

KfK 4946
Dezember 1991

**FILLC —
ein Unterprogramm zur
Übertragung nuklearer
Gruppen-Wirkungsquerschnitte
aus SIGMN-Blöcken in
Datenfelder von Neutronen-
Transportcodes**

K. Kufner, G. Buckel, W. Götzmann, E. Kiefhaber
Institut für Neutronenphysik und Reaktortechnik
Projekt Nukleare Sicherheitsforschung

Kernforschungszentrum Karlsruhe

KERNFORSCHUNGSZENTRUM KARSLRUHE
Institut für Neutronenphysik und Reaktortechnik
Projekt Nukleare Sicherheitsforschung

KfK 4946

**FILLC - EIN UNTERPROGRAMM ZUR ÜBERTRAGUNG NUKLEARER
GRUPPEN-WIRKUNGSQUERSCHNITTE AUS SIGMN-BLÖCKEN IN
DATENFELDER VON NEUTRONEN-TRANSPORTCODES**

K. Küfner, G. Buckel, W. Götzmann, E. Kiefhaber

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

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

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 7500 Karlsruhe 1

ISSN 0303-4003

Zusammenfassung

FILLC ist ein Service-Unterprogramm zur Verwendung in den KAPROS-Versionen von Multigruppen- S_N -Codes. Innerhalb von KAPROS erzeugen und modifizieren einige Module nukleare Gruppen-Wirkungsquerschnitte und legen sie in einem sogenannten SIGMN-Block ab. FILLC füllt ein in den S_N -Codes verwendetes Feld C für Gruppen-Wirkungsquerschnitte mit Daten aus einem SIGMN-Block. Die vorliegende Beschreibung dokumentiert dieses Interface mit der Versionsnummer 3.4.

FILLC - A SUBROUTINE TO TRANSFER NUCLEAR GROUP CROSS-SECTIONS FROM SIGMN BLOCKS TO DATA-ARRAYS OF NEUTRON TRANSPORT CODES

Abstract

FILLC is a utility subroutine for use in the KAPROS versions of multigroup S_N codes. Within KAPROS a couple of modules create and modify nuclear cross sections and store them in a so-called SIGMN file. FILLC stores group cross sections in an array C used by some S_N codes with data provided in a SIGMN file. This report documents this interface with release number 3.4.

Inhaltsverzeichnis

1. Einleitung	1
2. Die Multigruppen-Neutronen-Transportgleichung	3
3. Benutzung von FILLC	7
3.1 Aufrufparameter von FILLC	7
3.2 Eingabeparameter von FILLC	7
3.3 Ausgabeparameter von FILLC	9
3.3.1 Bedingungen an IHT, IUPS, IHS, IDOWS	10
3.3.2 Berechnung von IHT, IUPS, IHS, IDOWS	10
3.3.3 Interpretation von negativem IHT	11
3.3.4 Speicher-Reihenfolge im Feld	11
3.3.5 Hinweis zur Prüfung von Wirkungsquerschnittsrelationen	12
3.3.6 Berechnung des Feldes MOM0AD (Moment0Adresse)	14
3.3.7 Setzen des Fehlerindikators KONTR0	14
3.4 Ändern der Reihenfolge der Auswertetypen	17
3.5 Ausführen der Transportkorrektur	18
4. Ablauf von FILLC, Unterprogramme und Dokumentation	21
4.1 Unterprogramme von FILLC	21
4.2 KAPROS-2 und KAPROS-3 Versionen	21
4.3 Programminterne Dokumentation	22
4.4 FILLC COMMON-Blöcke	22
4.5 Änderungen gegenüber der bisherigen Version 3.0	23
4.6 Änderungen für bisherige Benutzer	25
5. Hinweise auf andere Wirkungsquerschnitts-Programme	27
Anhang A. Programmablaufplan für FILLC	29
Anhang B. Programmliste für das Unterprogramm FILTRA	31
Anhang C. Anwendungsbeispiel für FILLC	35
Literaturverzeichnis	39
Index	41

Tabellenverzeichnis

Tabelle	1. Dokumentation der Variablen der Aufrufliste	8
Tabelle	2. Berechnung von IHT,IHS,IUPS und IDOWS durch FILLC	11
Tabelle	3. Speicherreihenfolge im Feld C	13
Tabelle	4. Bedeutung der von FILLC gesetzten Fehlercodes	15
Tabelle	5. Bedeutung der von FILLC zurückgegebenen Warnungscodes	16
Tabelle	6. Bedeutung der im Unterprogramm FILTERA gesetzten Fehlercodes	16
Tabelle	7. Belegung der COMMON-Blöcke in FILLC	24

1. Einleitung

Die im Kernforschungszentrum Karlsruhe im Rahmen des Programmsystems KAPROS routinemäßig eingesetzten Multigruppen- S_N -Codes ONETRA /1/, TWOTRA /2/ und DIAMANT2 /3/ übernehmen in der Regel makroskopische (und mikroskopische zu Auswertezwecken) Gruppen-Wirkungsquerschnitte aus einer Datei in SIGMN-Struktur /4/. Für die Implementierung der S_N -Codes in KAPROS /5/ wurde dazu 1979 das Unterprogramm FILLC /6/ erstellt. Die letzte Beschreibung liegt als Primärbericht /7/ aus dem Jahre 1986 vor. Inzwischen wird FILLC auch in den Transport-Störungstheorie-Codes TP1 /8/, TP2 /9/ und TPTRIA /10/ sowie anderen neueren Programmen, z.B. TRANSX /11/, verwendet.

FILLC erfüllt folgende Bedingungen:

- es ist verträglich mit KAPROS, d.h. die Wirkungsquerschnitte werden aus einem KAPROS-Datenblock, $DBN = SIGMN, IND = 1$, gelesen;
- Bedeutung und Verwendung der Wirkungsquerschnitt-Typen aus dem SIGMN-Block sind klar definiert (diesem Zweck dient u.a. dieser Bericht);
- FILLC kann ungeändert für ONETRA /1/, TWOTRA /2/, DIAMANT2 /3/, TP1 /8/, TP2 /9/, TPTRIA /10/ und TRANSX /11/ sowie für alle KAPROS-Module, die Wirkungsquerschnitte im sogenannten "Los-Alamos"-Format (siehe "Speicher-Reihenfolge im Feld" auf Seite 11) verarbeiten, eingesetzt werden (Standardisierungseffekt);
- FILLC kann auch leicht dazu benutzt werden, um Wirkungsquerschnitte aus einem KAPROS-SIGMN-Block zu lesen und in einem geeigneten Format als Eingabedaten auszugeben (wichtig für die Verwendung in "stand-alone"-Programmen wie zum Beispiel TWODANT /14/);
- FILLC hat eine klare Fehlerdiagnostik und setzt für jeden aufgetretenen Fehler einen eigenen Fehlercode. Dadurch kann in dem FILLC rufenden Modul einfach über eine geeignete Reaktion auf aufgetretene Fehler entschieden werden.

Der vorliegende Bericht beschreibt die Version 3.4 von FILLC vom August 1991. Gegenüber den älteren Versionen /6/ und /7/ wurden eine Reihe von Änderungen und Erweiterungen vorgenommen. So ist es möglich, die Reihenfolge der Einspeicherung der Skalartypen zu beeinflussen; die Verarbeitung von 16 Bytes langen Wirkungsquerschnitt-Typnamen ist integriert; die Fehlerdiagnostik wurde verbessert und erweitert; während der normalen Verarbeitung wird Diagnostik-Ausdruck erzeugt; neue COMMON Blöcke (FILLCY; FILLCZ) wurden eingeführt; die maximale Anzahl für Skalartypen wurde auf 100 erhöht; für die Transportkorrektur wurden neue Optionen implementiert. Für Rechnungen mit 208 Energiegruppen wird eine Spezialbehandlung vorgenommen. Auch Version 3.4 ist vollständig in FORTRAN-77 geschrieben.

Member FILLC die aktuelle Version 3.4 und Member FILLCNEU eine Testversion für weitere Verbesserungen. Eine mit dem SIEMENS FORTRAN-77 Compiler übersetzte und gelinkte Version steht in der KAPROS-2-Bibliothek KAPROS.F77.KSLIB unter dem Namen FILLC bereit und kann mit "INCLUDE KSBIB(FILLC,WQRG)" in einem LINK-Step an KAPROS-Module angeschlossen werden (s. Beispiel in "Anhang C. Anwendungsbeispiel für FILLC" auf Seite 35). Für KAPROS-3 stehen entsprechend die SIEMENS FORTRAN-77 Version in KAPROS.KS3.LOAD und die IBM VS-FORTRAN Version in KAPROS.KS3V.LOAD als Member FILLC bereit. Hier entfällt der "INCLUDE"-Schritt, da FILLC automatisch zugelinkt wird.

Bisherige Benutzer müssen in ihrem Programm die geänderte COMMON-Block-Struktur berücksichtigen (s. "FILLC COMMON-Blöcke" auf Seite 22).

Nomenklatur

Im folgenden Text wird das Zeichen "b" verwendet, um erforderliche Leerzeichen zu kennzeichnen.

Die Begriffe "Feld C", "Matrix C" und "C" werden synonym verwendet. In den Programmlisten und einigen Tabellen wird auch der englische Begriff "Array C" verwendet. Nach Meinung der Autoren sollte das nicht zu Fehlinterpretationen führen.

Der Begriff "Auswertetyp" bzw. "skalärer Wirkungsquerschnitt" bezeichnet Wirkungsquerschnitte, die pro Material und Energiegruppe nur einen Wert haben. "Vektortypen" haben dagegen pro Material und Energiegruppe mehrere Werte.

2. Die Multigruppen-Neutronen-Transportgleichung

Wir betrachten die stationäre Multigruppen-Neutronen-Transportgleichung :

$$\vec{\Omega} \cdot \vec{\nabla} \psi(\vec{r}, \vec{\Omega}, g) + \sigma_t(\vec{r}, g) \psi(\vec{r}, \vec{\Omega}, g) = \sum_{h=1}^{IGM} \int_0^{4\pi} d\vec{\Omega}' T(\vec{r}, \vec{\Omega}, \vec{\Omega}', g, h) \psi(\vec{r}, \vec{\Omega}', h) + \chi(\vec{r}, g) \sum_{h=1}^{IGM} v\sigma_f(\vec{r}, h) \phi(\vec{r}, h) + Q(\vec{r}, \vec{\Omega}, g)$$

wobei folgende Bezeichnungen gewählt werden:

\vec{r}	Ortsvektor der Neutronen
$\vec{\Omega}, \vec{\Omega}'$	Einheitsvektoren für den Raumwinkel
g, h	Energiegruppenindizes
$\psi(\vec{r}, \vec{\Omega}, g)$	winkelabhängige Neutronenflußdichte (kurz: Winkelfluß)
$\phi(\vec{r}, g)$	winkelintegrierte Neutronenflußdichte (kurz: Skalarfluß)
$\sigma_t(\vec{r}, g)$	totaler Neutronenwirkungsquerschnitt in Gruppe g
$T(\vec{r}, \vec{\Omega}, \vec{\Omega}', g, h)$	Streu-Wirkungsquerschnitt von $(h, \vec{\Omega}')$ nach $(g, \vec{\Omega})$
$\chi(\vec{r}, g) :$	Spaltspektrum (evtl. ortsabhängig über die Mischungsabhängigkeit, aber unabhängig von h, d.h., von der Energie der die Spaltung auslösenden Neutronen).
$v\sigma_f(\vec{r}, g)$	Spaltproduktions-Wirkungsquerschnitt für Energiegruppe g
$Q(\vec{r}, \vec{\Omega}, g)$	sonstige (externe) Neutronenquellen
IGM	Anzahl der betrachteten Energiegruppen

Zur näherungsweisen Berücksichtigung anisotroper Streuprozesse kann $T(\vec{r}, \vec{\Omega}, \vec{\Omega}', g, h)$ in eine (nach dem Glied der Ordnung ISCT) abgebrochene Legendre-Reihe entwickelt werden /12/

$$T(\vec{r}, \vec{\Omega}, \vec{\Omega}', g, h) = \sum_{j=0}^{ISCT} \frac{(2j+1)}{4\pi} \sigma_j(\vec{r}, g, h) P_j(\mu) \quad \text{mit } \mu = \vec{\Omega} \cdot \vec{\Omega}'$$

wobei $P_j(\mu)$ die Legendre-Polynome sind.

