



ИНСТИТУТ ЯДЕРНОЙ ФИЗИКИ СО АН СССР

ИЯФ--90-96

А.Д. Букин, Н.А. Грозина, М.С. Дубровин,
И.Л. Кац, В.Н. Иванченко, В.А. Таюрский,
С.И. Эйдельман

**UNIMOD-2 — УНИВЕРСАЛЬНАЯ ПРОГРАММА
МОДЕЛИРОВАНИЯ ЭКСПЕРИМЕНТОВ
НА ВСТРЕЧНЫХ e^+e^- -ПУЧКАХ**

Часть 3. Руководство пользователя-программиста

ПРЕПРИНТ 90-96



НОВОСИБИРСК

ИНСТИТУТ ЯДЕРНОЙ ФИЗИКИ СО АН СССР

А.Д.Букин, Н.А.Грозина, М.С.Дубровин, И.Л.Кац,
В.Н.Иванченко, В.А.Таюрский, С.И.Эйдельман

UNIMOD2 - УНИВЕРСАЛЬНАЯ ПРОГРАММА МОДЕЛИРОВАНИЯ
ЭКСПЕРИМЕНТОВ НА ВСТРЕЧНЫХ e^+e^- -ПУЧКАХ
Часть 3. Руководство пользователя-программиста

ПРЕПРИНТ 90-96

НОВОСИБИРСК

АННОТАЦИЯ

Работа дополняет предыдущие описания новой версии программы моделирования UNIMOD2 для тех пользователей, которым для моделирования экспериментов с помощью UNIMOD2 потребуется писать свои подпрограммы.

ABSTRACT

This paper completes the previous descriptions of the new version of simulation program UNIMOD2 and is necessary for those users who would have to write their own subroutines for UNIMOD2 to simulate their experiments.

ОГЛАВЛЕНИЕ

1. Введение	5
2. Общие блоки программы	5
3. Правила написания типовых подпрограмм	24
3.1. Общие замечания	24
3.2. Генерация параметров первичных частиц	29
3.3. Программы моделирования процессов взаимодействия частиц с веществом	30
3.4. Распады нестабильных частиц	32
3.5. Программы моделирования геометрических блоков	33
3.6. Моделирование ионизационных потерь	35
3.7. Генерация угла многократного рассеяния	35
3.8. Вычисление магнитного поля	36
3.9. Моделирование проволочной камеры	36
3.10. Генерация амплитуды с проволочки	37
3.11. Генерация амплитуды в счётчике	38
3.12. Программы вычисления параметров события	39
3.13. Программы, анализирующие класс события по информации из выделенного блока	39
3.14. Моделирование дискриминаторов амплитуд	40
3.15. Моделирование блоков логической электроники	41
<i>Литература</i>	42

1. ВВЕДЕНИЕ

В Институте ядерной физики СО АН СССР началась эксплуатация программы UNIMOD2 для моделирования экспериментов на встречных e^+e^- -пучках. Ранее для моделирования экспериментов на детекторах НД (Нейтральный Детектор, [1]) и МД-1 (Магнитный Детектор, [2]) использовалась программа UNIMOD1 [3]. Основные причины для создания новой версии программы, краткое описание алгоритма моделирования приведены в [4]. Техническое руководство пользователя, содержащее начальные сведения, позволяющие использовать стандартную часть программы UNIMOD2 (без написания своих подпрограмм), имеется в [5]. Данная работа содержит дополнительные сведения, необходимые для написания своих подпрограмм, что необходимо при моделировании реальных детекторов в экспериментах по физике высоких энергий.

Программа UNIMOD2 работает в операционной системе СВМ (Система Виртуальных Машин) на ЭВМ серии ЕС. Терминология, используемая в документации по СВМ, в данной работе применяется без детальных пояснений. В случае возникновения трудностей с пониманием таких терминов следует ознакомиться с принципами работы СВМ, например, в [6].

Подпрограммы для UNIMOD2 написаны на языках Ассемблер и Фортран-77. Все подпрограммы пользователей должны быть совместимы со стандартом языка Фортран-77. Если пользователь не знаком с Фортраном-77, то для написания своих подпрограмм лучше (и проще) изучить описание Фортрана-77 (см., например, [7]), чем пытаться выяснить, совместим ли, например, PASCAL с Фортраном, и какие правила для этого надо соблюдать.

