

ISTITUTO NAZIONALE DI FISICA NUCLEARE

Sezione di Genova

INFN/AE-78/4  
8 Marzo 1978

M. R. Monge and S. Squarcia: PRISM PLOT ANALYSIS -  
MANUALE D'USO PER LA VERSIONE IMPLEMENTATA  
SUL CII 10070.

INFN/AE-78/4  
8 Marzo 1978

M. R. Monge and S. Squarcia: PRISM PLOT ANALYSIS - MANUALE D'USO PER LA VERSIONE IMPLEMENTATA SUL CII 10070.

## 1. - INTRODUZIONE. -

Nelle collisioni ad alte energie fra adroni uno stesso stato finale può essere raggiunto attraverso diversi stati intermedi che vengono convenzionalmente chiamati "canali" della reazione considerata. Nella quasi totalità dei casi questi stati intermedi sono costituiti da risonanze la cui vita media è così breve (dell'ordine di  $10^{-23}$  sec.) da non permettere di essere visualizzate in camera a bolle.

Sono perciò stati sviluppati moltissimi metodi per separare i vari canali che possono condurre ad un certo stato finale. Uno di questi è quello ideato al M. I. T. da Irwin Pless ed altri che prende il nome di "Prism Plot Analysis" (P. P. A.)<sup>(1)</sup>.

Il manuale qui di seguito riportato si riferisce ai programmi che realizzano questo tipo di analisi nella versione implementata sul CII 10070 del Centro di Calcolo dell'Università degli Studi di Genova nel corso del 1977.

Questo lavoro non vuole soltanto essere un manuale operativo per l'utente, ma anche, nei limiti imposti dallo spazio e dalla complessità dei programmi, una guida per chi volesse apportare le modifiche o i cambiamenti necessari per poterli utilizzare su un differente tipo di elaboratore.

I programmi implementari sul CII 10070 sono essenzialmente la versione originale scritta al M. I. T. opportunamente modificata. A parte l'ovvia traduzione delle routines scritte in assembler, le modifiche dei programmi scritti in FORTRAN riguardano principalmente la gestione I/O da disco, l'introduzione della gestione dinamica della memoria e l'aggiunta di alcune routines di utilità (READST, WRDST, CNXT) dipendenti dal tipo di DST (data summary tape) su cui viene effettuata l'analisi presso il Gruppo 2° della Sezione di Genova. Il DST a cui si fa riferimento in questo manuale è quello usato per i dati dell'esperienza  $\pi^+p$  a 11.2 GeV/c (BFGMOP - Collaborazione Bologna-Firenze-Genova-Milano-Oxford-Pavia): vedere appendice A per il suo formato.

Inoltre è stato introdotto l'uso di alcune routines di libreria del CERN (UZERO, UCOPY, CROSS, DOT). Nel capitolo seguente è riportata una breve descrizione della tecnica del P. P. A. Per ulteriori dettagli, che esulano dalla finalità di questo scritto, si rimanda alla numerosa bibliografia in merito (1, 2, 3, 4, 5).

## 2. - FINALITA' DEL PROGRAMMA. -

La tecnica del P. P. A. è un tipo di analisi multidimensionale basata sul confronto fra eventi reali ed eventi di tipo Montecarlo fatto nell'intera dimensionalità dello spazio fasi. Questo confronto permette di separare i vari canali attraverso cui si giunge ad un determinato stato finale. Le variabili di spazio fasi, scelte in modo da rendere possibile questa separazione, sono dette "coordinate di prisma" (1, 2, 7).

Il P. P. A. è un processo iterativo realizzato mediante calcolatore costituito dai seguenti passi:

- 1) si calcolano le coordinate di prisma per ogni evento costituente il campione che si vuole analizzare.
- 2) Si ipotizzano tutti i canali che possono essere presenti nell'insieme dei dati considerati e per ognuno di essi si genera un campione di eventi Montecarlo. Per la prima iterazione ogni campione è costituito dallo stesso numero di eventi: la distribuzione di massa è scelta a priori per tutte le iterazioni mentre le distribuzioni angolari di produzione e di decadimento sono isotrope per la prima iterazione e sono il risultato dell'iterazione precedente per quelle successive alla prima.  
Per ogni evento Montecarlo generato si calcolano ovviamente le coordinate di prisma.
- 3) Nello spazio fasi, attorno a ciascun evento Montecarlo, si costruisce una box (a  $3n-5$  dimensioni se le particelle nello stato finale sono  $n$ ) e si stabilisce quanti eventi reali rientrano in questa box ("tagging").  
In questo modo, alla fine del tagging, ogni evento reale avrà associato un numero di "hits" (colpi) ricevuti da ciascun "tipo" di eventi Montecarlo in base al quale è possibile stabilire la probabilità di appartenenza dell'evento a ciascun canale introdotto.
- 4) In base alle probabilità così calcolate si determinano la frequenza propria di ciascun canale e il numero di eventi Montecarlo da generare nell'iterazione successiva.  
Inoltre allo scopo di ottenere un campione di eventi sperimentali appartenenti ad un solo canale si considerano tutti gli eventi reali pesati ognuno con la probabilità del canale in considerazione.  
Si ottengono così le distribuzioni angolari da dare in ingresso per le successive generazioni di eventi Montecarlo e le distribuzioni di massa di ciascun canale.
- 5) Si ripetono i passi dal secondo al quarto fino a che le distribuzioni di massa e le distribuzioni angolari ottenute dopo un'iterazione non siano statisticamente identiche a quelle ottenute nell'iterazione precedente.

Si può dire che il metodo del P. P. A. abbia avuto successo se il campione Montecarlo corrisponde ora statisticamente al campione reale. In questo caso questo può essere pensato come costituito dai canali introdotti, con le percentuali ottenute nell'ultima iterazione.

## 3. - DESCRIZIONE DEL PROGRAMMA. -

### Parte Generale.

L'analisi del Prisma Plot (P. P. A.) consiste in un processo iterativo che è realizzato mediante cinque programmi (FELICIA, SR71A, FERRET, BADGER e SUMX-BADGER) secondo lo schema della Fig. 1.

1. FELICIA è il programma che trasforma in coordinate di prisma le coordinate di ciascun evento letto dal DST reale e riordina gli eventi secondo l'ordine crescente della prima coordinata di prisma.
2. SR71A è il programma che genera gli eventi di Montecarlo per ciascun tipo di canale (risonanze o altro) descritto mediante schede-parametro e schede di specificazione dell'evento.

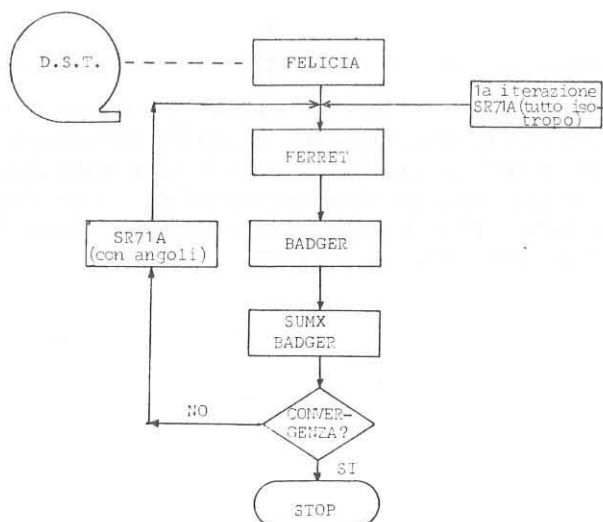


FIG. 1

3. FERRET è il programma che esegue il "tagging", cioè il confronto fra i dati reali ottenuti da FELICIA e gli eventi di tipo Montecarlo (entrambi descritti tramite coordinate di prisma) mediante opportune "boxes" costruite intorno a ciascun evento fittizio; viene così calcolato il numero di volte in cui ciascun evento reale è stato "hittato" (cioè colpito) vale a dire quante volte è rientrato nella "box" attorno a ciascun tipo di evento Montecarlo.
4. BADGER è il programma che converte il numero di "hits" associati ad ogni tipo di evento in probabilità. Più precisamente assegna i pesi a ciascun canale introdotto mediante SR71A e per ogni evento reale dà la probabilità che esso appartenga a ciascuno dei canali introdotti.
5. SUMX-BADGER è il programma che produce gli istogrammi delle grandezze fisiche che interessano, in particolare di tetramomenti, angoli di produzione, angoli di decadimento, ecc. Naturalmente, invece di SUMX si può usare qualsiasi altro programma di istogrammazione su dati contenuti in un DST.

#### 4. - FELICIA. -

##### 4. 1. - Main.

FELICIA è il programma che trasforma in coordinate di prisma le coordinate di ciascun evento reale letto dal DST e le converte da formato floating a formato intero. Il motivo di questa conversione è principalmente la maggior velocità dell'aritmetica intera rispetto all'aritmetica floating; questo permette di diminuire i tempi di calcolo soprattutto durante il tagging.

E' possibile con la presente versione del programma calcolare le coordinate di prisma nel caso di 3 oppure di 4 particelle uscenti.

Nel caso di 3 particelle uscenti (NP = 3), per cui sono necessarie 4 coordinate di prisma (3N-5), ogni evento è caratterizzato da 3 parole così strutturate:

1<sup>a</sup> parola: 1<sup>a</sup> e 2<sup>a</sup> coordinata di prisma dell'evento

2<sup>a</sup> parola: 3<sup>a</sup> e 4<sup>a</sup> coordinata di prisma dell'evento

3<sup>a</sup> parola: 1<sup>a</sup> coordinata di prisma e numero d'ordine del record che identifica l'evento considerato nel DST.