Setzt man diese Entwicklung in die Transportgleichung ein, so erhält man (unter Ausnutzung der Additionstheoreme für Legendre-Polynome) :

$$\vec{\Omega} \cdot \vec{\nabla} \psi(\vec{r}, \vec{\Omega}, g) + \sigma_t(\vec{r}, g) \psi(\vec{r}, \vec{\Omega}, g) = \sum_{h=1}^{IGM} \sum_{j=0}^{ISCT} (2j+1) \sigma_j(\vec{r}, g, h) \sum_{k=0}^j R(k, j) \psi(\vec{r}, h, k, j) + \chi(\vec{r}, g) \sum_{h=1}^{IGM} v \sigma_f(\vec{r}, h) \phi(\vec{r}, h) + Q(\vec{r}, \vec{\Omega}, g)$$

Dabei sind die $R(k, j)$ die Kugelfunktionen 2. Art ("spherical harmonics") /2/ und die Flußmomente definiert durch:

$$\psi(\vec{r}, g, k, j) = \frac{1}{2\pi} \int_{-1}^{+1} \int_0^\pi d\gamma d\eta R(k, j, \gamma, \eta) \psi(\vec{r}, \vec{\Omega}, g)$$

$$\gamma = \vec{e}_x \cdot \vec{\Omega}, \quad \eta \text{ definiert durch } \vec{e}_y \cdot \vec{\Omega} = \sqrt{(1 - \gamma^2)} \cos \eta$$

Dabei steht der Operator \cdot für das Skalarprodukt zweier Vektoren. Diese Formeln gelten für zweidimensionale Geometrien; in eindimensionaler Geometrie ergeben sich Vereinfachungen.

In FILLC wird folgende Zuordnung von SIGMN-Typnamen zu den Variablen der Transportgleichung getroffen:

<i>Symbol</i>	<i>Wirkungsquerschnitt</i>	<i>SIGMN-Typ</i>
σ_t	totaler Wirkungsquerschnitt	STOT
χ	Spaltspektrum, evtl. mischungsabhängig	CHI
$v\sigma_f$	Spaltproduktions-Wirkungsquerschnitt	NUSF
σ_0	Streumatrix (0.Moment)	SMTOT
σ_j	Legendre-Streu-Momente ($j = 1, ISCT$)	SMTO _j

Die gegenwärtig gültige Version 3.4 ist beschränkt auf die anisotropen Momente SMT01 bis SMT06; man beachte, daß die letzten zwei Zeichen im Typnamen Ziffern sind.

Einige skalare Wirkungsquerschnitte, wie STOT und NUSF, werden an festen Positionen des Feldes C gespeichert. Weitere skalare Wirkungsquerschnitte, die im oberen Teil des Feldes C eingespeichert werden können, dienen Auswertezwecken. Einzelheiten zur Speicherreihenfolge werden später beschrieben (siehe Tabelle 3 auf Seite 13).

In einigen Transportprogrammen, z.B. ONEDANT /13/ und TWODANT /14/, muß in der Eingabe spezifiziert werden, ob der Faktor $(2j + 1)$ bereits in den Zahlenwerten der höheren Momente enthalten ist oder nicht.

Die Transportprogramme werden nicht nur zur Modellierung des Transports von Neutronen eingesetzt, sondern auch für γ -Quanten. Die Kopplung von Neutronen- und γ -Feldern kann durch eine Modifikation der Bedeutung der Streumatrix erreicht werden. Auch (γ, n) -Prozesse können im Prinzip als "Aufwärtsstreuung" berücksichtigt werden. FILLC ist zur Wirkungsquerschnittsübernahme auch in solchen Rechnungen geeignet, vorausgesetzt, daß zuvor ein geeigneter SIGMN-Block erstellt wurde.

Es sei besonders darauf hingewiesen, daß ohne besondere Maßnahmen des Benutzers die sog. "Transportkorrektur" /12/ des totalen Wirkungsquerschnitts in FILLC unterbleibt. Das Unterprogramm FILTERA, das diese Korrektur ausführt, wird nur aufgerufen, wenn im COMMON-Block FILLCO die Variable NTRANS einen Wert $\neq 0$ hat. Durch seinen Aufruf werden die Werte für STOT und für SMTOT($g \rightarrow g$) abgeändert (siehe "Ausführen der Transportkorrektur" auf Seite 18). Über die Durchführung dieser Korrektur erfolgt eine Meldung im Bearbeitungsprotokoll.

3. Benutzung von FILLC

3.1 Aufrufparameter von FILLC

FILLC füllt das Feld C im sog. "Los Alamos" Format /2/ (s.a. Tabelle 3 auf Seite 13) mit den erforderlichen Wirkungsquerschnitts-Daten aus einem SIGMN-Block. Dabei werden die Wirkungsquerschnitts-Daten stets unter dem Datenblocknamen $DBN = SIGMN$, $IND = 1$ erwartet.

FILLC kommuniziert mit dem rufenden Programm in erster Linie durch die folgende Parameterliste :

```
SUBROUTINE FILLC (C, IHM, IGM, NUMTAB, CHI, IGP, VELOC,  
+                MATTAB, MOMOAD, NUMMAT, NFO, NPROT, ISCT,  
+                IHS, IHT, IUPS, IDOWS, MDCHI, KONTR0)
```

Tabelle 1 enthält eine Beschreibung der Aufrufparameter von FILLC. Die erste Spalte gibt den Variablennamen an sowie evtl. vorhandene Dimensionen; Spalte zwei gibt den Typ der Variablen an ($RA = REAL$ -Feld, $IA = INTEGER$ -Feld, $I = INTEGER$ -Zahl). Alle REAL-Zahlen sind von einfach genauer Wortlänge (4 Bytes). In der rechten Spalte wird angegeben, welche Variablen Eingabegrößen und welche Ausgabegrößen sind. **Die Ausgabegrößen dürfen nicht als Konstanten übergeben werden.**

Daneben definiert FILLC noch vier COMMON-Blöcke, über die Informationen weitergegeben werden. Diese werden später beschrieben (siehe "FILLC COMMON-Blöcke" auf Seite 22).

3.2 Eingabeparameter von FILLC

Die folgende Beschreibung erläutert die Eingabeparameter in Tabelle 1 etwas näher.

IGM, IGP, IHM, NUMMAT und NUMTAB sind Dimensionierungsgrößen und mit sinnvollen Werten anzuliefern. Besonders ist auf die unterschiedliche Bedeutung von NUMMAT und NUMTAB hinzuweisen: NUMMAT ist die Anzahl der im Feld C gespeicherten Mischungen. NUMTAB ist stets größer oder gleich NUMMAT; für jede anisotrop zu behandelnde Mischung (angezeigt durch ein Minuszeichen vor der Mischungsnummer im Feld MATTAB) muß NUMTAB gegenüber NUMMAT um ISCT erhöht werden. Der Wert IGP

Variable	Typ	Explanation	In/Output
C(IHM,IGM, NUMTAB)	RA	Array to contain cross-sections in the sequence defined in FILLC documentation	O
IHM	I	Number of rows of C ($IHM \geq IHS + IDOWS$)	I
IGM	I	Number of energy groups	I
NUMTAB	I	Number of cross-section tables to be stored; = NUMMAT if ISCT = 0 (isotropy)	I
CHI (IGP, NUMMAT)	RA	Fission spectrum (possibly mixture dependent if MDCHI \neq 0)	O
IGP	I	Number of energy groups + 1 or IGM	I
VELOC(IGM)	RA	Associated reciprocal velocities of each energygroup	O
MATTAB (NUMMAT)	IA	Mixture numbers in any order; a negative number indicates anisotropic treatment of scattering	I
MOM0AD (NUMMAT)	IA	Addresses of zeroth order moment of scattering matrix if ISCT > 0	O
NUMMAT	I	Number of different mixtures to be handled	I
NFO	I	Standard output unit (usually = 6)	I
NPROT	I	Output unit for error messages (usually = 42)	I
ISCT	I	Anisotropic scattering order ($\leq ISCTMX$)	I/O
IHS	I	Row number of selfscattering cross-section, SMTOT(g→g), in C ($IHS < IHM$)	I/O
IHT	I	Row number of total cross-section in C; $ABS(IHT) \geq 5$; $ABS(IHT) < IHS$	I/O
IUPS	I	Upscattering width ($0 \leq IUPS < IGM$)	I/O
IDOWS	I	Downscattering width ($0 \leq IDOWS < IGM$)	I/O
MDCHI	I	Use of mixture-dependent CHI if \neq 0	I
KONTRO	I	Error indicator	O

Tabelle 1. Dokumentation der Variablen der Aufrufliste

dient als Dimensionierungsvariable für das Feld CHI. In DIAMANT2 ist $IGP = IGM + 1$, in ONETRA und TWOTRA dagegen $IGP = IGM$. Gefüllt werden im Feld CHI jedoch nur (bei mischungsabhängigen Spektren: jeweils für jede Mischung) die ersten IGM Plätze. **Da CHI als zweidimensionales Feld definiert ist, muß seine erste Dimension (IGP) im rufenden Programm und FILLC gleich sein.**

ISCT gibt die Ordnung der Legendre-Entwicklung für anisotrope Streuung an (in Version 3.4 von FILLC ist ISCT begrenzt auf 6). MDCHI steuert, ob die Werte für das Spaltspektrum (Feld CHI) mischungsunabhängig ($MDCHI = 0$) oder mischungsabhängig ($MDCHI \neq 0$) gespeichert werden.

Das Feld MATTAB erfüllt zwei Funktionen. Zunächst können mit ihm Mischungen aus dem SIGMN-Block ausgewählt werden, d.h., die Wirkungsquerschnitte für Mischung Nummer i in der S_N -Rechnung entsprechen der Mischung MATTAB(i) im SIGMN-Block. Dadurch kann erreicht werden, daß auch für große SIGMN-Blöcke die Anzahl der im S_N -Code zu speichernden Werte dem jeweiligen Problem angepaßt werden kann. Sodann dient MATTAB auch zur Auswahl der anisotrop zu behandelnden Mischungsnummern. Ist MATTAB(i) negativ, so wird die Mischung -MATTAB(i) anisotrop (bis Grad ISCT) gerechnet. Dazu ist notwendig, daß unter den Typnamen SMT01, SMT02, ... die entsprechenden Momente im SIGMN-Block vorhanden sind. **Es wird vorausgesetzt, daß für alle Mischungen, die als anisotrop deklariert sind, ISCT, d.h. der Grad der Anisotropie, gleich ist.**

Die Ein- und Ausgabevariablen IHS, IHT und IDOWS werden im nächsten Kapitel näher erläutert. IUPS gibt die Anzahl der tatsächlich einzuspeichernden Aufwärtsstreuterme an. Insbesondere liegt für $IUPS = 0$ ein reines Abwärtsstreuproblem vor. NFO und NPROT sind Standarddruckausgabe- bzw. KAPROS-Protokolleinheit.

