899
-----
SUM:      -0.0000012      0.0000015      0.0000006      1876.5598174

RESONANT PARTICLE DETECTED : EO
DECAYS INTO: DELTA PIO

RESONANT PARTICLE DETECTED : PIO
DECAYS INTO: GAMMA GAMMA

RESONANT PARTICLE DETECTED : PIO
DECAYS INTO: GAMMA GAMMA

SCAN NO.2
DELTA    983.00000 | PX = -16.0080530 | PY = -459.7037451 | PZ = 91.7824939 | E = 1089.1725379 | P = 469.0499091
PIO     134.96449 | PX = 37.2319699 | PY = 351.8674775 | PZ = -211.8793795 | E = 433.9415019 | P = 412.4194621
GAMMA    0.0      | PX = 24.9192253 | PY = -52.8762664 | PZ = 0.1990608 | E = 58.4543152 | P = 58.4543152
GAMMA    0.0      | PX = 29.3646622 | PY = 65.2926408 | PZ = 82.9171277 | E = 109.5475349 | P = 109.5475349
GAMMA    0.0      | PX = -62.0021991 | PY = 74.2751879 | PZ = 86.5956935 | E = 129.8456405 | P = 129.8456405
GAMMA    0.0      | PX = -13.5056092 | PY = 21.1447203 | PZ = -49.6149809 | E = 55.5980845 | P = 55.5980845
-----
SUM:      -0.0000039      0.0000149      0.0000155      1876.5596150

RESONANT PARTICLE DETECTED : DELTA
DECAYS INTO: ETAO PIO

RESONANT PARTICLE DETECTED : PIO
DECAYS INTO: GAMMA GAMMA

SCAN NO.3
GAMMA    0.0      | PX = 24.9192253 | PY = -52.8762664 | PZ = 0.1990608 | E = 58.4543152 | P = 58.4543152
GAMMA    0.0      | PX = 29.3646622 | PY = 65.2926408 | PZ = 82.9171277 | E = 109.5475349 | P = 109.5475349
GAMMA    0.0      | PX = -62.0021991 | PY = 74.2751879 | PZ = 86.5956935 | E = 129.8456405 | P = 129.8456405
GAMMA    0.0      | PX = -13.5056092 | PY = 21.1447203 | PZ = -49.6149809 | E = 55.5980845 | P = 55.5980845
ETAO    548.80005 | PX = 235.1848873 | PY = -359.2014267 | PZ = 259.3528354 | E = 743.4937679 | P = 501.5989328
PIO     134.96449 | PX = -251.1929410 | PY = -100.5023393 | PZ = -167.5703373 | E = 345.6788192 | P = 318.2427246
GAMMA    0.0      | PX = -33.2706360 | PY = 222.1058704 | PZ = -88.0851646 | E = 241.2404384 | P = 241.2404384
GAMMA    0.0      | PX = 70.5026056 | PY = 129.7616046 | PZ = -123.7942134 | E = 192.7010605 | P = 192.7010605
-----
SUM:      -0.0000048      -0.0000083      0.0000212      1876.5596613

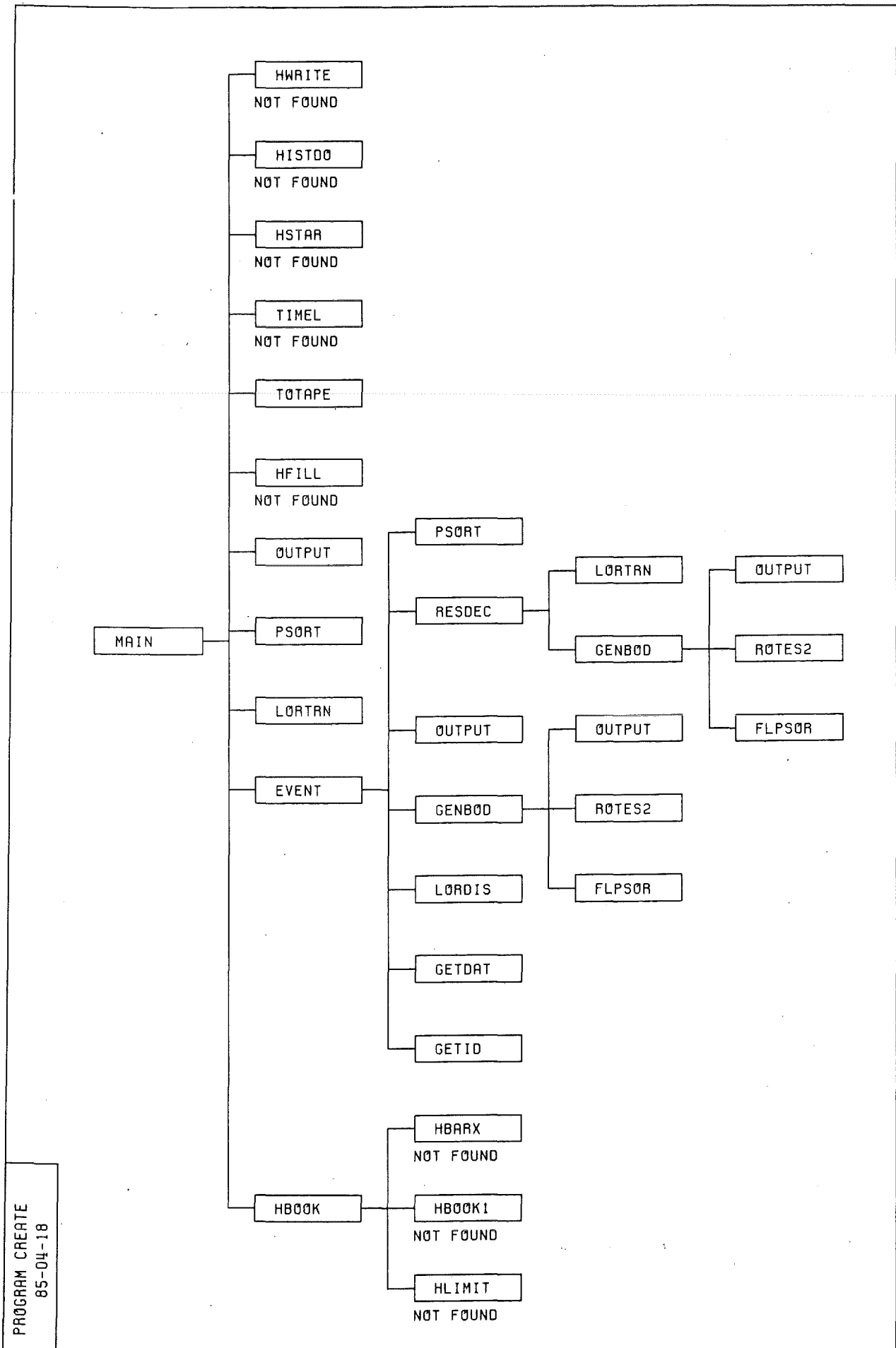
RESONANT PARTICLE DETECTED : ETAO
DECAYS INTO: GAMMA GAMMA

RESONANT PARTICLE DETECTED : PIO
DECAYS INTO: GAMMA GAMMA

SCAN NO.4
GAMMA    0.0      | PX = 24.9192253 | PY = -52.8762664 | PZ = 0.1990608 | E = 58.4543152 | P = 58.4543152
GAMMA    0.0      | PX = 29.3646622 | PY = 65.2926408 | PZ = 82.9171277 | E = 109.5475349 | P = 109.5475349
GAMMA    0.0      | PX = -62.0021991 | PY = 74.2751879 | PZ = 86.5956935 | E = 129.8456405 | P = 129.8456405
GAMMA    0.0      | PX = -13.5056092 | PY = 21.1447203 | PZ = -49.6149809 | E = 55.5980845 | P = 55.5980845
GAMMA    0.0      | PX = -33.2706360 | PY = 222.1058704 | PZ = -88.0851646 | E = 241.2404384 | P = 241.2404384
GAMMA    0.0      | PX = 70.5026056 | PY = 129.7616046 | PZ = -123.7942134 | E = 192.7010605 | P = 192.7010605
GAMMA    0.0      | PX = -113.0463162 | PY = 70.3960011 | PZ = 1.1546643 | E = 133.1780756 | P = 133.1780756
GAMMA    0.0      | PX = 348.2311815 | PY = -429.5973943 | PZ = 258.1981464 | E = 610.3156231 | P = 610.3156231
GAMMA    0.0      | PX = -142.9750937 | PY = -110.6110441 | PZ = -151.3928562 | E = 235.7935473 | P = 235.7935473
GAMMA    0.0      | PX = -108.2178456 | PY = 10.1087056 | PZ = -16.1704779 | E = 109.8852695 | P = 109.8852695
-----
SUM:      -0.0000250      0.0000258      -0.0000018      1876.5575876

```

ABLAUFDIAGRAMM



QUELLPROGRAMM

```
C*****
C*****
                                PROGRAM CREATE
C**                                *****
C**
C** THIS IS THE PROGRAM TO CREATE A SAMPLE OF DECAY CHANNELS
C** FROM PROTON-ANTIPROTON-ANNIHILATION AT REST ON LINE.
C** THE EVENTS ARE RANDOMIZED ACCORDING TO THE BRANCHING RATIOS
C** DEFINED IN ROUTINE 'GETID'.
C**
C**          >>>>  VERSION: DOUBLE PRECISION  <<<<
C**
C** ==>  USE OF SUBROUTINE 'TOTAPE' TO WRITE ON TAPE.
C**      DCB:BLKSIZE=32500,LRECL=25,DEN=4 (6250 BPI)
C**
C** ==>  OUTPUT SWITCHED TO FORTRAN UNIT #LUNOUT.
C**
C**      LANGUAGE   :FORTRAN77
C**      PROGRAMMER :MARCEL KUNZE/KFK
C**      VERS.4.0   :01-FEB-85
C**
C**      EXTENDED VERSION :
C**      -----
C**      PROGRAM 'CREATE' NOW ENABLES 'BARYONIUM-DECAY'.
C*****
C*****
C*
C* TYPE DECLARATIONS *
C*****
C*
C*      IMPLICIT REAL*8 (A-H,O-Z)
C*
C*      COMMON /STATIS/ WTMAX,FWT,KNT,NEV
C*      COMMON /RESNC/  YIELD,AMASS,GAMMA,NRES,RESPAR,EMIN
C*      COMMON /FLAGOP/ LIST,BARYON
C*
C*      LOGICAL LIST,TIMOUT,BARDEC,BARYON(50)
C*
C*      CHARACTER TYP(50)*5,INPTYP(50,100)*5
C*      CHARACTER TYPB(50)*5
C*
C*      DIMENSION YIELD(100),AMASS(50),BMASS(50),GAMMA(50)
C*      DIMENSION NRES(100),RESPAR(50,100),EMIN(100)
C*      DIMENSION PCM(5,50),PCMB(5,50),WTMAX(100),IDNUM(100)
C*
C*      NAMELIST /INPUT /  TECM,MAXGEN,LOOP,ISTORE,LUNOUT,BARDEC,
C*      &                    NCHAN,NLIST
C*
C*      DATA IDNUM/100*0/
C*      DATA PCM/250*0.DO/
C*      DATA TYP/50*' ' /,TYPB/50*' ' /,INPTYP/5000*' ' /
C*
C*      SET DEFAULT VALUES
C*      .....
C*
C*      TECM   = 1876.56
C*      LOOP   = 1
C*      MAXGEN = 100000
```

```
ISTORE = 0
LUNOUT = 0
BARDEC = .FALSE.
NCHAN = 200
NLIST = 100

C*
  FWT   = 1.DO
  KNT   = 0

C*
C*  INITIALIZE HBOOK
C*  .....
C*
  CALL HBOOK (NCHAN)

C*
C*  ADJUST TIME PARAMETER
C*  .....
C*
  TOSTRT = ZEIT(0)
  TIMEOUT = .FALSE.

C*
C*  START EXECUTION
C*  .....
C*
  PRINT 100
100  FORMAT ('1',125('*'),/,
&      '0', '*****',
&      ' ', '***** PROGRAM CREATE STARTED *****',
&      ' ', '*****',
&      '0',125('*'),////)

C*
C*  READ INPUT DATA
C*  .....
C*
  READ (5,INPUT)

C*
  NCHAN = 0

C*
200  READ (5,*,END=400) ID,YIELD(ID)
  READ (5,300) (INPTYP(I, ID),I=1,16,1)
300  FORMAT(16A5)
  NCHAN = MAX(NCHAN, ID)
  GOTO 200

C*
400  CONTINUE
  DO 700 ID=1,NCHAN,1
  PRINT 500, ID,TECM,YIELD(ID)
500  FORMAT ('0','CHANNEL NO.',I3,/,
&      ' ', 'TOTAL CM - ENERGY :',F10.5,5X,'YIELD :',F10.5,' %')
  PRINT 600,(INPTYP(I, ID),I=1,16,1)
600  FORMAT (' ', 'PARTICLES :',16A7)
700  CONTINUE

C*
  PRINT 800
800  FORMAT('1',125('*'),/,
&      ' ',55X,'PARTICLES' KINEMATICS:',/,
&      ' ',125('*'))

C*
C*  ADJUST RANDOM GENERATOR
```

```
C* .....
C*
DO 900 I=1,LOOP,1
X = RNDM2(I)
900 CONTINUE
C*
C*****
C** GRAND PARTICLE LOOP **
C*****
C*
DO 2200 WHILE (NEV.LT.MAXGEN.AND.(.NOT.TIMOUT))
C*
C* COUNT EVENTS ...
C*
NEV = NEV + 1
C*
C* SET LIST FLAG
C*
LIST = ((NEV.LE.NLIST).OR.(MOD(NEV,20000).EQ.0))
C*
CALL EVENT (ID,NPTOT,PCM,TYP,INPTYP,TECM,NCHAN)
C*
C* COUNT THE CHANNELS' FREQUENCY ...
C*
IDNUM(ID) = IDNUM(ID) + 1
C*
C*
C* IS THERE BARYONIUM?
C* .....
C*
IF (BARDEC) THEN
C*
C* LET THE BARYONIUM STATES DECAY ...
C*
I = 0
DO 1500 WHILE (I.LT.NPTOT)
I = I + 1
C*
IF (BARYON(I)) THEN
C*
IF (LIST) PRINT 1100,AMASS(I)
1100 FORMAT('0','BARYONIUM (' ,F6.1,') DECAYS :')
C*
DO 1200 J=1,NPTOT,1
BMASS(J) = AMASS(J)
1200 CONTINUE
C*
BARMAS = AMASS(I)
C*
CALL EVENT (IDB,NPTOTB,PCMB,TYPB,INPTYP,BARMAS,NCHAN)
C*
DO 1300 J=1,NPTOTB,1
TYP(NPTOT+J) = TYPB(J)
PCM(1,NPTOT+J) = PCMB(1,J)
PCM(2,NPTOT+J) = PCMB(2,J)
PCM(3,NPTOT+J) = PCMB(3,J)
PCM(4,NPTOT+J) = PCMB(4,J)
PCM(5,NPTOT+J) = PCMB(5,J)
```



```
      BMASS(NPTOT+J) = AMASS(J)
1300  CONTINUE
C*
C*      LORENTZ - TRANSFORMATION ...
C*
      CALL LORTRN (BARMAS,PCM(1,I),NPTOTB,PCM(1,NPTOT+1))
C*
      NPTOT = NPTOT + NPTOTB
C*
      DO 1400 J=1,NPTOT,1
      AMASS(J) = BMASS(J)
1400  CONTINUE
C*
      TYP(I) = 'PURGE'
      CALL PSORT (NPTOT,PCM,TYP)
C*
      ENDIF
C*      (* BARYON *)
C*
1500  CONTINUE
C*
      ENDIF
C*      (* BARDEC *)
C*
C*      PRINT FINAL EVENT DATA
C*      .....
C*
      IF (LIST) THEN
        PRINT *
        PRINT *, 'FINAL DATA:'
        CALL OUTPUT (ID,NPTOT,PCM,TYP,TECM)
      ENDIF
C*
C*      FILL THE HISTOGRAMS
C*      .....
C*
      DO 1600 I=1,NPTOT,1
C*
      IF (TYP(I).EQ.'PI+ ' .OR.TYP(I).EQ.'PI- ')
&CALL HFILL (01,PCM(4,I),0.,FWT)
C*
      IF (TYP(I).EQ.'GAMMA')
&CALL HFILL (02,PCM(4,I),0.,FWT)
C*
      IF (TYP(I).EQ.'K+ ' .OR.TYP(I).EQ.'K- ') THEN
        CALL HFILL (04,PCM(4,I),0.,FWT)
      ELSE
        IF (TYP(I)(1:1).EQ.'K') CALL HFILL (03,PCM(4,I),0.,FWT)
      ENDIF
C*
1600  CONTINUE
C*
C*      SAVE EVENT DATA ON FORTRAN UNIT #LUNOUT
C*      .....
C*
      IF (LUNOUT.GT.0) CALL TOTAPE (ID,NPTOT,PCM,TYP,TECM,LUNOUT)
C*
C*      RUN TIME CHECK
```

```
C* .....
C*
CALL TIMEL (TLEFT)
TIMOUT = (TLEFT.LE.5)
C*
2200 CONTINUE
C*
C* CLOSE FILE ON FORTRAN UNIT #LUNOUT
C* .....
C*
IF (LUNOUT.GT.0) END FILE LUNOUT
C*
C* PRINT A FINAL REPORT
C* .....
C*
ITIME = INT(ZEIT(TOSTRT))
MIN = ITIME/60
ISEC = MOD(ITIME,60)
PRINT 2300,MIN,ISEC,KNT,NEV
2300 FORMAT('0',///,
& ' EXECUTION TIME :',I3,' MINUTES ',I3,' SECONDS .',/,
& ' CREATED EVENTS :',I10,/,
& ' GOOD EVENTS :',I10,/)
PRINT 2400
2400 FORMAT('1',25X,'F I N A L R E P O R T :')
DO 2600 ID=1,NCHAN,1
PERCTG = IDNUM(ID)/FLOAT(NEV) * 100.0
PRINT 2500, ID, IDNUM(ID), PERCTG, WTMAX(ID)
2500 FORMAT(' ',5X,'CHANNEL NO.',I3,':',I8,' EVENTS.',
& '==>',F9.4,'% MAX.WEIGHT:',F20.15)
2600 CONTINUE
C*
C* PRINT THE HISTOGRAMS
C* .....
C*
CALL HSTAR(0)
CALL HISTDO
IF (ISTORE.GT.0) THEN
CALL HWRITE(0,ISTORE)
PRINT 2700,ISTORE
2700 FORMAT ('0','>>>> HISTOGRAMMS STORED ON FORTRAN UNIT #',I3)
ENDIF
C*
STOP
END
```

```
C*      *****
C*      SUBROUTINE HBOOK (NCHAN)
C*      *****
C*
C*      THIS ROUTINE SETS UP HISTOGRAM - BOOKING.
C*
C*      PROGRAMMER : MARCEL KUNZE
C*      LANGUAGE   : FORTRAN 77
C*      VERSION 1.0: 22-AUG-84
C*
C*      IMPLICIT REAL*8 (A-H,O-Z)
C*
C*      COMMON /          / HMEMOR(16000)
C*
C*      INITIALIZE HBOOK
C*      .....
C*
C*      IF (NCHAN.LE.0) NCHAN = 100
C*
C*      CALL HLIMIT (16000)
C*
C*      NAME THE HISTOGRAMS ...
C*
C*      CALL HBOOK1 (01, 'ENERGIE-SPEKTRUM DER GELADENEN PIONEN$',
C*      &              NCHAN,0.,1000.,0.)
C*      CALL HBOOK1 (02, 'ENERGIE-SPEKTRUM DER GAMMAS$',
C*      &              NCHAN,0.,1000.,0.)
C*      CALL HBOOK1 (03, 'ENERGIE-SPEKTRUM DER NEUTRALEN KAONEN$',
C*      &              NCHAN,0.,1000.,0.)
C*      CALL HBOOK1 (04, 'ENERGIE-SPEKTRUM DER GELADENEN KAONEN$',
C*      &              NCHAN,0.,1000.,0.)
C*
C*      DRAW ERROR BARS
C*
C*      CALL HBARX(0)
C*
C*      RETURN
C*      END
```

```
C* *****
C* SUBROUTINE EVENT (ID,NPTOT,PCM,TYP,INPTYP,TECM,NCHAN)
C* *****
C*
C* THIS ROUTINE PROVIDES KINEMATIC DATA (MOMENTA AND ENERGY)
C* FOR A SINGLE EVENT.
C*
C* LANGUAGE   : FORTRAN 77
C* PROGRAMMER: MARCEL KUNZE/KFK
C* VERS.2.0   : 15-JAN-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /RESNC/  YIELD,AMASS,GAMMA,NRES,RESPAR,EMIN
C* COMMON /STATIS/ WTMAX,FWT,KNT,NEV
C* COMMON /FLAGOP/ LIST,BARYON
C*
C* DIMENSION YIELD(100),AMASS(50),GAMMA(50)
C* DIMENSION NRES(100) ,RESPAR(50,100),EMIN(100)
C* DIMENSION OLDMAS(50),NRESO(100)
C* DIMENSION PCM(5,50),RPCM(5,50),WTMAX(100)
C*
C* CHARACTER TYP(50)*5,RTYP(50,50)*5
C* CHARACTER INPTYP(50,100)*5
C*
C* LOGICAL MISS,NOID,BADMAS,VETO,MAXTRY,RESONA,LIST,BARYON(50)
C*
C* GET AN EVENT-ID ACCORDING TO THE BRANCHING RATIOS
C* .....
C*
C* NOID = .TRUE.
C* DO 100 WHILE (NOID)
C* CALL GETID(ID,NCHAN,YIELD)
C* NOID = (INPTYP(1,ID).EQ.' ')
100 CONTINUE
C*
C* EVALUATE EVENT DATA
C* .....
C*
C* CALL GETDAT (ID,NPTOT,INPTYP(1,ID),RTYP)
C*
C* NPOLD      = NPTOT
C*
C* DO 200 I=1,NPOLD,1
C*   OLDMAS(I) = AMASS(I)
C*   NRESO(I)  = NRES(I)
200 CONTINUE
C*
C* START CREATING EVENT DATA (HIT-OR-MISS-MONTE-CARLO)
C* .....
C*
C* MISS = .TRUE.
C* DO 2400 WHILE (MISS)
C*   KNT = KNT + 1
C*
C* RESTORE INITIAL VALUES
C*
C* FWT      = 1.DO
C* NPTOT    = NPOLD
```

```

C*
DO 400 I=1,NPTOT,1
AMASS(I) = OLDMAS(I)
TYP(I)   = INPTYP(I,ID)
NRES(I)  = NRESO(I)
400      CONTINUE
C*
      NTRY   = 0
      MAXTRY = .FALSE.
500      VETO = .TRUE.
C*
      DO 1100 WHILE (VETO)
C*
C*      AFTER TRYING TOO MANY TIMES, CHOOSE ANOTHER EVENT-ID
C*
      IF (MAXTRY) THEN
        FWT   = 1.DO
        NTRY  = 0
        NSCAN = 0
        NOID  = .TRUE.
        DO 600 WHILE (NOID)
          CALL GETID(ID,NCHAN,YIELD)
          NOID = (INPTYP(1,ID).EQ.' ')
600      CONTINUE
          IF (LIST) PRINT *, 'NO CONVERGENCE. NEW ID CHOSEN:', ID
C*
C*      EVALUATE EVENT DATA
C*      .....
C*
          CALL GETDAT (ID,NPTOT,INPTYP(1,ID),RTYP)
C*
          NPOLD = NPTOT
C*
          DO 700 I=1,NPOLD,1
            OLDMAS(I) = AMASS(I)
            NRESO(I)  = NRES(I)
            TYP(I)    = INPTYP(I,ID)
700      CONTINUE
          ENDIF
C*
C*      RESONANT INTERMEDIATE STATE (GAMMA>0) ?
C*      .....
C*
          DO 900 I=1,NPTOT,1
C*
          IF (GAMMA(I).GT.0.0) THEN
            BADMAS = .TRUE.
            DO 800 WHILE (BADMAS)
              CALL LORDIS (OLDMAS(I),GAMMA(I),AMASS(I))
              BADMAS = (AMASS(I).LE.EMIN(I))
800      CONTINUE
            ENDIF
C*
900      CONTINUE
C*
C*      CHECK ENERGY CONSERVATION
C*      .....
C*
          ESUM = 0.0

```