Nel caso di 4 particelle uscenti (NP=4) per cui sono necessarie 7 coordinate di prisma, ogni evento è caratterizzato da 4 parole così strutturate:

1<sup>a</sup> parola: 7<sup>a</sup> e 6<sup>a</sup> coordinata di prisma dell'evento

2<sup>a</sup> parola: 5<sup>a</sup> e 4<sup>a</sup> coordinata di prisma dell'evento

3<sup>a</sup> parola: 3<sup>a</sup> e 2<sup>a</sup> coordinata di prisma dell'evento

4<sup>a</sup> parola: 1<sup>a</sup> coordinata di prisma e numero d'ordine del record che identifica l'evento considerato.

Gli eventi vengono quindi riordinati in ordine crescente rispetto alla prima coordinata di prisma per rendere più veloce il tagging infatti mediante ricerca binaria è possibile trovare rapidamente gli eventi reali la cui prima coordinata cada nella box costruita intorno all'evento Montecarlo e solo su questo numero limitato di eventi vengono fatti gli ulteriori confronti. Le coordinate di prisma per ciascun evento, compattate come detto sopra vengono quindi copiate su disco. In questo programma viene usata la gestione dinamica della memoria (LIMIT).

#### Subroutines richiamate.

1. CNXT in FORTRAN
2. READST in FORTRAN
3. LIMIT in METASYMBOL
4. STRH e EXTRH in METASYMBOL

#### Routines di libreria.

1. BUFFER OUT per la scrittura su disco
2. DOT che esegue il prodotto scalare fra due vettori
3. UZERO per l'azzeramento di un vettore.

#### Dati di input.

Dati letti da schede (3 schede successive):

- NP = numero particelle uscenti  
NPX(5) = numeri associati a ciascuna particella uscente nell'ordine in cui compaiono nel DST: i numeri di codice usati per ciascuna particella devono essere gli stessi usati in SR71A. Se esistono particelle identiche, esse devono essere poste come ultime (es. :  $\pi^+ p \rightarrow \frac{1}{p} \frac{2}{\pi^+} \frac{3}{\pi^+} \frac{4}{\pi^-}$  e NPX=1, 4, 2, 3)
- NEVT = numero di eventi contenuti nel DST che si vuole vengano presi in considerazione  
NTEST = numero di scritture di controllo richieste  
A, B = parametri per la conversione delle coordinate di prisma FX(I) da formato floating a formato intero secondo la formula  
$$MX(I) = (A(I) + FX(I)) * B(I)$$

Formato di lettura dei dati:

I1, 4X, 5I1, I5, I5/10F8. 0/10F8. 0

Dati letti da nastro:

DST di NEVT eventi reali (vedere READST)

#### Output.

Il programma crea un file su disco (DD-FELICIA) che contiene le coordinate di prisma di tutti gli eventi poste in ordine crescente rispetto alla prima coordinata. Il file contiene 240 parole per record cioè 80 o 60 eventi per record a seconda se le particelle uscenti siano 3 o 4.

#### 4. 2. - Subroutine LIMIT.

Questa subroutine è usata in tutti i programmi del P. P. A. eccetto SR71A e consente di effettuare la gestione dinamica della memoria. La sua utilità risiede nel fatto che vengono richieste al sistema solo le pagine di memoria effettivamente utilizzate e calcolabili a priori dall'utente in base al numero di dati (nel nostro caso numero di eventi) che devono stare in memoria, consentendo un notevole risparmio di quest'ultima.

Infatti, mentre il programma senza gestione dinamica della memoria conterrebbe un'area di caricamento dati dimensionata al massimo che si suppone utilizzabile, in questo caso invece con-

tiene solo un puntatore all'area di lavoro esterna al programma stesso: la LIMIT cioè permette di utilizzare come area di lavoro quella frazione della zona di memoria riservata al programma rimasta libera dopo il caricamento del programma stesso detta zona dinamica locale. La zona di memoria riservata al programma viene stabilita attraverso la scheda di controllo !LIMIT del CII 10070 e può essere variata dall'utente. Essa sarà costituita da un numero fisso di pagine di memoria in cui viene caricato il programma e da un numero di pagine variabile a seconda dei dati che si vogliono esaminare.

La subroutine LIMIT richiede tutte le pagine di memoria disponibili nella zona dinamica locale e determina gli indirizzi iniziale e finale dell'area di lavoro così ottenuta. Essa utilizza le procedure di sistema M:GL e M:GP (per la descrizione vedere il manuale "Procédures systèmes sous SIRIS 7/SIRIS 8" del CII 10070).

Sequenza chiamante.

CALL LIMIT (LDWN, LUP, JARRAY (1)).

Parametri di input.

JARRAY (1) = puntatore all'area di lavoro

Parametri di output.

LDWN = indice iniziale dell'area di lavoro  
LUP = indice finale dell'area di lavoro  
JARRAY(1) = indirizzo iniziale dell'area di lavoro.

4. 3. - Subroutine READST. -

Questa subroutine serve per la lettura del nastro in cui è contenuto il DST e perciò deve essere modificata a seconda del formato del DST stesso. Nella presente versione viene letto da nastro un record fisico di 1000 parole e viene posto in un buffer. Ad ogni chiamata della subroutine viene copiato un record logico di 125 parole dal buffer al vettore VETT.

Si legge da nastro il record fisico seguente quando sono stati copiati i 7 records logici del precedente.

Viene sempre scartata la prima parola del record fisico (parola di controllo contenente il numero di records logici per record fisico). Inoltre è possibile avere segnalazione di errore nella lettura del nastro.

Sequenza chiamante.

CALL READST (VETT, IFLG)

Parametri di input.

Nessuno.

Parametri di output.

VETT = vettore contenente il record logico che caratterizza un evento del DST.  
IFLG = flag di controllo: normalmente uguale a zero, è posto uguale a 1 se viene trovato un EOF.

Subroutines richiamate.

Routines di libreria:

1. BUFFER IN per la lettura da nastro
2. UCOPY per ricopiare N parole da un vettore in un altro.

Dati di input.

Dati letti da nastro:  
NEVT eventi reali (NEVT records logici) dal DST.

4. 4. - Subroutine CNXT. -

Questa subroutine serve per convertire, se è necessario, il DST proprio di una particolare esperienza del laboratorio dell'utilizzatore nel formato standard del DST del M. I. T. La routine CNXT qui descritta si riferisce ad un DST la cui principale differenza dal DST formato M. I. T. è che il fascio è diretto lungo l'asse x anzichè lungo l'asse z.

Tutti i calcoli nel P. P. A. sono fatti tenendo conto che il fascio sia diretto lungo l'asse z ed è quindi necessaria questa trasformazione. Per tutte le tracce vengono calcolate quindi le nuove componenti degli impulsi sia nel sistema del centro di massa sia in quello del laboratorio facendo in modo che anche in quest'ultimo sistema l'impulso della particella incidente sia tutto lungo l'asse z (componenti x e y nulle).

Sequenza chiamante.

CALL CNXT (VETT, CC, IFLAG).

Parametri di input.

VETT (125) = vettore contenente il record logico di 125 parole che caratterizza un certo evento letto dal DST.

Parametri output.

CC = vettore contenente le componenti dell'impulso, la massa e l'energia della particella incidente e di tutte le particelle uscenti.  
IFLAG = flag di controllo: solitamente è zero, viene posta uguale a 1 nei casi di errore sul tipo di evento (non è un evento con 3 o 4 particelle uscenti) oppure mancanza del barione.

Subroutines richiamate.

1. FUNCTION AMASS in FORTRAN
2. MPXMPY in FORTRAN.

Routines di libreria:

1. CROSS per il calcolo del prodotto vettore fra due vettori;
2. DOT per il calcolo del prodotto scalare fra due vettori;
3. LOTRAN per le trasformazioni di Lorentz;
4. UCOPY per ricopiare N parole di un vettore in un altro;
5. UZERO per l'azzeramento di un vettore.

Descrizione delle subroutines usate dalla CNXT.

1. FUNCTION AMASS

E' usata per il calcolo della massa, avendo letto energia ed impulso dal DST.

2. SUBROUTINE MPXMPY

Calcola gli impulsi di tutte le particelle uscenti nel nuovo sistema di riferimento usando i coseni direttori calcolati dalla CNXT.

Sequenza chiamante.

CALL MPXMPY (COSDR (1, 1), PC (1, J), PD (1, J), 3, 3, 1).



Parametri di input.

COSDR (3, 3) = vettore contenente i coseni direttori del nuovo sistema.  
PC(4, 5) = matrice contenente gli impulsi nel sistema del centro di massa e le masse delle particelle uscenti nel vecchio sistema di riferimento (beam lungo x).

Parametri di output.

PD(4, 5) = matrice contenente gli impulsi nel sistema del centro di massa e le masse delle particelle uscenti nel nuovo sistema di riferimento (beam lungo z).

4. 5. - Subroutine STRH, EXTRH.

a. Entry point STRH.

Questa subroutine serve per compattare le coordinate di prisma di ogni evento e il numero di record che lo caratterizza in NP parole (NP=numero di particelle uscenti cioè 3 oppure 4): ogni quantità è immagazzinata in una mezza parola come descritto nel par. 4. 1.

Sequenza chiamante.

CALL STRH (KARRAY, KAP, MX).

La STRH scrive la parola MX nella prima (o nella seconda) mezza parola dell'elemento KAP del vettore KARRAY.

Parametri di input.

KAP = indice che dà la prima parola libera dell'area di lavoro KARRAY;  
MX = parola contenente o una coordinata di prisma o il numero di record corrispondenti a quell'evento reale.

Parametri di output.

KARRAY= vettore contenente le coordinate di prisma e il numero di records compattati.

b. Entry point EXTRH.

Questa subroutine esegue esattamente il contrario della precedente: spacchetta per ogni evento le coordinate di prisma poste in forma compatta (2 coordinate per parola).

Sequenza chiamante.

CALL EXTRH (KARRAY, KAP, MX).

La EXTRH scrive nella parola MX la prima (o la seconda) mezza parola dell'elemento KAP del vettore KARRAY.

Parametri di input.