Die Variable NTRANS im COMMON-Block /FILLCO/ steuert, ob das Unterprogramm FILTRA (s. "Ausführen der Transportkorrektur" auf Seite 18) aufgerufen wird ($NTRANS \neq 0$) oder nicht ($NTRANS = 0$). **Daher muß diese Variable im rufenden Programm gesetzt werden.**

3.3 *Ausgabeparameter von FILLC*

Die folgende Beschreibung erläutert die Angaben in Tabelle 1 bezüglich der Ausgabeparameter etwas näher. Außer den Feldern C, CHI, VELOC und MOM0AD können von FILLC noch die Aufrufparameter IHT, IHS, IUPS, IDOWS, und KONTRO gesetzt bzw. verändert werden.

3.3.1 Bedingungen an IHT, IUPS, IHS, IDOWS

IHT legt die Anzahl der in C für die Speicherung von Skalararten verfügbaren Plätze fest. Die Differenz IHS-IHT-1 legt die für Aufwärtsstreuung reservierte Anzahl von Reihen in C fest (für reine Abwärtsstreuere Probleme ist IUPS = 0 und IHS = IHT + 1). Insbesondere muß $IUPS \leq IHS - IHT - 1$ gelten. IDOWS gibt die Anzahl der tatsächlich eingespeicherten Abwärtsstreuere (falls > 0) im Feld C an. **Daher muß $IHM \geq IHS + IDOWS$ sein.**

3.3.2 Berechnung von IHT, IUPS, IHS, IDOWS

Im folgenden Abschnitt ist IHT stets aufzufassen als ABS(IHT); die spezielle Bedeutung von $IHT < 0$ wird in " Interpretation von negativem IHT" auf Seite 11 beschrieben.

Für die implementierte Version von FILLC gilt die Einschränkung $IHT \leq 100$ (festgelegt durch den FORTRAN PARAMETER MAXSKA, siehe "FILLC COMMON-Blöcke" auf Seite 22); der Wert dieses PARAMETERS legt auch die Dimension der Felder SIGTYP (COMMON-Block /FILLCX/) und TYPEXT (COMMON-Block /FILLCY/) fest. Ferner wird der Wert des PARAMETERS als Variable MXSKA im COMMON-Block /FILLCO/ zurückgegeben.

Da die fünf Eingabegrößen IHT, IHS, IUPS, IDOWS und IHM nicht unabhängig voneinander sind, versucht FILLC, einen möglichst umfassenden Test auf Konsistenz durchzuführen und entsprechende Diagnostikmeldungen auszugeben.

Darüber hinaus ist es unter gewissen Umständen möglich, einige dieser Variablen von FILLC berechnen zu lassen. Wird beim Aufruf für IHS oder IDOWS ein Wert ≤ 0 , $IHT = 0$ oder $IUPS < 0$ eingegeben, so berechnet FILLC sinnvolle Werte. Der Fall $IHT < 0$ hat eine besondere Bedeutung und wird später beschrieben (siehe " Interpretation von negativem IHT" auf Seite 11). $IUPS = 0$ zeigt ein reines Abwärtsstreuere Problem an.

Ist $IHT = 0$ beim Aufruf, so setzt FILLC $IHT = 7$. Für IHS und IUPS gibt es 4 Kombinationen, die gemäß Tabelle 2 abgearbeitet werden, um zu sinnvollen Werten zu kommen.

Für $IDOWS \leq 0$ wird gesetzt : $IDOWS = IHM - IHS$. Für $IDOWS > 0$ wird getestet, ob $IHS + IDOWS \leq IHM$ gilt. Sollte dies nicht erfüllt sein, so besteht die Gefahr von Überspeicherungen. Daher setzt FILLC den Fehlercode KONTRO = 6 (s. Tabelle 4) und beendet die Verarbeitung.

Sind die so berechneten bzw. eingegebenen Werte für IDOWS bzw. IUPS kleiner als die im SIGMN-Block gespeicherten, dann setzt FILLC den Fehlercode KONTRO = 17 (bei IDOWS) bzw. 16 (bei IUPS) und beendet die Verarbeitung (s. Tabelle 4).

Wert beim Aufruf	Wirkung
$IHT = 0$	$IHT := 7$
$IHS \leq 0, IUPS < 0$	keine Berechnung möglich; $KONTRO = 5$
$IHS \leq 0, IUPS \geq 0$	$IHS := IHT + IUPS + 1$
$IHS > 0, IUPS < 0$	$IUPS := IHS - IHT - 1$
$IHS > 0, IUPS > 0$	Prüfung, ob $IUPS = IHS - IHT - 1$ bzw. $IHS = IHT + IUPS + 1$ gilt
$IDOWS > 0$	Prüfung auf $IHS + IDOWS \leq IHM$
$IDOWS \leq 0$	$IDOWS := IHM - IHS$

Tabelle 2. Berechnung von IHT,IHS,IUPS und IDOWS durch FILLC

3.3.3 Interpretation von negativem IHT

Ein negativer Eingabewert für IHT zeigt an, daß das rufende Programm erwartet, daß (n,2n)-Wirkungsquerschnitte im Feld C abgelegt werden sollen. Daher erfolgt lokal in FILLC die Verarbeitung mit dem Absolutbetrag von IHT wie oben beschrieben. Sollte der SIGMN-Typ SN2N nicht im SIGMN-Block vorhanden sein, erfolgt eine Fehlermeldung und FILLC setzt den Fehlercode $KONTRO = 15$ (s. Tabelle 4).

3.3.4 Speicher-Reihenfolge im Feld

Im "Los Alamos" Format wird für jede Mischung eine sogenannte "Tabelle" im Feld C angelegt, d.h., die Daten für die Mischung j werden in dem zweidimensionalen Sub-Feld $C(*,*,j)$ abgespeichert. Für Mischungen mit anisotroper Streuung werden zusätzliche Tabellen mit den höheren Streumomenten gespeichert, und zwar unmittelbar hinter den Daten des 0.Momentes. Das eindimensionale Sub-Feld $C(*,i,j)$ enthält die Daten für Mischung j und Energiegruppe i, $i=1$ entspricht dabei der Neutronengruppe mit der höchsten Energie. FILLC verwendet dieses Speicherschema für das Feld C.

Tabelle 3 zeigt die Reihenfolge (soweit zwingend vorgeschrieben) der einzelnen Skalartypen im Feld C (für eine feste Mischung j und eine gegebene Energiegruppe i). Für die implementierte Version von FILLC gilt die Einschränkung $IHT \leq 100$ (festgelegt durch den FORTRAN PARAMETER MAXSKA); der Wert dieses PARAMETERS legt auch die Di-

mension der Felder SIGTYP (COMMON-Block /FILLCX/) und TYPEXT (COMMON-Block /FILLCY/) fest. Ferner wird der aktuelle Wert des PARAMETERS als Variable MXSKA im COMMON-Block /FILLCO/ zurückgegeben.

In den Positionen 1 bis IPSFIS-1 wird versucht, zu verhindern, daß Typen, die in den Positionen IPSFIS bis IHT eingespeichert werden sollen, nochmals gespeichert werden.

Die Typnamen der Skalarmtypen werden in den COMMON-Blöcken FILLCX und FILLCY in der im Feld C verwendeten Reihenfolge zurückgegeben. Die Variablen IP... im COMMON FILLCO geben die Reihe an, in der der jeweilige Wirkungsquerschnitts-Typ gespeichert ist (siehe "FILLC COMMON-Blöcke" auf Seite 22).

Die Tabellen für höhere anisotrope Wirkungsquerschnitts-Momente enthalten in den Positionen 1 bis IHT stets den Wert 0.0.

Die Reihenfolge der Skalarmtypen im oberen Teil der Matrix C entspricht im Normalfall der Reihenfolge im Erklärungsteil des SIGMN-Blockes. Dabei sind allerdings die Zeilen IHT, IHT-1, IHT-2 stets reserviert für die Typnamen STOT, NUSF und SABS. Sodann ist die Zeile IHT-3 reserviert für den Typ SN2N, falls dieser vorhanden ist bzw. für den Typ STR, falls dieser vorhanden ist. Sind beide Typen vorhanden, dann ist auch die Zeile IHT-4 reserviert. Oberhalb davon werden (z.B. in den Zeilen IHT-5 und IHT-6) noch Plätze für die Querschnitte mit den Typnamen SCAPT (Einfang-Wirkungsquerschnitt) und SFISS (Spalt-Wirkungsquerschnitt) reserviert. Die Anzahl der für Auswertungen zur Verfügung stehenden freien Plätze ist somit IPSFIS-1 ($IHT-3 \geq IPSFIS \geq IHT-6$; vgl. auch Tabelle 3).

Die maximale Auf- und Abwärtsstrebweite, IUPS und IDOWS, ist für alle Streumatrizen gleich (s. aber KONTR0 = 16,17 in Tabelle 4 auf Seite 15).

Benutzern von ONEDANT /13/ und TWODANT /14/, die FILLC verwenden wollen, wird empfohlen, die Informationen über die dort verwendeten Speicherschemata für das Feld C in den angegebenen Programmbeschreibungen nachzulesen.

3.3.5 Hinweis zur Prüfung von Wirkungsquerschnittsrelationen

Bei einer Überprüfung der Wirkungsquerschnittsbilanz kann gelegentlich eine Abweichung festgestellt werden zwischen dem Wert von SABS, definiert als

$$SABS_g = SCAPT_g + SFISS_g$$

(wie in FILLC gemacht), und dem Wert SABS, definiert als

k	C(k,i,j)	Erläuterung
1 IPSFIS-1	Auswertetypen	IHT-IPSFIS Skalararten in der Reihenfolge, wie im Erklärungsteil des SIGMN-Blockes festgelegt (siehe aber dazu auch die Bemerkungen zur Speicherung in " Speicher-Reihenfolge im Feld" auf Seite 11 sowie in "Ändern der Reihenfolge der Auswertetypen" auf Seite 17)
IPSFIS	SFISS	Spalt-Wirkungsquerschnitt
IPSCAP	SCAPT	Einfang-Wirkungsquerschnitt
IPSN2N	SN2N	(n,2n)-Wirkungsquerschnitt (nur falls vorhanden)
IPSTR	STR	Transport-Wirkungsquerschnitt (nur falls vorhanden)
IHT-2	SABS	Absorptions-Wirkungsquerschnitt, berechnet als SCAPT+SFISS
IHT-1	NUSF	Spaltproduktions-Wirkungsquerschnitt
IHT	STOT	Totaler Wirkungsquerschnitt (u.U. auch Typ STRTR oder STR)
IHT + 1 IHS-1	SMTOT	Aufwärtsstreu-Wirkungsquerschnitt (nur falls IUPS ≠ 0 bzw. IHT < IHS-1) (i,j) → (i-IUPS, j) (i,j) → (i-1,j)
IHS	SMTOT	Selbststreu-Wirkungsquerschnitt (i,j) → (i, j)
IHS + i IHM	SMTOT	Abwärtsstreu-Wirkungsquerschnitt (i,j) → (i + 1, j) (i,j) → (i + IDOWS,j)
Anmerkung: Die Indizes i bzw. j stehen für Energiegruppe bzw. Mischungsnummer.		

Tabelle 3. Speicherreihenfolge im Feld C

$$SABS_g = SREM_g - \sum_{g' \neq g} SMTOT(g \rightarrow g').$$

Abgesehen von numerischen Effekten ist dies im allgemeinen auf den (n,2n)-Beitrag zurückzuführen, der in SMTOT mit dem Faktor 2 berücksichtigt ist. Diese Differenzen treten meist nur in einigen wenigen hochenergetischen Gruppen auf. Der Absorptionsquerschnitt SABS und der sogenannte Removal-Querschnitt SREM werden nicht zur Lösung der Transportgleichung verwendet, sondern gehen nur in Bilanzberechnungen ein.