```
DO 1000 I=1,NPTOT,1
ESUM = ESUM + AMASS(I)
1000 CONTINUE
C*
      NTRY = NTRY + 1
      MAXTRY = (NTRY.GT.100)
      VETO = (ESUM.GE.TECM)
1100 CONTINUE
C*
C* GO ON ...DO GENBOD
C*
      CALL GENBOD (ID,NPTOT,PCM,TYP,TECM,AMASS,WT)
      FWT = FWT * WT
C*
      RESONA = .TRUE.
      NSCAN = 0
C*
      DO 2300 WHILE (RESONA)
      NSCAN = NSCAN + 1
C*
      IF (LIST) THEN
        PRINT*
        PRINT*, 'SCAN NO. ', NSCAN
        CALL OUTPUT (ID,NPTOT,PCM,TYP,TECM)
      ENDIF
C*
C* RESONANT STATE ?
C* .....
C*
      NDEC = 0
      NPNEW = NPTOT
      DO 1800 I=1,NPTOT,1
C*
C* IF (NRES(I).GT.0) THEN
C*
      IF (LIST) THEN
        PRINT*
        PRINT*, 'RESONANT PARTICLE DETECTED : ', TYP(I)
        PRINT 1600, (RTYP(J,I), J=1, NRES(I), 1)
1600 FORMAT(' ', 'DECAYS INTO:', 16A7)
      ENDIF
C*
      NDEC = NDEC + 1
C*
      CALL RESDEC
& (ID,NRES(I),RPCM,RTYP(1,I),AMASS(I),PCM(1,I),RESPAR(1,I),RWT)
C*
C* FINAL WEIGHT...
C*
      FWT = FWT * RWT
C*
C* TRANSFER RESONANCE DATA IN ARRAY PCM...
C*
C* 1)PURGE RESONANT PARTICLE
C*
      TYP(I) = 'PURGE'
C*
C* 2)TRANSFER RESONANCE DATA IN 'PCM'
C*
```

```
DO 1700 J=1,NRES(I),1
TYP(NPNEW+J) = RTYP(J,I)
DO 1700 L=1,5,1
PCM(L,NPNEW+J) = RPCM(L,J)
1700 CONTINUE
C*
      NPNEW = NPNEW + NRES(I)
C*
ENDIF
C*
1800 CONTINUE
C* (* TEST PARTICLES *)
C*
      IF (NDEC.GT.0) THEN
          NPTOT = NPNEW
C*
          SORT THE PARTICLES
C*
          CALL PSORT (NPTOT,PCM,TYP)
C*
          EVALUATE RESONANCE DATA
C*
          CALL GETDAT (ID,NPTOT,TYP,RTYP)
C*
          DO 2200 WHILE (VETO)
C*
              AFTER TRYING TOO MANY TIMES, CHOOSE ANOTHER EVENT-ID
C*
              IF (MAXTRY) GOTO 500
C*
              RESONANT INTERMEDIATE STATE (GAMMA>0) ?
C* .....
C*
              DO 2000 I=1,NPTOT,1
C*
                  IF (GAMMA(I).GT.0.0) THEN
                      BADMAS = .TRUE.
                      DO 1900 WHILE (BADMAS)
                          CALL LORDIS (OLDMAS(I),GAMMA(I),AMASS(I))
                          BADMAS = (AMASS(I).LE.EMIN(I))
1900 CONTINUE
                      ENDIF
C*
2000 CONTINUE
C*
          CHECK ENERGY CONSERVATION
C* .....
C*
          ESUM = 0.0
          DO 2100 I=1,NPTOT,1
              ESUM = ESUM + PCM(4,I)
2100 CONTINUE
C*
          NTRY = NTRY + 1
          MAXTRY = (NTRY.GT.100)
          VETO = (ESUM.GE.TECM)
2200 CONTINUE
C*
      ELSE
```

```
        RESONA = .FALSE.
    ENDIF
C*
2300    CONTINUE
C*      (* RESONA *)
C*
C*      HIT OR MISS - MONTE CARLO
C*      .....
C*
C*      ADJUST THE CHANNELS' MAXIMUM WEIGHTS ...
C*
C*      IF (FWT.GT.WTMAX(ID)) WTMAX(ID) = FWT
C*
C*      RN   = DRNDM(I) * WTMAX(ID)
C*      MISS = (FWT.LT.RN)
C*      IF (LIST.AND.MISS) PRINT*, 'EVENT MISSED: TRY AGAIN.'
C*
2400    CONTINUE
C*
C*      SET WEIGHT OF THIS EVENT TO ONE...
C*
C*      FWT = 1.D0
C*
C*      RETURN
C*      END
C*      *****
C*      FUNCTION DRNDM(DUMMY)
C*      *****
C*      THIS ROUTINE ADAPTS RANDOM NUMBERS FOR REAL*8 USE.
C*
C*      IMPLICIT REAL*8 (A-H,O-Z)
C*      DRNDM = (RNDM2(DUMMY)*1.E8 + RNDM2(DUMMY))/1.E8
C*      RETURN
C*      END
```



```
C* *****
C* SUBROUTINE GETID(ID,NCHAN,YIELD)
C* *****
C*
C* THIS SUBROUTINE CREATES ID-NUMBERS ACCORDING TO THE
C* BRANCHING RATIOS FOR THE PBP-ANNIHILATION CHANNELS
C* SPECIFIED IN 'YIELD' .
C*
C* LANGUAGE   : FORTRAN 77
C* PROGRAMMER : MARCEL KUNZE/KFK
C* VERS.2.0   : 01-FEB-85
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* DIMENSION YIELD(100)
C*
C* LOGICAL SETUP
C*
C* DATA SETUP /.TRUE./
C*
C* IF (SETUP) THEN
C*     SUM = 0.DO
C*     DO 50 I=1,NCHAN,1
C*       SUM = SUM + YIELD(I)
50  C*     CONTINUE
C*
C*     SETUP = .FALSE.
C*
C* ENDIF
C*
C* CREATE A RANDOM NUMBER BETWEEN 0% AND SUM%
C*
C* X = RNDM2(DUMMY) * SUM
C*
C* LOOK FOR THE ID'S PERCENTAGE
C*
C* BRSUM = 0.DO
C* ID    = 0
C* DO 100 WHILE (ID.LE.NCHAN.AND.BRSUM.LT.X)
C*   ID    = ID + 1
C*   BRSUM = BRSUM + YIELD(ID)
100 C* CONTINUE
C*
C* RETURN
C* END
```

```
C* *****
C* SUBROUTINE GETDAT (ID,NPTOT,TYP,RTYP)
C* *****
C* THIS ROUTINE PROVIDES PARTICLE DATA FOR A SINGLE EVENT.
C*
C* LANGUAGE : FORTRAN 77
C* PROGRAMMER: MARCEL KUNZE/KFK
C* VERS.1.0 : 01-FEB-85
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /RESNC/ YIELD,AMASS,GAMMA,NRES,RESPAR,EMIN
C* COMMON /FLAGOP/ LIST,BARYON
C*
C* DIMENSION YIELD(100),AMASS(50),GAMMA(50)
C* DIMENSION NRES(100),RESPAR(50,100),EMIN(100)
C*
C* CHARACTER TYP(50)*5,RTYP(50,50)*5
C*
C* LOGICAL NOMORE,LIST,BARYON(50)
C*
C* EVALUATE EVENT DATA
C* .....
C*
C* I = 0
C* NOMORE = .FALSE.
C* DO 200 UNTIL (NOMORE)
C* I = I + 1
C* NOMORE = (I.GE.50.OR.TYP(I).EQ.' ')
C*
C* LOOK FOR THE PARTICLES
C*
C* IF (.NOT.NOMORE) THEN
C*
C* IF (TYP(I).EQ.'PURGE') THEN
C* BARYON(I) = .FALSE.
C* AMASS(I) = 0.0
C* GAMMA(I) = 0.0
C* NRES (I) = 0
C* ELSEIF (TYP(I).EQ.'GAMMA') THEN
C* BARYON(I) = .FALSE.
C* AMASS(I) = 0.0
C* GAMMA(I) = 0.0
C* NRES (I) = 0
C* ELSEIF (TYP(I).EQ.'PIO ') THEN
C* BARYON(I) = .FALSE.
C* AMASS(I) = 134.9645
C* GAMMA(I) = 0.0
C* NRES (I) = 2
C* RTYP(1,I) = 'GAMMA'
C* RTYP(2,I) = 'GAMMA'
C* RESPAR(1,I) = 0.0
C* RESPAR(2,I) = 0.0
C* ELSEIF (TYP(I).EQ.'PI+ ') THEN
C* BARYON(I) = .FALSE.
C* AMASS(I) = 139.5688
C* GAMMA(I) = 0.0
C* NRES (I) = 0
```

```
ELSEIF (TYP(I).EQ.'PI- ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 139.5688
  GAMMA(I) = 0.0
  NRES (I) = 0
ELSEIF (TYP(I).EQ.'RHO0 ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 769.0
  GAMMA(I) = 154.0
  NRES (I) = 2
  RTYP(1,I) = 'PI+ '
  RTYP(2,I) = 'PI- '
  RESPAR(1,I) = 139.5688
  RESPAR(2,I) = 139.5688
ELSEIF (TYP(I).EQ.'RHO- ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 769.0
  GAMMA(I) = 154.0
  NRES (I) = 2
  RTYP(1,I) = 'PI- '
  RTYP(2,I) = 'PIO '
  RESPAR(1,I) = 139.5688
  RESPAR(2,I) = 134.9645
ELSEIF (TYP(I).EQ.'RHO+ ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 769.0
  GAMMA(I) = 154.0
  NRES (I) = 2
  RTYP(1,I) = 'PI+ '
  RTYP(2,I) = 'PIO '
  RESPAR(1,I) = 139.5688
  RESPAR(2,I) = 134.9645
ELSEIF (TYP(I).EQ.'FO ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 1273.
  GAMMA(I) = 179.
  NRES (I) = 2
  RTYP(1,I) = 'PI+ '
  RTYP(2,I) = 'PI- '
  RESPAR(1,I) = 139.5688
  RESPAR(2,I) = 139.5688
ELSEIF (TYP(I).EQ.'OMEGA') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 782.6
  GAMMA(I) = 9.9
  NRES (I) = 3
  RTYP(1,I) = 'PI+ '
  RTYP(2,I) = 'PI- '
  RTYP(3,I) = 'PIO '
  RESPAR(1,I) = 139.5688
  RESPAR(2,I) = 139.5688
  RESPAR(3,I) = 134.9645
ELSEIF (TYP(I).EQ.'OMEGO') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 782.8
  GAMMA(I) = 9.9
  NRES (I) = 2
  RTYP(1,I) = 'PIO '
  RTYP(2,I) = 'GAMMA'
```

```
RESPAR(1,I) = 134.9645
RESPAR(2,I) = 0.0
ELSEIF (TYP(I).EQ. 'ETA  ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 548.8
  GAMMA(I) = 0.00083
  NRES (I) = 3
  RTYP(1,I) = 'PI+  '
  RTYP(2,I) = 'PI-  '
  RTYP(3,I) = 'PIO  '
  RESPAR(1,I) = 139.5688
  RESPAR(2,I) = 139.5688
  RESPAR(3,I) = 134.9645
ELSEIF (TYP(I).EQ. 'ETA0 ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 548.8
  GAMMA(I) = 0.00083
  NRES (I) = 2
  RTYP(1,I) = 'GAMMA'
  RTYP(2,I) = 'GAMMA'
  RESPAR(1,I) = 0.0
  RESPAR(2,I) = 0.0
ELSEIF (TYP(I).EQ. 'ETA" ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 957.57
  GAMMA(I) = 0.28
  NRES (I) = 3
  RTYP(1,I) = 'PI+  '
  RTYP(2,I) = 'PI-  '
  RTYP(3,I) = 'PIO  '
  RESPAR(1,I) = 139.5688
  RESPAR(2,I) = 139.5688
  RESPAR(3,I) = 134.9645
ELSEIF (TYP(I).EQ. 'KO  ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 497.67
  GAMMA(I) = 0.0
  NRES (I) = 0
ELSEIF (TYP(I).EQ. 'K+  ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 493.667
  GAMMA(I) = 0.0
  NRES (I) = 0
ELSEIF (TYP(I).EQ. 'K-  ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 493.667
  GAMMA(I) = 0.0
  NRES (I) = 0
ELSEIF (TYP(I).EQ. 'K1  ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 497.67
  GAMMA(I) = 0.0
  NRES (I) = 0
ELSEIF (TYP(I).EQ. 'K2  ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 497.67
  GAMMA(I) = 0.0
  NRES (I) = 0
ELSEIF (TYP(I).EQ. 'KO* ') THEN
```

```
BARYON(I) = .FALSE.
AMASS(I) = 891.8
GAMMA(I) = 50.8
NRES (I) = 2
RTYP(1,I) = 'K2 '
RTYP(2,I) = 'PI0 '
RESPAR(1,I) = 497.67
RESPAR(2,I) = 134.9645
ELSEIF (TYP(I).EQ.'K*0 ') THEN
BARYON(I) = .FALSE.
AMASS(I) = 891.8
GAMMA(I) = 50.8
NRES (I) = 2
RTYP(1,I) = 'K+ '
RTYP(2,I) = 'PI- '
RESPAR(1,I) = 493.667
RESPAR(2,I) = 139.5688
ELSEIF (TYP(I).EQ.'K*+ ') THEN
BARYON(I) = .FALSE.
AMASS(I) = 885.1
GAMMA(I) = 50.8
NRES (I) = 2
RTYP(1,I) = 'K0 '
RTYP(2,I) = 'PI+ '
RESPAR(1,I) = 497.67
RESPAR(2,I) = 139.5688
ELSEIF (TYP(I).EQ.'K*- ') THEN
BARYON(I) = .FALSE.
AMASS(I) = 885.1
GAMMA(I) = 50.8
NRES (I) = 2
RTYP(1,I) = 'K0 '
RTYP(2,I) = 'PI- '
RESPAR(1,I) = 497.67
RESPAR(2,I) = 139.5688
ELSEIF (TYP(I).EQ.'A1- ') THEN
BARYON(I) = .FALSE.
AMASS(I) = 1275.
GAMMA(I) = 315.
NRES (I) = 2
RTYP(1,I) = 'K0 '
RTYP(2,I) = 'K- '
RESPAR(1,I) = 497.67
RESPAR(2,I) = 493.667
ELSEIF (TYP(I).EQ.'A2+ ') THEN
BARYON(I) = .FALSE.
AMASS(I) = 1318.
GAMMA(I) = 110.
NRES (I) = 2
RTYP(1,I) = 'OMEGA'
RTYP(2,I) = 'PI+ '
RESPAR(1,I) = 782.6
RESPAR(2,I) = 139.5688
ELSEIF (TYP(I).EQ.'A2- ') THEN
BARYON(I) = .FALSE.
AMASS(I) = 1318.
GAMMA(I) = 110.
NRES (I) = 2
RTYP(1,I) = 'K0 '

```

```
RTYP(2,I) = 'K- '
RESPAR(1,I) = 497.67
RESPAR(2,I) = 493.667
ELSEIF (TYP(I).EQ.'B+ ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 1233.
  GAMMA(I) = 137.
  NRES (I) = 2
  RTYP(1,I) = 'OMEGA'
  RTYP(2,I) = 'PI+ '
  RESPAR(1,I) = 782.6
  RESPAR(2,I) = 139.5688
ELSEIF (TYP(I).EQ.'PHI ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 1019.61
  GAMMA(I) = 4.21
  NRES (I) = 2
  RTYP(1,I) = 'K1 '
  RTYP(2,I) = 'K2 '
  RESPAR(1,I) = 497.67
  RESPAR(2,I) = 497.67
ELSEIF (TYP(I).EQ.'E0 ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 1418.
  GAMMA(I) = 52.
  NRES (I) = 2
  RTYP(1,I) = 'DELTA'
  RTYP(2,I) = 'PIO '
  RESPAR(1,I) = 983.
  RESPAR(2,I) = 134.9645
ELSEIF (TYP(I).EQ.'DELTA') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 983.
  GAMMA(I) = 54.
  NRES (I) = 2
  RTYP(1,I) = 'ETA0 '
  RTYP(2,I) = 'PIO '
  RESPAR(1,I) = 548.8
  RESPAR(2,I) = 134.9645
ELSEIF (TYP(I).EQ.'GLUE ') THEN
  BARYON(I) = .FALSE.
  AMASS(I) = 800.
  GAMMA(I) = 20.

C*
C* SELECT DECAY MODE OF GLUEBALL
C*
C* X = RNDM2(I)

C*
C* IF ((X.GT.0.).AND.(X.LE.0.33)) THEN
  NRES (I) = 2
  RTYP(1,I) = 'PIO '
  RTYP(2,I) = 'PIO '
  RESPAR(1,I) = 134.9645
  RESPAR(2,I) = 134.9645
ELSEIF ((X.GT.0.33).AND.(X.LE.0.66)) THEN
  NRES (I) = 2
  RTYP(1,I) = 'GAMMA'
  RTYP(2,I) = 'GAMMA'
  RESPAR(1,I) = 0.0
```



```
      EMIN(I) = EMIN(I) + RESPAR(J,I)
300  CONTINUE
C*
      ENDIF
C*      (* .NOT. NOMORE *)
200  CONTINUE
C*
      NPTOT = I - 1
C*
      RETURN
      END
```



```
C* *****
C* SUBROUTINE LORDIS(E0,GAMMA,E)
C* *****
C*
C* THIS IS THE SUBROUTINE 'LORDIS' TO CREATE
C* LORENTZ DISTRIBUTED ENERGIES AROUND THE
C* ENERGIE E0.
C* (VIA INTEGRAL-TRANSFORMATION)
C* 1) X = INTEGRAL(-00/E)(L(E)*DE)
C* X: EQUAL DISTRIBUTED RANDOM NUMBER
C* E: LORENTZ DISTRIBUTED RANDOM ENERGIE
C* L(E) = GAMMA/(2*PI) * ((E0-E)**2+(GAMMA/2)**2)**(-1)
C* (LORENTZ - DISTRIBUTION)
C* SOLVING THE INTEGRAL EQUATION 1) FOR E LEADS TO
C* E = GAMMA/2. * TAN(PI*(X-0.5)) + E0
C*
C* ON INPUT:
C* =====
C* E0 : MAXIMUM OF THE DISTRIBUTION
C* GAMMA: FULL WIDTH AT HALF MAXIMUM ENERGY
C*
C* ON EXIT :
C* =====
C* E : LORENTZ DISTRIBUTED RANDOM ENERGY
C*
C* LANGUAGE : FORTRAN 77
C* PROGRAMMER: MARCEL KUNZE/KFK
C* VERS.1.0 : 10-JAN-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* PARAMETER (PI = 3.1415962653589793)
C*
C* X = RNDM2(I)
C* E = GAMMA/2. * TAN(PI*(X-0.5)) + E0
C*
C* RETURN
C* END
C* *****
C* FUNCTION TAN(X)
C* *****
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* TAN = DSIN(X) / DCOS(X)
C*
C* RETURN
C* END
```

```
C* *****
C* SUBROUTINE RESDEC (ID,NRES,RPCM,RTYP,RESMAS,RKIN,RESPAR,RWT)
C* *****
C*
C* THIS IS A SUBROUTINE TO CREATE PARTICLES' KINEMATICS
C* FROM RESONANT STATE.
C*
C* ON ENTRY:
C* NRES      -      NUMBER OF RESONANT PARTICLES
C* RESMAS    -      RESONANT MASS
C* RESPAR(50) -      ARRAY WITH MASSES OF RESONANT PARTICLES
C* RKIN(5)   -      KINEMATIC PARAMETERS OF RESONANT MASS
C*
C* ON EXIT:
C* RPCM(5,50) -      KINEMATIC PARAMETERS
C* RWT       -      WEIGHT OF THIS...
C*
C* LANGUAGE   : FORTRAN 77
C* PROGRAMMER : MARCEL KUNZE/KFK
C* VERS.1.0   : 07-DEC-83
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* DIMENSION RESPAR(50),RKIN(5)
C* DIMENSION RPCM(5,50)
C*
C* CHARACTER RTYP(50)*5
C*
C* CALL GENBOD (ID,NRES,RPCM,RTYP,RESMAS,RESPAR,RWT)
C*
C* NOW WE'VE TO DO THE LORENTZ TRANSFORMATION FROM CMS
C* OF RESONANT STATE TO LAB-SYSTEM.
C*
C* CALL LORTRN (RESMAS,RKIN,NRES,RPCM)
C*
C* RETURN
C* END
```

```
C* *****
C* SUBROUTINE GENBOD (ID,NPTOT,PCM,TYP,TECM,AMASS,WT)
C* *****
C*
C* SUBROUTINE TO GENERATE N-BODY EVENT
C* ACCORDING TO FERMI LORENTZ-INVARIANT PHASE SPACE
C* ADAPTED FROM FOWL (CERN W505) SEPT. 1974 BY F. JAMES
C* EVENTS ARE GENERATED IN THEIR OWN CENTER-OF-MASS,
C* BUT MAY BE TRANSFORMED TO ANY FRAME USING LOREN4
C*
C* INPUT TO SUBROUTINE IS
C* NPTOT = NUMBER OF OUTGOING PARTICLES (.LT. 50)
C* TECM = TOTAL ENERGY IN CENTER-OF-MASS
C* AMASS(I) = MASS OF ITH OUTGOING PARTICLE
C*
C* OUTPUT FROM SUBROUTINE IS
C* PCM(1,I) = X-MOMENTUM IF ITH PARTICLE
C* PCM(2,I) = Y-MOMENTUM IF ITH PARTICLE
C* PCM(3,I) = Z-MOMENTUM IF ITH PARTICLE
C* PCM(4,I) = ENERGY OF ITH PARTICLE
C* PCM(5,I) = MOMENTUM OF ITH PARTICLE
C* WT = WEIGHT OF EVENT
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* DIMENSION PCM(5,50),AMASS(50)
C* DIMENSION EMM(50)
C* DIMENSION RNO(100)
C* DIMENSION PD(50),EMS(50),SM(50)
C*
C* CHARACTER TYP(50)*5
C*
C* PARAMETER (TWOPI = 6.2831925307179586)
C*
C* INITIALIZATION
C*
C* IF (NPTOT .LT. 2) GO TO 1001
C* IF (NPTOT .GT.50) GO TO 1002
C* NPM1=NPTOT-1
C* NPM2=NPTOT-2
C* NPNM4 = 3*NPTOT - 4
C* EMM(1)=AMASS(1)
C* TM=0.0
C* DO 200 I=1,NPTOT
C* EMS(I)=AMASS(I)**2
C* TM=TM+AMASS(I)
200 SM(I)=TM
C*
C* CONSTANTS DEPENDING ON TECM
C*
C* TECMTM=TECM-TM
C* IF (TECMTM .LE. 0.0) GO TO 1000
C* EMM(NPTOT)=TECM
C*
C* CONSTANT MATRIX ELEMENT AS FUNCTION OF TECM
C*
C* EMMAX=TECMTM+AMASS(1)
C* EMMIN=0.0
C* WTMAX=1.0
```

```
DO 350 I=2,NPTOT
EMMIN=EMMIN+AMASS(I-1)
EMMAX=EMMAX+AMASS(I)
350 WTMAX=WTMAX*PDK(EMMAX,EMMIN,AMASS(I))
WTMAXQ=1.0/WTMAX
C*
C* FILL RNO WITH 3*NPTOT-4 RANDOM NUMBERS,
C* OF WHICH THE FIRST NPTOT-2 ARE ORDERED.
C*
DO 457 I= 1, NPNM4
457 RNO(I) = RNDM2(I)
IF(NPM2) 900,509,460
460 CONTINUE
CALL FLPSOR(RNO,NPM2)
DO 508 J=2,NPM1
508 EMM(J)=RNO(J-1)*(TECMTM)+SM(J)
509 WT=WTMAXQ
IR=NPM2
DO 530 I=1,NPM1
PD(I)=PDK(EMM(I+1),EMM(I),AMASS(I+1))
530 WT=WT*PD(I)
C*
C* COMPLETE SPECIFICATION OF EVENT (RAUBOLD-LYNCH METHOD)
C*
PCM(1,1)=0.0
PCM(2,1)=PD(1)
PCM(3,1)=0.0
DO 570 I=2,NPTOT
PCM(1,I)=0.0
PCM(2,I) = -PD(I-1)
PCM(3,I)=0.0
IR=IR+1
BANG=TWOPI*RNO(IR)
CB=DCOS(BANG)
SB=DSIN(BANG)
IR=IR+1
C=2.0*RNO(IR)-1.0
S=SQRT(1.0-C*C)
IF(I.EQ.NPTOT) GO TO 1567
ESYS=SQRT(PD(I)**2+EMM(I)**2)
BETA=PD(I)/ESYS
GAMA=ESYS/EMM(I)
DO 568 J=1,I
AA= PCM(1,J)**2 + PCM(2,J)**2 + PCM(3,J)**2
PCM(5,J) = SQRT(AA)
PCM(4,J) = SQRT(AA+EMS(J))
CALL ROTES2(C,S,CB,SB,PCM,J)
PSAVE = GAMA*(PCM(2,J)+BETA*PCM(4,J))
568 PCM(2,J)=PSAVE
GO TO 570
1567 DO 1568 J=1,I
AA=PCM(1,J)**2 + PCM(2,J)**2 + PCM(3,J)**2
PCM(5,J)=SQRT(AA)
PCM(4,J)=SQRT(AA+EMS(J))
CALL ROTES2(C,S,CB,SB,PCM,J)
1568 CONTINUE
570 CONTINUE
900 CONTINUE
```