KARRAY= vettore contenente le coordinate di prisma compattate.  
KAP = indice del vettore KARRAY.

Parametri di output.

MX = parola contenente una sola coordinata di prisma o il numero del record.



5. - SR71A.

5. 1. - Main.

E' il programma che genera gli eventi Montecarlo per ogni canale che si ipotizza presente nei dati della reazione considerata e fornisce in output un file contenente le coordinate di prisma (in formato floating) di ciascun evento generato.

Le informazioni necessarie al programma per generare gli eventi di ciascun tipo sono le seguenti:

- 1) le masse di tutte le particelle presenti nella reazione: quelle incidenti e quelle presenti nello stato finale. Inoltre le distribuzioni di massa per tutti gli stati intermedi;
- 2) l'impulso del fascio incidente;
- 3) il numero di eventi che si desiderano generare per il canale considerato;
- 4) una esatta descrizione dell'evento, compresi gli stati intermedi;
- 5) la distribuzione dell'angolo di produzione o la distribuzione del tetramomento trasferito  $t$  per tutti i canali;
- 6) le distribuzioni angolari di decadimento per tutti gli stati intermedi.

Per quanto riguarda il punto 1), nel caso di particelle stabili è sufficiente specificare il valore delle masse sperimentalmente determinato<sup>(10)</sup>. Nel caso di risonanze invece occorre dare il valore centrale di massa e l'ampiezza a metà del massimo  $\Gamma$ ; bisogna inoltre specificare la esatta forma della risonanza: solitamente quella più usata è la Breit-Wigner, ma mediante SR71A oltre a questa è possibile assegnare ad una risonanza la distribuzione gaussiana oppure, nel caso della  $\rho$  (773) o della  $\Delta$  (1236) una Breit-Wigner modificata Jackson. E' inoltre possibile dare in input masse mediante distribuzioni istogrammate lette da scheda bin per bin (esempio: proton diffraction dissociation).

Per quanto riguarda invece il punto 4), la descrizione dell'evento viene fatta nel seguente modo: si assegna la lettera A al vertice iniziale (vertice in cui la particella incidente interagisce con la targhetta), si assegnano lettere da B a T ai vertici in cui si ha un'interazione o un decadimento e lettere dalla U alla Z ai vertici in cui non si ha interazione. La reazione potrà così essere descritta alfabeticamente, come risulta dalla Fig. 2.

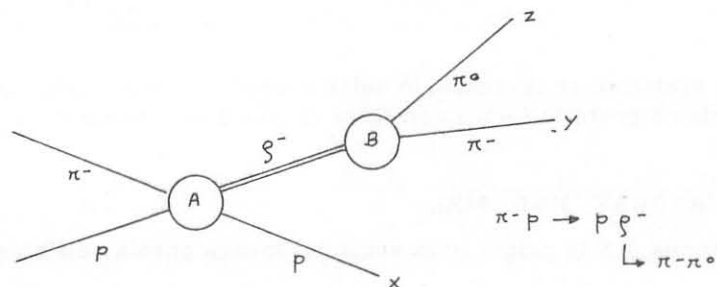


FIG. 2

La reazione mostrata in Fig. 2 viene descritta nel seguente modo: si assegna la lettera A al vertice in cui il  $\pi^-$  incidente interagisce col protone, la lettera B al vertice in cui la  $\rho^-$  decade e le lettere X, Y, Z per identificare le particelle nello stato finale cioè rispettivamente il protone, il  $\pi^-$  e il  $\pi^0$  uscenti. Si avrà così la descrizione:



Ogni combinazione di due lettere rappresenta senza ambiguità la traccia di uno stato intermedio o di uno stato finale della reazione.

Infine, a proposito dei punti 5) e 6), le distribuzioni angolari sono date mediante istogrammi letti da schede bin per bin. Nel caso degli angoli di decadimento è possibile la scelta del sistema di riferimento. Le distribuzioni angolari sono date solo nelle iterazioni successive alla prima nella

quale il processo viene considerato completamente isotropo.

Subroutines richiamate:

- |    |        |    |         |
|----|--------|----|---------|
| 1. | PAREAD | in | FORTRAN |
| 2. | SETUP  | "  | "       |
| 3. | HEDING | "  | "       |
| 4. | EVENT  | "  | "       |
| 5. | FUNC10 | "  | "       |
| 6. | GENFUN | "  | "       |
| 7. | OHIST  | "  | "       |

Viene inoltre richiamata la ITRNDM per l'inizializzazione del generatore di numeri casuali.

Dati di input.

Dati letti da schede:

Schede parametro e schede di specificazione dell'evento per ogni canale che si desidera introdurre alla prima iterazione. Nelle successive iterazioni a queste vanno aggiunte le schede per le distribuzioni angolari.

Descrizione dettagliata schede dati.

Le schede dati di SR71A possono essere divise in tre gruppi:

1. schede-parametro,
2. schede di specificazione dell'evento,
3. schede per le distribuzioni angolari.

1. Le schede-parametro contengono i parametri necessari alla generazione degli eventi di ciascun canale. Ciascun gruppo parametro inizia con una "head card" il cui formato è il seguente:

Colonne	Contenuto	Formato
1 - 20	NBRNCH	10I2 (effettivo (I2, 18X))
21 - 60	HEAD	10A4

dove

NBRNCH = numero negativo (solitamente -1) diverso da -9 che identifica la head card.

HEAD = intestazione BCD che specifica il tipo di canale generato.

Le schede parametro vere e proprie hanno invece il seguente formato:

Colonne	Contenuto	Formato
1 - 3	NN	I3
4	ATABLE=	A1
6 - 17	TEMP	F12.0
21 - 24	TNAME	A4
27 - 62	REMARK	9A4

dove

NN = numero di codice del parametro. E' compreso fra 1 e 99 secondo il seguente schema:

NN	Tipo del parametro
1 - 19	Numero di codice di massa per le particelle nello stato finale
20 - 29	Valore centrale per le masse dei canali a seconda delle distribuzioni: 20 - Breit-Wigner modificata Jackson per la risonanza $\Delta$ (1236). 21 - Breit-Wigner modificata Jackson per la risonanza $\rho$ (773). 22 - 25 - Breit-Wigner per le altre risonanze. 26 - 27 - distribuzioni di massa ottenute mediante istogrammi (es. : diffraction dissociation). 28 - 29 - distribuzione gaussiana.
30 - 39	Ampiezza della risonanza a metà del massimo: 30 si riferisce a 20, 31 a 21, etc.
40 - 93	Non usati.
94	Fattore di distribuzione di Fermi.
95-96-97	Coseni direttori del fascio incidente ( $\hat{x} = 0, \hat{y} = 0, \hat{z} = 1$ ).
98	Fattore di scala per la stampa delle masse (normalmente = 1 se le masse sono date in MeV/c <sup>2</sup> ).
99	Numero di eventi da generare per il canale considerato.

ATABLE =  $\beta$

TEMP = valore del parametro.

TNAME = I se il parametro è intero  
R se il parametro è reale.

REMARK = intestazione alfanumerica che specifica il parametro.

Nel caso che la variabile ATABLE non sia uguale a  $\beta$ , la scheda è trattata come scheda di controllo per la lettura della tavola PARS contenente i parametri (vedere PAREAD).

Ogni gruppo di schede-parametro è terminato da una "end card" che ha lo stesso formato delle schede parametro con

NN = 0

REMARK = END OF PARAMETER CARDS.

#### Note.

- Se si generano campioni Montecarlo di più canali di una stessa reazione in un'unica esecuzione del programma, non è necessario introdurre in tutti i gruppi di schede parametro le masse delle particelle nello stato finale, i coseni direttori del fascio e il fattore di scala per le masse: basta che esse siano presenti nel primo gruppo in quanto ogni gruppo "ricopre" con i propri parametri quelli del gruppo precedente.
- I numeri di codice delle particelle nello stato finale devono essere ordinati secondo i valori decrescenti delle masse.

## 2. Le schede di specificazione servono per la descrizione dell'evento attraverso i suoi stati intermedi.

Ogni gruppo di schede di specificazione inizia con una "head card" e termina con una "end card". Ogni scheda di specificazione ha il seguente formato:

Colonne	Contenuto	Formato
1	ID	A1
2 - 4	IP	I3
5 - 6	CM	A2
7 - 15	FP1	F9.3
16 - 24	FP2	F9.3
25 - 72	VAR	8A6

La "head card" dà informazioni sulla massa e sulla energia della particella incidente. In questa scheda

- ID = H
- IP = rappresenta il numero di codice associato alla massa della particella incidente definito in precedenza nella scheda parametro.
- CM = specifica il tipo di spazio fasi desiderato  
CM = 0 oppure 1 distribuzione invariante  
CM  $\geq$  2 distribuzione non invariante.
- FP1 = impulso della particella incidente se FP2 = 0; se FP2  $\neq$  0 contiene il limite superiore per l'energia.
- FP2 = se diverso da zero contiene il limite inferiore assumendo uno spettro bremsstrahlung per la particella incidente.
- VAR = contiene un commento BCD di intestazione dell'evento.

Le schede successive precisano i dettagli della reazione assegnando tracce e vertici. Le diverse schede si distinguono in base al codice di specificazione ID.

a) N - number card.

E' un modo alternativo per specificare il numero di eventi che devono essere generati. Ha il seguente formato

ID = N

IP = numeri di eventi da generare/100.

b) V - vertex card.

Specifica il numero di codice della particella "targhetta" nel vertice considerato. Ha il seguente formato:

ID = V

IP = numero di codice di massa assegnato alla particella "targhetta" nel vertice considerato che verrà specificato in VAR.

FP1 = se  $\neq$  0 specifica l'impulso di Fermi.

VAR = contiene la lettera che specifica il vertice.

c) T - track card.

Specifica le tracce. Ha il seguente formato:

ID = T

IP = numero di codice di massa assegnato alla particella corrispondente alla traccia specificata in VAR.