Sinnvoll ist auch eine weitere Konsistenzprüfung der Wirkungsquerschnitte, nämlich, ob

$$STOT_g = SCAPT_g + SFISS_g + \sum_{g'} SMTOT(g \rightarrow g').$$

gilt. Dies ist vor allem von Bedeutung wegen der in KAPROS möglichen weitgehenden Manipulierungsmöglichkeiten bei SIGMN-Blöcken.

Diese Überprüfungen werden aber nicht von FILLC ausgeführt, sondern verbleiben dem rufenden Programm als Aufgabe.

3.3.6 Berechnung des Feldes MOM0AD (Moment0Adresse)

Im Feld MATTAB wird jede anisotrop zu behandelnde Mischung durch ein Minuszeichen vor der Mischungsnummer gekennzeichnet. Da für jede anisotrope Mischung ISCT Tabellen zusätzlich gespeichert werden müssen, ergibt sich die Beziehung: NUMTAB = NUMMAT + n*ISCT, wobei n die Anzahl der anisotropen Mischungen ist. Geprüft wird in FILLC die Bedingung NUMTAB \geq NUMMAT + n*ISCT. Nach dem Aufruf von FILLC enthält MOM0AD(i) die Nummer der Tabelle in C, die dem 0. Moment der Streumatrix für die Mischung i zugeordnet ist. Für isotrope Streuung (ISCT=0) ist demnach MOM0AD(i) = i (beachte: MATTAB(i) muß nicht gleich i sein, siehe "Eingabeparameter von FILLC" auf Seite 7).

Beispiel: ISCT = 1, NUMMAT = 5, NUMTAB = 7

Eingabe: MATTAB = 1, -2, 3, -4, 5

Ergebnis: MOM0AD = 1, 2, 4, 5, 7

3.3.7 Setzen des Fehlerindikators KONTRO

KONTRO informiert das rufende Programm über evtl. aufgetretene Fehlerbedingungen. Eine Liste der derzeit möglichen Fehlercodes wird in Tabelle 4 gegeben. Fehlercodes kleiner 100 werden von FILLC, solche größer 100 vom Unterprogramm FILTRA gesetzt (siehe Tabelle 6). Die Fehlercodes 15, 16 und 17 wurden auf Benutzerwunsch eingeführt. Sie signalisieren Inkonsistenzen zwischen Aufruf-Parametern und dem aktuellen SIGMN-Block, die leicht übersehen werden können.

KON- TRO	Bedeutung
0	no error condition encountered during execution
1	Error during initialization of subroutine WQRG /15/
2	Inconsistent group numbers in input and SIGMN block
3	Not enough mixtures in SIGMN block
4	Input value of IHT not allowed
5	Input value of IUPS not allowed or inconsistent with IHS
6	Input value of IHM is too small (danger of subscript error for array C)
7	Input value of NUMTAB is too small (danger of subscript error for array C)
8	Input value of ISCT (order of anisotropy) is greater than allowed (e.g. greater ISCTMX)
9	Some important scalar type not found in SIGMN block; one or several of the types STOT,SCAPT,SFISS,NUSF,CHI are missing
10	Some dimensioning input values are not positive
11	NUMMAT is greater than NUMTAB, which is not allowed
12	Input value of IGP is not consistent with IGM
13	Error in WQSKAL-call
14	Error in WQVEKT-call
15	IHT < 0, but type SN2N not found in SIGMN block
16	Specified or calculated IUPS \leq IUPS in SIGMN block
17	Specified or calculated IDOWS \leq IDOWS in SIGMN block

Tabelle 4. Bedeutung der von FILLC gesetzten Fehlercodes

KON- TRO	Bedeutung
-1	Some specified scalar type not found in SIGMN block
-2	Some specified vector type not found in SIGMN block
-3	Input value of IHM is too big (storage wasting thereby)
-4	Values for velocities not found in SIGMN block
-5	More cross section types in FILLC XS TYPES E and/or FILLC XS TYPES than place to store them

Tabelle 5. Bedeutung der von FILLC zurückgegebenen Warnungscodes

KON- TRO	Bedeutung
113	General error in WQSKAL
114	SREM not found
115	Neither STRTR nor STR found
116	Transport cross section for some energy group not found
117	SREM for some energy group not found
118	Transport correction of type SMTOT($g \rightarrow g$) = STOT - SREM should be applied but STOT is not contained in the SIGMN-block

Tabelle 6. Bedeutung der im Unterprogramm FILTRA gesetzten Fehlercodes

KONTRO=0 bedeutet fehlerfreien Ablauf; KONTRO<0 zeigt eine Warnung an (siehe Tabelle 5), den Werten in C kann jedoch i.a. vertraut werden; KONTRO>0 zeigt einen echten Fehler an; es kann dann das Feld C überhaupt nicht oder nur unvollständig gefüllt sein und die Werte in C sind evtl. nicht vertrauenswürdig. Es bleibt dem Programm, von dem FILLC aufgerufen wird, überlassen, auf die fehlerhafte Situation angemessen zu reagieren.

3.4 Ändern der Reihenfolge der Auswertetypen

Falls die Reihenfolge der Typnamen im SIGMN-Block ungünstig ist, müßte normalerweise eine Vielzahl nicht interessierender Typen eingespeichert werden, um einige wenige gewünschte Auswertetypen im Feld C zu speichern. D.h., auch das Feld C müßte sehr groß dimensioniert werden (Aufrufparameter IHM, IHT). Darum wurde die Möglichkeit geschaffen, die Reihenfolge der Skalartypen oberhalb IPSFIS (d.h., die Reihenfolge von Wirkungsquerschnitten, die nicht zur Lösung der Transportgleichung benötigt werden) zu ändern. Dies ist hilfreich, wenn ein bereits bestehender SIGMN-Block mit vielen Typen in einer Transportrechnung verwendet werden soll, aber nur einige wenige spezielle Wirkungsquerschnittstypen zu Auswertezwecken benötigt werden.

Eine Meldung über die Änderung der Reihenfolge wird in die Ausgabe geschrieben. Ist keiner der beiden Datenblöcke vorhanden, übernimmt FILLC die Reihenfolge aus dem Erklärungsteil des SIGMN-Blockes und speichert die ersten IPSFISS-1 Wirkungsquerschnittstypen an den Anfang des Feldes C. Um die Reihenfolge zu ändern, werden die Typnamen in einem Eingabe-Datenblock $DBN = \text{FILLC}bXSbTYPESbE, IND = 1$ (für Typnamen mit 16 Bytes, das letzte E des Namens steht für "Extended") oder $DBN = \text{FILLC}bXSbTYPES, IND = 1$ (für Typnamen mit 8 Bytes) in der gewünschten Ordnung eingegeben. (Im obigen Text steht das Zeichen "b", um erforderliche Leerzeichen zu kennzeichnen.) FILLC stellt zunächst fest, ob eine Datenblock-Eingabe überhaupt vorhanden ist (in der Reihenfolge: erst 16 Bytes-; falls nicht vorhanden, dann auch 8 Bytes-Eingabe). Wird einer der beiden Blöcke gefunden, dann erhält die Variable II2 im COMMON-Block /FILLCZ/ den Wert 1, ansonsten den Wert 0.

Dann wird berechnet, wieviele Typnamen der gefundene Datenblock enthält und geprüft, ob genug Platz im oberen Teil des Feldes C ist. Ist dies nicht der Fall, dann werden nur die vorderen Typnamen, die im Datenblock angegeben sind, berücksichtigt. Die Typnamen (und natürlich später auch die Wirkungsquerschnitte) werden dann in die Zeilen 1 bis IPSFIS-1 eingespeichert. Werden in dem Datenblock weniger Typnamen spezifiziert als im Feld C Platz ist, so werden die spezifizierten Typnamen in den Bereich III bis IPSFIS-1 eingespeichert und der Bereich 1 bis III-1 wird mit Namen aus dem Erklärungsteil des SIGMN-Blockes gefüllt. III ist dabei eine Variable im COMMON-Block /FILLCZ/ (s. "FILLC COMMON-Blöcke" auf Seite 22).

Dazu ein Beispiel:

Es sei IHT=15 und 5 feste Typnamen vorgesehen.

In $DBN = \text{FILLC}bXSbTYPES, IND = 1$ seien 6 Typnamen angegeben.

Dann sind $15 - 5 = 10$ Zeilen frei für Auswertetypen.

6 Zeilen werden mit Werten gefüllt, die den Typnamen aus

DBN=FILLCtXSbTYPES entsprechen.

4 (=10-6) Zeilen werden mit Werten gefüllt, die Typnamen aus dem Erklärungsteil des SIGMN-Blockes entsprechen.

Insgesamt ergibt sich also folgende Reihenfolge:

Zeilen 1-4 gefüllt gemäß den Typnamen des Erklärungsteils des SIGMN-Blockes

Zeilen 5-10 gefüllt gemäß den Typnamen aus DBN=FILLCtXSbTYPES

Zeilen 11-15 gefüllt mit SN2N, STR, SABS, NUSF, STOT.

II1 aus Tabelle 7 auf Seite 24 erhält den Wert 5, II2 den Wert 1.

3.5 Ausführen der Transportkorrektur

Das Unterprogramm FILTRA führt eine Transportkorrektur (s. /11/ und /12/) für STOT durch. Es wird von FILLC aufgerufen, wenn im COMMON-Block /FILLCO/ die Variable NTRANS einen Wert $\neq 0$ hat. Für anisotrope Probleme (ISCT >0) und für Aufwärtsstreu-
probleme (IUPS >0 bzw. IHT $<IHS-1$) wird vor der Transportkorrektur eine spezielle Meldung ausgegeben, da in diesen Fällen der Benutzer die physikalische Angemessenheit der Korrektur abklären sollte.

Zum Zeitpunkt des Aufrufs von FILTRA ist der ursprüngliche SIGMN-Block noch verfügbar und das (von WQRG /15/ dynamisch angelegte) Arbeitsfeld ist noch definiert. Daher kann mit den diversen Entries von WQRG auf alle Daten des SIGMN-Blockes zugegriffen werden.

Das Quellprogramm des Unterprogramms FILTRA ist in "Anhang B. Programmliste für das Unterprogramm FILTRA" auf Seite 31 aufgelistet und kann zum besseren Verständnis der folgenden Ausführungen herangezogen werden.

Bisher sind zwei Optionen realisiert: für NTRANS = 1 erfolgt die Modifikation:

$$\begin{aligned} \text{STRTR} &\Rightarrow \text{STOT} & \text{und} & & \text{STRTR} - \text{SREM} &\Rightarrow \text{SMTOT}(g \rightarrow g) \\ \text{bzw. (falls STRTR nicht im SIGMN-Block)} & & & & & \\ \text{STR} &\Rightarrow \text{STOT} & \text{und} & & \text{STR} - \text{SREM} &\Rightarrow \text{SMTOT}(g \rightarrow g) \end{aligned}$$

wobei $A \Rightarrow B$ bedeutet: Wert A ersetzt Wert B.

Für NTRANS = -1 erfolgt die Modifikation:

$$\text{STOT} - \text{SREM} \Rightarrow \text{SMTOT}(g \rightarrow g)$$

Eine Sonderbehandlung für Wirkungsquerschnitte mit 208 Energiegruppen wird in Version 3.4 vorgenommen. Einzelheiten des Ablaufs können den mit "C XXXXXXXX" gekennzeichneten Stellen in "Anhang B. Programmliste für das Unterprogramm FILTRA" auf Seite 31 entnommen werden.

Nach Beendigung von FILTRA enthält das Feld C die entsprechend modifizierten Daten.

Hinweise zu den physikalischen Hintergründen für diese Modifikationen sowie auf weiterführende Literatur sind im Anhang von /11/ enthalten.

4. Ablauf von FILLC, Unterprogramme und Dokumentation

4.1 Unterprogramme von FILLC

“Anhang A. Programmablaufplan für FILLC” auf Seite 29 zeigt ein schematisches Block-Ablaufdiagramm von FILLC.