2. ОБЩИЕ БЛОКИ ПРОГРАММЫ

При описании общих блоков подразумевается, что в программе имеется неявное описание типа переменных:

```
IMPLICIT INTEGER *2 (U-W), REAL *8 (Q-S)
```

В Фортране-77 нельзя описывать в общих блоках массивы с неопределённой длиной, например, `WBL(20,UBLOCK)`. Однако, в некоторых общих блоках использована такая форма записи. Это означает в данном случае, что размерность массива `WBL` будет

описана в головной программе в соответствии со значением переменной UBLOCK.

В других случаях размерность массивов тоже определяется после ввода заказа на счёт, но ввести какую-либо описанную переменную в обозначение размерности не удаётся. В таких случаях размерность будет обозначаться точкой, например, QBLOCK(.). В своих программах размерность всех этих массивов можно описывать произвольными константами.

Далее следует описание тех общих блоков, которые представляют интерес для пользователей.

/UMB000/ UBLOCK, UCENTR, WBL(20,UBLOCK)

UCENTR - номер выделенного блока или нуль

WBL:

- 1-4 имя блока (до восьми символов) или пробелы, если имя не назначено,
- 5 тип блока (номер программы в списке геометрических программ в /UMB048/),
- 6 индекс начала описания размеров и положения блока в массиве QBLOCK (/UMB001/),
- 7 количество двойных слов в геометрическом описании,
- 8 номер блока, в который вложен данный, или 0, если нет охватывающего блока,
- 9 кол-во блоков, вложенных в данный, или 0, если нет вложенных блоков,
- 10 индекс в UCLIST (/UMB002/) - начало списка номеров блоков, вложенных в данный,
- 11 номер вещества или 0, если вакуум,
- 12 номер матрицы преобразования вектора к системе блока или 0, если основная система координат,
- 13 кол-во вложенных проволочных камер или нуль,
- 14 индекс начала списка номеров вложенных проволочных камер в массиве USHAML (/UMB005/),
- 15 номер комплекта чисел в /UMB033/, управляющих развитием каскада в данном блоке (пороговые энергии электронов, квантов, максимальный шаг и т.д.),
- 16 номер способа розыгрыша пробега заряженных частиц (1 или 2),
- 17 номер способа розыгрыша пробега нейтральных частиц (1 или 2),

18 индекс в UMATLS (/UMB032/) начала списка номеров веществ, содержащихся внутри данного блока (с учётом вложенных),

19 количество номеров в предыдущем списке. Если оба указанных выше способов розыгрыша пробегов равны 1, то кол-во номеров равно нулю. То же самое, если охватывающий блок имеет способы розыгрыша пробегов 2.

20 номер амплитуды в /UMB016/, которая регистрируется при прохождении заряженных частиц через данный счётчик, или 0, если блок представляет собой поглотитель.

/UMB001/ QBLOCK(.) - характеристики блоков и отсеков магнитного поля, определяющие положение, ориентацию, размеры. Количество чисел, описывающих блок, зависит от типа блока.

/UMB002/ UCLOS0, UCLIST(.)

UCLOS0 - кол-во номеров в начале UCLIST, представляющих список ни во что не вложенных блоков (нулевой уровень),

UCLIST - массив номеров блоков, описывающих вложенность блоков друг в друга (используется в /UMB000/).

/UMB003/ NMAT, AMAT(7,NMAT) - характеристики вещества

DIMENSION UAMAT(14,.)

EQUIVALENCE (UAMAT(1,1),AMAT(1,1))

UAMAT(1,.) - кол-во компонентов вещества. Для простых веществ равно 1.

UAMAT(2,.) - для сложных веществ, если не равно нулю, указывает на то, что пробег до точки взаимодействия надо моделировать по характеристикам эффективного вещества, которые располагаются после описания последнего компонента данного вещества. Индивидуальные характеристики компонентов учитываются при этом для розыгрыша продуктов взаимодействия.

AMAT(2,.) - ат.номер,

AMAT(3,.) - ат.вес,

AMAT(4,.) - плотность (г/см³),

AMAT(5,.) - ср.иониз.потенциал (кэВ),

AMAT(6-7,.) - название вещества (до восьми символов). Если имя

отсутствует (пробелы), то данное вещество входит в ближайшее из предыдущих веществ с именем в качестве компонента. Для всех компонентов сложного вещества плотность понимается как парциальная плотность, т.е. реальная плотность сложного вещества получается суммированием плотностей компонентов. При ссылках на сложное вещество его номер всегда равен номеру первого компонента (это значит, что в номерах веществ могут быть "просветы", например, 1, 2, 5, 6, 8, 15, 16,...).