VAR = è costituito da due lettere indicanti la traccia a cui è assegnato il codice associato alla massa.

Se quest'ultimo è compreso fra 20 e 29 si tratta di uno stato intermedio.

I vertici in cui il tetramomento viene conservato sono indicati con una lettera da A a T: il primo vertice deve essere A e i seguenti devono procedere in ordine alfabetico (es.: AB, CF, BK etc.). Per particelle che non subiscono un'ulteriore interazione o decadimento si usano le lettere dalla U alla Z. Tracce corrispondenti a particelle identiche possono essere incluse in un'unica scheda T.

d) G - Experimental facility.

Specifica la subroutine FUNC che deve essere chiamata dopo la generazione di ogni evento e per definire il file su cui saranno poste le coordinate di prisma di ciascun evento generato. Ha il seguente formato:

- ID = G  
IP = numero che specifica l'etichetta logica del file che conterrà le coordinate di prisma degli eventi generati per il canale considerato. Questo valore viene caricato nella variabile intera NTAPE.  
VAR = contiene un numero di 2 cifre che specifica la subroutine FUNC che verrà usata. Seguono le tracce delle particelle nello stato finale nello stesso ordine in cui sono date in FELICIA (vedere variabile NPX descritta in FELICIA).

Ciascun gruppo di schede di specificazione è terminato da una scheda con il seguente formato:

- ID = - (segno meno)  
VAR = END OF SPECIFICATIONS.

3. Le schede per le distribuzioni angolari servono, nelle iterazioni successive alla prima, a specificare gli angoli di produzione e di decadimento che vengono forniti mediante istogrammi. Nella prima iterazione, in cui tutte le distribuzioni angolari vengono assunte isotrope, basta porre dopo la scheda che termina le schede di specificazione una scheda bianca che cortocircuita la subroutine RDCARD che legge le schede di distribuzione angolare. Nelle iterazioni successive le schede necessarie sono le seguenti:

1a. scheda.

Contiene con formato 3I3, I11, 5F10.0

- J = numero di decodifica del vertice considerato (tra 1 e 20 per i vertici da A a T definiti in precedenza).  
K = numero corrispondente al tipo di distribuzione  
K = 1 distribuzione isotropa  
K = 2 distribuzione del tetramomento trasferito  $t$  del tipo  $e^{-at}$  ( $a$  è scelto dall'utente) per il primo vertice e serie di potenze in  $\cos \theta$  per gli altri vertici.  
K = 3 distribuzione di  $\cos \theta^*$  (coseno dell'angolo di produzione) e di  $\cos \theta$  e  $\varphi$  (distribuzioni angolari di decadimento) per gli altri vertici.  
L = numero corrispondente al sistema di riferimento  
L = 1 per il primo vertice o per distribuzioni isotrope agli altri vertici.  
L = 2 coseno dell'angolo di Gottfried-Jackson e angolo di Treiman-Yang per due particelle con asse  $z$  lungo il fascio (sistema di Jackson per risonanze mesoniche).  
L = 3 coseno angolo di Gottfried-Jackson e angolo di Treiman-Yang per due particelle con asse  $z$  lungo la targhetta (sistema di Jackson per risonanze barioniche).  
L = 4 coseno angolo di Gottfried-Jackson e angolo di Treiman-Yang per tre particelle con asse  $z$  lungo il fascio.  
L = 5 coseno angolo di Gottfried-Jackson e angolo di Treiman-Yang per tre particelle con asse  $z$  lungo la targhetta.  
L = 6 decadimento di due particelle con il centro di massa come asse  $z$ , vertice associato al fascio.  
L = 7 decadimento di due particelle con il centro di massa come asse  $z$ , vertice associato alla targhetta.  
L = 8 come L = 6 ma per tre particelle.  
L = 9 come L = 7 ma per tre particelle.  
II = se diverso da zero, dà il numero di distribuzioni differenti che devono essere lette per una stessa quantità.

EREAD(5) = valore massimo di una particolare quantità fisica (impulso longitudinale nel centro di massa per il primo vertice e il coseno dell'angolo di produzione per gli altri vertici) per cui occorre usare le corrispondenti distribuzioni angolari.

#### 2a. scheda.

Contiene con formato 2I3:

NP = numero di bins.

I = flag con il seguente significato

I = 1 ampiezza del bin uguale per tutti i bins

I  $\neq$  1 ogni bin ha ampiezza diversa.

#### 3a. scheda.

Se I = 1 contiene con formato 2F10, 3

XBL = estremo inferiore del bin

BW = ampiezza del bin

Se I  $\neq$  1 si hanno un insieme di schede contenenti con formato F20, 5 le ampiezze di tutti i bins (X(I), I = 1, NP).

#### 4a. scheda.

Contiene con formato A2

J = specifica il tipo di formato in cui sono perforate le schede che forniscono gli istogrammi

J = AA formato A

J = EE formato E

J = FF formato F.

#### 5a. scheda.

Contiene con formato 20A4 l'intestazione dell'istogramma che segue. Questa scheda viene letta solo nei casi

J = AA e J = EE.

#### 6a. scheda e seguenti.

Contengono l'istogramma perforato bin per bin con il seguente formato:

- a) J = AA : 20A4
- b) J = EE : 8E10, 3
- c) J = FF : F20, 5

Le schede sopra elencate devono essere presenti per ciascun vertice. Quando per uno stesso vertice si devono dare sia l'istogramma del coseno dell'angolo di Gottfried-Jackson sia l'istogramma dell'angolo di Treiman-Yang la prima scheda non va ripetuta. L'angolo di Treiman-Yang si deve dare in gradi: la trasformazione in radianti viene fatta dal programma.

Le schede per le distribuzioni angolari devono essere seguite da una scheda bianca che segnala alla subroutine RDCARD che sono finite le schede da leggere.

Ovviamente per ciascun tipo di evento che si desidera generare/tutti e tre i gruppi di schede dati. Se si vogliono generare più tipi di eventi in un unico run del programma basta dare i tre gruppi di schede per gli eventi del primo tipo seguite dai tre gruppi di schede per gli <sup>occorrono</sup> eventi del secondo tipo e così via, con l'avvertenza che, dopo l'ultimo gruppo di schede relative al-

L'ultimo canale è necessario un gruppo fittizio di schede-parametro costituito da una "head card" con NBRNCH = - 9 e da una "end card" per segnalare al programma che non ci sono più richieste. Se la distribuzione di massa di uno stato intermedio viene data mediante istogramma, dopo le schede di specificazione e prima delle schede per le distribuzioni angolari sono necessarie le schede che specificano la distribuzione bin per bin. Queste schede hanno lo stesso formato di quelle viste per le distribuzioni angolari, a partire dalla seconda in poi.

Nota.

Un esempio di schede-dati per SR71A è dato in Appendice C.

Output.

Il programma dà in uscita tanti files su disco quanti sono i canali introdotti. Ciascun file contiene le coordinate di prisma in formato floating per ogni evento Montecarlo generato.

5.2. - Breve descrizione delle subroutines essenziali.

a) PAREAD

E' la subroutine che legge le schede-parametro e le immagazzina nella tavola PARS controllando che non si eccedano i limiti. Inoltre stampa i parametri.

Sequenza chiamante

CALL PAREAD (NIT, NOT, NBRNCH, HEAD, PARS, LP, SNAME, REMARK, LMAP).

Parametri di input

NIT = 5 unità logica di input  
NOT = 6 unità logica di output  
LP = lunghezza della banca di parametri PARS  
LMAP= lunghezza del vettore REMARK.

Parametri di output

NBRNCH = contiene un numero negativo con cui inizia un gruppo di schede-parametro. Se è uguale a -9 il programma è finito.  
HEAD = contiene l'intestazione.  
PARS(1000) = tavola contenente i parametri.  
SNAME = contiene R se il parametro è reale, I se è intero.  
REMARK = contiene l'intestazione per ciascun parametro.

b) SETUP

E' la subroutine che legge le schede di specificazione per ciascun tipo di evento e segnala eventuali errori nelle schede suddette. Inoltre riempie le tavole ITABLE, RTABLE, OTABLE e KLIST necessarie per la descrizione e la generazione degli eventi.

Sequenza chiamante.

CALL SETUP (NERR).

Parametri di input

Nessuno.



Parametri di output

NERR = flag di errore. Se diverso da zero si è riscontrato un errore nelle schede di specificazione. A diversi valori di NERR corrispondono diversi errori come riportato nella apposita tabella .

c) HEDING

Questa routine fornisce in output la descrizione dell'evento letta dalla SETUP.

d) EVENT

E' la routine per la generazione degli eventi. Viene richiamata per ogni evento da generare.

Sequenza chiamante

CALL EVENT (NERR).

Parametri di input

NERR = flag col seguente significato  
NERR = -1 inizializzazione  
NERR = 0 generazione evento.

Parametri di output

NERR = è diverso da zero (positivo) se si è riscontrato qualche errore durante la generazione dell'evento.

Subroutines richiamate

- Durante la chiamata di inizializzazione (EVENT(-1)):  
1) EVENTS - calcola le quantità fisiche necessarie alla generazione dell'evento (massa per la risonanza, suo valore centrale, etc. ).

A sua volta la EVENTS richiama le seguenti routines

1) RANRES - costruisce la distribuzione di massa nel caso di risonanze o la legge da schede bin per bin nel caso di masse tipo diffraction dissociation.

2) PSEGEN - E' una chiamata di inizializzazione mediante cui, attraverso la subroutine RDCARD, vengono lette le schede che definiscono modalità e sistemi di riferimento con cui si forniscono le distribuzioni angolari; la RDCARD richiama a sua volta la subroutine READSH che legge gli istogrammi e li normalizza.