```
      RETURN
C      ERROR RETURNS
1000 PRINT 1100
      GO TO 1050
1001 PRINT 1101
      GO TO 1050
1002 PRINT 1102
1050 PRINT*, 'ID =', ID, ' NPTOT =', NPTOT, ' TECM =', TECM
      CALL OUTPUT (ID, NPTOT, PCM, TYP, TECM)
      STOP
1100 FORMAT (28H0 AVAILABLE ENERGY NEGATIVE )
1101 FORMAT (33H0 LESS THAN 2 OUTGOING PARTICLES )
1102 FORMAT (34H0 MORE THAN 50 OUTGOING PARTICLES )
      END
C* *****
SUBROUTINE FLPSOR(A,N)
C* *****
C*
C* SORT THE ONE-DIMENSIONAL FLOATING POINT ARRAY A(1),...,A(N)
C* BY INCREASING VALUES
C*
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION A(N)
      DIMENSION LT(20),RT(20)
      INTEGER R,RT
C*
      LEVEL=1
      LT(1)=1
      RT(1)=N
10  L=LT(LEVEL)
      R=RT(LEVEL)
      LEVEL=LEVEL-1
20  IF(R.LE.L) IF(LEVEL) 50,50,10
C*
C* SUBDIVIDE THE INTERVAL L,R
C* L : LOWER LIMIT OF THE INTERVAL (INPUT)
C* R : UPPER LIMIT OF THE INTERVAL (INPUT)
C* J : UPPER LIMIT OF LOWER SUB-INTERVAL (OUTPUT)
C* I : LOWER LIMIT OF UPPER SUB-INTERVAL (OUTPUT)
C*
      I=L
      J=R
      M=(L+R)/2
      X=A(M)
220 IF(A(I).GE.X) GO TO 230
      I=I+1
      GO TO 220
230 IF(A(J).LE.X) GO TO 231
      J=J-1
      GO TO 230
C*
231 IF(I.GT.J) GO TO 232
      W=A(I)
      A(I)=A(J)
      A(J)=W
      I=I+1
      J=J-1
      IF(I.LE.J) GO TO 220
C*
```

```

232 LEVEL=LEVEL+1
    IF((R-I).GE.(J-L)) GO TO 30
    LT(LEVEL)=L
    RT(LEVEL)=J
    L=I
    GO TO 20
30  LT(LEVEL)=I
    RT(LEVEL)=R
    R=J
    GO TO 20
50  RETURN
    END
C*  *****
C*  SUBROUTINE ROTES2(C,S,C2,S2,PR,I)
C*  *****
C*  THIS SUBROUTINE NOW DOES TWO ROTATIONS (XY AND XZ)
C*  IMPLICIT REAL*8 (A-H,O-Z)
C*  DIMENSION PR(50)
    K1 = 5*I - 4
    K2 = K1 + 1
    SA = PR(K1)
    SB = PR(K2)
    A      = SA*C - SB*S
    PR(K2) = SA*S + SB*C
    K2 = K2 + 1
    B = PR(K2)
    PR(K1) = A*C2 - B*S2
    PR(K2) = A*S2 + B*C2
    RETURN
    END
C*  *****
C*  FUNCTION PDK(A,B,C)
C*  *****
C*  PDK = SQRT(A*A+(B*B-C*C)**2/(A*A) - 2.0*(B*B+C*C))/2.0
C*
    REAL*16 A2,B2,C2,SQ
    REAL*8  PDK
C*
    A2 = QEXT (A*A)
    B2 = QEXT (B*B)
    C2 = QEXT (C*C)
    SQ = QSQRT(ABS(A2 + (B2-C2)**2/A2 - 2.0*(B2+C2)))
    PDK = 0.5*SQ
    RETURN
    END

```

```
C* *****
C* SUBROUTINE LORTRN (RESMAS,PCMR,NPAR,PCM)
C* *****
C*
C* THIS ROUTINE DOES THE LORENTZ-TRANSFORMATION OF A DECAY IN FLIGHT
C* FROM CM-SYSTEM OF THE DECAYING PARTICLE INTO LAB-SYSTEM
C* VER. 1.0/21-JAN-80(TK)
C*
C*
C* ON ENTRY:
C* =====
C*
C* RESMAS: MASS OF DECAYING PARTICLE
C* PCMR(1): X-MOMENTUM OF DECAYING PARTICLE IN LAB-SYSTEM
C* PCMR(2): Y-MOMENTUM OF DECAYING PARTICLE IN LAB-SYSTEM
C* PCMR(3): Z-MOMENTUM OF DECAYING PARTICLE IN LAB-SYSTEM
C* PCMR(4): ENERGY OF DECAYING PARTICLE IN LAB-SYSTEM
C* * PCMR(5): TOTAL MOMENTUM OF DECAYING PARTICLE IN LAB-SYSTEM
C* NPAR: NO OF SECONDARY-PARTICLES
C* PCM(1,I): X-MOMENTUM OF ITH PARTICLE IN CM-SYSTEM
C* PCM(2,I): Y-MOMENTUM OF ITH PARTICLE IN CM-SYSTEM
C* PCM(3,I): Z-MOMENTUM OF ITH PARTICLE IN CM-SYSTEM
C* PCM(4,I): ENERGY OF ITH PARTICLE IN CM-SYSTEM
C* * PCM(5,I): TOTAL MOMENTUM OF ITH PARTICLE IN CM-SYSTEM
C*
C* (*: VALUE NOT USED, BUT ARRAY-DIMENSIONS MUST BE AS SHOWN)
C*
C* ON EXIT:
C* =====
C*
C* PCM(1,I): X-MOMENTUM OF ITH PARTICLE IN LAB-SYSTEM
C* PCM(2,I): Y-MOMENTUM OF ITH PARTICLE IN LAB-SYSTEM
C* PCM(3,I): Z-MOMENTUM OF ITH PARTICLE IN LAB-SYSTEM
C* PCM(4,I): ENERGY OF ITH PARTICLE IN LAB-SYSTEM
C* PCM(5,I): TOTAL MOMENTUM OF ITH PARTICLE IN LAB-SYSTEM
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* DIMENSION PCMR(5),PCM(5,NPAR)
C*
C* RSMAS1 = 1./RESMAS
C*
C* ETA1 = PCMR(1)*RSMAS1
C* ETA2 = PCMR(2)*RSMAS1
C* ETA3 = PCMR(3)*RSMAS1
C* GAMMA = PCMR(4)*RSMAS1
C* GAMP1 = GAMMA+1.
C*
C* LOOP THROUGH NPAR PARTICLES
C*
C* DO 1000 I=1,NPAR,1
C*
C* LORENTZ-TRANSFORMATION OF PARTICLE-ENERGY FROM CM- TO LAB-SYS.
C*
C* FACT = ((ETA1*PCM(1,I))+(ETA2*PCM(2,I))+(ETA3*PCM(3,I)))
C* ELAB = GAMMA*PCM(4,I)+FACT
C*
C* LORENTZ-TRANSFORMATION OF PARTCLE-MOMENTA FROM CM- TO LAB-SYS.
```

```
C*      FACT = (PCM(4,I)+ELAB)/GAMP1
C*
      PCM(1,I) = PCM(1,I)+ETA1*FACT
      PCM(2,I) = PCM(2,I)+ETA2*FACT
      PCM(3,I) = PCM(3,I)+ETA3*FACT
      PCM(4,I) = ELAB
      PCM(5,I) = SQRT((PCM(1,I)*PCM(1,I))+(PCM(2,I)*PCM(2,I))+
+                (PCM(3,I)*PCM(3,I)))
1000 CONTINUE
C*
C*      RETURN TO CALLING PROGRAM
C*
      RETURN
C*      =====
      END
```



```
C* *****
C* SUBROUTINE PSORT(NPTOT,PCM,TYP)
C* *****
C* THIS ROUTINE SORTS AND REARRANGES KINEMATIC DATA (ARRAY 'PCM').
C*
C* PROGRAMMER : MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN 77
C* VERSION 1.0: 10-OCT-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /RESNC/ YIELD,AMASS,GAMMA,NRES,RESPAR,EMIN
C*
C* DIMENSION YIELD(100),AMASS(50),GAMMA(50)
C* DIMENSION NRES(100),RESPAR(5,100),EMIN(100)
C* DIMENSION PCM(5,50),PCMS(5,50),BMASS(50)
C*
C* CHARACTER*5 TYP(50),TYP5(50)
C*
C* 1.) SORT THE PARTICLES BY USE OF NAME
C* .....
C*
C* N = 0
C*
C* DO 100 I=1,NPTOT,1
C*
C* IF (TYP(I).NE.'PURGE') THEN
C*     N           = N + 1
C*     BMASS(N)   = AMASS(I)
C*     PCMS(1,N)  = PCM(1,I)
C*     PCMS(2,N)  = PCM(2,I)
C*     PCMS(3,N)  = PCM(3,I)
C*     PCMS(4,N)  = PCM(4,I)
C*     PCMS(5,N)  = PCM(5,I)
C*     TYP5 (N)   = TYP (I)
C* ENDIF
C*
C* 100 CONTINUE
C*
C* 2.REARRANGE ARRAY 'PCM'
C* .....
C*
C* NPTOT = N
C* DO 300 I=1,NPTOT,1
C*     AMASS(I) = BMASS(I)
C*     PCM(1,I) = PCMS(1,I)
C*     PCM(2,I) = PCMS(2,I)
C*     PCM(3,I) = PCMS(3,I)
C*     PCM(4,I) = PCMS(4,I)
C*     PCM(5,I) = PCMS(5,I)
C*     TYP (I) = TYP5 (I)
C* 300 CONTINUE
C*
C* TYP(NPTOT+1) = ' '
C*
C* RETURN
C* END
```

```
C* *****
C* SUBROUTINE TOTAPE (ID,NPTOT,PCM,TYP,TECM,LUNOUT)
C* *****
C* WRITE EVENT-DATA GENERATED BY CRETAP/HITMISS/DECAY ON TAPE
C* ==> OUTPUT SWITCHED TO FORTRAN UNIT #LUNOUT
C*
C* >>>>  VERSION:  SINGLE PRECISION  <<<<
C*          DCB: BLKSIZE=32500,LRECL=25
C*
C* (IN IBM-INTERNAL BINARY FLOATING-POINT-FORMAT: 4 BYTES)
C* .....
C*
C* LANGUAGE   : FORTRAN 77
C* PROGRAMMER : MARCEL KUNZE/KFK
C* VERS. 2.0   : 10-APR-84(MK)
C*
C* DIMENSION PCMS(5,25)
C* REAL*8     PCM (5,25),TECM
C*
C* CHARACTER TYP(25)*5
C*
C* DATA NCALL/0/
C*
C* NCALL = NCALL + 1
C*
C* WRITE THE EVENT-DESCRIPTOR-RECORD...
C*
C* TECMS = SNGL(TECM)
C* WRITE (LUNOUT,1010) NCALL, ID,NPTOT,TECMS
1010 FORMAT (A4,A4,A4,A4)
C*
C* DO 1025 J=1,NPTOT,1
C*
C* WRITE THE PARTICLE-KINEMATIC-RECORD
C*
C* PCMS(1,J) = SNGL(PCM(1,J))
C* PCMS(2,J) = SNGL(PCM(2,J))
C* PCMS(3,J) = SNGL(PCM(3,J))
C* PCMS(4,J) = SNGL(PCM(4,J))
C* PCMS(5,J) = SNGL(PCM(5,J))
C*
C* WRITE (LUNOUT,1015)
C* & PCMS(1,J),PCMS(2,J),PCMS(3,J),PCMS(4,J),PCMS(5,J),
C* & TYP(J)
1015 FORMAT (5A4,A5)
C*
1025 CONTINUE
RETURN
END
```

```
C* *****
C* SUBROUTINE OUTPUT (ID,NPTOT,PCM,TYP,TECM)
C* *****
C*
C* THIS ROUTINE PROVIDES OUTPUT DATA (MASSES, MOMENTA AND ENERGY)
C* FOR A SINGLE EVENT.
C*
C* LANGUAGE : FORTRAN 77
C* PROGRAMMER: MARCEL KUNZE/KFK
C* VERS.2.0 : 15-JAN-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /RESNC/ YIELD,AMASS,GAMMA,NRES,RESPAR,EMIN
C* COMMON /STATIS/ WTMAX,FWT,KNT,NEV
C*
C* DIMENSION YIELD(100),AMASS(50),GAMMA(50)
C* DIMENSION NRES(100) ,RESPAR(50,100),EMIN(100)
C* DIMENSION PCM(5,50),WTMAX(100)
C*
C* CHARACTER TYP(50)*5
C*
C* IF (NEVOLD.NE.NEV) THEN
C*   NEVOLD = NEV
C*   PRINT 300,NEV, ID,NPTOT,TECM
300   FORMAT ('0',131('*'),/,
&           ' ', '* EVENT NO. ', I8, 10X, 'CHANNEL NO. ', I3, 5X,
&           'NUMBER OF PARTICLES: ', I3, 5X,
&           'TOTAL ENERGY (CM): ', F10.2, 25X, ' *', /,
&           ' ', 131('*'))
C*   ENDIF
C*
C*   PXSUM = 0.0
C*   PYSUM = 0.0
C*   PZSUM = 0.0
C*   ESUM = 0.0
C*   DO 1300 I=1,NPTOT,1
C*     PXSUM = PXSUM + PCM(1,I)
C*     PYSUM = PYSUM + PCM(2,I)
C*     PZSUM = PZSUM + PCM(3,I)
C*     ESUM = ESUM + PCM(4,I)
C*     PRINT 1200, TYP(I), AMASS(I), (PCM(J,I), J=1,5)
1200   FORMAT (' ', A5, F16.5, ' | PX =', F15.7, ' | PY =', F15.7,
&           ' | PZ =', F15.7, ' | E =', F15.7,
&           ' | P =', F15.7)
1300   CONTINUE
C*     PRINT 1400, PXSUM, PYSUM, PZSUM, ESUM
1400   FORMAT (' ', 131('-'), /,
&           ' ', 'SUM: ', 24X, F15.7, 7X, F15.7, 7X, F15.7, 7X, F15.7)
C*
C*   RETURN
C*   END
```

ANHANG B. PROGRAMMPAKET DETEKTOR

Das Programmpaket 'DETEKTOR' wurde in der höheren Programmiersprache FORTRAN77 geschrieben und an der IBM 3081/Siemens 7890 Grossrechenanlage des Rechenzentrums im Kernforschungszentrum Karlsruhe getestet. Es simuliert ein 4π -Detektorsystem zum Nachweis geladener und neutraler Teilchen, die bei der Proton - Antiproton Annihilation entstehen.

PROGRAMMABLAUF

Im Hauptprogramm 'FOURPI' werden zunächst Variablen deklariert und mit Default- bzw. Startwerten versehen.⁹

Im Anschluß daran erfolgt die Initialisierung der Histogramme (Unterprogramm HBOOK). Zur weiteren Beschreibung sei auf den Abschnitt 'Programmtechnische Realisierung' dieser Arbeit verwiesen.

⁹ Ohne Zutun des Benutzers ist als Default das 'STANDARD-DETEKTOR-SYSTEM' implementiert.

EINGABEDATEN

Mit dem Programmpaket 'DETEKTOR' ist es dem Benutzer möglich, weitreichende Änderungen in der Geometrie des simulierten Detektorsystems vorzunehmen und deren Auswirkungen auf das Nachweisvermögen des Detektorsystems zu testen.

Die Kontrolle des Benutzers erstreckt sich auf folgende Eingabedaten:

Namelist 'INPUT'

Die Parameter zur Steuerung des Programmablaufs werden durch die Namelist 'INPUT' übergeben. Folgende Parameter sind definiert:

Gammadetektor

DMODUL Durchmesser eines Moduls (cm)
NMODUL Zahl der Module
RADIUS Innenradius der Kristallkugel (cm)
SIZE Oberfläche (Grad)
BSIZE Strahleinlaß(Grad)
MEDIUM Name des verwendeten Materials
ERESOL Energieauflösung (FWHM) (%)
EO Gültig bei dieser Energie (MeV)
EXPONT Exponent der Energieabhängigkeit
ANGRES Räumliches Auflösungsvermögen (Grad)
ETHRSH Nachweisschwelle (MeV)

Driftkammer

CHMLEN Länge der Kammer (cm)
CHMDIA Durchmesser der Kammer (cm)
SUPPLY Öffnung für Kabel etc. (Grad)
PRESOL Impulsauflösung (FWHM) (%)
PO Gültig bei diesem Impuls (MeV/c)
PTHRSH Nachweisschwelle (MeV/c)
BFIELD Stärke des Magnetfeldes (Tesla)
RESLEN Ortsauflösung in Strahlrichtung (FWHM) (cm)
RESDIA Ortsauflösung senkrecht (x,y) (FWHM) (cm)

Target

TGTDIA	Durchmesser (cm)
TGTLEN	Länge (cm)
TGTDEN	Dichte (g/cm ³)
TGTHOL	Größe der Targethalterung (Grad)
BMEANX	Maximum der Stopverteilung in x-Richtung (cm)
FWHMX	Breite der Stopverteilung in x-Richtung (cm)
BMEANY	Maximum der Stopverteilung in y-Richtung (cm)
FWHMY	Breite der Stopverteilung in y-Richtung (cm)
BMEANZ	Maximum der Stopverteilung in z-Richtung (cm)
FWHMZ	Breite der Stopverteilung in z-Richtung (cm)

Optionen

PIWIN	Breite des Fensters für die Rekonstruktion der neutralen Pionen (MeV)
ENECUT	Breite des Energiefensters (Energiesatzüberprüfung)
MOMCUT	Breite des Impulsfensters (Impulssatzüberprüfung)
MAXEVT	Zahl der zu verarbeitenden Annihilationen
NTAPE	Zahl der verwendeten Magnetbänder
LUNIN	Dateinummer der Magnetbanddatei
NLIST	Anzahl auszudruckender Ereignisse (Event-Dump)
HISPRT	Ausdrucken der Histogramme
NCHAN	Anzahl der Kanäle pro Histogramm
ISTORE	Dateinummer zum Ablegen der Histogramme
ID	Untersuchung eines spezifischen Kanals (0 : Alle Kanäle)

Ein Beispiel für eine Benutzereingabe könnte folgendermaßen aussehen:

```
&INPUT
MEDIUM = 'NA - JODID',
RESLEN = 0.46,
RESDIA = 0.1725,
BFIELD = 0.0,
ERESOL = 10.2,
EO = 130.,
EXPONT = 0.25,
NMODUL = 700,
NTAPE = 2,
NCHAN = 200,
ISTORE = 10
&END
```

Soll das 'STANDARD-DETEKTOR-SYSTEM' simuliert werden, so genügt als Eingabe:

```
&INPUT &END
```

da alle Detektorparameter mit den Werten des 'STANDARD-DETEKTOR-SYSTEMS' vorbelegt sind (siehe Quelltext 'Program FOURPI').

EVENTDUMP

Beim Entwurf des Programmpakets 'DETEKTOR' wurde darauf Wert gelegt, daß der Verlauf der Verarbeitung eines einzelnen Ereignisses durch den Benutzer leicht nachvollzogen werden konnte. Durch Angabe des 'NLIST'-Parameters erhält man folgende Druckerliste:

* EVENT NO. 30 CHANNEL NO. 2 NUMBER OF PARTICLES: 8 *

PI+ | PX = -223.2213440 | PY = -67.8496704 | PZ = 118.8237610 | E = 296.6982422 | P = 261.8212891
PI- | PX = 384.0202637 | PY = -103.4294739 | PZ = 333.1965332 | E = 537.2788086 | P = 518.8344727
GAMMA | PX = 33.6818695 | PY = 85.2217102 | PZ = -94.0963593 | E = 131.3443298 | P = 131.3443298
GAMMA | PX = -0.0945085 | PY = 102.1879425 | PZ = 37.4104156 | E = 108.8206024 | P = 108.8206024
GAMMA | PX = 36.7953033 | PY = 53.5496674 | PZ = 68.3807220 | E = 94.3259583 | P = 94.3259583
GAMMA | PX = 33.9596710 | PY = 143.3078918 | PZ = -49.4068451 | E = 155.3430023 | P = 155.3430023
GAMMA | PX = -69.0149994 | PY = -139.9760284 | PZ = -162.0298157 | E = 224.9667053 | P = 224.9667053
GAMMA | PX = -196.1264191 | PY = -73.0120544 | PZ = -252.2785187 | E = 327.7814941 | P = 327.7814941

VERTEX OF THIS EVENT: 0.006 0.176 -0.608 (CM)

A) FOLLOW THE CHARGED PARTICLES:

1 | DIR.COSINES: -0.862 -0.256 0.437 | PASSES AT POINT : -24.383 -7.237 12.374 (CM)
1 | MOMENTUM : -222.109 -66.050 117.448 | ENERGY: 294.904 | DELTA: 0.11 DEG.
2 | DIR.COSINES: 0.751 -0.196 0.630 | PASSES AT POINT : 21.250 -5.545 17.824 (CM)
2 | MOMENTUM : 377.615 -100.441 329.727 | ENERGY: 529.982 | DELTA: 0.99 DEG.

B) FOLLOW THE GAMMAS:

3 | DIR.COSINES: 0.252 0.644 -0.723 | HITS DETECTOR AT: 8.683 22.130 -24.848 (CM)
3 | DIR.COSINES: 0.233 0.668 -0.707 | DELTA: 1.97 DEG. ENERGY: 130.039
4 | DIR.COSINES: -0.001 0.945 0.326 | HITS DETECTOR AT: -0.024 32.504 11.227 (CM)
4 | DIR.COSINES: -0.008 0.940 0.342 | DELTA: 1.02 DEG. ENERGY: 113.466
5 | DIR.COSINES: 0.394 0.578 0.714 | HITS DETECTOR AT: 13.551 19.889 24.564 (CM)
5 | DIR.COSINES: 0.389 0.590 0.707 | DELTA: 0.84 DEG. ENERGY: 93.187
6 | DIR.COSINES: 0.216 0.918 -0.332 | HITS DETECTOR AT: 7.445 31.567 -11.431 (CM)
6 | DIR.COSINES: 0.251 0.905 -0.342 | DELTA: 2.18 DEG. ENERGY: 161.293
7 | DIR.COSINES: -0.304 -0.611 -0.731 | HITS DETECTOR AT: -10.442 -21.014 -25.138 (CM)
7 | DIR.COSINES: -0.348 -0.615 -0.707 | DELTA: 2.91 DEG. ENERGY: 217.790
8 | DIR.COSINES: -0.591 -0.215 -0.778 | HITS DETECTOR AT: -20.314 -7.388 -26.746 (CM)
8 | DIR.COSINES: -0.595 -0.244 -0.766 | DELTA: 1.83 DEG. ENERGY: 321.904

C) RECONSTRUCTION OF NEUTRAL PIONS:

INVARIANT MASS OF:GAMMAGAMMA
WINDOW SET FROM:85.00000000000000 TO 185.00000000000000
138.691430400655616
REC. MASS SORTED INTO:10
142.440896840369852
REC. MASS SORTED INTO:10
24.9813796410387781

D) RECONSTRUCTION OF CHARGED PIONS:

MOMENTUM MEASUREMENT IN CHAMBER.

E) SORT THE PARTICLES AND TEST EVENT:

PIO | PX = 29.8260552 | PY = 189.9273579 | PZ = -55.6559770 | E = 243.5054321 | P = 200.1489011
PIO | PX = 75.7499164 | PY = 196.3371511 | PZ = 13.5677514 | E = 254.4795074 | P = 210.8800859
PI- | PX = 377.6154280 | PY = -100.4414726 | PZ = 329.7267468 | E = 529.9821496 | P = 511.2745130
PI+ | PX = -222.1086270 | PY = -66.0500036 | PZ = 117.4476778 | E = 294.9038023 | P = 259.7860700
GAMMA | PX = -78.3479492 | PY = -138.4430827 | PZ = -159.0758631 | E = 217.7899170 | P = 224.9667053
GAMMA | PX = -194.8848631 | PY = -80.0720940 | PZ = -251.0957144 | E = 321.9042969 | P = 327.7814941

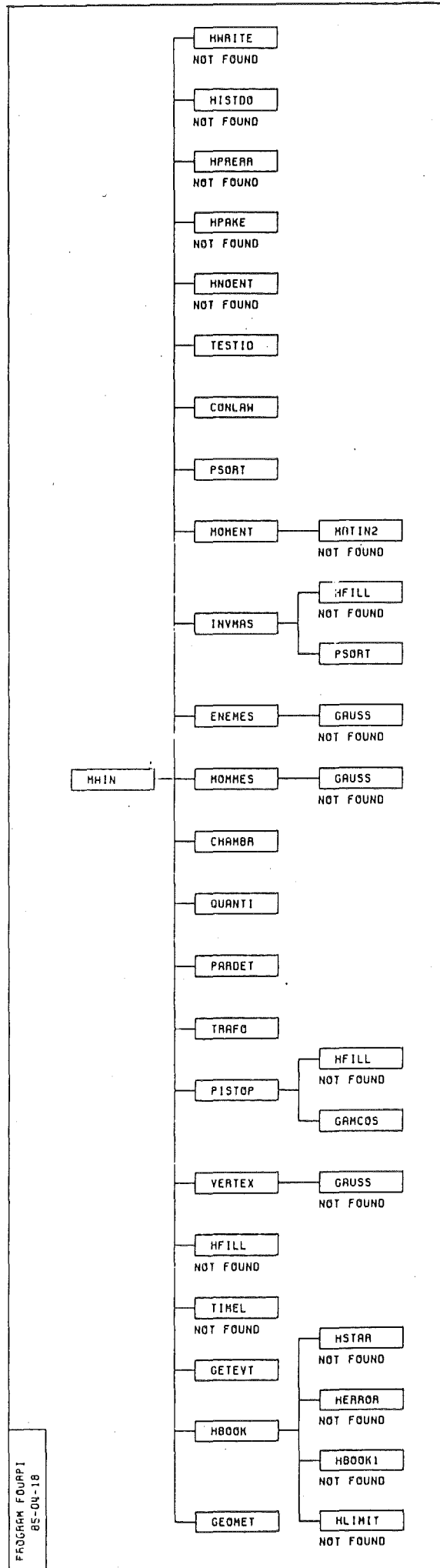
MISSING PX = -12.1500398 | PY = 1.2578561 | PZ = -5.0853785 | E = 13.9949532

PATTERN RECOGNITION : ID NO.72 F

F) CALCULATE INVARIANT MASSES :

ID=72 => E0
INVARIANT MASS OF:GAMMAGAMMA
WINDOW SET FROM:518.800112906564038 TO 578.800112906564038
24.9813796410387781
INVARIANT MASS OF:PIO ETA0
WINDOW SET FROM:923.00000000000000 TO 1043.00000000000000
INVARIANT MASS OF:PIO DELTA
WINDOW SET FROM:0.0 TO 2000.00000000000000

ABLAUFDIAGRAM



PROGRAM FOURPI
85-04-18

QUELLPROGRAMM