FILLC ruft fünf weitere eigene Unterprogramme auf: FILKON zum Setzen, Ändern und Ausdrucken der Fehlercodes; FILSEQ, um die Datenblöcke FILLCbXSbTYPES bzw. FILLCbXSbTYPESbE zu lesen und zu interpretieren; FILSET, um Felder mit 0.0 zu initialisieren; FILTES, um einen einfachen Konsistenztest für die Eingabevariablen (Dimensionsgrößen) durchzuführen und schließlich FILTRA, das die Ausführung der Transportkorrektur übernimmt (s. “Ausführen der Transportkorrektur” auf Seite 18). Es wird von FILLC nur aufgerufen, falls die Variable NTRANS im COMMON-Block /FILLCO/ einen Wert $\neq 0$ hat.

Das Unterprogramm WQRG und seine Entries von W. Höbel wird verwendet, um Daten aus dem SIGMN-Block zu lesen. In /15/ wird eine Beschreibung von WQRG gegeben und es sind alle in FILLC vorkommenden Aufrufe für WQRG-Funktionen beschrieben (WQETDT, WQFREE, WQSKAL, WQVEKT).

Daneben werden noch die KAPROS-Systemroutinen KSCC, KSGET und KSDLT verwendet.

4.2 KAPROS-2 und KAPROS-3 Versionen

Eine mit dem SIEMENS FORTRAN-77 Compiler übersetzte und gelinkte Version (ohne gelöste Referenzen auf externe Unterprogramme) steht in der KAPROS-2-Bibliothek KAPROS.F77.KSLIB unter dem Namen FILLC bereit und kann mit “INCLUDE KSBIB(FILLC,WQRG)” in einem LINK-Step an KAPROS-Module angeschlossen werden (s. Beispiel in “Anhang C. Anwendungsbeispiel für FILLC” auf Seite 35).

Für KAPROS-3 stehen entsprechend die SIEMENS FORTRAN-77 Version in KAPROS.KS3.LOAD und die IBM VS-FORTRAN Version in KAPROS.KS3V.LOAD als Member FILLC bereit. Hier entfällt der “INCLUDE”-Schritt, da sowohl FILLC als auch WQRG in der jeweils aktuellen Version automatisch zugelinkt werden.

4.3 Programminterne Dokumentation

Das Quellprogramm von FILLC enthält eine umfangreiche interne Dokumentation in Form von Kommentaren. Außerdem wurde bei der Wahl von Variablennamen versucht, möglichst mnemotechnisch vernünftige Zeichenketten zu wählen (leider läßt FORTRAN-77 nur maximal 6 Zeichen zu). Die Autoren hoffen, daß das Quellprogramm von FILLC von jedem Benutzer leicht verstanden und notfalls auch abgeändert werden kann. Es erscheint jedoch wünschenswert, daß stets **eine** Referenzversion für die allgemeine Benutzung verfügbar ist.

4.4 FILLC COMMON-Blöcke

Die vier COMMON-Blöcke

```
/FILLCO/ MXSKA, IPSTOT, IPSTR, IPSN2N, IPSCAP, IPSFIS, NTRANS
```

```
/FILLCX/ SIGTYP(MAXSKA)
```

```
/FILLCY/ TYPEXT(MAXSKA), C1, C2, C3, C4, C5, C6, C7, C8, C9, C10
```

```
/FILLCZ/ R1, R2, R3, R4, R5, II1, II2, II3, II4, II5
```

stehen dem rufenden Programm zur Verfügung. Sie werden in Tabelle 7 auf Seite 24 erläutert. In dem Feld SIGTYP (CHARACTER*8) werden die ersten 8 Zeichen des Typnamens der Skalarmtypen, in Feld TYPEXT (CHARACTER*8) die letzten 8 Zeichen, jeweils in der im Feld C verwendeten Reihenfolge zurückgegeben. Die Variablen IP... geben die Zeile in C an, in der der jeweilige Wirkungsquerschnitts-Typ gespeichert ist. MAXSKA = 100 ist die maximal erlaubte Anzahl zu speichernder Skalarmtypen und wird durch ein PARAMETER-Statement festgelegt. Der aktuelle Wert von MAXSKA wird als Variable MXSKA im COMMON-Block FILLCO zurückgegeben. Falls in dem FILLC rufenden Programm NTRANS \neq 0 gesetzt wird, so erfolgt in FILLC ein Aufruf des Unterprogramms FILTRA (s. "Ausführen der Transportkorrektur" auf Seite 18). Die Variablen C1 bis C5 sind CHARACTER*16 Variable, C6 bis C10 CHARACTER*8 Variable und dienen derzeit als Platzhalter für spätere Erweiterungen. Das gleiche gilt für die REAL (einfach genau) Variablen R1 bis R5.

Von den INTEGER Variablen II1 bis II5 werden bisher nur II1 und II2 verwendet. II1 gibt die Startposition im Feld SIGTYP an, ab der die Namen von Skalarmtypen abgeändert wurde (s. "Ändern der Reihenfolge der Auswertetypen" auf Seite 17). II2 erhält den Wert 1, falls externe FILLC-Eingabe, d.h. einer der beiden Datenblöcke DBN = FILLC δ XS δ TYPE δ E,

IND = 1 (für Typnamen mit 16 Bytes, das letzte E des Namens steht für "Extended") oder DBN = FILLC6XS6TYPES, IND = 1, in der KAPROS-Lifeline gefunden wurde.

4.5 Änderungen gegenüber der bisherigen Version 3.0

Gegenüber den älteren Versionen /6/ und /7/ wurden eine Reihe von Änderungen und Erweiterungen vorgenommen. So ist es möglich, die Reihenfolge der Einspeicherung der Skalarmtypen zu beeinflussen; die Verarbeitung von 16 Bytes langen Wirkungsquerschnitt-Typnamen ist integriert; die Fehlerdiagnostik wurde verbessert und erweitert; während der normalen Verarbeitung wird Diagnostik-Ausdruck erzeugt; neue COMMON Blöcke (FILLCY; FILLCZ) wurden eingeführt; die maximale Anzahl für Skalarmtypen wurde auf 100 erhöht; für die Transportkorrektur wurden neue Optionen implementiert.

Für Rechnungen mit 208 Energiegruppen wurde beim Speichern der Streumatrix sowie bei der Ausführung der Transportkorrektur eine Spezialbehandlung eingeführt (s. "Anhang B. Programmliste für das Unterprogramm FILTRA" auf Seite 31).

FILLC führt eine Vielzahl von Konsistenztests mit seinen Aufrufparametern durch. Bei auftretenden Fehlern und/oder Inkonsistenzen werden in der neuen Version grundsätzlich Meldungen ausgegeben und Fehlercodes gesetzt. Außerdem werden standardmäßig Meldungen während des normalen Ablaufs ausgegeben. Daher kann der Benutzer leicht verfolgen, was in der aktuellen Anwendung eventuell zu Schwierigkeiten führte.

Benutzern von S_N -Codes wird dringend geraten, das Verarbeitungsprotokoll von FILLC genau zu studieren. Die von FILLC erzeugte Druckausgabe erscheint in der Ausgabeliste etwas abgehoben von der Ausgabe des rufenden Transport-Programms. Außerdem erfolgt auch eine Nachricht über den Aufruf von und die Rückkehr aus FILLC im KAPROS-Protokoll, wodurch die Fehlermeldungen und Warnungen des KAPROS-Kerns, die sich auf FILLC beziehen, klar abgegrenzt werden.

Variable	Inhalt
COMMON-Block FILLCO	
MXSKA	Wert des aktuellen Dimensions-PARAMETERs MAXSKA
IPSTOT	Relative Position des Wertes von STOT
IPSTR	Relative Position des Wertes von STR
IPSN2N	Relative Position des Wertes von SN2N
IPSCAP	Relative Position des Wertes von SCAPT
IPSFIS	Relative Position des Wertes von SFISS
NTRANS	Transportkorrektur ausführen oder nicht
COMMON-Block FILLCX	
SIGTYP (MAXSKA)	Die ersten 8 Bytes des Typnamens in der verwendeten Reihenfolge (bzw. gegebenenfalls Leerzeichen)
COMMON-Block FILLCY	
TYPEXT (MAXSKA)	Die letzten 8 Bytes des Typnamens in der verwendeten Reihenfolge bzw. 8 b (Blanks) bei Verwendung von 8-Byte-Typnamen
C1 .. C5	Platzhalter des Typs CHARACTER*16 für künftige Erweiterungen
C6 .. C10	Platzhalter des Typs CHARACTER*8 für künftige Erweiterungen
COMMON-Block FILLCZ	
R1 .. R5	Platzhalter des Typs REAL (einfach genau) für künftige Erweiterungen
II1	Startzeile in Feld C, ab der die Reihenfolge der Skalartypen geändert wurde (s. "Ändern der Reihenfolge der Auswertetypen" auf Seite 17)
II2	= 1, falls ein Datenblock zur Änderung der Reihenfolge der Skalartypen gefunden wurde (s. "Ändern der Reihenfolge der Auswertetypen" auf Seite 17); = 0 sonst
II3 .. II5	Platzhalter des Typs INTEGER für künftige Erweiterungen

Tabelle 7. Belegung der COMMON-Blöcke in FILLC

4.6 *Änderungen für bisherige Benutzer*

Für bisherige Anwender ist es wichtig, vor allem bei Einbindung des übersetzten FILLC-Codes aus der KAPROS-Bibliothek, die Länge der COMMON-Blöcke anzupassen (speziell diejenige des bisher schon vorhandenen /FILLCX/-Blockes). Dieser Block hat nun eine Länge von MAXSKA*8 BYTES (MAXSKA=100 anstelle von 30 in der bisherigen Version). Außerdem kann es in einer bestehenden Anwendung zu Fehlern kommen, wenn dort der Datenblockname FILLCbXSbTYPESbE bereits verwendet wird, da FILLC beim Vorhandensein eines so benannten Datenblocks spezielle Arbeitsschritte ausführt (siehe "Ändern der Reihenfolge der Auswertetypen" auf Seite 17).

Änderungen bei den Ergebnissen kann es bei Verwendung anisotroper Streu-Wirkungsquerschnitte bzw. bei Aufwärtsstreuung geben (wegen der in Version 3.4 evtl. durchgeführten Transportkorrektur) sowie bei SIGMN-Blöcken in 208-Energiegruppen-Struktur (Sonderbehandlung in Version 3.4).

5. Hinweise auf andere Wirkungsquerschnitts-Programme

Es wird eine (sicher unvollständige) Liste von verwandten Programmen gegeben, die häufig im Zusammenhang mit Transportrechnungen verwendet werden bzw. von grundsätzlichem Interesse sind.

(i) GRUCAL /4/

Der Code GRUCAL dient innerhalb von KAPROS als Standardprogramm zur Erstellung von (sogenannten effektiven, d.h., Resonanz-selbstabgeschirmten) makroskopischen und mikroskopischen Wirkungsquerschnitten für Mischungen und deren Einzelbestandteile. Die neueste Dokumentation und die Eingabevorschriften können über das KAPROS INFO-System erhalten werden.

(ii) SIGMUT /16/

Mit Hilfe von SIGMUT können SIGMN-Blöcke manipuliert werden; z.B. können die Typnamen geändert werden oder neue Typen (z.B. Daten für verzögerte Neutronen) eingefügt werden. Vor einer Umprogrammierung von FILLC sollte stets geprüft werden, ob nicht SIGMUT die gleiche Aufgabe schneller und "durchsichtiger" erfüllen kann.