- /UMB004/ NSHAMV, WSHAM(13,..)
- NSHAMV - количество проволочных камер,
WSHAM(1-3,..) - имя камеры (до шести символов),
WSHAM(4,..) - номер блока, в который вложена камера. Реакция камеры на движение заряженной частицы моделируется только при нахождении частицы внутри указанного блока.
WSHAM(5,..) - тип камеры (номер программы в /UMB049/, обслуживающей эту камеру),
WSHAM(6,..) = 0, если амплитуды не генерируются,
> 0 - количество амплитуд с групп проволочек, при этом в WSHAM(7,..) - индекс первой амплитуды в /UMB019/,
< 0 - номер программы в /UMB022/ генерации амплитуд с обратным знаком, при этом в WSHAM(7,..) - индекс амплитуды с первой проволочки в /UMB020/ (амплитуды генерируются с каждой проволочки и записываются в /UMB020/ подряд).
WSHAM(7,..) - вспомогательный индекс (см. WSHAM(6,..)),
WSHAM(8,..) - индекс начала геометрического описания поверхности, образуемой проволочками, и ориентации проволочек в массиве QSHAM в /UMB006/,
WSHAM(9,..) - количество двойных слов в этом описании,
WSHAM(10,..) - номер матрицы в /UMB026/ преобразования вектора к системе камеры или нуль,
WSHAM(11,..) - количество проволочек,
WSHAM(12,..) - индекс начала "да-нет"-ной информации о срабатывании проволочек в /UMB018/. Левая граница поля всегда выровнена по границе полуслова.
WSHAM(13,..) - номер временных ворот в /UMB058/.

/UMB005/ USHAML(.) - Списки номеров вложенных в блоки камер.

/UMB006/ QSHAM(.) - Описание поверхностей, образуемых проволочками камер, и ориентации проволочек.

/UMB007/ UMFIEL, UMF(10,.)

UMFIEL - количество отсеков магнитного поля,
 UMF(1-2,.) - имя отсека,
 UMF(3,.) - тип геометрии отсека (номер программы в /UMB048/),
 UMF(4,.) - номер охватывающего отсека или 0,
 UMF(5,.) - индекс начала списка отсеков, вложенных в данный (в массиве UFLIST в /UMB008/),
 UMF(6,.) - количество вложенных отсеков или 0,
 UMF(7,.) - индекс начала геометрического описания отсека в QBLOCK в /UMB001/,
 UMF(8,.) - количество двойных слов в этом описании,
 UMF(9,.) - номер матрицы преобразования вектора к системе отсека из /UMB026/ или нуль,
 UMF(10,.) - номер описания магнитного поля в данном отсеке (индекс в UMDES в /UMB010/).

/UMB008/ UMUNCL, UFLIST(.)

UMUNCL - количество отсеков поля, не вложенных в другие отсеки (список номеров этих отсеков приводится в самом начале UFLIST).

Далее в UFLIST следуют списки номеров отсеков, вложенных в другие отсеки (пользоваться через UMF(2-3,)).

/UMB009/ NRECO, WORDS(NRECO) - параметры процедуры UNIMOD2
 CHARACTER * 8 WORDS

/UMB010/ FIELD0, NFDESC, UMDES(12, NFDESC)
 DIMENSION NUMDES(6,.)
 EQUIVALENCE (UMDES(1,1), NUMDES(1,1))

FIELD0 - общий множитель ко всем полям в детекторе
 UMDES(1-2,.) - имя описания
 UMDES(3,.) - тип описания (целое число)
 0 - нестандартное описание (структуру "знает" только подпрограмма пользователя)
 1 - однородное поле
 2 - куб. сплайн дефекта 2

UMDES(4,.) - индекс в массиве FSPLI в /UMB011/ - начало списка узлов сплайна.

UMDES(5,.) - индекс в FSPLI - начало списка коэффициентов сплайна.

UMDES(6,.) - количество коэффициентов сплайна для одной компоненты магнитного поля.

NUMDES(4,.) - количество узлов сплайна по оси X

NUMDES(5,.) - количество узлов сплайна по оси Y

NUMDES(6,.) - количество узлов сплайна по оси Z

Последние числа имеют указанный смысл только при описании сплайном (тип описания 2).