```
C*****
C*****
PROGRAM FOURPI
C**
C**          >>>>  VERSION...: DOUBLE PRECISION  <<<<<
C**          LANGUAGE...:FORTRAN 77
C**          PROGRAMMER:MARCEL KUNZE/KFK
C**          VERS 8.0/05-JAN-85(MK)
C**
C** THIS IS A PROGRAM TO SIMULATE A 4-PI DETECTOR (CRYSTAL-BALL) .
C** ONE BEAM INJECTION HOLE (Z-AXIS) RECTANGULAR TO THE SLOT FOR
C** CHAMBER SUPPLY .
C**
C** ==>  DETEKTOR-PARAMETERS: VIA NAMELIST 'INPUT' :
C**
C**
C** DETECTOR SIZE
C** *****
C** DMODUL : DIAMETER OF MODUL ..... (CM)
C** NMODUL : NUMBER OF MODULES ..... (-)
C** RADIUS : INNER RADIUS OF DETECTOR SYSTEM .... (CM)
C** SIZE   : SIZE OF DETEKTOR AREA ..... (DEGREES)
C** BSIZE  : SIZE OF BEAM INJECTION HOLE ..... (DEGREES)
C**
C**
C** GAMMA DETECTOR
C** *****
C** MEDIUM : MEDIUM ..... (NAME)
C** ERESOL : ENERGY RESOLUTION (FWHM) ..... (}
C** EO     : VALID FOR THIS ENERGY ..... (MEV)
C** EXPONT : EXPONENT FOR ENERGY DEPENDENCE..... (-)
C** ANGRES : ANGLE RESOLUTION ..... (DEGREES)
C** ETHRSH : ENERGY THRESHOLD ..... (MEV)
C**
C**
C** CHARGED PARTICLE DRIFT CHAMBER
C** *****
C** CHMLEN : LENGTH OF CHAMBER ..... (CM)
C** CHMDIA : DIAMETER OF CHAMBER ..... (CM)
C** SUPPLY : SLOT FOR CABLES ETC. .... (DEGREES)
C** PRESOL : MOMENTUM RESOLUTION FOR PIONS (FWHM) (}
C** PO     : VALID FOR THIS MOMENTUM ..... (MEV/C)
C** PTHRSH : MOMENTUM THRESHOLD FOR PIONS ..... (MEV/C)
C** BFIELD : STRENGTH OF MAGNETIC FIELD ..... (TESLA)
C** RESLEN : RESOLUTION IN Z - DIRECTION (FWHM) (CM)
C** RESDIA : RESOLUTION IN X,Y - DIRECTION (FWHM) (CM)
C**
C**
C** BEAM DIMENSION
C** *****
C** BMEANX : MEAN VALUE OF GAUSS DISTRIBUTION(X). (CM)
C** FWHMX  : FWHM (X) ..... (CM)
C** BMEANY : MEAN VALUE OF GAUSS DISTRIBUTION(Y). (CM)
C** FWHMY  : FWHM (Y) ..... (CM)
C** BMEANZ : MEAN VALUE OF GAUSS DISTRIBUTION(Z). (CM)
C** FWHMZ  : FWHM (Z) ..... (CM)
C**
C*****
C*****
```

```
C**
C**          CYLINDRICAL TARGET
C**          *****
C**          TGT DIA : DIAMETER ..... (CM)
C**          TGT LEN : LENGTH ..... (CM)
C**          TGT DEN : DENSITY OF TARGET-MEDIUM ..... (G/CM**3)
C**          TGT HOL : DIMENSION OF TARGET-HOLDER ..... (DEGREES)
C**
C**          OPTIONS
C**          *****
C**          PIWIN  : WINDOW FOR P10-MASS ..... (MEV)
C**          ENECUT : WIDTH OF ENERGY CUT ..... (MEV)
C**          MOMCUT : WIDTH OF MOMENTUM CUT ..... (MEV/C)
C**          MAXEVT : NUMBER OF EVENTS TO BE PROCESSED .
C**          NTAPE  : NUMBER OF TAPES TO BE PROCESSED ..
C**          LUNIN  : FORTRAN UNIT FOR DATA INPUT .....
C**          NLIST  : NUMBER OF EVENTS TO BE DUMPED ....
C**          HISPRT : PRINT HISTOGRAMS ON LINE PRINTER..
C**          NCHAN  : NUMBER OF BINS FOR THE HISTOGRAMS.
C**          ISTORE : UNIT FOR SAVING THE HISTOGRAMS ... (0 : NOP)
C**          ID     : SELECT SPECIAL CHANNEL ..... (0 : ALL CH.)
C**
C**
C**          *****
C**          THE FOLLOWING COMPUTATIONAL MODI ARE POSSIBLE :
C**          A) ANGLE RESOLUTION ==> RADIUS      + NUMB. OF MODULES
C**          B) RADIUS           ==> ANGLE RES. + NUMB. OF MODULES
C**          C) NUMB. OF MODULES ==> ANGLE RES. + RADIUS
C**          *****
C*          VARIABLE TABLE ..... *
C**          *****
C*
C* A) GAMMA - COUNTERS
C*
C*          NGATOT : NUMBER OF GAMMAS FROM TAPE (TOTAL)
C*          NGAMMA : NUMBER OF GAMMAS FOR CURRENT EVENT
C*          NPIGAM : NUMBER OF GAMMAS FROM : PI- + P -> N + GAMMA
C*          NPIPIO : NUMBER OF GAMMAS FROM : PI- + P -> N + P10
C*          NGADEV : NUMBER OF GAMMAS DETECTED FOR CURRENT EVENT
C*          NGADET : NUMBER OF GAMMAS DETECTED (TOTAL)
C*          NOREGA : NUMBER OF GAMMAS THAT COULD NOT BE RECONSTRUCTED
C*          NSIGAR : NUMBER OF SINGLE GAMMAS AFTER RECONSTRUCTION
C*          INHOLE : NUMBER OF LOST GAMMAS (CURRENT EVENT)
C*          ILOST  : NUMBER OF LOST GAMMAS
C*          IBARGA : NUMBER OF MONOENERGETIC GAMMAS (BARYONIUM)
C*
C* B) EVENT - COUNTERS
C*
C*          KNT     : NUMBER OF EVENTS
C*          NOTRIG  : NUMBER OF EVENTS WITHOUT TRIGGER
C*          IBARST  : NUMBER OF BARYONIUM STATES (NEUTRAL+CHARGED)
C*
C* C) PION - COUNTERS
C*
C*          NPITAP  : NUMBER OF P10'S ON TAPE
C*          NPITOT  : NUMBER OF P10'S FROM 'TRIGGER - EVENTS'
C*          NPIREC  : NUMBER OF RECONSTRUCTED P10'S IN TOTAL
C*
C* D) FLAGS
```

```

C*
C*   OFF      : STOP PROCESSING EVENTS BECAUSE ...
C*   EOF      : END OF INPUT FILE REACHED
C*   TIMEOUT  : RUNTIME LIMIT REACHED
C*   MAXIMO   : MAXIMUM NUMBER OF EVENTS PROCESSED
C*   NIXEVT   : EVENT WITH IDENTIFIER OUT OF RANGE
C*   MISMAT   : EVENT WITH IDENTIFIER MISMATCH
C*   GOTID    : EVENT WITH TRUE IDENTIFIER FOUND
C*
C*   VETO     : A CHARGED PARTICLE WENT OFF THE CHAMBER
C*   DETECT   : A GAMMA HAS BEEN DETECTED
C*   TRIGGR   : THE DETECTOR IS TRIGGERED
C*   BARYON   : EVENT WITH BARYONIUM STATE (NEUTRAL + CHARGED)
C*   MOGABA   : EVENT WITH BARYONIUM STATE (NEUTRAL ONLY)
C*   NOCHAR   : EVENT WITH NO CHARGED PARTICLES
C*   CALCUL   : RECONSTRUCTION OF CHARGED PARTICLES KINEMATICS
C*
C*   PTTGT    : TARGET IS POINT-LIKE
C*   SUTGT    : TARGET IS TOO LARGE
C*   NOCHAM   : NO CHAMBER INSTALLED (ONLY ANTI-COUNTERS)
C*   FIELD    : MAGNETIC FIELD TURNED ON (MOMENTUM MEASUREMENT)
C*
C*   LIST     : EVENT DUMP TURNED ON
C*   HISPRT   : PRINT THE CREATED HISTOGRAMS ON LINE PRINTER
C*
C*****
C*   TYPE DECLARATIONS.....*
C*****
C*
C*   IMPLICIT REAL*8 (A-H,O-Z)
C*           REAL*8 MOMCUT
C*           REAL*4 TRIGNO,ERRID
C*
C*   COMMON /FLAGOP/ LIST
C*   COMMON /GEO   / DMODUL,CZDETE,CZBEAM,CZTARG,CZCHAM,CYSUPP,
C*   &           RADIUS,ANGRES,NMODUL
C*   COMMON /PIMDAT/ NPIGAM,NPIPIO
C*   COMMON /TARGET/ BMEANX,FWHMX,BMEANY,FWHMY,BMEANZ,FWHMZ,
C*   &           TGTRAD,TGTHLF,TGTDEN
C*
C*   INTEGER PILOST,PART(4)
C*
C*   LOGICAL TIMEOUT,OFF,EOF,MAXIMO,NIXEVT
C*   LOGICAL TRIGGR,DETECT,MISMAT,GOTID
C*   LOGICAL PTTGT,SUTGT,NOCHAM,FIELD
C*   LOGICAL LIST,SELECT,VETO,HISPRT
C*   LOGICAL NOCHAR,CALCUL,NIXDO
C*   LOGICAL INVMOX,INVMOY,INVMOZ
C*   LOGICAL INVENE,INVMOM
C*   LOGICAL BARYON,MOGABA
C*
C*   CHARACTER TYP(50)*5,PARTIC(3)*5,MEDIUM*20
C*
C*   DIMENSION PCM(5,50)
C*   DIMENSION TRIGNO(0:100)
C*   DIMENSION ERRID (0:100)
C*
C*****
C*   SET INITIAL VALUES .....*
```

```
C*****
C*
PARAMETER (PI = 3.1415962653589793)
PARAMETER (PIMASS = 139.5688)
C*
DATA KNT/0/ ,IBARST/0/,PILOST/0/,
& NOTRIG/0/,NGATOT/0/,NSIGAR/0/,NOREGA/0/,
& ILOST/0/ ,IBARGA/0/,
& TRIGNO/101*0./,ERRID/101*0./,
& PCM/250*0.DO/,
& TYP/50*' /,
& MEDIUM/' /
C*
DATA PRCTG1/0./,PRCTG2/0./,PRCTG3/0./,PRCTG4/0./,
& PRCTG6/0./,PRCTG7/0./
C*
DATA EOF /.FALSE./
DATA OFF /.FALSE./
C*
NAMELIST /INPUT / SIZE,BSIZE,
& MEDIUM,ANGRES,ERESOL,EO,EXPONT,ETHRSH,
& CHMLN,CHMDIA,PRESOL,SUPPLY,PO,PTHRSH,
& BFIELD,RESLEN,RESDIA,PIWIN,
& MAXEVT,ISTORE,ID,NTAPE,LUNIN,
& DMODUL,RADIUS,NMODUL,
& BMEANX,FWHMX,BMEANY,FWHMY,BMEANZ,FWHMZ,
& TGT DIA,TGTLEN,TGTDEN,TGTHOL,
& NLIST,NCHAN,ENECUT,MOMCUT,HISPR
C*
NPIGAM = 0
NPIPIO = 0
C*
C*****
C* SET DEFAULT PARAMETERS FOR STANDARD DETECTOR SYSTEM ... *
C*****
C*
C* GAMMA DETECTOR
C*
C*
C*
MEDIUM = 'CS - JODID'
SIZE = 360.0
BSIZE = 5.0
ERESOL = 7.0
EXPONT = 0.25
EO = 100.0
ETHRSH = 10.0
DMODUL = 3.0
ANGRES = 5.0
RADIUS = 0.0
NMODUL = 0
C*
C* CHARGED PARTICLE CHAMBER
C*
C*
C*
CHMLN = 40.0
CHMDIA = 40.0
BFIELD = 1.0
RESLEN = 0.5
RESDIA = 0.025
```

```
PRESOL = 5.0
PO      = 200.0
PTHRSH = 50.0
SUPPLY  = 10.0

C*
C* TARGET
C* .....
C*
C* TGT DIA = 3.0
C* TGT LEN = 3.0
C* TGT DEN = 0.0708
C* TGT HOL = 0.0

C*
C* BEAM
C* ....
C*
C* BMEANX = 0.0
C* FWHMX  = 1.0
C* BMEANY = 0.0
C* FWHMY  = 1.0
C* BMEANZ = 0.0
C* FWHMZ  = 1.0

C*
C* PIO RECONSTRUCTION
C* .....
C*
C* PIWIN = 50.0

C*
C* OPTIONS
C* .....
C*
C* MAXEVT =1600000
C* NTAPE  = 1
C* LUNIN  = 2
C* ID     = 0
C* ISTORE = 0
C* NLIST  = 20
C* NCHAN  = 100
C* ENECUT = 50.
C* MOMCUT = 50.
C* HISPRT = .TRUE.

C*
C* *****
C* START THE SIMULATION PROGRAM ..... *
C* *****
C*
C* RESET RUNTIME FLAG
C* .....
C*
C* TOSTRT = ZEIT(0)

C*
C*
C* READ DETEKTOR PARAMETERS
C* .....
C*
C* READ(5,INPUT)

C*
C* TARGET DIMENSIONS
```

```

C* .....
C*
IF (TGTDIA.GE.0.5.AND.TGTLEN.GE.0.5) THEN
  PTTGT = .FALSE.
  TGTRAD = TGTDIA/2.
  TGTHLF = TGTLEN/2.
ELSE
  PTTGT = .TRUE.
  BMEANX = 0.
  HWHMX = 0.
  BMEANY = 0.
  FWHMY = 0.
  BMEANZ = 0.
  FWHMZ = 0.
  TGTHLF = 0.
  TGTRAD = 0.
ENDIF

C*
CZTARG = COS(TGTHOL/360.*PI)

C*
C*
C* CHARGED PARTICLE CHAMBER DIMENSIONS
C* .....
C*
FIELD = (BFIELD.GT.0.0)
IF (.NOT.FIELD) PRESOL = 0.DO

C*
IF (CHMDIA.EQ.0.0.OR.CHMLN.EQ.0.0) THEN
  NOCHAM = .TRUE.
  CZCHAM = 1.DO
ELSE
  NOCHAM = .FALSE.
  ALPHA = ATAN(CHMDIA/CHMLN)
  CZCHAM = COS(ALPHA)
ENDIF

C*
CHMRAD = 0.5 * SQRT(CHMLN**2 + CHMDIA**2)
CYSUPP = COS(SUPPLY/360.*PI)

C*
C* PRINT SIMULATION PARAMETERS
C* .....
C*
PRINT 100
100 FORMAT ('1',125('*'),/,
&          '0', '*****',
&          ' ', '***** PROGRAM FOURPI STARTED *****',
&          ' ', '*****',
&          '0',125('*'))
PRINT 200
200 FORMAT ('0', 'A 4-PI-DETECTOR-SYSTEM (CRYSTAL-BALL) IS SIMULATED '
&          ', 'ACCORDING TO THE FOLLOWING PARAMETERS:')
C*
PRINT 300, SIZE, BSIZE, TGTHOL, SUPPLY
C*
300 FORMAT('0', 'GEOMETRICAL DATA', /
&          ', ', '*****', /
&          ', ', 'SIZE OF DETECTOR .....: ', F9.4, ' DEGREES', /
&          ', ', 'INJECTION HOLE FOR BEAM .....: ', F9.4, ' DEGREES', /
&          ', ', 'TARGET HOLDER .....: ', F9.4, ' DEGREES', /

```



```

&      , ' ', 'SLOT FOR CHAMBER SUPPLY & READ OUT: ', F9.4, ' DEGREES')
C*
PRINT 400, MEDIUM, ERESOL, EO, EXPONT, ETHRSH, ANGRES
400  FORMAT('0', 'GAMMA DETECTOR', /,
&      ' ', '*****', /,
&      ' ', 'MEDIUM .....: ', A20, /,
&      ' ', 'ENERGY RESOLUTION (FWHM) .....: ', F9.4, ' %',
&      ' ', 'AT ', F5.1, ' MEV, PROP. TO 1/E**', F5.2, /,
&      ' ', 'ENERGY THRESHOLD .....: ', F9.4, ' MEV', /,
&      ' ', 'ANGLE RESOLUTION .....: ', F9.4, ' DEGREES')
C*
PRINT 500, TGTLEN, TGTDIA, BMEANX, FWHMX, BMEANY, FWHMY, BMEANZ, FWHMZ,
&      TGTDEN
500  FORMAT('0', 'TARGET', /,
&      ' ', '*****', /,
&      ' ', 'LENGTH OF TARGET .....: ', F9.4, ' CM', /,
&      ' ', 'DIAMETER OF TARGET .....: ', F9.4, ' CM', /,
&      ' ', 'MAIN VALUE OF VERTEX (X) .....: ', F9.4, ' CM', /,
&      ' ', 'FWHM (X) .....: ', F9.4, ' CM', /,
&      ' ', 'MAIN VALUE OF VERTEX (Y) .....: ', F9.4, ' CM', /,
&      ' ', 'FWHM (Y) .....: ', F9.4, ' CM', /,
&      ' ', 'MAIN VALUE OF VERTEX (Z) .....: ', F9.4, ' CM', /,
&      ' ', 'FWHM (Z) .....: ', F9.4, ' CM', /,
&      ' ', 'DENSITY OF TARGET MATERIAL .....: ', F9.4, ' G/CM**3')
C*
IF (PTTGT) PRINT 600
600  FORMAT ('0', 'TARGET ASSUMED TO BE POINT-LIKE .')
C*
PRINT 700, CHMLN, CHMDIA, RESLEN, RESDIA, BFIELD, PRESOL,
&      PO, PO, PO, PTHRSH
700  FORMAT('0', 'CHARGED PARTICLE DRIFT CHAMBER', /,
&      ' ', '*****', /,
&      ' ', 'LENGTH OF CHAMBER .....: ', F9.4, ' CM', /,
&      ' ', 'DIAMETER OF CHAMBER .....: ', F9.4, ' CM', /,
&      ' ', 'RESOLUTION IN Z - DIRECTION .....: ', F9.4, ' CM', /,
&      ' ', 'RESOLUTION IN X,Y - DIRECTION .....: ', F9.4, ' CM', /,
&      ' ', 'MAGNETIC FIELD .....: ', F9.4, ' TESLA', /,
&      ' ', 'MOMENTUM RESOLUTION FOR PIONS .....: ', F9.4, ' %',
&      ' ', 'AT ', F5.1, ' MEV/C PROP. TO 1/3. * ( 1 + P / ', F5.1,
&      ' + ', F5.1, ' / P ).', /,
&      ' ', 'MOMENTUM THRESHOLD .....: ', F9.4, ' MEV/C')
C*
PRINT*
PRINT*, 'NOTE : ALL RESOLUTIONS ARE FWHM.'
C*
IF (NOCHAM) PRINT 800
800  FORMAT ('0', 'NO CHARGED PARTICLE CHAMBER INSTALLED .')
C*
PRINT 900, PIWIN
900  FORMAT('0', 'PIO RECONSTRUCTION PARAMETERS', /,
&      ' ', '*****', /,
&      ' ', 'WINDOW FOR PIO-MASS .....: ', F9.4, ' MEV')
C*
PRINT 1000, MAXEVT, LUNIN, NTAPE, NLIST, ISTORE, NCHAN, ID, ENECUT, MOMCUT
1000 FORMAT('0', 'OPTIONS', /,
&      ' ', '*****', /,
&      ' ', 'NUMBER OF EVENTS TO BE PROCESSED ..: ', I8, /,
&      ' ', 'FORTRAN UNIT FOR DATA INPUT .....: ', I8, /,
&      ' ', 'NUMBER OF TAPES USED .....: ', I8, /,

```

```

&      ' ', 'NUMBER OF EVENTS TO BE DUMPED .....: ', I8, /,
&      ' ', 'UNIT FOR SAVING THE HISTOGRAMS ....: ', I8, /,
&      ' ', 'NUMBER OF BINS FOR THE HISTOGRAMS : ', I8, /,
&      ' ', 'SELECT SPECIAL CHANNEL .....: ', I8, /,
&      ' ', 'WIDTH OF ENERGY CUT .....: ', F8.3, ' MEV', /,
&      ' ', 'WIDTH OF MOMENTUM CUT .....: ', F8.3, ' MEV/C')

C*
C*
C*      GET THE DETECTOR'S GEOMETRY
C*      .....
C*
      CZDETE = COS( SIZE/360.*PI)
      CZBEAM = COS(BSIZE/360.*PI)
C*
      CALL GEOMET
C*
      IF (NOCHAM) THEN
          CHMRAD = RADIUS
          SUPPLY = 0.0
      ENDIF
C*
      SUTGT = (TGTHLF.GT.RADIUS.OR.TGTRAD.GT.RADIUS)
      IF (SUTGT) THEN
          PRINT 1100
1100      FORMAT ('0', '>>> TARGET IS TOO LARGE TO FIT INTO DETECTOR .')
          STOP
      ENDIF
C*
C*      INITIALIZE HISTOGRAM BOOKING
C*      .....
C*
      CALL HBOOK (NCHAN)
C*
C*****
C**      GRAND PARTICLE LOOP                                     **
C*****
C*
      PRINT 1300
1300      FORMAT ('1', 'PROCESSING THE EVENTS', /,
&      ' ', '*****')
C*
      DO 3400 UNTIL (OFF)
C*
      SELECT = .TRUE.
C*
      DO 1400 WHILE (SELECT)
C*
      KNTP1 = KNT + 1
C*
      SET LIST-EVENT FLAG ...
C*
      LIST = (KNT.LT.NLIST)
C*
C*
C*      READ THE PARTICLES' KINEMATICS FROM FORTRAN UNIT #LUNIN
C*      .....
C*
      CALL GETEVT (IDEVNT,NPTOT,PCM,TYP,EOF,LUNIN)
      IF (EOF.AND.NTAPE.GT.1) THEN

```

```
      EOF = .FALSE.
      NTAPE = NTAPE - 1
ENDIF
C*
C*
C* TIMING
C* .....
C*
      CALL TIMEL (TLEFT)
      TIMEOUT = (TLEFT.LE.5.)
      NIXEVT = (IDEVNT.LE.0)
      MAXIMO = (KNT.EQ.MAXEVT)
C*
      OFF = (TIMEOUT.OR.NIXEVT.OR.MAXIMO.OR.EOF)
C*
C*
C* SELECT SPECIAL CHANNEL ???
C* .....
C*
      SELECT = ((ID.GT.0).AND.(IDEVNT.NE.ID).AND.(.NOT.OFF))
1400 CONTINUE
C*
      IF (.NOT.OFF) THEN
      KNT = KNTP1
C*
C* COUNT THE EVENTS ...
C*
      CALL HFILL(24,FLOAT(IDEVNT),DUMMY,1.DO)
C*
C* CLASSIFY THE EVENT
C* .....
C*
C* IS IT AN EVENT WITH BARYONIUM STATE ???
C*
      BARYON = (IDEVNT.GE.54.AND.IDEVNT.LE.64)
C*
C* IS IT AN EVENT WITH MONOENERGETIC BARYONIUM GAMMA ???
C*
      MOGABA = (IDEVNT.GE.54.AND.IDEVNT.LE.58)
C*
      IF (.NOT.PTTGT) THEN
C*
C* CHOOSE A VERTEX FOR THIS EVENT
C* .....
C*
      CALL VERTEX (X,Y,Z)
      CALL HFILL (3,X,DUMMY,1.DO)
      CALL HFILL (4,Y,DUMMY,1.DO)
      CALL HFILL (5,Z,DUMMY,1.DO)
      IF (LIST) PRINT 1500,X,Y,Z
1500 FORMAT('0','VERTEX OF THIS EVENT:',3F10.3,' (CM)')
C*
      ENDIF
C*
C* EXAMINE THE CHARGED PARTICLES
C* .....
C*
      IF (LIST) PRINT 1600
1600 FORMAT ('0','A) FOLLOW THE CHARGED PARTICLES:',/,
```