- Durante la generazione dell'evento (EVENT (0)):

1) HISMAP } Queste due routines attribuiscono la massa agli stati intermedi: la prima è  
2) INVMAP } usata nel caso di masse date mediante istogrammi letti bin per bin; la seconda è usata nel caso di distribuzioni standard. Questo valore di massa viene scelto probabilisticamente tra i valori della distribuzione assegnata mediante estrazione di numeri a caso fra -1 e 1.

3) PSEGEN - Calcola gli impulsi e l'energia per l'evento generato a partire dai parametri precedentemente calcolati. Chiama la subroutine RDECAY che, in maniera analoga a quanto visto per l'assegnazione di massa degli stati intermedi, sceglie la direzione degli impulsi partendo dalle distribuzioni angolari. Le quantità calcolate dalla PSEGEN vengono poste nella matrice OTABLE (7, 50) che è strutturata nel modo seguente:

OTABLE (I, J) contiene per ogni traccia J dell'ultimo evento generato:

per I = 1 PX  
I = 2 PY  
I = 3 PZ  
I = 4 E (energia)  
I = 5 M (massa invariante)

I = 6      numero di codice del vertice superiore della traccia  
I = 7      numero di codice del vertice inferiore della traccia.

e) FUNC 10

E' la routine che scrive i files di output di SR71A. A partire dagli impulsi e dalle energie delle particelle dell'evento generato calcola le coordinate di prisma per l'evento e le scrive su disco in records di 800 parole. Con la presente versione è possibile calcolare le coordinate di prisma per tre, quattro o cinque particelle nello stato finale.

Sequenza chiamante

CALL FUNC 10 (KR, KCM).

Parametri di input

KR = è un parametro che serve per identificare in OTABLE le particelle nello stato finale.  
KCM = flag: se è positiva o nulla la routine è chiamata per calcolare le coordinate di prisma e scrivere il file di output; se è minore di zero è solo una chiamata di inizializzazione per scrivere un'intestazione.

Parametri di output

Nessuno.

5. 3. - Codice di errore.

Quando, durante la lettura delle schede di specificazione o durante la generazione dell'evento, si verificano degli errori, la variabile NERR assume valori diversi che vengono stampati in output. Ecco il significato:

NERR = 1      i vertici non procedono in ordine alfabetico  
      = 2      ci sono più di 8 particelle in un unico vertice  
      = 3      trovato uno stato intermedio dove ci si aspettava uno stato finale  
      = 4      codice di vertice illegale in una T card  
      = 5      due particelle diverse sono considerate come incidenti in un unico vertice  
      = 7      codice di vertice illegale in una V card  
      = 10     codice di una scheda di specificazione illegale  
      = 12     reazione ben specificata ma output non richiesto  
      = 18     errore nel numero di funzione di una G card  
      = 20     errore nell'elenco delle tracce in una G card  
      = 21     massa efficace errata o energia non conservata (masse o impulsi troppo grandi per l'energia assegnata)  
      = 22     impulsi nulli  
      = 23     non ci sono stati intermedi: canale atteso ma non assegnato.

Se NERR = 100 vuol dire che ci sono ulteriori schede parametro che devono essere ancora lette.

Nota.

La presente versione di SR71A contiene numerose variabili e vettori definiti ma non usati (ad es. KTABLE e MTABLE), interi gruppi di istruzioni che non vengono eseguite ed alcune routines (ad es. l'HISCAP) che non vengono mai richiamate. Ciò deriva dal fatto che SR71A è una versione, modificata per il P. P. A. di NVERTX<sup>(6)</sup>, programma per la generazione di eventi Montecarlo scritto ad Harvard il quale prevede numerosissime subroutines in più rispetto a SR71A.

Non si sono volute abolire tali parti in vista di una loro possibile utilizzazione da parte dello utente: naturalmente può essere cura dell'utente stesso apportare modifiche ottimizzando il programma a seconda delle proprie esigenze.

La modifica fondamentale rispetto a NVERTX è l'introduzione degli angoli di produzione e di decadimento come input proveniente dall'iterazione precedente, oltre naturalmente al calcolo delle coordinate di prisma per gli eventi generati.

## 6. - FERRET.

### 6.1. - Main.

FERRET è il programma che esegue il tagging cioè opera il confronto mediante l'uso di opportune box tra gli eventi reali e gli eventi Montecarlo generati per ciascun canale. Il procedimento è quello descritto a pag. 2: attorno ad ogni evento Montecarlo viene costruita una "box" (scatola) di lati variabili a discrezione dell'utente<sup>(7)</sup>. Ogni evento reale viene quindi confrontato e "hittato" se le sue coordinate di prisma cadono entro la box costruita intorno all'evento Montecarlo.

Il programma fornisce in uscita un file con records di 258 parole (DD-FERRET) in cui è contenuto il numero di volte in cui ciascun evento reale è stato "hittato" (colpito) vale a dire quante volte esso è rientrato nella box per ciascun tipo di evento Montecarlo. Il programma principale è semplicemente costituito dalla chiamata ad una subroutine diversa a seconda del numero di particelle uscenti. Verrà qui descritta in dettaglio la subroutine FERRET 3 implementata per l'analisi della reazione  $\pi^- p \rightarrow p \pi^- \pi^0$  a 11,2 GeV/c e che può essere utilizzata per l'analisi di ogni stato finale 3 corpi.

La FERRET 4 per stato finale 4 corpi ed ogni subroutine per un numero maggiore di particelle nello stato finale sono completamente analoghe.

### 6.2. - Subroutine FERRET 3.

La subroutine FERRET 3 costituisce a tutti gli effetti il MAIN del programma che stiamo considerando. Più in particolare essa legge l'output di FELICIA e per ogni evento Montecarlo di ciascun canale chiama la subroutine che esegue il confronto tra gli eventi reali e l'evento Montecarlo considerato (le cui coordinate di prisma vengono precedentemente convertite con lo stesso procedimento usato in FELICIA). Calcola quindi il numero degli "hits" per ciascun evento reale nella box-size considerata ed il numero di eventi reali hittati per ciascun canale creato con eventi Montecarlo. Naturalmente può accadere che un evento reale non venga hittato da nessun tipo di evento Montecarlo; questo può dipendere dalle misure troppo ridotte della box.

Per ovviare a tale inconveniente nella presente versione del programma è possibile dare più box-sizes di dimensioni crescenti in modo tale che gli eventi reali che non sono stati hittati nella prima box siano confrontati utilizzando una box di dimensioni maggiori e quindi possano essere eventualmente hittati. Questa possibilità permette la scelta della dimensione più adatta per la box. Anche in questo programma viene usata la gestione dinamica della memoria (LIMIT).

#### Subroutines richiamate

1. FERRET        in    METASYMBOL
2. LIMIT        in    METASYMBOL
3. SETPR        in    METASYMBOL

#### Routines di libreria

1. BUFFER IN    per la lettura di un file da disco
2. BUFFER OUT   per la scrittura di un file su disco.

### Dati di input

Dati letti da schede:

NBATCH = numero di canali considerati (al massimo 10)  
NTST = numero di scritture di controllo richieste  
NCNT = test per la subroutine PRINT  
NDST (10) = etichette logiche associate ai files di input contenenti le coordinate degli eventi Montecarlo generati per ciascun canale  
NADD (4) = dimensioni delle 4 componenti delle box considerate  
A, B = parametri di conversione da floating a intero delle coordinate di prisma (devono essere gli stessi usati in FELICIA).

Formato di lettura dei dati

3I3/10I3/4I10/(10F8.0)/(10F8.0).

Nel caso in cui si vogliono usare più box-sizes, le schede contenenti le dimensioni delle box successive alla prima devono essere poste per ultime nel blocco di schede-dati perchè vengono lette solo alla fine dell'esecuzione del programma per la prima box.

L'ordine delle schede dati deve perciò essere il seguente:

1a. scheda NBATCH, NTST, NCNT  
2a. " NDST (10)  
3a. " NADD (4) = dimensioni della prima box  
4a. " A  
5a. " B  
6a. " NADD (4) = dimensioni della seconda box  
seguenti NADD (4) = dimensioni delle box seguenti.

Dati letti da disco:

1. File di output da FELICIA (DD-FELICIA) contenente le coordinate di prisma per gli eventi reali già convertite in formato intero.
2. Files di output di SR71A (uno per ciascun canale considerato) contenenti le coordinate di prisma per gli eventi Montecarlo in formato floating. Questi files possono essere al massimo 10 perchè questo è il numero massimo di canali consentito. In effetti la limitazione a 10 canali è solo dovuta a motivi di tempi di calcolo i quali per FERRET aumentano enormemente all'aumentare del numero di eventi Montecarlo<sup>(7)</sup>. L'unico vincolo di programmazione è il dimensionamento di NDST.

### Output.

Il programma fornisce in uscita un file su disco contenente il numero degli "hits" ottenuti da ciascun evento reale (DD-FERRET). Gli hits di tutti gli eventi reali sono dati per ciascun canale e per ciascuna box presente. Ciascun gruppo è separato da un record di divisione IC(258) in cui la terza parola contiene una flag con il seguente significato:

IC (3) = 0 separa due sequenze di records contenenti il numero di hits per ogni evento Montecarlo di ciascun canale.  
IC(3) = -1 separa sequenze di records di box differenti.  
IC(3) = -2 indica che tutti gli eventi sono stati hittati e quindi che questo è l'ultimo record (EOF).

Ovviamente per tutti i records intermedi IC(3) > 0 (vedere BADGER per il contenuto).