(iii) FLUMMI, ELAS, SIGMOD /12/

Diese Programme von H.W.Wiese bieten die Möglichkeit, SIGMN-Blöcke so abzuändern, daß die Wirkungsquerschnitte (unter den üblicherweise verwendeten Typennamen) eine bessere Behandlung der anisotropen Streuung erfahren. Leider ist das Verfahren auf Abwärtsstreuprobleme beschränkt und nur als stand-alone Version (d.h., nicht als KAPROS-Modul) verfügbar. Das Programm FLUMMI erzeugt eine "ELAS"-Datei, während ELAS und SIGMOD die Daten im SIGMN-Block modifizieren.

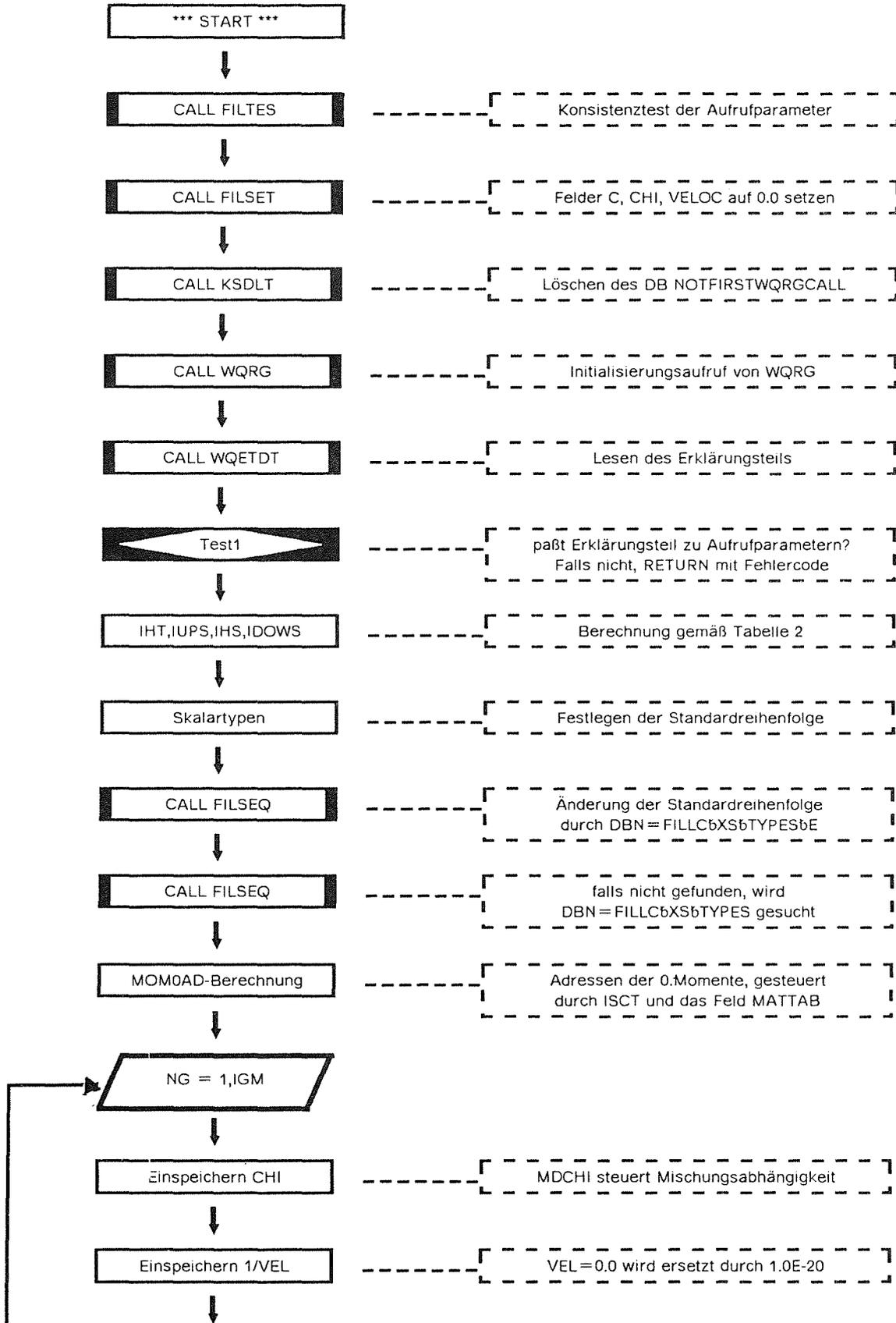
(iv) KARBUS /17/

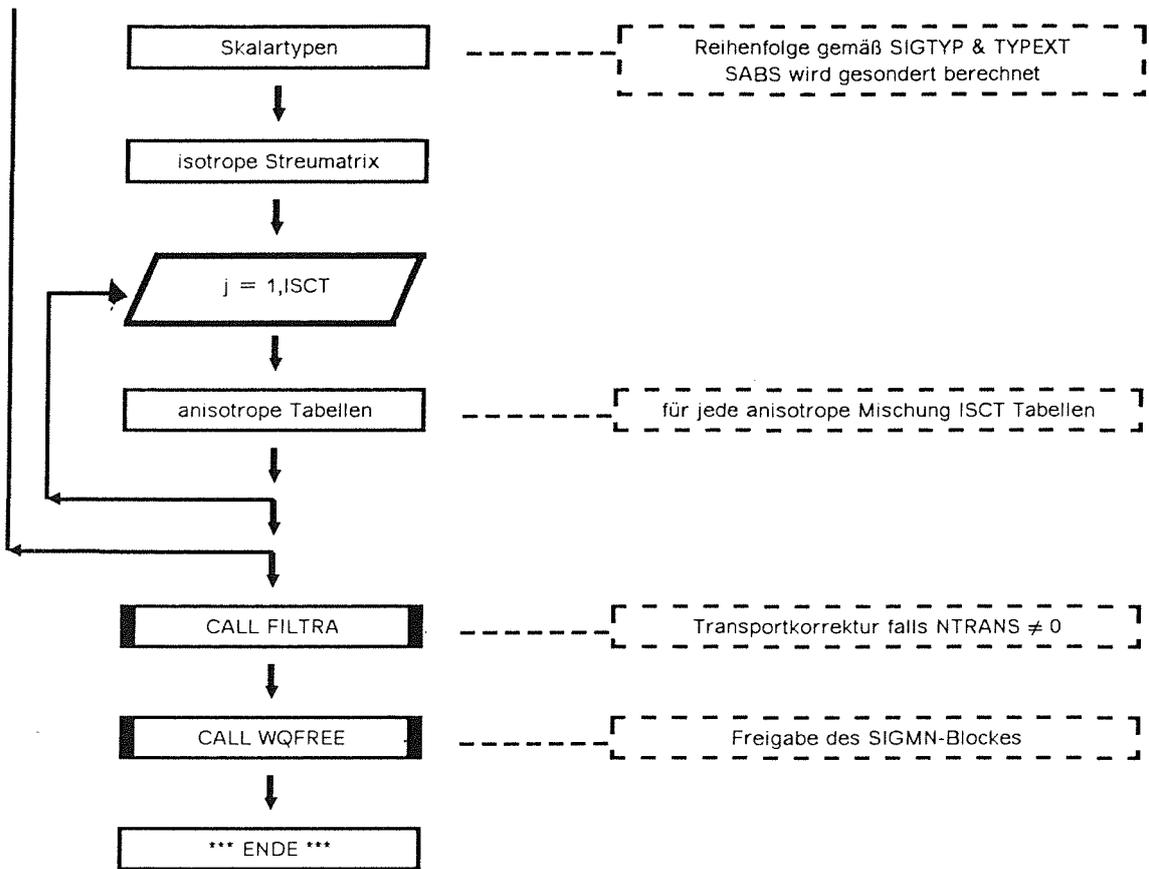
Die KAPROS-Prozedur KARBUS ermöglicht die Erzeugung von SIGMN-Blöcken mit Hilfe von Zellrechnungen. Sie wurde zur Behandlung von Aufwärts-Streuproblemen (Leichtwasser- und sogenannte Fortgeschrittene Druckwasser-Reaktoren) entwickelt und benutzt SIGMUT zur Erstellung des SIGMN-Blocks.

(v) TRANSX /11/

Die KAPROS-Prozedur TRANSX ist ein Hilfsprogramm zum Umgang mit SIGMN-Blöcken und zur Bereitstellung von makroskopischen Wirkungsquerschnitten für stand-alone Programme, die für Transport- und Diffusionsrechnungen im KfK eingesetzt werden.

Anhang A. Programmablaufplan für FILLC





Anhang B. Programmliste für das Unterprogramm FILTRA

```

C
C
C ***** THIS SUBROUTINE PERFORMS TRANSPORT CORRECTIONS *****
C
C      SUBROUTINE FILTRA (C,IHM,IGM,NUMTAB,MATTAB,MOMOAD,NUMMAT,
C      +                 NFO,IHS,IHT,IUPS,KONTRO,A,ISCT)
C
C EXPLANATION OF MAIN VARIABLES:
C
C C(IHM,IGM,NUMTAB): ARRAY TO CONTAIN CROSS-SECTIONS IN THE
C                   SEQUENCE DEFINED IN FILLC DOCUMENTATION
C IHM                : NUMBER OF ROWS OF C (IHM.GE.IHS+IDOWS)
C IGM                : NUMBER OF ENERGY GROUPS
C NUMTAB             : NUMBER OF CROSS-SECTION TABLES TO BE
C                   STORED; =NUMMAT IF ISCT=0 (ISOTROPY)
C MATTAB(NUMMAT)    : MIXTURE NUMBERS IN ANY ORDER;
C                   A NEGATIVE NUMBER INDICATES ANISOTROPY
C MOMOAD(NUMMAT)    : ADDRESSES OF ZEROth ORDER MOMENT OF SCAT-
C                   TERING CROSS-SECTION IF ISCT>0
C NUMMAT             : NUMBER OF DIFFERENT MIXTURES TO BE
C                   HANDLED
C NFO                : STANDARD OUTPUT UNIT (=6 NORMALLY)
C ISCT               : ANISOTROPIC SCATTERING ORDER(.LE.ISCTMX)
C IHS                : ROW NUMBER OF SELFSCATTERING CROSS-
C                   SECTION IN C (IHS.LT.IHM)
C IHT                : ROW NUMBER OF TOTAL CROSS-SECTION IN C
C                   (IHT.GE.5; IHT.LT.IHS)
C IUPS               : UPSCATTERING WIDTH (0.LE.IUPS.LE.IGM)
C KONTRO             : ERROR INDICATOR
C =113: GENERAL ERROR IN WQSKAL
C =114: SREM NOT FOUND
C =115: NEITHER STRTR NOR STR FOUND
C =116: TRANSPORT CROSSSECTION FOR AN ENERGY GROUP NOT FOUND
C =117: SREM FOR AN ENERGY GROUP NOT FOUND
C =118: TRANSPORT CORRECTION OF TYPE SMTOT(G-G) = STOT - SREM SHOULD
C       BE APPLIED BUT STOT IS NOT CONTAINED IN THE SIGMN-BLOCK
C A : ORIENTATION ARRAY FOR WORK WITH WQRG C
C
C      PARAMETER (ISCTMX=6, MAXSKA=100)
C
C      REAL C(IHM,IGM,NUMTAB),A(1)
C      INTEGER MATTAB(NUMMAT),MOMOAD(NUMMAT)
C
C      CHARACTER*8  SIGTYP(MAXSKA),TYPEXT(MAXSKA),C6,C7,C8,C9,C10
C      CHARACTER*16 CSTR,CSTRTR,CSREM,CSTOT,C1,C2,C3,C4,C5,CGR208(5)
C
C      COMMON /FILLCO/ MXSKA,IPSTOT,IPSTR,IPSN2N,IPSCAP,IPSFIS,NTRANS
C      COMMON /FILLCX/ SIGTYP
C      COMMON /FILLCY/ TYPEXT,C1,C2,C3,C4,C5,C6,C7,C8,C9,C10
C
C      DATA CSTR      /'STR
C      +              /'SREM
C
C