```

& RESLEN, RESDIA, DELTA, DETECT)
C*
C* IF (.NOT.DETECT) TYP(I) = 'PURGE'
C*
C* IF (LIST)
& PRINT 1900, I, PCM(1, I), PCM(2, I), PCM(3, I), PCM(4, I),
& DELTA*180./PI
1900 & FORMAT(' ', '#', I3, 3X, ' | MOMENTUM :', 3F10.3, ' | ',
& 10X, 'ENERGY:', F10.3, ' | DELTA:',
& F6.2, ' DEG.')
```

```

C*
C* ENDIF
C* (* OF .NOT.VETO *)
C*
C* ENDIF
C* (* OF .NOCHAM *)
C*
C* ENDIF
C* (* OF .NOT.DETECT *)
C*
C* ENDIF
C* (* OF CHARGED PARTICLES *)
C*
C* I = I + 1
2000 CONTINUE
C*
C* NOCHAR = (NCHAR.EQ.0)
C* CALCUL = (NCHAR.EQ.2.OR.NCHAR.EQ.4)
C* NIXDO = (.NOT.(NOCHAR.OR.CALCUL))
C* IF (NOCHAM.OR.(.NOT.FIELD)) VETO = (NIXDO.OR.VETO)
C*
C* IF (VETO) THEN
C*
C* IF (LIST) THEN
C* IF (NOCHAM.OR.(.NOT.FIELD).AND.NIXDO) THEN
C* PRINT*, 'EVENT VETOED . NUMBER OF PARTICLES NOT 0,2 OR 4.'
C* ELSE
C* PRINT*, 'EVENT VETOED . CHARGED PARTICLE(S) LOST.'
C* ENDIF
C* ENDIF
C*
C* PILOST = PILOST + 1
C* TRIGNO(IDEVNT) = TRIGNO(IDEVNT) + 1.
C* ELSE
C*
C* EXAMINE THE GAMMAS
C* .....
C*
C* IF (LIST) PRINT 2100
2100 & FORMAT ('0', 'B) FOLLOW THE GAMMAS:', '/',
& ' ', '*****')
```

```

C*
C* NGADEV = 0
C* NGAMMA = 0
C* INHOLE = 0
C* TRIGGR = .FALSE.
C*
C* DO 2400 I=1, NPTOT, 1
C* IF (TYP(I).EQ.'GAMMA') THEN
```

```
NGAMMA      = NGAMMA + 1
C*
IF (.NOT.PTTGT)
& CALL TRAFO(PCM(1,I),X,Y,Z,RADIUS)
IF (LIST) THEN
    CX      = PCM(1,I) / PCM(5,I)
    CY      = PCM(2,I) / PCM(5,I)
    CZ      = PCM(3,I) / PCM(5,I)
    XDETEC  = CX * RADIUS
    YDETEC  = CY * RADIUS
    ZDETEC  = CZ * RADIUS
    PRINT 2200,I,CX,CY,CZ,XDETEC,YDETEC,ZDETEC
2200    FORMAT(' ', '#', I3, 3X, ' | DIR.COSINES: ', 3F10.3, ' | ',
&          ', 'HITS DETECTOR AT: ', 3F10.3, ' (CM)')
ENDIF
C*
CALL PARDET (PCM(1,I),DETECT)
C*
TRIGGR = (TRIGGR.OR.DETECT)
C*
IF (.NOT.DETECT) THEN
    INHOLE = INHOLE + 1
    TYP(I) = 'PURGE'
ELSE
C*
C*      MONTE - CARLO ENERGY MEASUREMENT
C*      .....
C*
CALL ENEMES (PCM(4,I),ERESOL,E0,EXPONT,ETHRSH,DETECT)
C*
TRIGGR = (TRIGGR.OR.DETECT)
C*
IF (.NOT.DETECT) THEN
    TYP(I) = 'PURGE'
ELSE
C*
C*      MONTE - CARLO ANGLE RESOLUTION
C*      .....
C*
CALL QUANTI (PCM(1,I),DELTA)
    DELTA = DELTA * 180./PI
    CALL HFILL(12,DELTA,DUMMY,1.DO)
C*
IF (LIST) THEN
    CX = PCM(1,I)/PCM(5,I)
    CY = PCM(2,I)/PCM(5,I)
    CZ = PCM(3,I)/PCM(5,I)
    PRINT 2300,I,CX,CY,CZ,DELTA,PCM(4,I)
2300    FORMAT(' ', '#', I3, 3X, ' | DIR.COSINES: ', 3F10.3, ' | ',
&          'DELTA: ', F6.2, ' DEG. ', 3X, 'ENERGY: ', F10.3)
C*
ENDIF
C*
C*      GAMMA COUNTING AND HISTOGRAM BOOKING
C*
NGADEV = NGADEV + 1
CALL HFILL (01,PCM(4,I),DUMMY,1.DO)
```

```

                CALL HFILL (17,PCM(3,I)/PCM(5,I),DUMMY,1.DO)
C*
                ENDIF
C* =>            (* .NOT.DETECT BY ENERGY MEASUREMENT *)
                ENDIF
C* =>            (* .NOT.DETECT BY SOLID ANGLE *)
                ENDIF
C* =>            (* NO GAMMA *)
2400            CONTINUE
C*
C*            KEEP A RUNNING SUM OF
C*            .....
C*
C*            THE GAMMAS IN TOTAL ...
C*
                NGATOT = NGATOT + NGAMMA
C*
C*            THE PIO'S IN TOTAL ...
C*
                NPITAP = NPITAP + NGAMMA/2
C*
                IF (TRIGGR.AND.NGADEV.GT.1) THEN
C*
C*            RECONSTRUCTION OF NEUTRAL PIONS
C*            .....
C*
                IF (LIST) PRINT 2500
2500            FORMAT ('0','C) RECONSTRUCTION OF NEUTRAL PIONS: ',/,
&                ' ', '*****')
C*
C*            MAXIMUM NUMBER OF PIOS TO BE RECONSTRUCTED IS ...
C*
                NPIO = NGADEV/2
C*
                PARTIC(1) = 'GAMMA'
                PARTIC(2) = 'GAMMA'
                PARTIC(3) = ' '
                CALL INVMAS (PCM,TYP,NPTOT,10,135.DO,PIWIN,'PIO ',PARTIC)
                ENDIF
C*
C*
C*            CALCULATE CHARGED PARTICLES' KINEMATICS
C*            .....
C*
                IF (LIST) PRINT 2600
2600            FORMAT ('0','D) RECONSTRUCTION OF CHARGED PIONS: ',/,
&                ' ', '*****')
C*
C*            IF ((NOCHAM.OR.(.NOT.FIELD)).AND.CALCUL) THEN
C*
                CALL MOMENT (PCM,TYP,NPTOT)
C*
                ELSE
                IF (LIST) PRINT*, 'MOMENTUM MEASUREMENT IN CHAMBER.'
                ENDIF
C*            (* CALCUL *)
C*
C*            CHECK ENERGY CONSERVATION

```

```
C* .....
C*
IF (LIST) PRINT 2700
2700 FORMAT ('0','E') SORT THE PARTICLES AND TEST EVENT: ',/,
&      ' ', '*****')
C*
CALL PSORT (PCM,TYP,NPTOT)
C*
IF (LIST) THEN
  DO 2900 I=1,NPTOT,1
  PRINT 2800,TYP(I),(PCM(J,I),J=1,5)
2800   FORMAT(' ',A5,' | PX =',F15.7,' | PY =',F15.7,
&           ' | PZ =',F15.7,' | E =',F15.7,
&           ' | P =',F15.7)
2900   CONTINUE
ENDIF
C*
C* TEST CONSERVATION LAWS
C*
CALL CONLAW (PCM,NPTOT,DELTA,DELTAX,DELTAY,DELTAZ)
C*
IF (LIST) PRINT 3000,DELTAX,DELTAY,DELTAZ,DELTA
3000   FORMAT(' ',115('-'),/,
&           ' ', 'MISSING PX =',F15.7,' | PY =',F15.7,
&           ' | PZ =',F15.7,' | E =',F15.7,/)
C*
CALL HFILL (20,DELTA ,DUMMY,1.DO)
CALL HFILL (21,DELTAX,DUMMY,1.DO)
CALL HFILL (22,DELTAY,DUMMY,1.DO)
CALL HFILL (23,DELTAZ,DUMMY,1.DO)
C*
C* CHECK ENERGY CONSERVATION
C*
INVENE = (ABS(DELTA ).GT.ENECUT)
C*
C* CHECK MOMENTUM CONSERVATION
C*
INVMOX = (ABS(DELTAX).GT.MOMCUT)
INVMOY = (ABS(DELTAY).GT.MOMCUT)
INVMOZ = (ABS(DELTAZ).GT.MOMCUT)
INVMOM = (INVMOX.OR.INVMOY.OR.INVMOZ)
C*
VETO   = (INVENE.OR.INVMOM)
C*
IF (VETO) THEN
C*
  IF (LIST) THEN
    IF (INVENE) PRINT*, 'ENERGY CONSERVATION NOT VALID.'
    IF (INVMOX) PRINT*, 'MOMENTUM CONSERVATION (X) NOT VALID.'
    IF (INVMOY) PRINT*, 'MOMENTUM CONSERVATION (Y) NOT VALID.'
    IF (INVMOZ) PRINT*, 'MOMENTUM CONSERVATION (Z) NOT VALID.'
  ENDIF
C*
ELSE
C*
  HISTOGRAM BOOKING
C*
  .....
C*
  DO 3100 I=1,NPIO,1
```



```

3100 CALL HFILL (25,FLOAT(IDEVNT),DUMMY,1.DO)
      CONTINUE
C*
      PART(1) = 0
      PART(2) = 0
      PART(3) = 0
      PART(4) = 0
      DO 3200 I=1,NPTOT,1
C*
      IF (TYP(I).EQ.'PI+ ') THEN
          CALL HFILL (32,PCM(4,I),DUMMY,1.DO)
          PART(1) = PART(1) + 1
C*
      ELSEIF (TYP(I).EQ.'PI- ') THEN
          CALL HFILL (32,PCM(4,I),DUMMY,1.DO)
          PART(2) = PART(2) + 1
C*
      ELSEIF (TYP(I).EQ.'PIO ') THEN
          CALL HFILL (31,PCM(4,I),DUMMY,1.DO)
          CALL HFILL (26,FLOAT(IDEVNT),DUMMY,1.DO)
          PART(3) = PART(3) + 1
C*
      ELSEIF (TYP(I).EQ.'GAMMA') THEN
          CALL HFILL (02,PCM(4,I),DUMMY,1.DO)
          PART(4) = PART(4) + 1
          INXPOS = I
C*
      ENDIF
C*
3200 CONTINUE
C*
      BOOK THE SINGLE GAMMAS AFTER RECONSTRUCTION
C*
      IF (PART(4).EQ.1) THEN
          CALL HFILL (11,PCM(4,INXPOS),0.,1.DO)
          CALL HFILL (18,PCM(3,INXPOS)/PCM(5,INXPOS),DUMMY,1.DO)
      ENDIF
C*
C* TEST EVENT ID
C* .....
C*
      CALL TESTID (PART,IDFIX,IDEVNT,MISMAT)
C*
      CALL HFILL (27,FLOAT(IDFIX),DUMMY,1.DO)
      IF (MISMAT) ERRID(IDFIX) = ERRID(IDFIX) + 1.DO
      GOTID = (.NOT.MISMAT)
      IF (LIST) PRINT*, 'PATTERN RECOGNITION : ID NO.',IDFIX,GOTID
C*
C* CALCULATION OF INVARIANT MASSES
C* .....
C*
      IF (LIST.AND.IDFIX.GT.0) PRINT 3300
3300 FORMAT ('0','F) CALCULATE INVARIANT MASSES :',/,
      &      ' ', '*****')
C*
      IF (IDFIX.EQ.8) THEN
          IF (LIST) PRINT*, 'ID=',IDFIX, ' => RHO OR FO'
          PARTIC(1) = 'PI- '
          PARTIC(2) = 'PI+ '

```

```

PARTIC(3) = '      '
CALL INVMAS (PCM,TYP,NPTOT,15,0.DO,0.DO,'RHO  ',PARTIC)
IF (MISMAT)
& CALL HFILL (28,FLOAT(IDEVNT),DUMMY,1.DO)
C*
ELSEIF (IDFIX.EQ.16) THEN
  IF (LIST) PRINT*,'ID=',IDFIX,' => ETA OR OMEGA '
  PARTIC(1) = 'PIO  '
  PARTIC(2) = 'PI-  '
  PARTIC(3) = 'PI+  '
  CALL INVMAS (PCM,TYP,NPTOT,16,0.DO,0.DO,'ETA  ',PARTIC)
  IF (MISMAT)
& CALL HFILL (29,FLOAT(IDEVNT),DUMMY,1.DO)
C*
ELSEIF (IDFIX.EQ.54) THEN
  IF (LIST) PRINT*,'ID=',IDFIX,' => BARYONIUM '
  PARTIC(1) = 'PIO  '
  PARTIC(2) = 'PI-  '
  PARTIC(3) = 'PI+  '
  CALL INVMAS (PCM,TYP,NPTOT,19,0.DO,0.DO,'BARYO',PARTIC)
  IF (MISMAT)
& CALL HFILL (30,FLOAT(IDEVNT),DUMMY,1.DO)
C*
ELSEIF (IDFIX.EQ.65) THEN
  IF (LIST) PRINT*,'ID=',IDFIX,' => OMEGA  '
  PARTIC(1) = 'PIO  '
  PARTIC(2) = 'GAMMA'
  PARTIC(3) = '      '
  CALL INVMAS (PCM,TYP,NPTOT,13,782.8d0,100.DO,'OMEGO',PARTIC)
C*
  IF (LIST) PRINT*,'ID=',IDFIX,' => GLUON  '
  PARTIC(1) = 'PIO  '
  PARTIC(2) = 'PIO  '
  PARTIC(3) = '      '
  CALL INVMAS (PCM,TYP,NPTOT,33,0.DO,0.DO,'GLUE ',PARTIC)
C*
ELSEIF (IDFIX.EQ.66) THEN
  IF (LIST) PRINT*,'ID=',IDFIX,' => ETA  '
  PARTIC(1) = 'GAMMA'
  PARTIC(2) = 'GAMMA'
  PARTIC(3) = '      '
  CALL INVMAS (PCM,TYP,NPTOT,14,0.DO,0.DO,'ETAO ',PARTIC)
C*
ELSEIF (IDFIX.EQ.69) THEN
  IF (LIST) PRINT*,'ID=',IDFIX,' => GLUON  '
  PARTIC(1) = 'PIO  '
  PARTIC(2) = 'PIO  '
  PARTIC(3) = '      '
  CALL INVMAS (PCM,TYP,NPTOT,34,0.DO,0.DO,'GLUE ',PARTIC)
C*
ELSEIF (IDFIX.EQ.70) THEN
  IF (LIST) PRINT*,'ID=',IDFIX,' => EO  '
  PARTIC(1) = 'GAMMA'
  PARTIC(2) = 'GAMMA'
  PARTIC(3) = '      '
  CALL INVMAS (PCM,TYP,NPTOT,35,548.8d0,30.DO,'ETAO ',PARTIC)
  PARTIC(1) = 'PIO  '
  PARTIC(2) = 'ETAO '
  PARTIC(3) = '      '

```

```
CALL INVMAS (PCM,TYP,NPTOT,36,983.d0,60.DO,'DELTA',PARTIC)
PARTIC(1) = 'PIO '
PARTIC(2) = 'DELTA'
PARTIC(3) = ' '
CALL INVMAS (PCM,TYP,NPTOT,37,0.DO,0.DO,'EO ',PARTIC)
C*
ELSEIF (IDFIX.EQ.71) THEN
  IF (LIST) PRINT*,'ID=',IDFIX,' => EO'
  PARTIC(1) = 'GAMMA'
  PARTIC(2) = 'GAMMA'
  PARTIC(3) = ' '
  CALL INVMAS (PCM,TYP,NPTOT,38,548.8d0,30.DO,'ETA0 ',PARTIC)
  PARTIC(1) = 'PIO '
  PARTIC(2) = 'ETA0 '
  PARTIC(3) = ' '
  CALL INVMAS (PCM,TYP,NPTOT,39,983.d0,60.DO,'DELTA',PARTIC)
  PARTIC(1) = 'PIO '
  PARTIC(2) = 'DELTA'
  PARTIC(3) = ' '
  CALL INVMAS (PCM,TYP,NPTOT,40,0.DO,0.DO,'EO ',PARTIC)
C*
ELSEIF (IDFIX.EQ.72) THEN
  IF (LIST) PRINT*,'ID=',IDFIX,' => EO'
  PARTIC(1) = 'GAMMA'
  PARTIC(2) = 'GAMMA'
  PARTIC(3) = ' '
  CALL INVMAS (PCM,TYP,NPTOT,41,548.8,30.DO,'ETA0 ',PARTIC)
  PARTIC(1) = 'PIO '
  PARTIC(2) = 'ETA0 '
  PARTIC(3) = ' '
  CALL INVMAS (PCM,TYP,NPTOT,42,983.0,60.DO,'DELTA',PARTIC)
  PARTIC(1) = 'PIO '
  PARTIC(2) = 'DELTA'
  PARTIC(3) = ' '
  CALL INVMAS (PCM,TYP,NPTOT,43,0.DO,0.DO,'EO ',PARTIC)
C*
ENDIF
C*
GAMMA STATISTICS
C*
.....
C*
IF (.NOT.TRIGGR) THEN
  TRIGNO(IDEVNT) = TRIGNO(IDEVNT) + 1.
C*
ELSE
C*
  COUNT THE LOST GAMMAS ...
C*
  ILOST = ILOST + INHOLE
C*
  COUNT THE BARYONIUM STATES ...
C*
  IF (BARYON) IBARST = IBARST + 1
C*
  COUNT BARYONIUM STATES WITH MONOENERGETIC GAMMA...
C*
  IF (MOGABA) IBARGA = IBARGA + 1
C*
```

```

                ENDIF
C* =>          (* .NOT.TRIGGR *)
C*

                ENDIF
C* =>          (* OF TEST OF CONSERVATION LAWS *)
C*

                ENDIF
C* =>          (* VETO *)
                ENDIF
C* =>          (* OFF *)
3400 CONTINUE
C*
C*****
C*   END OF GRAND PARTICLE LOOP ... *
C*****
C*
                IF (TIMEOUT) PRINT 3500
3500  FORMAT ('0','>>> RUNNING OUT OF TIME.....')
C*
C*   PRINT A SIMULATION REPORT
C*   .....
C*
                PRINT 3600
3600  FORMAT ('1','*****',/,
&      ' ', '*           RESULTS OF THE SIMULATION          *',/,
&      ' ', '*****')
                ITIME = INT(ZEIT(TOSTRT))
                MIN   =   ITIME/60
                ISEC  = MOD(ITIME,60)
C*
                PRINT 3700,MIN,ISEC
3700  FORMAT ('0',///,
&      ' ', '>>> RUNNING TIME IN TOTAL .....:',
&      ',I4,' MINUTES ',I3,' SECONDS .')
C*
                DO 3800 ID=1,70,1
                NOTRIG = NOTRIG + INT(TRIGNO(ID))
3800  CONTINUE
                NTRG = KNT - NOTRIG
C*
                CALL HNOENT (01,NGADET)
                CALL HNOENT (02,NOREGA)
                CALL HNOENT (11,NSIGAR)
                CALL HNOENT (25,NPITOT)
                CALL HNOENT (26,NPIREC)
C*
                IF (KNT.GT.0) THEN
                    PRCTG1 = (FLOAT(PILOST)/FLOAT(NOTRIG))*100.
                    PRCTG2 = (FLOAT(NOTRIG)/FLOAT(KNT))*100.
                    PRCTG3 = (FLOAT(NTRG) /FLOAT(KNT))*100.
                ENDIF
                IF (NGADET.GT.0) THEN
                    PRCTG4 = (FLOAT(NOREGA)/FLOAT(NGADET))*100.
                    PRCTG6 = (FLOAT(NSIGAR)/FLOAT(NGADET))*100.
                ENDIF
                IF (NOREGA.GT.0) THEN
                    PRCTG7 = (FLOAT(NSIGAR)/FLOAT(NOREGA))*100.
                ENDIF
C*

```



```
C*   HISTOGRAMS' OPERATIONS                                     *
C*****
C*
      CALL HPAKE (24,TRIGNO)
      CALL HPRERR(24,'YES')
      CALL HPAKE (27,ERRID )
      CALL HPRERR(27,'YES')

C*
C*   PRINT AND SAVE THE HISTOGRAMS
C*   .....
C*
      IF (HISPRT)
&CALL HISTDO
C*
      IF (ISTORE.GT.0) THEN
      CALL HWRITE (0,ISTORE)
      PRINT 4800,ISTORE
4800  FORMAT ('0','>>>> HISTOGRAMS STORED ON FORTRAN UNIT #',I4)
      ENDIF
C*
      STOP
      END
```

```
C*      *****
C*      SUBROUTINE HBOOK (NCHAN)
C*      *****
C*
C*      THIS ROUTINE SETS UP HISTOGRAM - BOOKING.
C*
C*      PROGRAMMER : MARCEL KUNZE
C*      LANGUAGE   : FORTRAN 77
C*      VERSION 1.0: 22-AUG-84
C*
C*      IMPLICIT REAL*8 (A-H,O-Z)
C*
C*      COMMON /      / HMEMOR(64000)
C*      COMMON /GEO   / DMODUL,CZDETE,CZBEAM,CZTARG,CZCHAM,CYSUPP,
C*      &              RADIUS,ANGRES,NMODUL
C*      COMMON /TARGET/ BMEANX,FWHMX,BMEANY,FWHMY,BMEANZ,FWHMZ,
C*      &              TGTRAD,TGTHLF,TGTDEN
C*
C*      TGT DIA = 2. * TGTRAD
C*      TGTLEN = 2. * TGTHLF
C*
C*      IF (NCHAN.LE.0) NCHAN = 100
C*
C*      *****
C*      INITIALIZE HBOOK ..... *
C*      *****
C*
C*      CALL HLIMIT (16000)
C*
C*      NAME THE HISTOGRAMS ...
C*
C*      CALL HBOOK1 (01,
C*      & 'GAMMA-SPEKTRUM IM DETEKTORSYSTEM VOR PIO-REKONSTRUKTION$',
C*      &          NCHAN,0.,1000.)
C*      CALL HERROR (01)
C*      CALL HSTAR  (01)
C*
C*      CALL HBOOK1 (02,
C*      & 'GAMMA-SPEKTRUM IM DETEKTORSYSTEM NACH PIO-REKONSTRUKTION$',
C*      &          NCHAN,0.,1000.)
C*      CALL HERROR (02)
C*      CALL HSTAR  (02)
C*
C*      CALL HBOOK1 (03, 'VERTEX-VERTEILUNG IN X-RICHTUNG$',
C*      &          100,-TGT DIA,TGT DIA)
C*      CALL HERROR (03)
C*      CALL HSTAR  (03)
C*
C*      CALL HBOOK1 (04, 'VERTEX-VERTEILUNG IN Y-RICHTUNG$',
C*      &          100,-TGT DIA,TGT DIA)
C*      CALL HERROR (04)
C*      CALL HSTAR  (04)
C*
C*      CALL HBOOK1 (05, 'VERTEX-VERTEILUNG IN Z-RICHTUNG$',
C*      &          100,-TGTLEN,TGTLEN)
C*      CALL HERROR (05)
C*      CALL HSTAR  (05)
C*
C*      CALL HBOOK1 (06, 'PI- STOP-VERTEILUNG X-RICHTUNG$',
```

& 100,-TGTDIA,TGTDIA)
CALL HERROR (06)
CALL HSTAR (06)
C*
CALL HBOOK1 (07, 'PI- STOP-VERTEILUNG Y-RICHTUNG\$',
& 100,-TGTDIA,TGTDIA)
CALL HERROR (07)
CALL HSTAR (07)
C*
CALL HBOOK1 (08, 'PI- STOP-VERTEILUNG Z-RICHTUNG\$',
& 100,-TGTLEN,TGTLEN)
CALL HERROR (08)
CALL HSTAR (08)
C*
CALL HBOOK1 (09,
& 'ENERGIESPEKTRUM DER GESTOPPTEN NEGATIVEN PIONEN\$',
& 100,130.,170.)
CALL HERROR (09)
CALL HSTAR (09)
C*
CALL HBOOK1 (10,
& 'MASSENSPEKTRUM DER REKONSTRUIERTEN NEUTRALEN PIONEN\$',
& NCHAN,0.,250.)
CALL HERROR (10)
CALL HSTAR (10)
C*
CALL HBOOK1 (11,
& 'ENERGIESPEKTRUM DER EINZELGAMMAS NACH REKONSTRUKTION\$',
& NCHAN,0.,1000.,0.)
CALL HERROR (11)
CALL HSTAR (11)
C*
CALL HBOOK1 (12, 'WINKELAUFLÖSUNG DES DETEKTORS\$',
& 100,0.,ANGRES,0.)
CALL HERROR (12)
CALL HSTAR (12)
C*
CALL HBOOK1 (13,
& 'PLOT DER INVARIANTEN MASSE : OMEGA (2 PIO GAMMA)\$',
& NCHAN,0.,2000.,0.)
CALL HERROR (13)
CALL HSTAR (13)
C*
CALL HBOOK1 (14,
& 'PLOT DER INVARIANTEN MASSE : ETA (PIO 2 GAMMA)\$',
& NCHAN,0.,2000.,0.)
CALL HERROR (14)
CALL HSTAR (14)
C*
CALL HBOOK1 (15,
& 'PLOT DER INVARIANTEN MASSE : RHO/FO (PI+ PI- PIO)\$',
& NCHAN,0.,2000.,0.)
CALL HERROR (15)
CALL HSTAR (15)
C*
CALL HBOOK1 (16,
& 'PLOT DER INVARIANTEN MASSE : ETA/OMEGA (2 PI+ 2 PI- PIO)\$',
& NCHAN,0.,2000.,0.)
CALL HERROR (16)

CALL HSTAR (16)

C*
CALL HBOOK1 (17,
& 'RICHTUNGSKOSINUS DER NACHGEWIESENEN GAMMAS IN Z-RICHTUNG\$',
& 100,-1.,1.,0.)

C*
CALL HBOOK1 (18,
& 'RICHTUNGSKOSINUS DER EINZELGAMMAS IN Z-RICHTUNG\$',
& 100,-1.,1.,0.)

C*
CALL HBOOK1 (19,
& 'PLOT DER INVARIANTEN MASSE : BARYONIUM (PI+ PI- PIO GAMMA)\$',
& NCHAN,0.,2000.,0.)
CALL HERROR (19)
CALL HSTAR (19)

C*
CALL HBOOK1 (20,
& 'ERFUELLUNG DER ENERGIE-ERHALTUNG\$',
& 100,-200.,800.,0.)
CALL HERROR (20)
CALL HSTAR (20)

C*
CALL HBOOK1 (21,
& 'ERFUELLUNG DER IMPULS-ERHALTUNG IN X-RICHTUNG\$',
& 100,-500.,500.,0.)
CALL HERROR (21)
CALL HSTAR (21)

C*
CALL HBOOK1 (22,
& 'ERFUELLUNG DER IMPULS-ERHALTUNG IN Y-RICHTUNG\$',
& 100,-500.,500.,0.)
CALL HERROR (22)
CALL HSTAR (22)

C*
CALL HBOOK1 (23,
& 'ERFUELLUNG DER IMPULS-ERHALTUNG IN Z-RICHTUNG\$',
& 100,-500.,500.,0.)
CALL HERROR (23)
CALL HSTAR (23)

C*
CALL HBOOK1 (24, 'ANZAHL DER EINGEGEBENEN ANNIHILATIONS-KANAELES\$',
& 100,0.,100.,0.)

C*
CALL HBOOK1 (25, 'ANZAHL DER EINGEGEBENEN NEUTRALEN PIONEN\$',
& 100,0.,100.,0.)

C*
CALL HBOOK1 (26, 'ANZAHL DER REKONSTRUIERTEN NEUTRALEN PIONEN\$',
& 100,0.,100.,0.)

C*
CALL HBOOK1 (27, 'ANZAHL DER ERKANNTEN ANNIHILATIONS-KANAELES\$',
& 100,0.,100.,0.)

C*
CALL HBOOK1 (28, 'ID MISMATCHES : RHO/FO (PI+ PI- PIO)\$',
& 100,0.,100.,0.)

C*
CALL HBOOK1 (29, 'ID MISMATCHES : ETA/OMEGA(2 PI+ 2 PI- PIO)\$',
& 100,0.,100.,0.)

C*