При типе 0 в UMDES(5,.) записывается номер подпрограммы пользователя в /UMB051/, в UMDES(6,.) - индекс первого параметра для этой программы в FSPLI, в UMDES(7,.) - количество этих параметров.

При типе 1 в UMDES(4,.) заносится индекс в FSPLI первой компоненты поля.

/UMB011/ FSPLI(.) - параметры для описания поля

- 1) для программы пользователя (тип 0) структура описания не фиксирована (программа пользователя сама должна понимать занесённые числа).
- 2) в случае однородного поля всё описание состоит из трёх чисел - проекций поля на оси X, Y, Z.
- 3) в случае сплайна сначала приводятся координаты узлов сплайна по оси X, затем - по оси Y, затем - по оси Z. Затем коэффициенты сплайна в узлах для X-ой проекции поля, потом - для Y-ой и, наконец, для Z-ой проекции поля. В каждом узле для каждой проекции приводятся четыре коэффициента: значение функции в узле и три частных производные функции по X, Y, Z. Порядок перечисления узлов следующий: сначала для первого узла по Y и Z перечисляются коэффициенты для всех узлов по X, затем для второго узла по Y и первого узла по Z снова перебираются узлы по оси X и т.д.

/UMB012/ UTYP, UCOUN, UBOX, UMAT, ENER, TIME, QR(3), QV(3), QSPIN(3), ZET, GAM

UTYP - тип частицы (целое число),

UCOUN - номер блока, в котором находится частица (0,1,2,...),

UBOX - номер отсека в объёме поля, в котором находится частица

(0, 1, 2, ...).

UMAT - номер вещества, в котором находится частица (0, 1, ...).

Для UCOUN, UBOX, UMAT номер 0 означает отсутствие номера блока, отсека, вещества.

ENERG- полная энергия частицы в МэВ,

TIME - время от начала розыгрыша события в нсек.

QR - координаты радиус - вектора частицы,

QV - единичный вектор направления скорости в пространстве

QSPIN- направление вектора поляризации частицы в пространстве (модуль не больше 1 или QSPIN(1)=?. если поляризация не задана .

BET = SQRT(1-масса**2/ENERG**2)

GAM = ENERG / масса

/UMB013 UTYPE, UCOUN, UBOX, UMAT, ENERG, TIME, QR(3),
QV(3), QSPIN(3), BET, GAM

Общий олоп характеристик частицы в конце элементарного перемещения. Смысл всех чисел такой же, как в /UMB012 .

/UMB014/ QDL, QDRINT, DENER, DENRS, EXCIT, DTIME, WINT, WSTOP, WRANG,
WSPARK, WAMPL

QDL - длина очередного перемещения частицы вдоль траектории

QDRINT- остаток расстояния до точки взаимодействия

DENER - уменьшение энергии частицы при перемещении (за счет ионизационных потерь и приравненных к ним)

DENRS - ограничение на изменение энергии на очередном перемещении частицы

EXCIT - энергия взаимодействия, пошедшая на возбуждение среды

DTIME - добавка к времени жизни частицы (нсек)

WINT - номер типа "точечного" взаимодействия, случившегося в конце перемещения, или 0, если взаимодействия нет

WSTOP - целое число, характеризующее тип "сработавшего" ограничения на величину элементарного перемещения:

=1 максимальное перемещение заряженной частицы в магнитном поле,

=2 максимальное перемещение заряженной частицы в веществе

=3 максимальное изменение энергии заряженной частицы

=4 граница блока

=5 предел по времени чувствительности детектора

=6 граница отсека магнитного поля
 =7 "точечное" взаимодействие
 =8 ионизационный пробег

WRANG - способ розыгрыша пробега на очередном перемещении частицы (1 или 2)

WSPARK- количество номеров проволочек, добавленных в /UMB018/ на очередном перемещении

WAMPL - количество амплитуд с проволочек, добавленных в /UMB020/ на очередном перемещении

/UMB015/ NPAR1,NPAR2,NZAP1,NZAP2,PARTC1(28,64),PARTC2(28,64)

DIMENSION QPART1(14,64),QPART2(14,64)

DIMENSION WPART1(56,64),WPART2(56,64)

EQUIVALENCE (QPART1(1,1),PARTC1(1,1)),(WPART1(1,1),PARTC1(1,1))

EQUIVALENCE (QPART2(1,1),PARTC2(1,1)),(WPART2(1,1),PARTC2(1,1))

NPAR1- номер частицы, которую надо "проводить" через детектор,

NPAR2- кол-во частиц в истории события к текущему моменту,

NZAP1- номер буфера в массиве PARTC1

NZAP2- номер буфера в массиве PARTC2

WPART1(1,..) - номер поколения (целое число 1, 2, 3, ...)