```

```

+   CSTRTR  /'STRTR           '/,
+   CSTOT   /'STOT           '/,
+   CGR208  /'STRBO         ',
+           'DIFK           ',
+           'STRTR         ',
+           'STR           ',
+           'DIFKO         '/,
+   IEINS   /1             /

C
  KONTRA = 0
  ITYP   = 0

C
  IF (MOMOAD(NUMMAT).NE.NUMMAT.OR.IUPS.GT.0) THEN
    WRITE (NFO,4500)
C
    GO TO 700
  END IF
  CALL WQSKAL (IEINS,CSREM,1,1,LPCR,*600,*50)
  IF (NTRANS.EQ.-1) THEN
    CALL WQSKAL(IEINS,CSTOT,1,1,LPCR,*600,*800)
    KONTRA=2
    GO TO 100
  END IF
  GO TO 100
50 CALL FILKON(NFO,KONTRO,114)
  WRITE (NFO,1000)
100 CALL WQSKAL (IEINS,CSTRTR,1,1,LPCR,*600,*150)
  GOTO 300
150 CALL WQSKAL (IEINS,CSTR,1,1,LPCR,*600,*200)
  IF (NTRANS.NE.-1) KONTRA = 1
  GOTO 250
200 CALL FILKON(NFO,KONTRO,115)
  WRITE (NFO,2000)
  GOTO 700
250 WRITE (NFO,3000)
300 IF (KONTRO.EQ.114) GOTO 700
  IF (KONTRA.EQ.0) SIGTYP(IHT) = CSTRTR(1:8)
  IF (KONTRA.EQ.1) SIGTYP(IHT) = CSTR(1:8)
  IF (KONTRA.EQ.2) SIGTYP(IHT) = CSTOT(1:8)
  WRITE (NFO,3250)
  DO 330 J=1,IHT,10
    JE = MIN(J + 9,IHT)
    WRITE (NFO,3500)(SIGTYP(K),K=J,JE)
330  WRITE (NFO,3500)(TYPEXT(K),K=J,JE)
  DO 450 NG=1,IGM
CÜÜÜ  IF (NG.GT.(IGM-IUPS)) GO TO 450
  IF (KONTRA.EQ.0) CALL WQSKAL (IEINS,CSTRTR,1,NG,LPCR,*600,*550)
  IF (KONTRA.EQ.1) CALL WQSKAL (IEINS,CSTR,1,NG,LPCR,*600,*550)
  IF (KONTRA.EQ.2) CALL WQSKAL (IEINS,CSTOT,1,NG,LPCR,*600,*800)
C XXXXXXXX SPECIAL TREATMENT FOR 208 GROUP STRUCTURE  START *****
  IF (IGM.EQ.208) THEN
    IF (ISCT.EQ.1) GO TO 345
    DO 340 I = 1,5
      CALL WQSKAL (IEINS,CGR208(I),1,NG,LPCR,*600,*335)
      ITYP = I
      IF (NG.EQ.1) WRITE (NFO,7500) CGR208(ITYP)
      GO TO 345
335  IF (I.EQ.3) WRITE (NFO,5500)

```

```

340     CONTINUE
        WRITE (NFO,6500)
        KONTRO = 9
        GO TO 700
    END IF
345     CONTINUE
C XXXXXXXX SPECIAL TREATMENT FOR 208 GROUP STRUCTURE   END   *****
    DO 350 N=1,NUMMAT
        NM = IABS(MATTAB(N))
        NA = MOMOAD(N)
        C(IHT,NG,NA) = A(LPCR+NM)
        IF (ITYP.EQ.2.OR.ITYP.EQ.5)
+           C(IHT,NG,NA) = 1./(3.*C(IHT,NG,NA))
350     CONTINUE
        CALL WQSKAL (IEINS,CSREM,1,NG,LPCR2,*600,*500)
        IF (NTRANS .EQ. -1)CALL WQSKAL(IEINS,CSTOT,1,NG,LPCR3,*800,*800)
        DO 400 N=1,NUMMAT
            NM = IABS(MATTAB(N))
            NA = MOMOAD(N)
            IF (NTRANS .EQ. -1)
1              C(IHS,NG,NA) = A(LPCR3 + NM)-A(LPCR2 + NM)
400         IF (NTRANS .EQ. 1)
1              C(IHS,NG,NA) = C(IHT,NG,NA)-A(LPCR2+NM)
450     CONTINUE
        WRITE (NFO,4000)
        GOTO 700
500     CALL FILKON(NFO,KONTRO,117)
        WRITE (NFO,5000) NG
        GOTO 700
550     CALL FILKON(NFO,KONTRO,116)
        WRITE (NFO,6000) NG
        GOTO 700
600     CALL FILKON(NFO,KONTRO,113)
        WRITE (NFO,7000)
        GO TO 700
800     CALL FILKON(NFO,KONTRO,118)
        WRITE (NFO,8000)
700     RETURN
1000    FORMAT ('0***** TYPE SREM NOT FOUND')
2000    FORMAT ('0***** NEITHER TYPE STRTR NOR TYPE STR FOUND')
3000    FORMAT ('0***** TRANSPORT CORRECTION USES TYPE STR ',
+             'BECAUSE TYPE STRTR IS NOT FOUND')
3250    FORMAT ('0***** STORAGE SEQUENCE CHANGED BY FILTRA TO'//)
3500    FORMAT (' *****',10(A8,2X))
4000    FORMAT ('0***** TRANSPORT CORRECTION APPLIED TO CROSS SECTION ',
+             'MATRIX C IN SUBROUTINE FILTRA WITHOUT ERROR')
4500    FORMAT ('0***** TRANSPORT CORRECTION USED FOR ANISOTROPIC ',
+             'AND/OR UPSCATTERING PROBLEM, CHECK RESULTS CAREFULLY ')
5000    FORMAT ('0***** WQSKAL ERROR FOR SREM IN ENERGY GROUP',I10)
5500    FORMAT ('0***** NO APPROPRIATE TRANSPORT CROSS SECTION ',
+             'SUITABLE FOR TRANSPORT CALCULATIONS '/
+             ' IN 208 ENERGY GROUPS HAS BEEN FOUND IN SIGMN-BLOCK.')
6000    FORMAT ('0***** WQSKAL ERROR FOR TRANSPORT XS IN ENERGY ',
+             'GROUP',I10)
6500    FORMAT ('0***** NOT ANY TRANSPORT CROSS SECTION SUITABLE FOR ',
+             '208 ENERGY GROUPS FOUND IN THE SIGMN-BLOCK.'/
+             'OTHE CALCULATION IS STOPPED!')

```

```
7000 FORMAT ('0***** ERROR IN WQSKAL ')
7500 FORMAT ('0***** 208 ENERGY GROUP CALCULATIONS ARE PERFORMED ',
+          'USING TYPE ',A16,' AS TOTAL CROSS SECTION.')
8000 FORMAT ('0***** TRANSPORT CORRECTION OF TYPE ',
+          'SMTOT(G-G) = STOT - SREM SHOULD BE APPLIED BUT ',
+          'STOT IS NOT CONTAINED IN THE SIGMN-BLOCK')
END
```

Anhang C. Anwendungsbeispiel für FILLC

Der folgende Job zeigt wie FILLC verwendet wird, um einen SIGMN-Block zu lesen und anschließend im Kartenformat auf eine Datei auszugeben. Grundkenntnisse in der Benutzung von KAPROS werden dabei vorausgesetzt. Verwendet wird KAPROS-2 und der SIEMENS-FORTRAN77-Compiler.

Man beachte insbesondere die “*INCLUDE”-Anweisungen im KAPROS-LINK-Step.

Zunächst wird der Modul GRUCAL aufgerufen, um aus der KAPROS-Dateneingabe (DBN=MISCH und DBN=GRUCAL) einen SIGMN-Block mit Namen DBN=SIGMN in der Lifeline zu erzeugen (das wird durch die Anweisung “*GO SM=GRUCAL” bewirkt).

Anschließend wird der gerade vorher neu erzeugte Modul WQPUN aufgerufen, in dem FILLC den SIGMN-Block liest und in das Feld C abspeichert. Das Unterprogramm FILPUN gibt dann das Feld C in einer bestimmten Reihenfolge und einem bestimmten Format aus auf die Einheit FT22F001.

Das abgedruckte Beispiel steht in der Datei INR986.FILLC.DATA als Member WQPUNCH.

```
// ----> Job-Karte <----
//*****
//*   DIESER JOB GIBT   WIRKUNGSQUERSCHNITTE EINES SIGMN BLOCKS      *
//*   IM DIAMANT2 CARD IMAGE FORMAT AUS   AUF DIE DATEI SCRATCH.CNTL *
//*                                           06.06.86                *
//*****
//VORSTEP EXEC F7C,
//      PARM.C='LANGLVL(77),NOMAP'
//C.SYSLIN DD UNIT=SYSSQ,DCB=LRECL=3200,
// SPACE=(TRK,(200,10,5)),
// DISP=(MOD,PASS),DSN=&&TEMP1(FILPUN)
//C.SYSIN DD *
      SUBROUTINE MAIN
C
      PARAMETER (MAXTAB=10,MAXGRP=26,
+              MAXSKA=30,ISCTMX=6,MAXROW=MAXSKA+MAXGRP)
C
      DIMENSION C(MAXROW*MAXGRP*MAXTAB),CHI(MAXGRP,1),
+              MOMOAD(MAXTAB),MATTAB(MAXTAB),VELOC(MAXGRP)
C
      COMMON /FILLCO/ MXSKA,IPSTOT,IPSTR,IPSN2N,IPSCAP,IPSFIS,NTRANS
      COMMON /FILLCX/ SIGTYP
      CHARACTER*8 SIGTYP(MAXSKA)
C
      CALL KSINIT(TC,DT,N1,N2,N3)
C
C   DIE FOLGENDEN VARIABLEN MUESSEN DEM SIGMN-BLOCK ANGEPASST WERDEN:
C   NUMMAT : MAXIMAL 10  ANZAHL DER MISCHUNGEN
C   NUMTAB : MAXIMAL 10  ANZAHL DER TABELLEN (ISCT BEACHTEN!)
C   IGM    : MAXIMAL 26  GRUPPENZAHL
C   ISCT   : MAXIMAL 6   ANISOTROPIE GRAD
C
```

```

C   DIE AUSGABE ERFOLGT IM CARD IMAGE FORMAT AUF EINHEIT FT22F001
C
  NUMMAT=1
  NUMTAB=1
  IGM=26
  ISCT=0
  IF (NUMMAT.GT.MAXTAB) THEN
    WRITE (6,(''OMISCHUNGSANZAHL ZU GROSS''))
    GOTO 500
  END IF
  IF (NUMTAB.GT.MAXTAB) THEN
    WRITE (6,(''OTABELLENANZAHL ZU GROSS''))
    GOTO 500
  END IF
  IF (IGM.GT.MAXGRP) THEN
    WRITE (6,(''OGRUPPENZAHL ZU GROSS''))
    GOTO 500
  END IF
  IF (ISCT.GT.ISCTMX) THEN
    WRITE (6,(''OANISOTROPIE-INDEX ZU GROSS''))
    GOTO 500
  END IF
  MATTAB(1)=1
  NTRANS = 1
  NFO=6
  NPROT=42
  IUPS=0
  MDCHI=0
  IHT=7
  IHS=IHT+1
  IHM=IGM+7
  IDOWS=IGM-1
  IGP=IGM+1
  CALL FILLC(C,IHM,IGM,NUMTAB,CHI,IGP,VELOC,MATTAB,MOMOAD,
+           NUMMAT,NFO,NPROT,ISCT,IHS,IHT,IUPS,IDOWS,MDCHI,KONTRO)
  CALL FILPUN(C,IHM,IGM,NUMTAB,CHI,IGP,VELOC,MATTAB,MOMOAD,
+           NUMMAT,NFO,NPROT,ISCT,IHS,IHT,IUPS,IDOWS,MDCHI,KONTRO)
500 CONTINUE
  RETURN
  END

C
C   SUBROUTINE FILPUN - OPEN END SUBROUTINE FOR XS HANDLING
C
  SUBROUTINE FILPUN (C,IHM,IGM,NUMTAB,CHI,IGP,VELOC,MATTAB,MOMOAD,
+                   NUMMAT,NFO,NPROT,ISCT,IHS,IHT,IUPS,IDOWS,MDCHI,
+                   KONTRO,A)
C
  DIMENSION C(IHM,IGM,NUMTAB),CHI(IGP,NUMMAT),VELOC(IGM),
+           MATTAB(NUMMAT),MOMOAD(NUMMAT)
C
  PARAMETER (ISCTMX=6, MAXSKA=30)
  CHARACTER*8 SIGTYP(MAXSKA),STR,STRTR,SREM
C
  COMMON /FILLCO/ MXSKA,IPSTOT,IPSTR,IPSN2N,IPSCAP,IPSFIS,NTRANS
  COMMON /FILLCX/ SIGTYP
C
C CAPABILITY: PUNCH OUT CHI AND MATRIX C IN CARD IMAGE FORMAT