```
CALL HBOOK1 (30, 'ID MISMATCHES : BARYONIUM( PI+ PI- PIO GAMMA)$',
&          100,0.,100.,0.)
C*
CALL HBOOK1 (31, 'ENERGIE-SPEKTRUM DER NEUTRALEN PIONEN$',
&          NCHAN,0.,1000.,0.)
CALL HERROR(31)
CALL HSTAR (31)
C*
CALL HBOOK1 (32, 'ENERGIE-SPEKTRUM DER GELADENEN PIONEN$',
&          NCHAN,0.,1000.,0.)
CALL HERROR(32)
CALL HSTAR (32)
C*
CALL HBOOK1 (33,
& 'PLOT DER INVARIANTEN MASSE : GLUEBALL (2 PIO GAMMA)$',
&          NCHAN,0.,2000.,0.)
CALL HERROR (33)
CALL HSTAR (33)
C*
CALL HBOOK1 (34,
& 'PLOT DER INVARIANTEN MASSE : GLUEBALL (2 PIO PI- )$',
&          NCHAN,0.,2000.,0.)
CALL HERROR (34)
CALL HSTAR (34)
C*
CALL HBOOK1 (35,
& 'PLOT DER INVARIANTEN MASSE : ETA (3 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
CALL HERROR (35)
CALL HSTAR (35)
C*
CALL HBOOK1 (36,
& 'PLOT DER INVARIANTEN MASSE : DELTA (3 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
CALL HERROR (36)
CALL HSTAR (36)
C*
CALL HBOOK1 (37,
& 'PLOT DER INVARIANTEN MASSE : EO (3 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
CALL HERROR (37)
CALL HSTAR (37)
C*
CALL HBOOK1 (38,
& 'PLOT DER INVARIANTEN MASSE : ETA (4 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
CALL HERROR (38)
CALL HSTAR (38)
C*
CALL HBOOK1 (39,
& 'PLOT DER INVARIANTEN MASSE : DELTA (4 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
CALL HERROR (39)
CALL HSTAR (39)
C*
CALL HBOOK1 (40,
& 'PLOT DER INVARIANTEN MASSE : EO (4 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
CALL HERROR (40)
```

```
C*      CALL HSTAR (40)

      CALL HBOOK1 (41,
& 'PLOT DER INVARIANTEN MASSE : ETA (PI+ PI- 2 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
      CALL HERROR (41)
      CALL HSTAR (41)

C*      CALL HBOOK1 (42,
& 'PLOT DER INVARIANTEN MASSE : DELTA (PI+ PI- 2 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
      CALL HERROR (42)
      CALL HSTAR (42)

C*      CALL HBOOK1 (43,
& 'PLOT DER INVARIANTEN MASSE : EO (PI+ PI- 2 PIO 2 GAMMA)$',
&          NCHAN,0.,2000.,0.)
      CALL HERROR (43)
      CALL HSTAR (43)

C*      RETURN
      END
```

```

C*      *****
C*      SUBROUTINE GEOMET
C*      *****
C*
C*      THIS ROUTINE CALCULATES THE DETECTOR'S GEOMETRY.
C*
C*      PROGRAMMER: MARCEL KUNZE/KFK
C*      LANGUAGE   : FORTRAN 77
C*      VERS.2.0/29-JUN-84
C*
C*      INPUT                OUTPUT
C*      *****            *****
C*      A) ANGLE RESOLUTION  ==>  RADIUS      + NUMB. OF MODULES
C*      B) RADIUS            ==>  ANGLE RES.  + NUMB. OF MODULES
C*      C) NUMB. OF MODULES  ==>  ANGLE RES.  + RADIUS
C*
C*      'DMODUL' HAS A CONSTANT VALUE ...
C*      THE SINGLE MODULES HAVE HEXAGONAL FRONT SHAPE.
C*
C*      .....
C*
C*      IMPLICIT REAL*8 (A-H,O-Z)
C*
C*      LOGICAL ASSUMP
C*
C*      COMMON /GEO   / DMODUL,CZDETE,CZBEAM,CZTARG,CZCHAM,CYSUPP,
C*      &             RADIUS,ANGRES,NMODUL
C*
C*      PARAMETER (PI = 3.1415962653589793)
C*
C*      COMPUTE THE MODULE'S FRONT AREA ...
C*
C*      RMODUL = DMODUL/2.
C*      AREA   = DMODUL**2
C*
C*      COMPUTE DELTA OMEGA OF ...
C*      .....
C*
C*      1.DETEKTOR WITHOUT HOLES
C*      ODETE = 2. * PI      * (1. - CZDETE)
C*      2.BEAM INJECTION HOLE
C*      OBEAM = 2. * PI      * (1. - CZBEAM)
C*      3.CHAMBER SUPPLY & READ OUT
C*      OCHMB = 2. * PI      * (1. - CYSUPP)
C*      4.TARGET
C*      OTARG = 2. * PI      * (1. - CZTARG)
C*      5.DETECTOR
C*      OMEGA = ODETE - OBEAM - OCHMB - OTARG
C*
C*      THE AVAILABLE DETECTOR SURFACE IS ...
C*
C*      SURFAC = OMEGA * RADIUS**2
C*
C*      COMPUTE INNER RADIUS AND ANGLE RESOLUTION (NMODUL GIVEN) ???
C*      .....
C*      IF (RADIUS.EQ.0.AND.ANGRES.EQ.0.0) THEN
C*          IF (OMEGA.GT.0.0) RADIUS = SQRT(NMODUL*AREA/OMEGA)
C*          IF (RADIUS.GT.0.0) ANGRES = 2.* ASIN(RMODUL/RADIUS) * 180./PI
C*      ENDIF

```



```
OCHMB = OCHMB/PI
OTARG = OTARG/PI
OMEGA = OMEGA/PI
PRINT 400,ODETE,OBEAM,OCHMB,OTARG,OMEGA
400  FORMAT ('0','DETECTOR SURFACE...:',F8.4,' PI',/,
&      ' ','- BEAM INJ. HOLE...:',F8.4,' PI',/,
&      ' ','- CHAMBER SUPPLY...:',F8.4,' PI',/,
&      ' ','- TARGET HOLDER...:',F8.4,' PI',/,
&      ' ','-----',/,
&      ' ','REMAINING SURFACE...:',F8.4,' PI')
PERCTG = OMEGA/4. * 100.
PRINT 500,PERCTG
500  FORMAT ('0','THIS IS ',F5.1,' % OF THE FULL SPHERE.')
C*
IF (OMEGA.LE.0) THEN
  PRINT 600,OMEGA
600  FORMAT ('0','>>>> AVAILABLE SOLID ANGLE NEGATIVE:',F8.2)
  PRINT *, ' PROGRAM EXECUTION TERMINATED.'
  STOP
ENDIF
C*
IF (ASSUMP) PRINT 700
700  FORMAT ('0','THE FOLLOWING ASSUMPTION WAS MADE:',/,
&      ' ','ONE MODULE HAS TO BE OMITTED FOR HOLDING TARGET .')
C*
RETURN
END
```

```
C* *****
C* SUBROUTINE GETEVT (ID,NPTOT,PCM,TYP,EOF,LUNIN)
C* *****
C*
C* THIS IS A SUBROUTINE TO READ THE EVENT DATA WRITTEN BY USE OF
C* SUBROUTINE TOTAPE FROM FORTRAN UNIT #LUNIN .
C*
C* LANGUAGE   : FORTRAN 77
C* PROGRAMMER: MARCEL KUNZE/KFK
C* VERS 2.0   : 24-MAY-84
C*
C* TYPE DECLARATIONS
C* .....
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*          REAL*4 TECMS,PCMS(5,50)
C*
C* COMMON /FLAGOP/ LIST
C*
C* LOGICAL LIST,ILLEGL,HARDWA,UNIDEN,EOF
C*
C* CHARACTER TYP(50)*5
C*
C* DIMENSION PCM(5,50)
C*
C* DATA KNT /0/
C* DATA NPERR,NRERR,NHERR /0,0,0/
C*
C* CHECK STATUS
C* .....
C*
C* IF (EOF) THEN
C*   ID = 0
C*
C* ELSE
C*
C*   IF (KNT.EQ.0) PRINT 100,LUNIN
100  FORMAT ('0','>>> FORTRAN UNIT #',I2,' OPENED FOR READING.')
```

```
C*
C*   READ ONE EVENT'S KINEMATICS
C*   .....
C*
C*   READ (LUNIN,200,IOSTAT=IOS,END=1100) NCALL,ID,NPTOT,TECMS
200  FORMAT (A4,A4,A4,A4)
C*   TECM = DBLE(TECMS)
C*
C*   HARDWA = (IOS.GT.0)
C*   IF (HARDWA) THEN
C*     IF (NHERR.LT.100) PRINT 300,NCALL
300  FORMAT('0','>>> HARDWARE ERROR OCCURED AT CALL NO.',I8)
C*     NHERR = NHERR + 1
C*     GOTO 1300
C*   ENDIF
C*
C*   ILLEGL = (NPTOT.LE.0.OR.NPTOT.GT.50)
C*   IF (ILLEGL) THEN
```

```

      IF (NRERR.LT.100) PRINT 400,NCALL,NPTOT
400   FORMAT('0','>>> EVENT NO.',I8,' HAS INVALID LENGTH:',I8)
      NRERR = NRERR + 1
      GOTO 1300
    ENDIF
C*
    NUP = 0
    DO 600 I=1,NPTOT,1
    READ (LUNIN,500,END=1100) (PCMS(J,I),J=1,5),TYP(I)
500   FORMAT (5A4,A5)
      PCM(1,I) = DBLE(PCMS(1,I))
      PCM(2,I) = DBLE(PCMS(2,I))
      PCM(3,I) = DBLE(PCMS(3,I))
      PCM(4,I) = DBLE(PCMS(4,I))
      PCM(5,I) = DBLE(PCMS(5,I))
      UNIDEN = (TYP(I).EQ. ' ')
      IF (UNIDEN) THEN
        NPERR = NPERR + 1
        NUP = NUP + 1
      ENDIF
600   CONTINUE
C*
      UNIDEN = (NUP.GT.0.AND.NPERR.LT.200)
      IF (UNIDEN) THEN
        PRINT 700,NUP,NCALL
700   FORMAT('0','>>> ',I2,' UNIDENTIFIED PARTICLE(S)',
&          ' AT CALL NO.',I8)
      ENDIF
C*
C*
C*   LIST EVERY 10000TH EVENT
C*   .....
C*
      KNT = KNT + 1
C==> LIST =(MOD(KNT,10000).EQ.0)
      IF (LIST.OR.UNIDEN) THEN
        PRINT 800,KNT,ID,NPTOT
800   FORMAT ('0',73('*'),/,
&          ' ', '* EVENT NO.',I8,10X,'CHANNEL NO.',I3,5X,
&          'NUMBER OF PARTICLES:',I3,'*',/,
&          ' ',73('*'))
        DO 1000 I=1,NPTOT,1
        PRINT 900,TYP(I),(PCM(J,I),J=1,5)
900   FORMAT(' ',A5,' | PX =',F15.7,' | PY =',F15.7,
&          ' | PZ =',F15.7,' | E =',F15.7,
&          ' | P =',F15.7)
1000  CONTINUE
      ENDIF
C*
      GOTO 1300
C*
C*
C*   CLOSE INPUT FILE
C*   .....
C*
1100  PRINT 1200,LUNIN,NCALL,NHERR,NRERR,KNT,NPERR
1200  FORMAT ('0','>>> END OF FILE ON FORTRAN UNIT #',I8,/,
&          ' ', 'NUMBER OF EVENTS ON FILE ...:',I8,/,
&          ' ', 'HARDWARE ERRORS .....:',I8,/,

```



```
&      ' ', ' READ ERRORS OCCURED .....: ', I8, /,  
&      ' ', ' EVENTS READ .....: ', I8, /,  
&      ' ', ' UNIDENTIFIED PARTICLES .....: ', I8, /)  
C*  
C*  RESET STATISTICS ...  
C*  
      EOF = .TRUE.  
      NHERR = 0  
      NRERR = 0  
      NPERR = 0  
      KNT = 0  
C*  
      ENDIF  
C*  
1300 RETURN  
      END
```

```
C* *****
C* SUBROUTINE VERTEX (X,Y,Z)
C* *****
C*
C* THIS SUBROUTINE CALCULATES A VERTEX FOR ANTIPROTON-STOP IN
C* A ZYLINDRICAL TARGET ACCORDING TO MONTE CARLO METHODS.
C*
C* PROGRAMMER: MARCEL KUNZE/KFK
C* LANGUAGE : FORTRAN77
C* VERS.1.0 : 26-JUN-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* LOGICAL NIXOUT
C*
C* PARAMETER (ISEED = 31332547)
C*
C* COMMON /TARGET/ BMEANX,FWHMX,BMEANY,FWHMY,BMEANZ,FWHMZ,
C* & TGTRAD,TGTHLF,TGTDEN
C*
C* NOW CALCULATE THE VERTEX-COORDINATES
C* .....
C*
C* NIXOUT = .FALSE.
C* DO 100 UNTIL (NIXOUT)
C*   CALL GAUSS (ISEED,0.5*FWHMZ,BMEANZ,Z)
C*   NIXOUT = (ABS(Z).LT.TGTHLF)
100 CONTINUE
C*
C* NIXOUT = .FALSE.
C* DO 200 UNTIL (NIXOUT)
C*   CALL GAUSS (ISEED,0.5*FWHMZ,BMEANZ,Z)
C*   CALL GAUSS (ISEED,0.5*FWHMZ,BMEANZ,Z)
C*   XY = SQRT(X**2+Y**2)
C*   NIXOUT = (XY.LT.TGTRAD)
200 CONTINUE
C*
C* RETURN
C* END
```

```

C* *****
C* SUBROUTINE TRAFO (PCM,X1,Y1,Z1,R)
C* *****
C*
C* THIS SUBROUTINE WILL CALCULATE THE DIRECTIONAL COSINES
C* TO THE POINT OF INTERSECTION BETWEEN A LINE (CX,CY,CZ,X1,Y1,Z1)
C* AND A SPHERE .
C*
C* LINE      :   |X1|           |CX|   ->
C*              |Y1| + SIGMA * |CY| = X
C*              |Z1|           |CZ|
C*
C* SPHERE    :                2      -> 2
C*                R      = |X|
C*
C* PROGRAMMER : MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN77
C* VERSION 1.0: 27-JUN-84
C* .....
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /FLAGOP/ LIST
C*
C* DIMENSION PCM(5)
C*
C* LOGICAL LIST
C*
C* CALCULATE THE DIRECTIONAL COSINES
C* .....
C*
C* CX      = PCM(1) / PCM(5)
C* CY      = PCM(2) / PCM(5)
C* CZ      = PCM(3) / PCM(5)
C*
C* ATMSU   = X1*CX + Y1*CY + Z1*CZ
C* A       = X1*X1 + Y1*Y1 + Z1*Z1
C* DISCRM  = ATMSU**2 - (A - R**2)
C*
C* IF (DISCRM.LT.0) THEN
C*   IF (LIST) PRINT 100
100   FORMAT ('0','NO POINT OF INTERSECTION.')
C* ELSE
C*   SIGMA  = -ATMSU + SQRT(DISCRM)
C*
C* POINT OF INTERSECTION
C*
C* X       = X1 + SIGMA * CX
C* Y       = Y1 + SIGMA * CY
C* Z       = Z1 + SIGMA * CZ
C*
C* NEW DIRECTIONAL COSINES
C*
C* CXNEW   = X/R
C* CYNEW   = Y/R
C* CZNEW   = Z/R
C*

```

```
PCM(1) = CXNEW * PCM(5)
PCM(2) = CYNEW * PCM(5)
PCM(3) = CZNEW * PCM(5)
C*
ENDIF
C*
RETURN
END
```

```
C* *****
C* SUBROUTINE PISTOP (PCM,TYP,NPTOT,X,Y,Z)
C* *****
C*
C* THIS ROUTINE SIMULATES A PI- STOPPING IN TARGET.
C*
C* PROGRAMMER: MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN 77
C* VERS.1.0   : 27-JUN-84
C*
C* ADAPTED FROM ROUTINE PIMSTP (TK).
C*
C* THE FOLLOWING STEPS ARE PERFORMED :
C* 1. DECIDE WETHER THE REACTON PI- P -> N GAMMA OR PI- P -> N PIO
C*    SHOULD BE SIMULATED (USE PANOFSKI-RATIO IN VARIABLE PANOFR)
C* 2. FOR THE REACTION PI- P -> N GAMMA DO THE FOLLOWING:
C*    NEGLECT THE NEUTRON AND
C* 2.1 GENERATE A DIRECTION FOR THE GAMMA (USING 'GAMCOS')
C* 2.2 SET THE ENERGY OF THIS GAMMA TO 129.41 MEV
C* 3. FOR THE REACTION PI- P => N PIO DO THE FOLLOWING:
C*    NEGLECT THE NEUTRON AND
C* 3.1 GENERATE A DIRECTION FOR THE PIO
C* 3.2 LET THE PIO DECAY INTO TWO GAMMAS IN ITS CENTER-OF-MASS-SYSTEM
C* 3.3 TRANSFORM THE TWO GAMMAS INTO LAB-SYSTEM
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /FLAGOP/ LIST
C* COMMON /PIMDAT/ NPIGAM,NPIPIO
C* COMMON /TARGET/ BMEANX,FWHMX,BMEANY,FWHMY,BMEANZ,FWHMZ,
C* & TGTRAD,TGTHLF,TGTDEN
C*
C* CHARACTER TYP(50)*5
C*
C* LOGICAL LIST
C*
C* DIMENSION PCM(5,50)
C* DIMENSION ETA(3),PPHOT1(3),PPHOT2(3)
C*
C* PARAMETER (PANOFR = 0.608)
C*
C* PARAMETER (HLFPIO = 67.48255 , PIOM1 = 0.7409353E-02 ,
C* & EPIO = 137.85 , PPIO = 28.05)
C*
C* LOOK FOR THE REACTION : PI- STOP PROTON IN TARGET
C* .....
C*
C* NPART = NPTOT
C* DO 200 J=1,NPART,1
C* IF (TYP(J).EQ.'PI-') THEN
C*   ENERGY = PCM(4,J) - 139.5688
C*   RANGE   = PIRANG (ENERGY,TGTDEN)
C*   CXPI    = PCM(1,J) / PCM(5,J)
C*   CYPI    = PCM(2,J) / PCM(5,J)
C*   CZPI    = PCM(3,J) / PCM(5,J)
C*   XSTOP   = X + RANGE * CXPI
C*   YSTOP   = Y + RANGE * CYPI
C*   ZSTOP   = Z + RANGE * CZPI
C*   IF (ABS(ZSTOP).LT.TGTHLF)THEN
```

```

XY = SQRT(XSTOP**2 + YSTOP**2)
IF (XY.LT.TGTRAD) THEN
C*
C*
C*
PI- STOPS IN THE TARGET...
C*
C*
IF (LIST) PRINT 100,J,TYP(J),ENERGY,RANGE,XSTOP,YSTOP,ZSTOP
100 FORMAT('0','PARTICLE#',I3,A7,' WITH KINETIC ENERGY',
&          F10.3,' MEV HAS A RANGE OF',F10.4,' CM',/,
&          ' AND STOPS AT POSITION :',3F10.3,' (CM)')
C*
C*
PURGE PI- ...
C*
TYP(J) = 'PURGE'
C*
1. SELECT THE REACTION
C*
DECPRB = RNDM2(DUMMY)
IF (DECPRB.GE.PANOFR) THEN
C*
2. SIMULATE THE REACTION P PI- => N GAMMA
C*
IF (LIST) PRINT *,'STOPPED PI- DOES THE REACTION: N + GAMMA'
C*
NPIGAM      = NPIGAM + 1
NPTOT       = NPTOT  + 1
TYP(NPTOT) = 'GAMMA'
C*
2.1 GENERATE A DIRECTION FOR THE GAMMA
C*
CALL GAMCOS (XCOS,YCOS,ZCOS)
C*
2.2 SET ENERGY TO 129.41 MEV
C*
PCM(1,NPTOT) = 129.41 * XCOS
PCM(2,NPTOT) = 129.41 * YCOS
PCM(3,NPTOT) = 129.41 * ZCOS
PCM(4,NPTOT) = 129.41
PCM(5,NPTOT) = 129.41
C*
ELSE
C*
3. SIMULATE THE REACTION P PI- => N PIO
C*
.....
C*
IF (LIST) PRINT *,'STOPPED PI- DOES THE REACTION: N + PIO.'
NPIPIO = NPIPIO + 2
C*
3.1 CALCULATE KINEMATIC FOR PIO
C*
.....
C*
CALL GAMCOS (XCOS,YCOS,ZCOS)
PCMX      = XCOS * PPIO
PCMY      = YCOS * PPIO
PCMZ      = ZCOS * PPIO
C*
ETA(1) = PCMX * PIOM1
ETA(2) = PCMY * PIOM1
ETA(3) = PCMZ * PIOM1
GAMMA  = EPIO * PIOM1

```