WPART1(2,..) - номер частицы - родителя. Для первого поколения этот номер равен 0.

WPART1(3,..) - причина "смерти" частицы. Положительное число означает номер "точечного" процесса взаимодействия, завершившего путь частицы. 0 - частица ещё не рассматривалась, (-1) - конец ионизационного пробега (в том числе достижение пороговой энергии), (-2) - выход за пределы разрешённого временного интервала, (-3) - слишком много перемещений, (-4) - принято решение не моделировать оставшиеся частицы события, (-5) - распад нестабильной частицы.

WPART1(4,..) - тип частицы

PARTC1(3,..) - энергия частицы в начале траектории

PARTC1(4,..) - время старта частицы (нсек)

QPART1(3-5,..) - начальная точка траектории

QPART1(6-8,..) - начальное направление вектора скорости (единичный вектор)

QPART1(9-11,..)- начальное направление вектора поляризации

или первое число = 2.

QPART1(12-14,.) - конечная точка траектории

В PARTC2, QPART2, WPART2 - такое же расположение информации, как в вышеописанных массивах. Вся история развития события группируется в блоки по 64 частицы ($112 \cdot 64 = 7168$ байт), которые в случае превышения 128 частиц в событии записываются на магнитный диск (временно). NZAP1 и NZAP2 - порядковые номера таких блоков (буферов), находящиеся в оперативной памяти в описываемом общем блоке. Перемещение блоков частиц на диск и обратно возможно с помощью стандартной программы.

Запись на диск каждого события очевидным образом ограничивает возможное количество частиц в событии доступным местом на диске.

С другой стороны, внутреннее представление целых чисел длиной в 2 байта, используемых для номера частицы-родителя, ограничивает число частиц в каждом событии величиной 32767, и, следовательно, максимальный потребный объем диска под историю события = $7K \cdot 512 = 3584K$. Если NPAR2 > 32000, то история события после 128-ой частицы выбрасывается и полагается WPART1(2,.) = -1 для всех частиц с номером больше 128.

/UMB016/ NAMPL, UAMPL(6, NAMPL)

UAMPL(1-3,.) - имя амплитуды в счётчиках (до шести символов)

UAMPL(4,.) - номер программы преобразования энерговыделения в амплитуду. Список программ в общем блоке UMB017.

UAMPL(5,.) - номер временных ворот в /UMB058/ или нуль

UAMPL(6,.) - индекс первой аналоговой величины в "амплитуде" в /UMB020/

/UMB017/ UCONV, UNAMC(4,UCONV)

UNAMC(1-3,.) - имена программ преобразования энерговыделения в амплитуду.

UNAMC(4,.) - количество аналоговых величин, генерируемых программой (сюда могут входить также времена)

/UMB018/ WIRES(.) - "да-нет"-ная информация о работающих проволочках в камерах. Информация, относящаяся к любой

камере, начинается с левой границы полуслова, т.е. списание сработавших проволочек в камере всегда занимает целое количество полуслов.

/UMB019/ NWAMP, UAMPWR(8,NWAMP) - амплитуды с камер.

UAMPWR(1,..) - номер программы (в /UMB022/), используемой для генерации амплитуды с отдельной проволочки,

UAMPWR(2,..) - количество проволочек с последовательными номерами, которые дают вклад в данную амплитуду,

UAMPWR(3,..) - номер первой проволочки в камере из этого интервала (нумерация проволочек в каждой камере начинается с единицы),

UAMPWR(4,..) - индекс первого слова в /UMB020/, где записывается аналоговая информация,

UAMPWR(5-8,..) - имя амплитуды (до восьми символов).

/UMB020/ AMPVA(.) - аналоговые сигналы с камер и счетчиков.

До начала события всё заполнено нулями.

/UMB021/ NSPARK, USPARW(2,..) - список сработавших на очередном перемещении частицы проволочек (номер камеры, номер проволочки).