### 6. 3. - Subroutine FERRET.

E' la subroutine fondamentale che esegue il tagging. Il confronto avviene nel modo seguente: si considera la prima coordinata di prisma dell'evento Montecarlo considerato a cui si aggiunge e si toglie la prima dimensione della box. Vengono determinati mediante confronti gli indirizzi minimo e massimo degli eventi reali le cui prime coordinate rientrano nell'intervallo costruito attorno alla prima coordinata dell'evento Montecarlo. Quindi per tutti questi eventi reali si controlla che anche le rimanenti coordinate rientrino nell'intervallo ottenuto aggiungendo e togliendo la corrispondente dimensione della box dalla coordinata corrispondente dell'evento Montecarlo. Se questo avviene per tutte le coordinate l'evento reale in considerazione è stato "hittato" dall'evento Montecarlo e quindi viene posto in MX(4) il numero di hit definito come

$$h = \frac{d_{\max}^2}{d_{MR}^2}$$

dove

$$d_{MR}^2 = \sum_i |r_i - m_i|^2$$

è la distanza fra l'evento Montecarlo (con coordinate di prisma  $m_i$ ) e l'evento reale considerato (di coordinate di prisma  $r_i$ ) e

$$d_{\max}^2 = \left( \sum_i b_i^2 \right) + 1$$

è la distanza massima possibile fra evento Montecarlo ed evento reale ( $b_i$  = dimensioni della box).

#### Sequenza chiamante

CALL FERRET.

#### Parametri di input

Nessuno.

#### Parametri di output

Nessuno.

#### Subroutines richiamate

1. PRINT in METASYMBOL

Questa subroutine ha lo scopo di scrivere il contenuto di tutti i registri generali come stampa di controllo.

### 6. 4. - Subroutine SETPR.

E' la subroutine che esegue il calcolo di alcune variabili facenti parte del COMMON/BUDDA/ e che vengono utilizzate dalla subroutine FERRET.

#### Sequenza chiamante

CALL SETPR (MX(KST), MONTE, NADD, LEV).

#### Parametri di input

XXM = indirizzo iniziale dell'area di lavoro MX

XXMONTE = indirizzo iniziale degli eventi Montecarlo

XNADD(4) = dimensioni della box  
NLEV = numero totale di eventi reali.

Parametri di output

Nessuno.

COMMON.

COMMON/BUDDA/ : MX = indirizzo iniziale dell'area di lavoro.  
NADD(4) = dimensioni della box.  
MAXWGT = distanza massima ottenuta come somma dei quadrati delle quattro dimensioni della box + 1.  
MONTA = indirizzo del primo evento Montecarlo.  
MONTE(4) = coordinate dell'evento Montecarlo.  
LEV = è l'indice che nell'area di lavoro identifica il massimo valore della prima coordinata di prisma degli eventi reali.  
ML = minimo valore della prima coordinata di prisma per gli eventi reali.  
MU = massimo valore della prima coordinata di prisma per gli eventi reali.

7. - BADGER.

7.1. - Main.

BADGER è il programma che converte il numero di "hits" assegnati da FERRET in probabilità. Più precisamente BADGER calcola ed assegna il "peso" a ciascun canale introdotto con SR71A e la "probabilità" che ciascun evento reale appartenga ad ognuno di questi canali. Fornisce quindi in uscita un nuovo DST contenente queste probabilità per ogni evento e perora i pesi per ciascun canale e il numero di eventi Montecarlo da generare per ciascun canale nella successiva iterazione. I pesi per ogni evento reale si ottengono calcolando

$$W_{ij} = \frac{NH_{ij} * W_j}{NH_i}$$

dove  $W_{ij}$  = peso dell'i-esimo evento reale per il j-esimo canale

$NH_{ij}$  = numero di hits dell'i-esimo evento per il j-esimo canale

$W_j$  = peso del j-esimo canale nell'iterazione precedente

$NH_i$  = numero totale di hits per l'i-esimo evento reale

mentre i pesi di ciascun canale vengono calcolati tramite

$$W_j = \frac{\sum_i W_{ij}}{\sum_{ij} W_{ij}}$$

I calcoli di  $W_{ij}$  e  $W_j$  vengono ripetuti iterativamente per un numero di volte fissato dall'utente tramite il parametro NIT. Una possibile scelta di questo parametro può essere quella che permette la stabilizzazione dei pesi di ciascun canale. I pesi  $W_j$  nella prima iterazione vengono assunti uguali per ogni canale.

Anche in BADGER viene usata la gestione dinamica della memoria (LIMIT). Inoltre a seconda delle dimensioni dell'area di lavoro vengono compiuti più o meno passi di calcolo: in ogni passo vengono calcolati i pesi solo per il numero di eventi reali che possono rientrare nell'area di lavoro (tenendo conto che per ogni evento è necessario un numero di parole pari al numero di canali più uno, per la flag indicante la box). E' quindi possibile usare l'area di lavoro variabile data dalla LIMIT mediante due tipi diversi di gestione: o la gestione dinamica della memoria vera e propria con cui si ha un'occupazione di memoria maggiore ma un unico passo di calcolo o la gestione classica di BADGER con cui si ha una minore occupazione di memoria ma più passi di calcolo.

#### Subroutines richiamate

1. LIMIT in METASYMBOL
2. READST in FORTRAN
3. WRDST in FORTRAN

#### Routines di libreria:

1. BUFFER IN per la lettura di file da disco
2. BUFFER OUT per la scrittura di file su disco
3. UZERO per l'azzeramento di un vettore.

#### Dati di input

Dati letti da schede:

- NTEST = numero di scritture di controllo richieste  
IHP = flag di conversione per la scrittura su nastro:  
se IHP  $\neq$  0 il nastro viene scritto in formato HP  
NIT = numero di iterazioni che si desiderano per il calcolo dei pesi  
NTOTAL = numero totale di eventi Montecarlo che si vogliono far creare da SR71A nella iterazione successiva  
TEMP(I) = pesi iniziali per ciascun canale: sono tutti uguali nella 1<sup>a</sup> iterazione mentre sono l'output di BADGER per quelle successive.

Formato di lettura dei dati:

10I10/12, I8/(8F10.0).

Dati letti da disco:

File di output di FERRET (DD-FERRET) contenente il numero di hits ottenuti da ciascun evento reale per ciascun canale preso in considerazione. Il contenuto dei vari records è specificato dalla terza parola del record:

IC(3) > 0 indica un record vero e proprio contenente gli hits. E' una parola impacchettata contenente il numero di particelle uscenti, una flag (K) indice di particelle ambigue e il numero di eventi per record:  
 $IC(3) = \text{numero particelle} * 2^{17} + K * 2^{16} + \text{numero eventi per record.}$   
La flag per le particelle ambigue ha il seguente significato:  
K = 0 non esistono particelle identiche nello stato finale  
K = 1 esistono due particelle identiche nello stato finale.

IC(3) = 0 record di separazione fra le varie sequenze di records di eventi Montecarlo di canali distinti

IC(3) = -1 record di separazione fra le differenti box-sizes

IC(3) = -2 EOF.

Dati letti da nastro:

DST di tutti gli eventi reali: ovviamente è lo stesso di FELICIA (vedere READST) a parte la rota-



zione degli assi eventualmente necessaria. Infatti, come abbiamo visto nelle descrizioni della subroutine CNXT i programmi del P. P. A. sono predisposti con il fascio lungo l'asse z. Nel caso questo non avvenga, per poter calcolare gli angoli corretti, bisogna trasformare il DST in modo che l'impulso del fascio sia lungo questo asse.

#### Output.

Il programma dà in uscita su nastro un DST pesato (vedere WRDST). Inoltre perfora una scheda contenente i pesi iniziali da dare in input a BADGER nell'iterazione successiva e le schede contenenti il numero di eventi Montecarlo da creare nell'iterazione successiva per ciascun canale.

#### Descrizione delle variabili fondamentali.

LDW	=	indirizzo iniziale dell'area di lavoro
LUP	=	indirizzo finale dell'area di lavoro
LDW1	=	LDW + 1
J X I	=	LUP - LDW = dimensioni dell'area di lavoro
NETYPE	=	indice sul numero di canali presenti
NBOX	=	indice sul numero di box
NFILE	=	indice sui files di lettura
NBREC	=	numero di records scritti sullo scratch file per ogni box size
NEVREC	=	numero di eventi per record scritti sullo scratch file
NRPASS	=	numero di records dello scratch file che possono entrare nell'area di lavoro JARRAY tenendo conto che l'area di lavoro è costituita da JXI parole e che sono necessarie NNEXT parole per evento
NNEXT	=	numero di parole necessarie per evento: una parola è riservata per il numero di box, le successive per il numero di hits associato a ciascun canale generato
NEPASS	=	numero di eventi dello scratch file che possono essere contenuti nell'area di lavoro
NEPASS	=	numero di passi necessari per considerare tutti gli eventi reali presenti tenendo conto delle dimensioni dell'area di lavoro
NFLAG	=	flag per le particelle identiche: è zero se non esistono particelle identiche altrimenti è 1.

#### Nota.

Lo scratch file è un file di lavoro in cui viene ricopiato per comodità di calcolo il file DD-FERRET, escluse le parole di controllo.

#### 7. 2. - Subroutine READST.

E' la stessa usata in FELICIA a cui si rimanda per la descrizione.

#### 7. 3. - Subroutine WRDST.

E' la subroutine che serve per scrivere su nastro il DST pesato output di BADGER. Anche questa subroutine, come la READST, dipende dal tipo di DST utilizzato e deve quindi essere modificata a seconda del DST stesso. Qui di seguito è descritta la versione della subroutine che si riferisce al DST di BFGMOP.

In ogni record fisico del nastro sono contenuti 5 records logici più una parola di controllo contenente il numero di records logici per quel record fisico. Ogni record logico è fissato a 190 parole e contiene i dati per un evento reale nel seguente formato:

- dalla parola 1 alla parola 125 contiene le quantità del DST formato Genova versione P.P.A. (cioè con il beam lungo z: viene ricopiato dal DST opportunamente modificato contenente gli stessi

- eventi di FELICIA) (vedere appendice A);
- nella parola 126 contiene la flag indicante il numero di box;
  - dalla parola 127 alla  $127 + (n - 1)$  contiene i pesi dati da BADGER per gli n canali introdotti con SR71A (nello stesso ordine dato in FERRET);
  - dalla parola  $127 + n$  alla parola  $127 + 2n - 1$  contiene gli hits ottenuti dall'evento per ciascuno degli n canali introdotti;
  - nella parola  $127 + 2n$  contiene la flag indicante la box successiva, se esiste, dopo di cui si trovano i pesi e gli hits relativi a quella box nello stesso modo visto per la prima box.