```
GAMP1 = GAMMA + 1.
C*
C* 3.2 DECAY OF FLYING P10 IN P10-CENTER-OF-MASS-SYSTEM
C* .....
C*
C* CALL GAMCOS (XCOS,YCOS,ZCOS)
C*
C* PPHOT1(1) = XCOS * HLFPIO
C* PPHOT1(2) = YCOS * HLFPIO
C* PPHOT1(3) = ZCOS * HLFPIO
C*
C* PPHOT2(1) = -PPHOT1(1)
C* PPHOT2(2) = -PPHOT1(2)
C* PPHOT2(3) = -PPHOT1(3)
C*
C* A = (ETA(1)*PPHOT1(1))+(ETA(2)*PPHOT1(2))+(ETA(3)*PPHOT1(3))
C*
C* LORENTZ-TRANSFORMATION OF PHOTON MOMENTA ...
C*
C* FIRST PHOTON
C* .....
C*
C* NPTOT      = NPTOT + 1
C* TYP(NPTOT) = 'GAMMA'
C*
C* EPHOTN = GAMMA * HLFPIO + A
C* FACTOR = (HLFPIO + EPHOTN) / GAMP1
C*
C* PCM(1,NPTOT) = PPHOT1(1) + ETA(1) * FACTOR
C* PCM(2,NPTOT) = PPHOT1(2) + ETA(2) * FACTOR
C* PCM(3,NPTOT) = PPHOT1(3) + ETA(3) * FACTOR
C* PCM(4,NPTOT) = EPHOTN
C* PCM(5,NPTOT) = EPHOTN
C*
C* SECOND PHOTON
C* .....
C*
C* NPTOT      = NPTOT + 1
C* TYP(NPTOT) = 'GAMMA'
C*
C* EPHOTN = GAMMA * HLFPIO - A
C* FACTOR = (HLFPIO + EPHOTN) / GAMP1
C*
C* PCM(1,NPTOT) = PPHOT2(1) + ETA(1) * FACTOR
C* PCM(2,NPTOT) = PPHOT2(2) + ETA(2) * FACTOR
C* PCM(3,NPTOT) = PPHOT2(3) + ETA(3) * FACTOR
C* PCM(4,NPTOT) = EPHOTN
C* PCM(5,NPTOT) = EPHOTN
C*
C*
C* ENDIF
C*
C* CALL HFILL (06,XSTOP,DUMMY,1.DO)
C* CALL HFILL (07,YSTOP,DUMMY,1.DO)
C* CALL HFILL (08,ZSTOP,DUMMY,1.DO)
C* CALL HFILL (09,PCM(4,J),DUMMY,1.DO)
C*
C*
C* ENDIF
C* ENDIF
```

```
200  ENDIF  
C*  CONTINUE  
    RETURN  
    END
```



```
C* *****
C* FUNCTION PIRANG (EKIN,TGTDEN)
C* *****
C* THIS FUNCTION CALCULATES THE RANGE OF A PI- WITH ENERGY 'EKIN'
C* IN A TARGET WITH DENSITY 'TGTDEN'.
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /FLAGOP/ LIST
C*
C* DIMENSION A(70),B(70),C(70),D(70)
C*
C* LOGICAL LIST
C*
C* DATA A /0.0      ,0.1364E-01,0.4893E-01,0.1031E+00,0.1747E+00,
& 0.2626E+00,0.4832E+00,0.7584E+00,0.1083E+01,0.1453E+01,
& 0.1863E+01,0.2312E+01,0.2795E+01,0.3310E+01,0.3854E+01,
& 0.4425E+01,0.5961E+01,0.7629E+01,0.9409E+01,0.1128E+02,
& 0.1323E+02,0.1525E+02,0.1732E+02,0.1945E+02,0.2162E+02,
& 0.2382E+02,0.2606E+02,0.2832E+02,0.3061E+02,0.3292E+02,
& 0.3525E+02,0.3995E+02,0.4470E+02,0.4949E+02,0.5430E+02,
& 0.5913E+02,0.6398E+02,0.6883E+02,0.7368E+02,0.7854E+02,
& 0.8340E+02,0.8825E+02,0.9310E+02,0.9795E+02,0.1028E+03,
& 0.1076E+03,0.1124E+03,0.1172E+03,0.1221E+03,0.1269E+03,
& 0.1316E+03,0.1364E+03,0.1412E+03,0.1459E+03,0.1507E+03,
& 0.1554E+03,0.1601E+03,0.1648E+03,0.1695E+03,0.1742E+03,
& 0.1789E+03,0.1836E+03,0.1882E+03,0.1929E+03,0.1975E+03,
& 0.2022E+03,0.2068E+03,0.2114E+03,0.2160E+03,0.2206E+03/
C* DATA B /0.3725E-08,0.1273E-01,0.2249E-01,0.3157E-01,0.3995E-01,
& 0.4781E-01,0.6222E-01,0.7519E-01,0.8695E-01,0.9769E-01,
& 0.1075E+00,0.1165E+00,0.1248E+00,0.1325E+00,0.1396E+00,
& 0.1461E+00,0.1605E+00,0.1727E+00,0.1827E+00,0.1912E+00,
& 0.1987E+00,0.2048E+00,0.2102E+00,0.2148E+00,0.2188E+00,
& 0.2222E+00,0.2252E+00,0.2278E+00,0.2301E+00,0.2319E+00,
& 0.2336E+00,0.2364E+00,0.2385E+00,0.2400E+00,0.2411E+00,
& 0.2419E+00,0.2424E+00,0.2427E+00,0.2429E+00,0.2429E+00,
& 0.2428E+00,0.2426E+00,0.2423E+00,0.2423E+00,0.2414E+00,
& 0.2412E+00,0.2412E+00,0.2400E+00,0.2401E+00,0.2395E+00,
& 0.2390E+00,0.2386E+00,0.2376E+00,0.2376E+00,0.2371E+00,
& 0.2361E+00,0.2361E+00,0.2356E+00,0.2347E+00,0.2342E+00,
& 0.2341E+00,0.2336E+00,0.2327E+00,0.2322E+00,0.2320E+00,
& 0.2316E+00,0.2306E+00,0.2306E+00,0.2300E+00,0.0 /
C* DATA C /0.3867E-02,0.2497E-02,0.2383E-02,0.2160E-02,0.2028E-02,
& 0.1901E-02,0.1703E-02,0.1538E-02,0.1402E-02,0.1281E-02,
& 0.1179E-02,0.1072E-02,0.1004E-02,0.9138E-03,0.8538E-03,
& 0.7902E-03,0.6470E-03,0.5761E-03,0.4208E-03,0.4264E-03,
& 0.3221E-03,0.2954E-03,0.2368E-03,0.2279E-03,0.1725E-03,
& 0.1627E-03,0.1371E-03,0.1295E-03,0.9497E-04,0.9141E-04,
& 0.8042E-04,0.5850E-04,0.4577E-04,0.2879E-04,0.2662E-04,
& 0.1521E-04,0.1025E-04,0.4171E-05,0.3395E-05,-.2444E-05,
& -.7713E-06,-.9062E-05,-.7612E-05,0.1004E-04,-.5459E-04,
& 0.4395E-04,-.4566E-04,-.1079E-04,0.1443E-04,-.4637E-04,
& 0.2169E-04,-.3983E-04,-.1171E-04,0.1242E-04,-.3727E-04,
& -.1254E-04,0.1335E-04,-.3998E-04,-.2616E-05,-.2165E-04,
& 0.1568E-04,-.3994E-04,-.2833E-05,-.2111E-04,0.1533E-04,
& -.3776E-04,-.1248E-04,0.1358E-04,-.4055E-04,0.0 /
C* DATA D /-.2283E-03,-.1892E-04,-.3728E-04,-.2196E-04,-.2113E-04,
& -.1649E-04,-.1374E-04,-.1136E-04,-.1005E-04,-.8542E-05,
```

```
&      -.8948E-05, -.5646E-05, -.7496E-05, -.5006E-05, -.5294E-05,  
&      -.4775E-05, -.2361E-05, -.5179E-05, 0.1875E-06, -.3477E-05,  
&      -.8906E-06, -.1954E-05, -.2968E-06, -.1847E-05, -.3267E-06,  
&      -.8518E-06, -.2525E-06, -.1152E-05, -.1188E-06, -.3662E-06,  
&      -.3653E-06, -.2122E-06, -.2830E-06, -.3605E-07, -.1903E-06,  
&      -.8269E-07, -.1013E-06, -.1294E-07, -.9731E-07, 0.2788E-07,  
&      -.1382E-06, 0.2416E-07, 0.2942E-06, -.1077E-05, 0.1642E-05,  
&      -.1493E-05, 0.5812E-06, 0.4204E-06, -.1013E-05, 0.1134E-05,  
&      -.1025E-05, 0.4687E-06, 0.4022E-06, -.8282E-06, 0.4121E-06,  
&      0.4315E-06, -.8888E-06, 0.6228E-06, -.3172E-06, 0.6222E-06,  
&      -.9271E-06, 0.6185E-06, -.3047E-06, 0.6073E-06, -.8848E-06,  
&      0.4214E-06, 0.4343E-06, -.9022E-06, 0.6759E-06, 0.0      /
```

C*

```
IF (EKIN.GE.0.0.AND.EKIN.LT.10.0) THEN  
  H = MOD(EKIN,2.)  
  I = INT(EKIN/2.) + 1  
ELSEIF (EKIN.GE.10.0.AND.EKIN.LT.50.0) THEN  
  H = MOD(EKIN-10.,4)  
  I = INT((EKIN-10.)/4.) + 6  
ELSEIF (EKIN.GE.50.0.AND.EKIN.LT.200.0) THEN  
  H = MOD(EKIN-50,10.)  
  I = INT((EKIN-50.)/10.) + 16  
ELSEIF (EKIN.GE.200.0.AND.EKIN.LT.1000.0) THEN  
  H = MOD(EKIN-200.,20.)  
  I = INT((EKIN-200.)/20.) + 31  
ELSE  
  IF (LIST) PRINT*, 'PI- : KINETIC ENERGY OUT OF RANGE:', EKIN  
ENDIF
```

C*

```
PIRANG = (D(I)*H**3 + C(I)*H**2 + B(I)*H + A(I)) / TGTDEN
```

C*

```
RETURN  
END
```

```
C* *****
C* SUBROUTINE CHAMBR (PCM,VETO)
C* *****
C* THIS ROUTINE DETECTS THE CHARGED PARTICLES.
C*
C* LANGUAGE...:FORTRAN 77
C* PROGRAMMER:MARCEL KUNZE/KFK
C* VERS 1.0/21-AUG-84(MK)
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* DIMENSION PCM(5)
C*
C* LOGICAL VETO,LIST
C*
C* COMMON /FLAGOP/ LIST
C* COMMON /GEO / DMODUL,CZDETE,CZBEAM,CZTARG,CZCHAM,CYSUPP,
C* & RADIUS,ANGRES,NMODUL
C*
C* WENT THE CHARGED PARTICLE OFF THE CHAMBER ?
C*
C* CX = PCM(1) / PCM(5)
C* CY = PCM(2) / PCM(5)
C* CZ = PCM(3) / PCM(5)
C*
C* WENT THE CHARGED PARTICLE OFF THE CHAMBER ?
C*
C* VETO = (ABS(CZ).GT.CZCHAM)
C*
C* IF (LIST.AND.VETO)
C* &PRINT*,'PARTICLE WENT OFF THE CHAMBER.'
C*
C* RETURN
C* END
```

```
C* *****
C* SUBROUTINE MOMMES
C* &(PCM,PRESOL,PO,PTHRSH,BFIELD,CHMDIA,RESLEN,RES DIA,DELTA,DETECT)
C* *****
C*
C* THIS ROUTINE DOES THE MOMENTUM-MEASUREMENT FOR CHARGED PARTICLES .
C*
C* PROGRAMMER : MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN 77
C* VERSION 3.0: 20-DEC-84
C*
C* IMPLICIT REAL*8 (A-H,M-Z)
C*
C* DIMENSION PCM(5)
C*
C* LOGICAL DETECT
C*
C* PARAMETER (ISEED = 1236554153)
C* PARAMETER (PIMASS = 139.5688 )
C* PARAMETER (PI    = 3.1415962653589793)
C*
C* MOMENTUM - MEASUREMENT
C* .....
C*
C* IF (BFIELD.GT.0.0) THEN
C*
C*     PPURP = SQRT(PCM(1)**2 + PCM(2)**2)
C*     GAMMA = 1./3. * (PPURP/PO + PO/PPURP + 1.) * PRESOL
C*
C*     DO 200 I=1,3,1
C*       HWHM = ABS(0.5 * (GAMMA/300.) * PCM(I))
C*       CALL GAUSS (ISEED,HWHM,PCM(I),PCM(I))
C*     200 CONTINUE
C*
C*     PCM(5) = SQRT(PCM(1)**2 + PCM(2)**2 + PCM(3)**2)
C*
C* ENDIF
C*
C* DETECT = (ABS(PCM(5)).GT.PTHRSH)
C*
C* IF (DETECT) THEN
C*
C*   CALCULATE DIRECTIONAL COSINES
C*
C*   CX = PCM(1) / PCM(5)
C*   CY = PCM(2) / PCM(5)
C*   CZ = PCM(3) / PCM(5)
C*
C*   TRANSFORM TO SPHERICAL COORDINATES
C*   .....
C*
C*   1) COMPUTE ANGLE THETA ...
C*
C*   IF (CZ.GT. 1.D0) CZ = 1.D0
C*   IF (CZ.LT.-1.D0) CZ = -1.D0
C*   THETA = ACOS(CZ)
C*
C*   2) COMPUTE ANGLE PHI ...
C*
```

```

IF (CX.EQ.0.0) THEN
  IF (CY.GE.0.0) THEN
    PHI = PI / 2.
  ELSE
    PHI = 1.5 * PI
  ENDIF
ELSE
  PHI = ATAN(CY/CX)
  IF (CX.LT.0.0) THEN
    PHI = PI + PHI
  ELSE
    IF (CY.LT.0.0) PHI = 2. * PI + PHI
  ENDIF
ENDIF

C*
C*   COMPUTE HWHM
C*   .....
C*
C*   1) THETA (Z - DIRECTION)
C*
C*   HWHMTH = RESLEN/CHMDIA * SIN(THETA)
C*
C*   2) PHI (X,Y - DIRECTION)
C*
C*   HWHMPH = RESDIA/CHMDIA / SIN(THETA)
C*
C*   CALL GAUSS (ISEED,HWHMTH,THETA,THETA)
C*   CALL GAUSS (ISEED,HWHMPH,PHI ,PHI )
C*
C*   TRANSFORM TO DIRECTIONAL COSINES
C*   .....
C*
C*   CXNEW = SIN(THETA) * COS(PHI)
C*   CYNEW = SIN(THETA) * SIN(PHI)
C*   CZNEW = COS(THETA)
C*
C*   DIRCOS = CX*CXNEW + CY*CYNEW + CZ*CZNEW
C*   IF (DIRCOS.GT. 1.DO) DIRCOS = 1.DO
C*   IF (DIRCOS.LT.-1.DO) DIRCOS = -1.DO
C*
C*   DELTA = ACOS(DIRCOS)
C*
C*   CX = CXNEW
C*   CY = CYNEW
C*   CZ = CZNEW
C*
C*   ELSE
C*     PCM(5) = 0.0
C*   ENDIF
C*   (* DETECT *)
C*
C*   ENERGY AND MOMENTUM
C*   .....
C*
C*   PCM(1) = CX * PCM(5)
C*   PCM(2) = CY * PCM(5)
C*   PCM(3) = CZ * PCM(5)
C*   MOQUAD = PCM(1)**2 + PCM(2)**2 + PCM(3)**2
C*   PCM(4) = SQRT(MOQUAD + PIMASS**2)

```

```
C*   PCM(5) = SQRT(MOQUAD)
      RETURN
      END
```

```
C* *****
C* SUBROUTINE MOMENT (PCM,TYP,NPTOT)
C* *****
C* THIS ROUTINE DOES RECONSTRUCTION OF KINEMATICS OF
C* TWO OR FOUR CHARGED PARTICLES.
C*
C* LANGUAGE...:FORTRAN 77
C* PROGRAMMER:MARCEL KUNZE/KFK
C* VERS 1.0 :29-NOV-84(MK)
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /FLAGOP/ LIST
C*
C* DIMENSION PCM(5,50),CX(50),CY(50),CZ(50)
C* DIMENSION A2(2,3),A4(4,5),A0(3,4),E(4),P(4),INDEX(4)
C*
C* INTEGER PIPOS(10)
C*
C* LOGICAL TWOPI,FORPI,IMPROV,NOERR,LIST
C*
C* CHARACTER TYP(50)*5
C*
C* PARAMETER (PIMASS = 139.56)
C* PARAMETER (TECM = 1876.56)
C*
C* NOERR = .TRUE.
C*
C* NCHAR = 0
C*
C* PXSUM = 0.
C* PYSUM = 0.
C* PZSUM = 0.
C* ENESUM = 0.
C*
C* DO 100 I=1,NPTOT,1
C*
C* CALCULATE MOMENTUM AND ENERGY SUMS OF NEUTRALS
C*
C* IF (TYP(I).EQ.'GAMMA'.OR.TYP(I).EQ.'PIO ') THEN
C*     PXSUM = PXSUM + PCM(1,I)
C*     PYSUM = PYSUM + PCM(2,I)
C*     PZSUM = PZSUM + PCM(3,I)
C*     ENESUM = ENESUM + PCM(4,I)
C* ENDIF
C*
C* LOCATE CHARGED PARTICLES POSITIONS (DIRECTIONAL COSINES)
C*
C* IF (TYP(I).EQ.'PI+ '.OR.TYP(I).EQ.'PI- ') THEN
C*     NCHAR = NCHAR + 1
C*     PIPOS(NCHAR) = I
C*     CX(I) = PCM(1,I) / PCM(5,I)
C*     CY(I) = PCM(2,I) / PCM(5,I)
C*     CZ(I) = PCM(3,I) / PCM(5,I)
C* ENDIF
C*
C* 100 CONTINUE
C*
C* TWOPI = (NCHAR.EQ.2)
```

```

FORPI = (NCHAR.EQ.4)
C*
IF (LIST) THEN
  PRINT*, 'GIVEN VALUES:'
  PRINT*, 'MOMENTUM SUM OF NEUTRALS IN X : ', PXSUM
  PRINT*, 'MOMENTUM SUM OF NEUTRALS IN Y : ', PYSUM
  PRINT*, 'MOMENTUM SUM OF NEUTRALS IN Z : ', PZSUM
  PRINT*, 'ENERGY SUM OF NEUTRALS : ', ENESUM
  PRINT*
  PRINT*, 'NUMBER OF CHARGED PARTICLES : ', NCHAR
  PRINT*
ENDIF
C*
PXSUM = -PXSUM
PYSUM = -PYSUM
PZSUM = -PZSUM
ENESUM = TECM - ENESUM
C*
C* CALCULATE MOMENTUM OF TWO CHARGED PARTICLES
C* .....
C*
IF (TWOPI) THEN
C*
C* SOLVE KINEMATICS EQUATION
C*
DO 200 J=1,2,1
  A2(1,J) = CX(PIPOS(J))
  A2(2,J) = CY(PIPOS(J))
200 CONTINUE
C*
  A2(1,3) = PXSUM
  A2(2,3) = PYSUM
C*
CALL MATIN2 (A2,2,2,MDIM,1,INDEX,NERROR,DETERM)
C*
  P(1) = A2(1,3)
  P(2) = A2(2,3)
C*
  NOERR = (NERROR.LE.0)
C*
ENDIF
C*
C* CALCULATE MOMENTUM OF FOUR CHARGED PARTICLES
C* .....
C*
IF (FORPI) THEN
  DO 300 J=1,4,1
    A0(1,J) = CX(PIPOS(J))
    A0(2,J) = CY(PIPOS(J))
    A0(3,J) = CZ(PIPOS(J))
    E(J) = ENESUM/4.
    IF(E(J).LT.PIMASS) E(J) = PIMASS
    P(J) = SQRT(E(J)**2-PIMASS**2)
    A4(4,J) = P(J)/E(J)
300 CONTINUE
C*
  IMPROV = .TRUE.
  ITERAT = 0
C*

```



```
DO 1000 WHILE (IMPROV.AND.NOERR.AND.ITERAT.LT.10)
  ECHARG = ENESUM
  DO 400 J=1,4,1
  ECHARG = ECHARG - E(J) + A4(4,J)*P(J)
400 CONTINUE
C*
  A4(1,5) = PXSUM
  A4(2,5) = PYSUM
  A4(3,5) = PZSUM
  A4(4,5) = ECHARG
C*
C* UNSAVE MATRIX A4
C*
  DO 500 I=1,3,1
  DO 500 J=1,4,1
  A4(I,J) = A0(I,J)
500 CONTINUE
C*
  CALL MATIN2 (A4,4,4,MDIM,1,INDEX,NERROR,DETERM)
C*
  NOERR = (NERROR.LE.0)
C*
  IF (NOERR) THEN
    DO 600 J=1,4,1
    P(J) = A4(J,5)
    E(J) = SQRT(P(J)**2 + PIMASS**2)
    A4(4,J) = P(J)/E(J)
600 CONTINUE
C*
    PX = 0.
    PY = 0.
    PZ = 0.
    EN = 0.
    DO 700 J=1,4,1
    PX = PX + P(J) * A0(1,J)
    PY = PY + P(J) * A0(2,J)
    PZ = PZ + P(J) * A0(3,J)
    EN = EN + SQRT(P(J)**2 + PIMASS**2)
700 CONTINUE
C*
    CHISQ = (PX-PXSUM)**2 + (PY-PYSUM)**2 + (PZ-PZSUM)**2 +
    & (EN-ENESUM)**2
    IMPROV = (CHISQ.GT.1.)
C*
    ITERAT = ITERAT + 1
  ENDIF
C*
1000 CONTINUE
C*
  IF (LIST.AND.ITERAT.EQ.10)
  & PRINT*, 'ITERATION DID NOT CONVERGE.'
C*
C*
  ENDIF
C*
C* CALCULATE KINEMATICS
C* .....
C*
  IF (NOERR.AND.(TWOPI.OR.FORPI)) THEN
```

```

DO 1200 I=1,NCHAR,1
PCM(1,PIPOS(I)) = P(I) * CX(PIPOS(I))
PCM(2,PIPOS(I)) = P(I) * CY(PIPOS(I))
PCM(3,PIPOS(I)) = P(I) * CZ(PIPOS(I))
PCM(4,PIPOS(I)) = SQRT(PIMASS**2 + P(I)**2)
PCM(5,PIPOS(I)) = P(I)
C*
      IF (LIST) PRINT 1100,PIPOS(I),
&      PCM(1,PIPOS(I)),PCM(2,PIPOS(I)),PCM(3,PIPOS(I)),
&      PCM(5,PIPOS(I)),PCM(4,PIPOS(I))
C*
1100      FORMAT(' ', '#', I3, 3X, '| MOMENTUM   : ', 3F10.3, 5X,
&              '|P| = ', F10.3, ' | ENERGY: ', F10.3)
C*
1200      CONTINUE
C*
      ELSE
C*
      IF (LIST) PRINT*, 'ERROR OCCURED.'
C*
      IF (TWOPI) THEN
          TYP(PIPOS(1)) = 'PURGE'
          TYP(PIPOS(2)) = 'PURGE'
      ENDIF
C*
      IF (FORPI) THEN
          TYP(PIPOS(1)) = 'PURGE'
          TYP(PIPOS(2)) = 'PURGE'
          TYP(PIPOS(3)) = 'PURGE'
          TYP(PIPOS(4)) = 'PURGE'
      ENDIF
C*
      ENDIF
C*
      RETURN
      END

```

```
C* *****
C* SUBROUTINE PARDET (PCM,DETECT)
C* *****
C*
C* THIS ROUTINE FOLLOWS THE PARTICLES' WAY AND A DECISION IS MADE
C* WHETHER THEY ARE DETECTED OR NOT DETECTED (LOGICAL:DETECT).
C*
C* LANGUAGE...:FORTRAN 77
C* PROGRAMMER:MARCEL KUNZE/KFK
C* VERS 1.0/21-AUG-84(MK)
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* DIMENSION PCM(5)
C*
C* LOGICAL LIST
C* LOGICAL BEAM,SUPP,TARG,DETECT
C*
C* COMMON /FLAGOP/ LIST
C* COMMON /GEO / DMODUL,CZDETE,CZBEAM,CZTARG,CZCHAM,CYSUPP,
C* & RADIUS,ANGRES,NMODUL
C*
C* CALCULATE THE DIRECTIONAL COSINES
C*
C* CX = PCM(1) / PCM(5)
C* CY = PCM(2) / PCM(5)
C* CZ = PCM(3) / PCM(5)
C*
C* WENT THE PARTICLE INTO THE BEAM INJECTION HOLE????
C*
C* BEAM = (CZ.GT.CZBEAM)
C*
C* WENT THE PARTICLE INTO THE CHAMBER SUPPLY ???
C*
C* SUPP = (CY.GT.CYSUPP)
C*
C* WENT THE PARTICLE INTO THE TARGET HOLDER ???
C*
C* TARG = (CZ.LT.(-CZTARG))
C*
C* DETECT = (.NOT.(BEAM.OR.SUPP.ÓR.TARG))
C*
C* IF (LIST) THEN
C*   IF (BEAM) PRINT *,'PARTICLE WENT INTO BEAM INJ. HOLE.'
C*   IF (SUPP) PRINT *,'PARTICLE WENT INTO CHAMBER SUPPLY.'
C*   IF (TARG) PRINT *,'PARTICLE WENT INTO TARGET HOLDER.'
C* ENDIF
C*
C* RETURN
C* END
```