/UMB022/ UTYPAM(4,..) - имена программ, генерирующих амплитуды на отдельной проволочке при прохождении одной частицы.

UTYPAM(1-3,..)- имя программы

UTYPAM(4 ,..)- количество слов аналоговой информации, генерируемое при срабатывании одной проволочки.

/UMB023/ NTYPAR, UTYPES(16,NTYPAR)

DIMENSION TYPES(8,NTYPAR)

EQUIVALENCE (UTYPES(1,1),TYPES(1,1))

NTYPAR - количество типов частиц,

TYPES(1-2,..)- названке частицы,

TYPES(3,..)- масса частицы в МэВ,

TYPES(4,..)- заряд частицы в единицах заряда позитрона,

TYPES(5,..)- спин частицы,

TYPES(6,..)- полный магнитный момент частицы в единицах ядерного магнетона,

TYPES(7,..)- произведение скорости света на время жизни частицы (см) или 1.E+50, если частица стабильная,

UTYPES(15,..)- количество мод распада,

UTYPES(16,..)- начало списка мод распада в /UMB024/.

/UMB024/ UMODES(4,.)
 DIMENSION PMODES(2,.)
 EQUIVALENCE (PMODES(1,1),UMODES(1,1))
 PMODES(1,.) - вероятность распада по данному каналу (сумма вероятностей равна 1),
 UMODES(3,.) - количество вторичных частиц в данном канале,
 UMODES(4,.) - индекс начала списка типов вторичных частиц в /UMB025/.

/UMB025/ UTPLST (.) - списки типов частиц по разным каналам распада.

/UMB026/ NMATRIX, NMBUF1, QMATRIX(13,NMATRIX) - преобразования координат из общей системы в локальную.

NMATRIX - количество локальных систем координат
 NMBUF1 - не используется
 QMATRIX(1,.) - имя локальной системы
 QMATRIX(2-4,.) - вектор смещения (см)
 QMATRIX(5-13,.) - матрица поворота (по столбцам). Если вектор в общей системе $RO(i)$, то вектор в локальной системе $R(j)$ находится суммированием по k величины $QMATRIX(3*j+k+1,.) * (RO(k) - QMATRIX(k+1,.)$. Матрица поворота всегда ортогональная, поэтому обратное преобразование соответствует транспонированной матрице. Вектор $RO(i)$ в общей системе находится суммированием по k величины $QMATRIX(3*k+i+1,.) * R(k)$ и добавлением $QMATRIX(i+1,.)$.

/UMB027/ QVECTO(6,.) - вектора положения частицы и направления скорости (попарно) для всех систем координат. Применено или нет преобразование к данной системе - указывается в общем блоке /UMB028/.

/UMB028/ USIGNM(.) - указатели, применено(=1) или нет (=0) преобразование к данной системе координат векторов QR и QV . Если преобразование к текущим координатам скорости частицы применено, то в /UMB027/ находятся их величины в соответствующей локальной системе координат.

/UMB029/ UINTR,UFALS,UNMBR,UREGPR,UBUF1,UMATIN,PTOT,PINTR(UINTR),
 ADECA

UINTR - количество "точечных" взаимодействий,
 UFALS - признак "ложного" взаимодействия, который программа генерации продуктов реакции может выставить. Частица продолжает движение.
 UNMBR - номер точечного взаимодействия, которому выпало произойти (в блоке /UMB030/).
 UREGPR- режим работы программы взаимодействия (1 - следует вычислять коэффициент поглощения, 2 - генерировать продукты взаимодействия)
 UBUF1 - р е з е р в
 UMATIN- номер вещества, для которого ведется расчёт коэффициента поглощения или генерация продуктов распада. Используется программами обслуживания взаимодействий без анализа, простое это вещество или компонент сложного.
 PTOT - сумма коэффициентов поглощения PINTR
 PINTR - коэффициенты поглощения для каждого типа взаимодействия (в ед. 1/см)
 ADECA - коэффициент поглощения (1/см) за счёт распада.
 /UMB030/ UNAMIN(3,UINTR) - имена программ моделирования процессов взаимодействия
 /UMB031/ ULSINT(NTYPAR,UINTR)
 ULSINT(j,i) - указывает для i-ого взаимодействия и частицы типа j, подвержена ли эта частица данному взаимодействию (=1) или нет (=0).
 /UMB032/ UMATLS(.) - списки номеров веществ, вложенных в блоки, для которых возможна генерация пробега типа 2.
 /UMB033/ NCUTS,CUTS(10,.)-управляющие числа для развития каскада
 DIMENSION LCUTS(10,.)
 EQUIVALENCE (CUTS(1,1),LCUTS(1,1))
 NCUTS - количество комплектов чисел,
 CUTS(1,.) - имя комплекта чисел,
 CUTS(2,.) - максимальный шаг частиц, на которые влияет магнитное поле,
 CUTS(3,.) - максимальный шаг заряженных частиц в веществе в г/см*2,
 CUTS(4,.) - максимально допустимое изменение энергии частицы на пробеге до точки взаимодействия (МэВ),