Con la presente versione della subroutine è possibile far scrivere il nastro in formato HP (a seconda del valore della variabile IHP).

#### Sequenza chiamante

CALL WRDST (IFLAG, IC).

#### Parametri di input

- IFLAG = flag di controllo: se IFLAG = 0 <sup>viene</sup> ricopiato il contenuto del vettore di lavoro IC nel buffer di scrittura IDUMP e viene incrementato il numero di records; al quinto record logico viene scritto un record fisico su nastro. Se IFLAG = 1 sono finiti gli eventi reali da considerare per cui si scrive l'ultimo record fisico su nastro e si mette l'EOF;
- IC = vettore di lavoro contenente tutti i dati da porre nel DST.

#### Parametri di output

Nessuno.

#### Dati di input

Nessuno

#### Output

Il programma dà in uscita un DST pesato il cui formato è quello dato nella descrizione della subroutine.

#### Subroutines richiamate

L'unica subroutine richiamata è la SIGHP in METASYMBOL se IHP è diverso da zero. In questo caso infatti si desidera il nastro scritto in formato HP e la SIGHP è la routine di conversione da formato SIGMA7 a formato HP.

#### Routines di libreria

1. BUFFER OUT per la scrittura su nastro
2. UCOPY per ricopiare N parole da un vettore in un altro.

#### Segnalazione di errore

Vengono segnalati gli errori di scrittura su nastro, specificando il record in cui si è verificato l'errore.

#### 8. - SUMX-BADGER. -

Si indica qui col nome di SUMX-BADGER l'analisi fisica (in particolare distribuzioni di massa efficace, angoli di produzione e di decadimento) effettuata sui dati reali, ciascuno pesato, canale per canale, con le probabilità fornite in BADGER. Qualsiasi programma di analisi statistica di

dati sperimentali può essere utilizzato in questa fase al posto del programma SUMX<sup>(8)</sup>,<sup>(9)</sup> da noi impiegato.

Quest'ultimo passo del processo iterativo è l'unico indipendente dal metodo del P. P. A. ed è il solo che può fornire all'utilizzatore informazioni riguardo alla correttezza o meno delle quantità ottenute per l'iterazione in corso. Solo in base all'analisi di questi risultati si può stabilire quando, entro gli errori desiderati, il P. P. A. abbia avuto successo.

#### BIBLIOGRAFIA. -

- (1) J. E. Brau, F. T. Dao, M. F. Hodous, I. A. Pless and R. A. Singer; Prism Plot: a new analysis of multibody final states. *Phys. Rev. Letters* 27, 1481 (1971).
- (2) H. E. Haber; Prism Plot Analysis of the five body final states at 3.92 GeV/c - Master thesis, MIT (1973) (unpublished).
- (3) A. Ferrando; Prism Plot Analysis of multibody final states. Proc. of the 1st Intern. Winter Meeting on Fundamental Physics - Formigal (Spain, 1973).
- (4) Deutschmann et al., *Nucl. Phys.* B86, 221 (1975).
- (5) Deutschmann et al., *Nucl. Phys.* B99, 397 (1975).
- (6) NVERTX manual; Private communication from Heidelberg, NVERTX fu scritto nel 1965 da C. A. Bordner, A. E. Brenner and E. E. Ronat del dipartimento di fisica di Harvard e successivamente modificato da D. T. Gillespie, nel 1969.
- (7) Maria Roberta Monge, Tesi di laurea, Istituto Scienze Fisiche Università di Genova - A. A. 1976/77.
- (8) SUMX User's Manual - CERN Computer Program Library Long Write Up Y 200.
- (9) "SUMS7 - SUMX for SIGMA7" (Genova - INFN Gruppo 2, pubblicazione interna).
- (10) Review of Particle Properties - Particle Data Group, April 1976.

\*\*\* APPENDICE A \*\*\*

FORMATO DST ESPERIENZA PAIMENO=PROTONI A 11.2 GEV/C

STRUTTURA RECORD FISICO

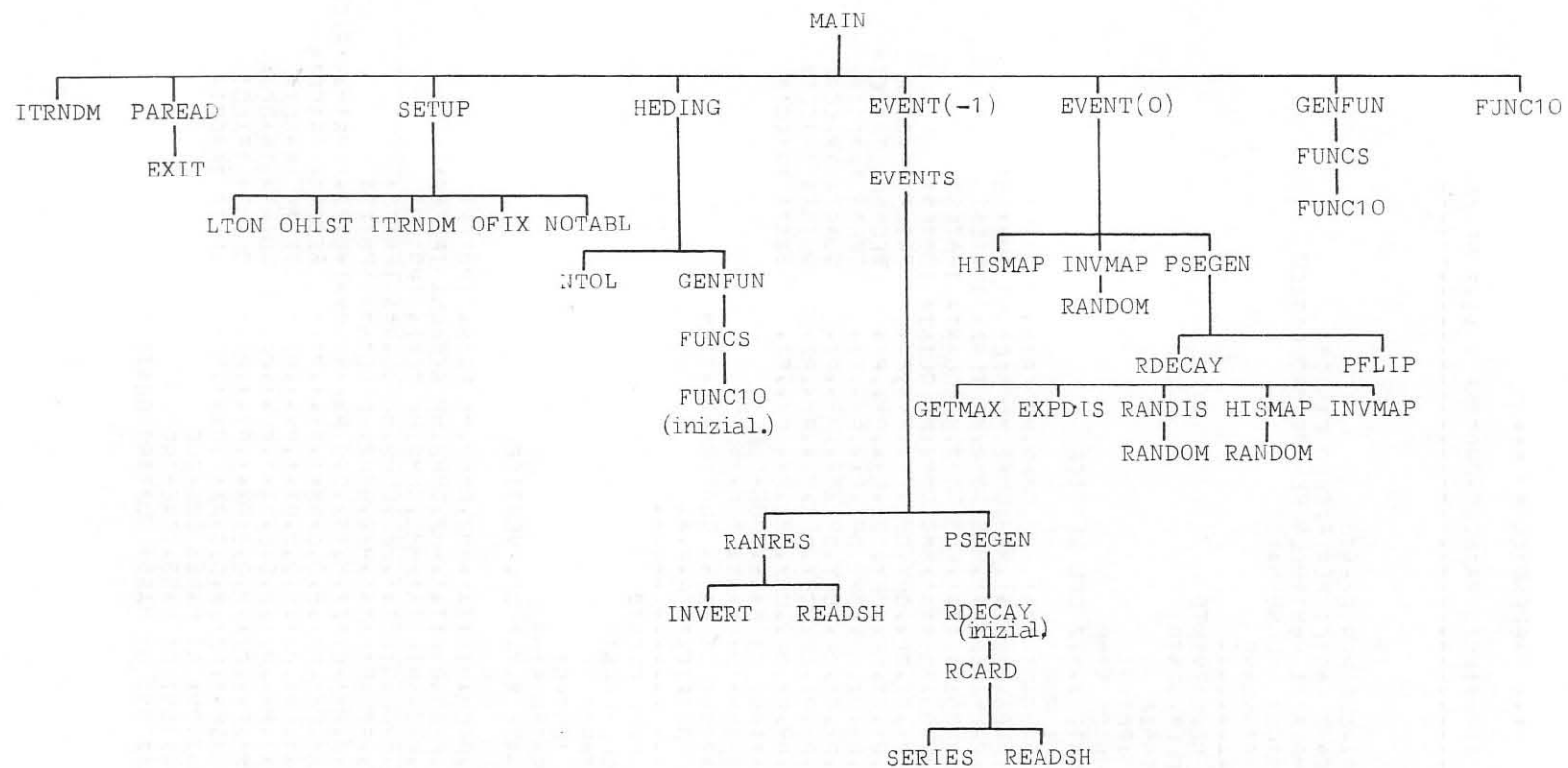
125 PAROLE PER EVENTO
7 RECORD LOGICI PER RECORD FISICO
1 PAROLA DI CONTROLLO PER RECORD FISICO
5 IPOTESI NEL SOMMARIO

STRUTTURA RECORD LOGICO STANDARD

1 NUMERO DELL'EVENTO
2 TIPO DELL'EVENTO
3 LABORATORIO
4 NUMERO IPOTESI
5 FLAG PROTON STOP
6,7,8 COORDINATE X,Y,Z DEL VERTICE
9 NUMERO DI RUN
10-14 CODICE,PROBABILITA',MM2,DMM2,ME PRIMA IPOTESI
15-19 CODICE,PROBABILITA',MM2,DMM2,ME SECONDA IPOTESI
20-24 CODICE,PROBABILITA',MM2,DMM2,ME TERZA IPOTESI
25-29 CODICE,PROBABILITA',MM2,DMM2,ME QUARTA IPOTESI
30-34 CODICE,PROBABILITA',MM2,DMM2,ME QUINTA IPOTESI
35-45 E,L,PX,PY,P7,DPX,DPY,DPZ,PX\*,PY\*,PZ\* PRIMA TRACCIA
46-56 E,L,PX,PY,P7,DPX,DPY,DPZ,PX\*,PY\*,PZ\* SECONDA TRACCIA
57-67 E,L,PX,PY,P7,DPX,DPY,DPZ,PX\*,PY\*,PZ\* TERZA TRACCIA
68-78 E,L,PX,PY,P7,DPX,DPY,DPZ,PX\*,PY\*,PZ\* QUARTA TRACCIA
79-89 E,L,PX,PY,P7,DPX,DPY,DPZ,PX\*,PY\*,PZ\* QUINTA TRACCIA
90-100 E,L,PX,PY,P7,DPX,DPY,DPZ,PX\*,PY\*,PZ\* SESTA TRACCIA
101-110 COMBINAZIONI DI MASSA DUE-DUE
111-120 COMBINAZIONI DI MASSA TRE-TRE
121-125 COMBINAZIONI DI MASSA QUATTRO-QUATTRO

STRUTTURA RECORD LOGICO MODIFICATO P.P.A.