```

C OUTPUT ON LOGICAL UNIT 22 (DD-CARD REQUIRED)

C

CALL KSDD(1,22,-1,DUMMY,IQ)

IF (IQ.LE.0) GOTO 10

KONTRO = 101

CALL KSCC(1,IQ)

IQ = 0

GOTO 200

10 WRITE (22,1000) (SIGTYP(I),I=1,IHT)

WRITE (22,1100)

IF (MDCHI.EQ.0) ICHI = 1

IF (MDCHI.NE.0) ICHI = NUMMAT

DO 50 J = 1,ICHI

WRITE (22,1200) (CHI(I,J),I=1,IGM)

50 CONTINUE

DO 100 J=1,NUMTAB

WRITE (22,1400) J

WRITE (22,1200) ((C(K,I,J),K=1,IHM),I=1,IGM)

100 CONTINUE

CALL KSDD(0,22,-1,DUMMY,IQ)

IF (IQ.LE.0) GOTO 200

KONTRO = 101

CALL KSCC(1,IQ)

IQ = 0

200 RETURN

C FORMAT DECLARATIONS

1000 FORMAT (' ')

1100 FORMAT (' ')

1200 FORMAT(5(1PD13.5))

1300 FORMAT (' ')

1400 FORMAT (' ')

C1000 FORMAT ('*\$ SEQUENCE OF SCALAR TYPES IS AS FOLLOWS: '/

C + ('*\$ ',7A9))

C1100 FORMAT ('*\$ FISSION SPECTRA ')

C1200 FORMAT(5(1PD13.5))

C1300 FORMAT ('*\$ X-SECTIONS FOR ENERGY GROUP ',I3)

C1400 FORMAT ('*\$ X-SECTIONS FOR XS TABLE NR. ',I3)

END

/*

//TESTFILL EXEC KSCLG7

//K.LOAD DD DSN=&&TEMP1,DISP=(OLD,PASS),VOL=REF=*.VORST.C.SYSLIN

//K.FT22F001 DD DISP=SHR,DSN=INR986.SCRATCH.CNTL

//K.FT01F001 DD DISP=SHR,DSN=INR909.KSDA.KFKINR

//K.FT10F001 DD DISP=SHR,DSN=INR909.GRSTAB

//K.FT11F001 DD DISP=SHR,DSN=INR909.F26

//K.SYSIN DD *

*LINK MAP,LIST

INCLUDE LOAD(FILPUN)

INCLUDE KSBIB(WQRG,FILLC)

ENTRY MAIN

NAME WQPUN

\$\$

*\$

*KSIOX DBN=MISCH,IND=1,TYP=CARD,PMN=KETT

1

*\$ TEILCHENZAHLDICHTEN FUER GODIVA

```

2      '      '      '      '      '      '
'U 235      '      300.0  0.045447E+00
'U 238      '      300.0  0.002561E+00
*$$
*KSI0X DBN=GRUCAL,IND=1,TYP=CARD,PMN=PRDUM
'GRUCAL      ' 'KFKINR      '      '      '      '      ' 10 11
'MISCH      ' 'DATBLOCK'
'TYP      ' 'ALLEIN      ' 12
'CHI      ' 'NUSF      ' 'SCAPT      ' 'SFISS      ' 'SBE      ' 'SREM      '
'STR      ' 'STRTR      ' '1/V      ' 'STOT      ' 'SN2N      ' 'SMTOT      '
'GRUCEND      '
*$$
*GO SM=GRUCAL
*GO SM=WQPUN
/*
//

```

Literaturverzeichnis

- /1/ V.Brandl, unveröffentlichter Bericht, 1980
- /2/ V.Brandl, unveröffentlichter Bericht, 1981
- /3/ K.Küfner, J.Burkhard, R.Heger, "An Updated FORTRAN-77 Version of the 2-d Static Neutron Transport Code DIAMANT2 for Regular Triangular Geometry", KfK 4133, 1987
- /4/ D.Woll, "GRUCAL - Ein Programmsystem zur Berechnung makroskopischer Gruppenkonstanten ", KfK 2108, 1975
- /5/ G.Buckel, W.Höbel, "Das Karlsruher Programmsystem KAPROS - Teil I: Einführung für Benutzer und Programmierer", KfK 2253, 1976
- /6/ K.Küfner, unveröffentlichter Bericht, 1980
- /7/ K.Küfner, unveröffentlichter Bericht, 1986
- /8/ K.Kobayashi, "TP1 - Computer Program for the Calculation of Reactivity and Kinetic Parameters by One-Dimensional Neutron Transport Perturbation Theory", KfK 2738, 1979
- /9/ K.Kobayashi, "TP2 - Computer Program for the Calculation of Reactivity and Kinetic Parameters by Two-Dimensional Neutron Transport Perturbation Theory", KfK 2787, 1979
- /10/ K.Kobayashi, G.Buckel, K.Küfner, "TPTRIA - A Computer Program for the Reactivity and Kinetic Parameters for Two-Dimensional Triangular Geometry by Transport Perturbation Theory", KfK 4116, 1986
- /11/ G.Buckel, W.Götzmann, E.Kiefhaber, K.Küfner, "TRANSX - Ein Transformationsprogramm zur Bereitstellung makroskopischer Gruppenkonstanten aus SIGMN-Blöcken für Neutronen-Transport- und Diffusionsprogramme", KfK 4977, 1991
- /12/ H.W.Wiese, unveröffentlichter Bericht, 1973
- /13/ R.D.O'Dell , F.W.Brinkley, Jr., D.R.Martin, "User's Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport", LA-9184M, 1982
- /14/ R.E.Alcouffe, F.W.Brinkley, D.R.Marr, R.D.O'Dell, "User's Guide for TWODANT: A Code Package for Two-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport", LA-100494M, 1984
- /15/ W.Höbel, unveröffentlichter Bericht, 1981
- /16/ H.Bachmann, H.Gabaieff, D.Woll, unveröffentlichter Bericht, 1979
- /17/ C.Broeders, "KAPROS-Prozedur KARBUS", persönliche Mitteilung

Volker Brandl stellte dankenswerterweise die Urversion des Unterprogramms FILTRA zur Verfügung. Für Unterstützung und fruchtbare Diskussionen danken wir R.Böhme, C.Broeders, I.Broeders, G.Henneges, W.Höbel, F.Kappler, W.Maschek, E.Stein, E.Wiegner und H.W.Wiese.

- b 2, 17
- A**
- Ablaufdiagramm 21, 29
 Abwärtsstreuung 9, 10, 12
 Anisotropie 3, 4, 7, 9, 12, 14
 Anwendungsbeispiele 35
 Aufrufparameter 7
 siehe auch Ausgabeparameter, Eingabeparameter
 Aufwärtsstreuung 5, 9, 10, 12
 Ausgabeparameter 9
 Auswertetyp 2
- B**
- Beispiele 35
 Bibliotheken 2, 21
- C**
- C 2, 11, 12, 17
 CHI 4, 9
 COMMON-Blöcke 5, 9, 10, 11, 12, 17, 18, 21, 22, 23
- D**
- Dateien 2, 21
 Datenblocknamen 1, 7, 17, 22, 29
 Dimensionierungsgrößen 7, 9, 14
- E**
- Eingabeparameter 7, 9
 Einschränkungen 4, 9, 10, 11
- F**
- Fehlerbedingungen 11, 14, 15
 Fehlercodes 15, 16
 Fehlermeldungen von FILTRA 16
 Feld C 2, 11, 12, 17
 FILLC 35
 FILLCbXSbTYPES 17, 22, 29
 FILLCbXSbTYPESbE 17, 22, 29
 FILLCO, FILLCX, FILLCZ 22
 FILLCO, FILLCX 5, 9, 10, 11, 12, 18, 21, 23
 FILLCZ 17
 FILTRA 9, 18, 21, 22, 27, 31
 Flußmomente 4
- G**
- γ -Quanten 5
- I**
- IDOWS 9, 10, 12
 IGM 7
 IGP 7, 9
 IHM 7
 IHS 9, 10
 IHT 9, 10, 11
 IPSCAP 12, 22, 23
 IPSFIS 12, 22, 23
 IPSN2N 12, 22, 23
 IPSTOT 12, 22, 23
 IPSTR 12, 22, 23
 ISCT 3, 5, 7, 9, 14, 18
 IUPS 9, 10, 12, 18
- K**
- KAPROS 2, 21
 KAPROS-2 2, 21
 KAPROS-3 2, 21
 Konsistenzprüfung 12, 13, 14
 KONTRO 9, 10, 11, 14, 15, 16
- L**
- Legendre-Entwicklung 3, 4, 9
 Linken 2, 21
 Los Alamos Format 7, 11

M

Matrix C 2, 11, 12, 17
MATTAB 7, 9, 14
MAXSKA 10, 11
MDCHI 9
Mischungsauswahl 9
Modifikationen 1, 23
Modulbibliotheken 2, 21
Momente der Streumatrix 14
MOMOAD 7, 14

N

Neutronen-Transportgleichung 3
Nomenklatur 2
NOTFIRSTWQRCALL 29
NTRANS 5, 9, 21, 22
NUMMAT 7, 14
NUMTAB 7, 14
NUSF 4, 12

P

Programmliste 31, 35

S

SABS 12, 13
SCAPT 12, 13
SFISS 12, 13
SIGTYP 10, 11, 12, 22, 23
skalärer Wirkungsquerschnitt 2
SMTOT 4, 5, 12, 13, 18
SN2N 11, 12
Spaltspektrum 4, 9
Speicherreihenfolge 10, 11, 12, 13, 17, 22
SREM 5, 12, 13, 18

STOT 4, 5, 12, 13, 18
STR 18
Streubreiten 9, 10, 12
Streumatrix 4, 5, 12, 13, 14
STRTR 5, 18

T

Transportkorrektur 5, 9, 16, 18, 21, 22, 31
TYPEXT 10, 11, 12, 22, 23

U

Unterprogramme 21

V

Vektortyp 2
Verfügbarkeit 2, 21

W

Warnungen von FILLC 16
Warnungs-Codes 16
Wirkungsquerschnittsrelationen 12, 13, 14
WQRC 21

Z

208-Gruppen-Struktur 1, 19, 23, 25, 31

(γ, n)-Prozesse 5