CUTS(5,.) - максимальная доля энергии, которую может потерять частица на элементарном перемещении до точки взаимодействия (величина от 0 до 1),

CUTS(6,.) - пороговая энергия γ -квантов (МэВ) в процессе тормозного излучения,

CUTS(7,.) - пороговая кинетическая энергия электронов и позитронов в процессе рассеяния на электронах атомов,

CUTS(8,.) - ограничение на средний угол многократного рассеяния (градусы),

LCUTS(9,.) - количество пороговых кинетических энергий для разных частиц. При понижении энергии частицы ниже пороговой она выбывает из каскада, остаток её кинетической энергии принимается за энерговыделение в точке остановки.

LCUTS(10,.) - индекс описания первой пороговой энергии в /UMB060/. Частица с энергией ниже пороговой не рассматривается. Если частица заряженная или фотон, то её кинетическая энергия добавляется в энергию возбуждения среды EXCIT в /UMB014/ и далее может использоваться в регистрирующих приборах. Энергия заряженной частицы уменьшается до энергии покоя. Если для данного типа частиц имеется программа моделирования взаимодействия в остановке, то управление передается ей, в противном случае для нестабильных частиц вызывается программа моделирования распада.

/UMB034/ BFIELD, BOVO, QFAZE, RFIELD(3), RVOXB(3), RBLOC(6,.)

BFIELD - абсолютная величина магнитного поля (кГс),

BOVO - косинус угла между полем и скоростью частицы,

QFAZE - множитель перед пройденным расстоянием для получения приращения фазы винтовой линии,

RFIELD - единичный вектор направления магнитного поля,

RVOXB - вектор, равный векторному произведению единичного вектора скорости частицы на единичный вектор RFIELD,

RBLOC - вектора RFIELD и RVOXB, преобразованные в локальные системы координат. Проведены эти преобразования или нет, можно узнать по указателям в USIGNM в /UMB028/.

/UMB035/ NMAMAX, NMUSED, ALFI(UINTR,.)

NMAMAX - максимальное количество веществ в списке "вложенных" веществ,
NMUSED - количество веществ, для которых в ALFI есть вычисленные коэффициенты поглощения,
ALFI - коэффициенты поглощения для каждого процесса и каждого компонента вещества
/UMB036/ **NMATM, ALFASM, ALFIM(UINTR,.)**
NMATM - номер вещества, коэффициент поглощения которого используется для розыгрыша пробега,
ALFASM - суммарный коэффициент поглощения, который был использован для розыгрыша пробега,
ALFIM - парциальные коэффициенты поглощения по процессам и разным веществам (необходимо при розыгрыше пробега типа 2).
/UMB037/ **NEVENT, NCHARG, NNEUT, NPTOT, NTSTAR, NTIMEV, QRANDS, QPULST(3), QE2TOT, JSURV**
NEVENT - номер события,
NCHARG - количество заряженных частиц в событии,
NNEUT - количество нейтральных частиц в событии,
NPTOT - полное количество частиц в событии,
NTSTAR - остаток времени счёта в момент, когда событие стартовало (мсек),
NTIMEV - полное время CPU на это событие (мсек),
QRANDS - случайное число, с которого стартовало событие,
QPULST - вектор импульса системы начальных электрона и позитрона,
QE2TOT - полная энергия системы начальных электрона и позитрона в МэВ,
JSURV - номер просмотра буфера частиц (1 или 2).
/UMB038/ **SLUM, QSRAND, RINTRG(3), SINTRG(3), QNAMCT, VTAPE(3), VTAPE2(3), NRUN, ENERGY, E1SPR, IREND, NEVMAX, NTSAVE, TIMAX, NTIMAX, NSLONG, NRANLG(2), NTISUM, NPARMX, NPARSM, JSINT, INTOLD, LPRINT**
SLUM - суммарное значение величины, используемой для вычисления интеграла сечения.
QSRAND - стартовое случайное число для всего счёта.
RINTRG - координаты центра области взаимодействия (см)
SINTRG - размеры центра области взаимодействия (ср.-квадр., см)

QNAMCT - имя каталога файлов на магнитных лентах для записи результатов.