1 NUMERO DELL'EVENTO
2 TIPO DELL'EVENTO
3 LABORATORIO
4 NUMERO IPOTESI
5 FLAG PROTON STOP
6,7,8 COORDINATE X,Y,Z DEL VERTICE
9 NUMERO DI RUN
10-14 CODICE,PROBABILITA',MM2,DMM2,ME PRIMA IPOTESI
15-19 CODICE,PROBABILITA',MM2,DMM2,ME SECONDA IPOTESI
20-24 CODICE,PROBABILITA',MM2,DMM2,ME TERZA IPOTESI
25-29 CODICE,PROBABILITA',MM2,DMM2,ME QUARTA IPOTESI
30-34 CODICE,PROBABILITA',MM2,DMM2,ME QUINTA IPOTESI
35-45 E',PX'=0,PY'=0,PZ',MASSA,0,0,PX\*'=0,PY\*'=0,PZ\*'=0 PRIMA TRACCIA
46-56 E',L,PX',PY',PZ',M,0,0,PX\*',PY\*',PZ\*'=0 SECONDA TRACCIA
57-67 E',L,PX',PY',PZ',M,0,0,PX\*',PY\*',PZ\*'=0 TERZA TRACCIA
68-78 E',L,PX',PY',PZ',M,0,0,PX\*',PY\*',PZ\*'=0 QUARTA TRACCIA
79-89 E',L,PX',PY',PZ',M,0,0,PX\*',PY\*',PZ\*'=0 QUINTA TRACCIA
90-100 E',L,PX',PY',PZ',M,0,0,PX\*',PY\*',PZ\*'=0 SESTA TRACCIA
101-110 COMBINAZIONI DI MASSA DUE-DUE
111-120 COMBINAZIONI DI MASSA TRE-TRE
121-125 COMBINAZIONI DI MASSA QUATTRO-QUATTRO



80

\*\*\* APPENDICE B \*\*\*

FLOW-CHART GENERALIZZATO DEL PROGRAMMA SR71A

\*\*\* APPENDICE C \*\*\*

ESEMPLI DI SCHEDE DATI PER SR71A

```

-----
-1          TENTATIVO N 1
  1  938.      R  MASSA DFI PROTONE
  2  139.      R  MASSA DFI PAIMENO
  3  134.      R  MASSA DFI PAIZERO
 21  773.      R  MASSA DFILA RHOMENO
 31  152.      R  GAMMA RHOMENO
 95   0.       R  X DIRECTION COSINE
 96   0.       R  Y DIRECTION COSINE
 97   1.       R  Z DIRECTION COSINE
 98   1.       R  SCALE FACTOR FOR PRINTOUT MASSES
 99          32000 I  NUMBER OF TYPE 1 EVENTS
000          FND OF PARAMETER CARDS
H  2  1  11200.0  RHOMENO
V  1          A
T  1          AX
T  21         AB
T  2          BY
T  3          BZ
G  10         10 AX BY RZ
-          FND OF SPECIFICATIONS
  1  3  1
 20  1
      0.8      0.01
EE
ANGOLO DI PRODUZIONE RHO
.000F 00 .291E-01 .000F 00 .727E 00 .141E 01 .183E 01 .147F 01 .529E 01
.545F 01 .588E 01 .354F 01 .548E 01 .794E 01 .916E 01 .166F 02 .223E 02
.343F 02 .738E 02 .133F 03 .269E 03
  2  3  2
 50  1
      -1.0     0.04
EE
COSENO ANGOLO DI GOTTFRIED-JACKSON PESATO RHO
.220F 02 .190E 02 .210F 02 .211E 02 .202E 02 .200E 02 .123F 02 .180E 02
.121F 02 .175E 02 .137F 02 .147E 02 .155E 02 .123E 02 .125F 02 .118E 02
.103F 02 .123E 02 .127F 02 .135E 02 .896E 01 .774E 01 .699F 01 .787E 01
.954F 01 .560E 01 .730F 01 .102E 02 .786E 01 .799E 01 .557F 01 .774E 01
.514F 01 .839E 01 .117F 02 .944E 01 .941E 01 .884E 01 .756F 01 .105E 02
.103F 02 .101E 02 .722F 01 .933E 01 .662E 01 .721E 01 .862F 01 .121E 02
.138E 02 .883E 01
 36  1
      0.0      10.0
EE
ANGOLO DI TREIMAN-YANG PESATO RHO
.800F 01 .800E 01 .800F 01 .745E 01 .179E 02 .865E 01 .145F 02 .170E 02
.124F 02 .151E 02 .230F 02 .175E 02 .243E 02 .165E 02 .173F 02 .204E 02
.140F 02 .117E 02 .825F 01 .163E 02 .145E 02 .239E 02 .178F 02 .200E 02
.191F 02 .173E 02 .193F 02 .174E 02 .190E 02 .171E 02 .139F 02 .129E 02
.727F 01 .800E 01 .800F 01 .800E 01
-1          TENTATIVO N 2

```

```

26 2000.      R   P.D.D.
36 1.         R   FLAG PER P.D.D.
99          59000 I   NUMBER OF TYPE 2 EVENTS
000
H 2 1 11200.0  PRATEX DIFFRACTION DISSOCIATION
V 1          A
T 2          AX
T 26         AB
T 1          BY
T 3          BZ
G 16         10 BY AX BZ
-           END OF SPECIFICATIONS
50 1
1000.      40.

```

```

EE
PRATEX DIFFRACTION DISSOCIATION(P-PIZERB)
.000F 00 .536E 00 .352F 01 .486E 01 .935E 01 .102E 02 .160F 02 .179E 02
.289F 02 .432E 02 .375F 02 .486E 02 .565E 02 .573E 02 .601F 02 .654E 02
.682F 02 .514E 02 .415F 02 .373E 02 .347E 02 .353E 02 .420F 02 .228E 02
.245F 02 .328E 02 .163F 02 .289E 02 .171E 02 .927E 01 .165F 02 .278E 02
.140F 02 .165E 02 .110F 02 .122E 02 .788E 01 .757E 01 .629F 01 .693E 01
.955F 01 .805E 01 .736F 01 .547E 01 .195E 01 .450E 01 .317F 01 .309E 01
.247F 01 .173E 01
1 3 1
40 1
-1.0      0.01

```

```

EE
ANGLE DI PRODUZIONE P.D.D.
.304F 03 .222E 03 .167F 03 .948E 02 .674E 02 .423E 02 .481F 02 .345E 02
.229F 02 .137E 02 .106F 02 .118E 02 .606E 01 .467E 01 .583F 01 .427E 01
.540F 01 .400E 01 .433F 01 .104E 01 .338E 01 .258E 01 .127F 01 .263E 01
.601F 00 .428E 01 .135F 01 .113E 00 .887E 00 .000E 00 .200F 01 .190E 01
.107F 00 .385E 02 .100F 01 .209E 01 .100E 01 .468E 00 .000F 00 .000E 00
2 3 3
50 1
-1.0      0.04

```

```

EE
CORSENA ANGLE DI GOTTFRIED-JACKSON PESAT6 P.D.D.
.995F 02 .640E 02 .489F 02 .456E 02 .348E 02 .392E 02 .182F 02 .243E 02
.210F 02 .213E 02 .211F 02 .166E 02 .169E 02 .156E 02 .142F 02 .117E 02
.142F 02 .700E 01 .152F 02 .104E 02 .983E 01 .761E 01 .105F 02 .115E 02
.194F 02 .109E 02 .103F 02 .136E 02 .150E 02 .888E 01 .154F 02 .126E 02
.734F 01 .124E 02 .769F 01 .113E 02 .151E 02 .144E 02 .172F 02 .167E 02
.138F 02 .161E 02 .150F 02 .165E 02 .250F 02 .255E 02 .343F 02 .442E 02
.445F 02 .587E 02
36 1
0.0      10.0

```

```

EE
ANGLE DI TREIMAN-YANG PESAT6 P.D.D.
.238F 02 .172E 02 .176F 02 .172E 02 .183E 02 .175E 02 .210F 02 .270E 02
.396F 02 .397E 02 .295F 02 .416E 02 .320F 02 .464F 02 .308F 02 .453E 02
.435F 02 .481E 02 .489F 02 .429E 02 .458F 02 .508E 02 .402F 02 .341E 02
.386F 02 .279E 02 .261F 02 .193E 02 .257E 02 .223E 02 .215E 02 .163E 02
.217F 02 .160E 02 .203F 02 .263E 02

```

-9  
000



<u>INDICE</u>	Pagina
1. - INTRODUZIONE	1
2. - FINALITA' DEL PROGRAMMA	2
3. - DESCRIZIONE DEL PROGRAMMA	2
4. - FELICIA	
4. 1. - Main	3
4. 2. - Subroutine LIMIT	4
4. 3. - Subroutine READST	5
4. 4. - Subroutine CNXT	6
4. 5. - Subroutines STRH, EXTRH	7
5. - SR71A	
5. 1. - Main	8
5. 2. - Breve descrizione delle subroutines essenziali	14
5. 3. - Codice di errore	16
6. - FERRET	
6. 1. - Main	17
6. 2. - Subroutine FERRET 3	17
6. 3. - Subroutine FERRET	19
6. 4. - Subroutine SETPR	19
7. - BADGER	
7. 1. - Main	20
7. 2. - Subroutine READST	22
7. 3. - Subroutine WRDST	22
8. - SUMX-BADGER	23
BIBLIOGRAFIA	24
APPENDICE A	25
APPENDICE B	26
APPENDICE C	27