```
C* *****
C* SUBROUTINE ENEMES (ENERGY,ERESOL,E0,EXPONT,ETHRSH,DETECT)
C* *****
C* THIS ROUTINE DOES THE ENERGY-MEASUREMENT.
C*
C* PROGRAMMER : MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN 77
C* VERSION 1.0: 21-AUG-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* LOGICAL DETECT,NOGOOD
C*
C* PARAMETER (ISEED = 1236554153)
C*
C* ENERGY - MEASUREMENT
C* .....
C*
C* IF (ENERGY.GT.0.0) THEN
C*   HWHM = 0.5 * (ERESOL/100.) * (E0/ENERGY)**EXPONT * ENERGY
C*   NOGOOD = (HWHM.GT.0.0)
C*
C*   DO 100 WHILE (NOGOOD)
C*     CALL GAUSS (ISEED,HWHM,ENERGY,ENERGY)
C*     NOGOOD = (ENERGY.LE.0.0)
100 CONTINUE
C*   ELSE
C*     ENERGY = 0.0
C*   ENDIF
C*
C* DETECT = (ENERGY.GT.ETHRSH)
C*
C* RETURN
C* END
```

```
C* *****
C* SUBROUTINE QUANTI (PCM,DELTA)
C* *****
C* THIS SUBROUTINE CALCULATES THE ANGLE RESOLUTION OF A DETECTOR.
C*
C* PROGRAMMER: MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN 77
C* VERS. 1.0  : 10-JUL-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* COMMON /GEO / DMODUL,CZDETE,CZBEAM,CZTARG,CZCHAM,CYSUPP,
& RADIUS,ANGRES,NMODUL
C*
C* DIMENSION THETAQ(361),PHIQ(361)
C* DIMENSION PCM(5)
C*
C* LOGICAL SETUP
C*
C* PARAMETER (PI = 3.1415962653589793)
C* DATA SETUP /.TRUE./
C*
C* IF (SETUP) THEN
C*
C*     RMODUL = 0.5 * DMODUL
C*     PIHALF = 0.5 * PI
C*     THRHLF = 1.5 * PI
C*     TWOPI  = 2.0 * PI
C*
C*     N      = INT(180./ANGRES) + 1
C*     ANGLE  = ANGRES * PI/180.
C*     DO 100 I=1,N,1
C*         THETAQ(I) = (I-1) * ANGLE
C*         R          = SIN(THETAQ(I)) * RADIUS
C*         IF (R.GT.RMODUL) THEN
C*             PHIQ(I) = 2. * ASIN(RMODUL/R)
C*         ELSE
C*             PHIQ(I) = TWOPI
C*         ENDIF
100    CONTINUE
C*     SETUP = .FALSE.
C* ENDIF
C*
C* CALCULATE THE DIRECTIONAL COSINES
C* .....
C*
C*     CX = PCM(1) / PCM(5)
C*     CY = PCM(2) / PCM(5)
C*     CZ = PCM(3) / PCM(5)
C*
C* TRANSFORM TO SPHERICAL COORDINATES
C* .....
C*
C* 1) COMPUTE ANGLE THETA ...
C*
C* IF (CZ.GT. 1.D0) CZ = 1.D0
C* IF (CZ.LT.-1.D0) CZ = -1.D0
C* THETA = ACOS(CZ)
```

```
C*
C* 2) COMPUTE ANGLE PHI ...
C*
  IF (CX.EQ.0.0) THEN
    IF (CY.GE.0.0) THEN
      PHI = PIHALF
    ELSE
      PHI = THRHLF
    ENDIF
  ELSE
    PHI = ATAN(CY/CX)
    IF (CX.LT.0.0) THEN
      PHI = PI + PHI
    ELSE
      IF (CY.LT.0.0) PHI = TWOPI + PHI
    ENDIF
  ENDIF

C*
C* COMPUTE DISCRETE ANGLE
C* .....
C*
C* 1) THETA
C*
  INDEX = INT(THETA/ANGLE + 0.5) + 1
  THETA = THETAQ(INDEX)

C*
C* 2) PHI
C*
  MODPOS = INT(PHI/PHIQ(INDEX) + 0.5)

C*
  PHI = MODPOS * PHIQ(INDEX)

C*
C* TRANSFORM TO DIRECTIONAL COSINES
C* .....
C*
  CXNEW = SIN(THETA) * COS(PHI)
  CYNEW = SIN(THETA) * SIN(PHI)
  CZNEW = COS(THETA)

C*
  DIRCOS = CX*CXNEW + CY*CYNEW + CZ*CZNEW
  IF (DIRCOS.GT. 1.DO) DIRCOS = 1.DO
  IF (DIRCOS.LT.-1.DO) DIRCOS = -1.DO
  DELTA = ACOS(DIRCOS)

C*
  PCM(1) = CXNEW * PCM(5)
  PCM(2) = CYNEW * PCM(5)
  PCM(3) = CZNEW * PCM(5)

C*
  RETURN
  END
```

```
C* *****
C* SUBROUTINE GAMCOS (XCOS,YCOS,ZCOS)
C* *****
C* THIS ROUTINE GENERATES DIRECTIONAL COSINES.
C*
C* PROGRAMMER: MARCEL KUNZE
C* LANGUAGE   : FORTRAN 77
C* VERS 1.0   : 27-JUN-84
C*
C* ADAPTED FROM ROUTINE 'GAMCOS' (TK).
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* LOGICAL NOGOOD
C*
C* GENERATE A SET OF TWO RANDOM-NUMBERS
C*
C* NOGOOD = .TRUE.
C* DO 100 WHILE (NOGOOD)
C*   R1 = 2. * RNDM(DUMMY) - 1.
C*   R2 = 2. * RNDM(DUMMY) - 1.
C*
C* CHECK THEM. IF NOT USEFUL TRY THE NEXT SET
C*
C*   S = (R1*R1) + (R2*R2)
C*   NOGOOD = (S.GE.1)
100 CONTINUE
C*
C* CALCULATE ...
C*
C*   SQ = SQRT(1. - S)
C*   R12 = 2. * R1
C*   R22 = 2. * R2
C*
C* ...THE DIRECTIONAL COSINES
C*
C*   XCOS = R12 * SQ
C*   YCOS = R22 * SQ
C*   ZCOS = 1. - (2.*S)
C*
C* RETURN
C*
C* END
```

```

C* *****
C* SUBROUTINE INVMAS (PCM,TYP,NPTOT,IHIST,RMASS,WINDOW,RESNAM,PARTIC)
C* *****
C*
C* THIS ROUTINE CALCULATES THE INVARIANT MASS OF THE PARTICLES
C* SPECIFIED BY PARTIC (E.G. DATA PARTIC /'PI+ ','PI- ',.../ ).
C*
C* PROGRAMMER: MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN 77
C* VERS.1.0/24-AUG-84.
C*
C* IMPLICIT REAL*8 (A-H,M,O-Z)
C*
C* COMMON /FLAGOP/ LIST
C*
C* DIMENSION PCM(5,50)
C* DIMENSION PSUM1(3),PSUM2(3),PSUM3(3)
C*
C* LOGICAL LIST,HIST
C* LOGICAL ONEBOD,TWOBOD,THRBOD,EQUL12,EQUL23,GOON
C*
C* CHARACTER*5 TYP(50)
C* CHARACTER*5 RESNAM,PARTIC(3),PARTS
C*
C* 1.SORT THE PARTICLES
C* .....
C*
C* CALL PSORT (PCM,TYP,NPTOT)
C*
C* DO 100 J=1,2,1
C* DO 100 I=1,2,1
C*
C* IF (PARTIC(I).LT.PARTIC(I+1)) THEN
C*     PARTS      = PARTIC(I)
C*     PARTIC(I)  = PARTIC(I+1)
C*     PARTIC(I+1) = PARTS
C* ENDIF
C*
C* 100 CONTINUE
C*
C* 2.SET FLAGS
C* .....
C*
C* IF (LIST) PRINT*,'INVARIANT MASS OF:',PARTIC
C*
C* HIST  = (IHIST.GT.0)
C* ONEBOD = (PARTIC(1).NE.' ')
C* TWOBOD = (PARTIC(2).NE.' ')
C* THRBOD = (PARTIC(3).NE.' ')
C* EQUL12 = (PARTIC(1).EQ.PARTIC(2))
C* EQUL23 = (PARTIC(2).EQ.PARTIC(3))
C*
C* IF (RMASS.LE.0.0) THEN
C*     WINMAX = 2000.DO
C*     WINMIN = 0.DO
C* ELSE
C*     WINMAX = RMASS + WINDOW
C*     WINMIN = RMASS - WINDOW
C* ENDIF

```



```

IF (LIST) PRINT*, 'WINDOW SET FROM:', WINMIN, ' TO ', WINMAX
C*
C* 2.CALCULATE THE INVARIANT MASS
C* .....
C*
IF (ONEBOD) THEN
IF (TWOBOD.AND..NOT.THRBOD) THEN
C*
C* TWO PARTICLES
C*
DO 300 I=1,NPTOT-1,1
IF (TYP(I).EQ.PARTIC(1)) THEN
PSUM1(1) = PCM(1,I)
PSUM1(2) = PCM(2,I)
PSUM1(3) = PCM(3,I)
ESUM1 = PCM(4,I)
IF (EQU12) THEN
JMIN = I + 1
ELSE
JMIN = 1
ENDIF
GOON = .TRUE.
DO 200 J=JMIN,NPTOT,1
IF (TYP(J).EQ.PARTIC(2).AND.GOON) THEN
PSUM2(1) = PSUM1(1) + PCM(1,J)
PSUM2(2) = PSUM1(2) + PCM(2,J)
PSUM2(3) = PSUM1(3) + PCM(3,J)
ESUM2 = ESUM1 + PCM(4,J)
CMASS = MASS(ESUM2,PSUM2)
IF (LIST) PRINT*,CMASS
IF (CMASS.GT.WINMIN.AND.CMASS.LT.WINMAX) THEN
IF (HIST) CALL HFILL (IHIST,CMASS,DUMMY,1.DO)
IF (LIST) PRINT*, 'REC. MASS SORTED INTO:', IHIST
IF (WINDOW.GT.0.) THEN
GOON = .FALSE.
TYP(I) = 'PURGE'
TYP(J) = RESNAM
PCM(1,J) = PSUM2(1)
PCM(2,J) = PSUM2(2)
PCM(3,J) = PSUM2(3)
PCM(4,J) = ESUM2
PCM(5,J) = SQRT(PCM(1,J)**2+PCM(2,J)**2+PCM(3,J)**2)
ENDIF
ENDIF
ENDIF
200 CONTINUE
ENDIF
300 CONTINUE
C*
ELSE
C*
C* THREE PARTICLES
C*
DO 600 I=1,NPTOT-2,1
IF (TYP(I).EQ.PARTIC(1)) THEN
PSUM1(1) = PCM(1,I)
PSUM1(2) = PCM(2,I)
PSUM1(3) = PCM(3,I)
ESUM1 = PCM(4,I)

```

```

IF (EQU12) THEN
  JMIN = I + 1
ELSE
  JMIN = 1
ENDIF
GOON = .TRUE.
DO 500 J=JMIN,NPTOT-1,1
IF (TYP(J).EQ.PARTIC(2).AND.GOON) THEN
  PSUM2(1) = PSUM1(1) + PCM(1,J)
  PSUM2(2) = PSUM1(2) + PCM(2,J)
  PSUM2(3) = PSUM1(3) + PCM(3,J)
  ESUM2    = ESUM1    + PCM(4,J)
  IF (EQU23) THEN
    KMIN = J
  ELSE
    KMIN = 1
  ENDIF
  DO 400 K=KMIN,NPTOT,1
  IF (TYP(K).EQ.PARTIC(3).AND.GOON) THEN
    PSUM3(1) = PSUM2(1) + PCM(1,K)
    PSUM3(2) = PSUM2(2) + PCM(2,K)
    PSUM3(3) = PSUM2(3) + PCM(3,K)
    ESUM3    = ESUM2    + PCM(4,K)
    CMASS    = MASS(ESUM3,PSUM3)
    IF (LIST) PRINT*,CMASS
    IF (CMASS.GT.WINMIN.AND.CMASS.LT.WINMAX) THEN
      IF (HIST) CALL HFILL (IHIST,CMASS,DUMMY,1.DO)
      IF (LIST) PRINT*, 'REC. MASS SORTED INTO:', IHIST
      IF (WINDOW.GT.0.) THEN
        GOON = .FALSE.
        TYP(I) = 'PURGE'
        TYP(J) = 'PURGE'
        TYP(K) = RESNAM
        PCM(1,K) = PSUM3(1)
        PCM(2,K) = PSUM3(2)
        PCM(3,K) = PSUM3(3)
        PCM(4,K) = ESUM3
        PCM(5,K) = SQRT(PCM(1,K)**2+PCM(2,K)**2+PCM(3,K)**2)
      ENDIF
    ENDIF
  ENDIF
  ENDIF
  ENDIF
  CONTINUE
400
  ENDIF
  CONTINUE
500
  ENDIF
  CONTINUE
600
  ENDIF
  CONTINUE
C*   (* TWOBOD.OR.THRBOD *)
  ENDIF
C*   (* ONEBOD *)
C*
  RETURN
  END

```

```
C* *****  
C* REAL FUNCTION MASS (ENERGY,MOMENT)  
C* *****  
C* THIS FUNCTION CALCULATES THE INVARIANT MASS  
C* IF ENERGY AND MOMENTUM ARE GIVEN.  
C*  
C* IMPLICIT REAL*8 (A-H,M-Z)  
C*  
C* DIMENSION MOMENT(3)  
C*  
C* MOQUAD = MOMENT(1)**2 + MOMENT(2)**2 + MOMENT(3)**2  
C* MAQUAD = ENERGY**2 - MOQUAD  
C*  
C* IF (MAQUAD.GT.0.0) MASS = SQRT(MAQUAD)  
C*  
C* RETURN  
C* END
```

```
C* *****
C* SUBROUTINE PSORT(PCM,TYP,NPTOT)
C* *****
C*
C* THIS ROUTINE SORTS AND REARRANGES KINEMATIC DATA (ARRAY 'PCM').
C*
C* PROGRAMMER : MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN 77
C* VERSION 1.0: 10-OCT-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* DIMENSION PCM(5,50),PCMS(5)
C*
C* CHARACTER*5 TYP(50),TYP5
C*
C* LOGICAL SORT
C*
C* 1.) PURGE PARTICLES
C* .....
C*
C* I      = 0
C* DO 200 WHILE (I.LT.NPTOT)
C*   I = I + 1
C*   IF (TYP(I).EQ.'PURGE') THEN
C*     NPTOT = NPTOT - 1
C*     DO 100 J=I,NPTOT,1
C*       PCM(1,J) = PCM(1,J+1)
C*       PCM(2,J) = PCM(2,J+1)
C*       PCM(3,J) = PCM(3,J+1)
C*       PCM(4,J) = PCM(4,J+1)
C*       PCM(5,J) = PCM(5,J+1)
C*       TYP(J)  = TYP(J+1)
100    CONTINUE
C*   ENDIF
200    CONTINUE
C*
C* 2.) SORT THE PARTICLES BY USE OF NAME
C* .....
C*
C* SORT = .TRUE.
C* DO 400 WHILE (SORT)
C*
C* LASTCL = 0
C*
C* DO 300 I=1,NPTOT-1,1
C*
C* IF (TYP(I).LT.TYP(I+1)) THEN
C*   LASTCL = I
C*
C*   PCMS(1) = PCM(1,I)
C*   PCMS(2) = PCM(2,I)
C*   PCMS(3) = PCM(3,I)
C*   PCMS(4) = PCM(4,I)
C*   PCMS(5) = PCM(5,I)
C*   TYP5    = TYP (I)
C*
C*   PCM(1,I) = PCM(1,I+1)
C*   PCM(2,I) = PCM(2,I+1)
```

```
PCM(3,I) = PCM(3,I+1)
PCM(4,I) = PCM(4,I+1)
PCM(5,I) = PCM(5,I+1)
TYP(I)   = TYP(I+1)
C*
PCM(1,I+1) = PCMS(1)
PCM(2,I+1) = PCMS(2)
PCM(3,I+1) = PCMS(3)
PCM(4,I+1) = PCMS(4)
PCM(5,I+1) = PCMS(5)
TYP(I+1)   = TYPS
ENDIF
C*
300 CONTINUE
C*
SORT = (LASTCL.GT.0)
C*
400 CONTINUE
C*
RETURN
END
```

```
C* *****
C* SUBROUTINE TESTID (PART, ID, IDEVNT, MISMAT)
C* *****
C*
C* THIS ROUTINE SEARCHES FOR WELL-KNOWN DECAY-CHANNELS
C* VIA PATTERN-RECOGNITION.
C*
C* PROGRAMMER : MARCEL KUNZE
C* LANGUAGE   : FORTRAN 77
C* VERSION 1.0: 10-OCT-84
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* INTEGER PART(4)
C*
C* LOGICAL TEST, MISMAT
C*
C* DIMENSION MASK(76)
C*
C* DATA MASK / 1120,1130,1140,2220,2230,0020,0030,1110,1110,1110,
&             2200,2200,2200,2200,2200,2210,2210,2210,2210,2210,
&             2210,2210,3300,3310,3310,3310,3310,1110,2210,9999,
&             9999,9999,9999,9999,9999,9999,9999,9999,9999,9999,
&             9999,9999,9999,9999,9999,9999,9999,9999,9999,9999,
&             9999,9999,9999,1111,1111,1111,1111,1111,1000,1000,
&             0100,0010,0010,0010,0021,0012,0011,0021,0120,0032,
&             0042,1122,0030,0040,0050,9999/
C*
C* THE PATTERN TO SEARCH FOR IS ...
C* NPATN = 1000*PART(1)+100*PART(2)+10*PART(3)+PART(4)
C*
C* ID   = 0
C* TEST = .FALSE.
C*
C* DO 100 UNTIL (TEST.OR.ID.GT.75)
C*     ID   = ID + 1
C*     TEST = (NPATN.EQ.MASK(ID))
100 CONTINUE
C*
C* CHECK EVENT - ID
C*
C* MISMAT = (MASK(ID).NE.MASK(IDEVNT))
C*
C* IF (ID.GT.75) THEN
C*     ID   = 0
C*     MISMAT = .FALSE.
C* ENDIF
C*
C* RETURN
C* END
```

```
C* *****
C* SUBROUTINE CONLAW (PCM,NPTOT,DELTA,DELTAX,DELTAY,DELTAZ)
C* *****
C*
C* THIS ROUTINE CALCULATES THE MISSING ENERGY AND THE MISSING
C* MOMENTUM OF AN EVENT.
C*
C* PROGRAMMER : MARCEL KUNZE/KFK
C* LANGUAGE   : FORTRAN 77
C* VERSION 2.0: 05-JAN-85
C*
C* IMPLICIT REAL*8 (A-H,O-Z)
C*
C* DIMENSION PCM(5,50)
C*
C* PARAMETER (PBPMAS = 1876.56)
C*
C* SUMMATION
C* .....
C*
C* DELTAX = 0.DO
C* DELTAY = 0.DO
C* DELTAZ = 0.DO
C* ENESUM = 0.DO
C*
C* DO 100 I=1,NPTOT,1
C*   DELTAX = DELTAX + PCM(1,I)
C*   DELTAY = DELTAY + PCM(2,I)
C*   DELTAZ = DELTAZ + PCM(3,I)
C*   ENESUM = ENESUM + PCM(4,I)
100 CONTINUE
C*
C* DELTA = PBPMAS - ENESUM
C*
C* RETURN
C* END
```

ABBILDUNGSVERZEICHNIS

Abbildung	1. Bild der Annihilation im Quarkmodell	v
Abbildung	2. Annihilationskanäle des Proton - Antiprotonsystems . . .	3
Abbildung	3. Dalitz-Plot eines Systems aus drei Pionen	6
Abbildung	4. Annihilation als Abfolge von Zweikörperzerfällen	7
Abbildung	5. Energiespektrum der geladenen Pionen	13
Abbildung	6. Energiespektrum der neutralen Pionen	14
Abbildung	7. Energiespektrum der Gammas	15
Abbildung	8. Standard Detektor System	16
Abbildung	9. Detektor-System (praxisnahe Geometrie)	19
Abbildung	10. Stopdichtenverteilung der Antiprotonen im Target . . .	20
Abbildung	11. Stopverteilung der geladenen Pionen im Target	21
Abbildung	12. Energiespektrum der stoppenden geladenen Pionen . . .	22
Abbildung	13. Spektrum der invarianten Masse der rekonstruierten neu- tralen Pionen	24
Abbildung	14. Inklusives Gesamtenergiespektrum der rekonstruierten neutralen Pionen	26
Abbildung	15. Gemessenes Energiespektrum der geladenen Pionen . . .	28
Abbildung	16. Gemessenes Energiespektrum der Gammas	28
Abbildung	17. Inklusives Energiespektrum der Gammas nach Rekonstruktion der neutralen Pionen	29
Abbildung	18. Inklusives Energiespektrum der Einzelgammas nach Rekonstruktion der neutralen Pionen	30
Abbildung	19. Invariante Masse des radiativ zerfallenden Omega . . .	32
Abbildung	20. Invariante Masse des radiativ zerfallenden Eta	33
Abbildung	21. Invariante Masse des neutralen Rho	34
Abbildung	22. Invariante Masse des Omega	35
Abbildung	23. Invariante Masse hypothetischer Baryoniumzustände . .	37
Abbildung	24. Invariante Masse des radiativ zerfallenden Glueballs .	38
Abbildung	25. Invariante Masse des E(1418) im Kanal ID=70	40
Abbildung	26. Invariante Masse des E(1418) im Kanal ID=71	40
Abbildung	27. Invariante Masse des E(1418) im Kanal ID=72	41
Abbildung	28. Energiespektrum der rekonstruierten neutralen Pionen (Modularität: 1000)	43
Abbildung	29. Energiespektrum der rekonstruierten neutralen Pionen (Modularität: 600)	43
Abbildung	30. Energiespektrum der rekonstruierten neutralen Pionen (Modularität: 150)	44
Abbildung	31. Energiespektrum der rekonstruierten neutralen Pionen (Modularität: 45)	44
Abbildung	32. Verringerung der Energieauflösung auf 25%	45
Abbildung	33. Verringerung der Flächenbedeckung auf 95%	46
Abbildung	34. Verringerung der Flächenbedeckung auf 85%	47
Abbildung	35. Verringerung der Impulsauflösung auf 15%	48
Abbildung	36. Invariante Masse des neutralen Rho (Detektor ohne Magnetfeld)	50
Abbildung	37. Invariante Masse des Omega (Detektor ohne Magnetfeld)	51
Abbildung	38. Invariante Masse des hypothetischen Baryoniums (Detektor ohne Magnetfeld)	51

LITERATURVERZEICHNIS

- AMS84 C.Amsler
Private Mitteilung
- AST80 Asterix-Kollaboration
A Study of Proton - Antiproton Interactions at rest in a H -
Target at LEAR
CERN PSCC 80-101 (1980)
- BKS78 Basel-Karlsruhe-Stockholm-Straßburg-Thessaloniki-Kollaboration
Search for strongly bound states ...
CERN PSCC 78-28/M 12
- BKS78 Basel-Karlsruhe-Stockholm-Straßburg-Thessaloniki-Kollaboration
Proton-Antiproton Annihilations at rest into $\pi^0 \omega, \pi^0 \eta, \pi^0 \chi, \pi^0$
 π^0 and $\pi^0 \eta$
Nuclear Physics B228 (1983) 424-438
- BUR70 E.H.S Burhop (Hrsg.)
High Energy Physics IV
Pure And Applied Physics 25-IV
- FRA83 M.E.B Franklin et al.
SLAC-Pub-3092, LBL-15969 (1983)
- GRA84 Hans Graßmann
Ein CsJ Testkalorimeter bei Energien zwischen 1 und 20 GeV
Diplomarbeit
Universität Erlangen-Nürnberg (1984)
- HBO78 R.Brun, I.Ivanchenko, P.Palazzi
HBOOK - Histogramming, Fitting and Data Presentation Package
(Vers.3.0)
CERN - Data Handling Division
DD/77/9
- IBM83 CERN - User Support Group
IBM User's Guide
DD/US/4
- JAM68 F.James
Monte-Carlo-Phase-Space
CERN 68-15

- KOC76 H.Koch
Nuclear and Particle Physics at intermediate Energies
Plenum Publishing Company, New York (1976)
- KÖH80 Thomas Köhler
Monte-Carlo-Simulation eines Detektorsystems
Diplomarbeit, Universität Karlsruhe (1980)
- KÖH82 Thomas Köhler
XHBPACK, Programmbeschreibung
KfK, unveröffentlichter Bericht
- PAV78 P.Pavlopoulos
Suche nach tiefgebundenen Zuständen im Proton -
Antiproton-System
Dissertation, Universität Karlsruhe (1978)
erschieden als KfK-Ext. 3/78-4
- PER72 Donald H. Perkins
Introduction to High Energy Physics
Addison-Wesley Publishing Company (1972)
- REV82 Particle Data Group
Review Of Particle Properties
CERN (1982)
- RIC83 B.Richter
Suche nach gebundenen Baryoniumzuständen in seltenen
Annihilationskanälen mittels γ - Spektroskopie
Dissertation, Universität Karlsruhe (1983)
- SPU77 J.Spuller et al.
A Measurement of the Panofsky-Ratio
Phys.Let. 67B (1977) 479-482
- WAG80 A.Wagner
Central Detectors
DESY 80-83