VTAPE1 - логическое имя магнитной ленты

VTAPE2 - логическое имя запасной магнитной ленты

NRUN - цифровая часть имени файла с результатами моделирования для записи на маг.ленту. Если имеет нулевое значение - запись не заказана.

ENERGY - энергия одного пучка (МэВ).

E1SPR - разброс энергий в одном пучке (МэВ).

IEND - необязательное средство для общения с оператором. При единичном значении этого флага счёт заканчивается. Нормальное значение переменной IEND - нуль.

NEVMAX - максимальное потребное кол-во событий. По достижении этой статистики счёт прекращается.

NTSAVE - запас по времени для записи результатов на маг.ленту и распечатки (сек. процессорного времени). Генерация нового события не начинается, если оставшееся время до конца счёта меньше суммы максимального времени на событие, зарегистрированного на предыдущих событиях, и NTSAVE.

TIMAX - ограничение по времени в мсек на время чувствительности детектора от начала события.

NTIMAX - время генерации в мсек самого долгосчётного события.

NSLONG - номер самого долгосчётного события.

NRANLG - случайное число, с которого начался розыгрыш самого долгосчётного события.

NTISUM - суммарное время в мсек, затраченное на генерацию всех событий.

NPARMX - максимальное количество частиц, достигнутое в каком-либо событии.

NPARSM - суммарное количество всех частиц во всех событиях.

JSINT - величина интервала в секундах процессорного времени между выходами на точку программы NEXTEP. Если JSINT<0, то выхода на точку NEXTEP не делается.

INTOLD - остаток времени счёта в момент последнего выхода на точку NEXTEP (сек).

LPRINT - целое число 0,1,2,.. - уровень печати о развитии события

/UMB039/ NDISCR, UDISCR(10,..)
 DIMENSION PDISCR (5,..)
 EQUIVALENCE (UDISCR(1,1),PDISCR(1,1))
 NDISCR - количество дискриминаторов
 UDISCR(1-2,..) - собственное имя дискриминатора
 UDISCR(3,..) - выход дискриминатора (0 или 1)
 UDISCR(4,..) - тип дискриминатора, целое число. Является
 одновременно номером программы в /UMB040/,
 обслуживающей этот тип дискриминаторов.
 UDISCR(5,..) - количество входных амплитуд.
 UDISCR(6,..) - индекс описания первой амплитуды в /UMB041/
 PDISCR(4-5,..) - два порога дискриминатора. Использование
 зависит от программы, моделирующей дискриминатор.

/UMB040/ NPROGD, UNAMDS(4,NPROGD)
 NPROGD - количество типов дискриминаторов
 UNAMDS(1-3,..) - название программы
 UNAMDS(4,..) - целое число, возможное количество входов в
 дискриминатор (0, 1, 2,...). Если это число = -1, то
 программа может работать с любым числом входных
 импульсов.

/UMB041/ ULISD(.) - списки номеров амплитуд в /UMB020/ для
 дискриминаторов

/UMB042/ NLOG, ULOGS(6,NLOG)
 NLOG - количество логических блоков быстрой электроники
 ULOGS(1-2,..) - название блока
 ULOGS(3,..) - тип блока (номер программы)
 ULOGS(4,..) - выход блока. Если блок сработал - единица, иначе
 нуль
 ULOGS(5-6,..) - количество входов и индекс первого входа в
 ULISBL в /UMB044/.

/UMB044/ ULISBL(2,..) - списки входов для логических блоков
 электроники. Первое число - номер общего блока: 1 -
 общий блок дискриминаторов /UMB039/, 2 - общий блок
 логической электроники /UMB042/. Второе число -
 индекс.

В виде исключения из ранее декларированных правил
 именовании программ и общих блоков целая серия программ и
 общих блоков включена со своими прежними именами - некоторая

выборка из существующего гистограммного комплекса. Ниже перечисляются общие блоки, представляющие интерес для пользователей программы UNIMOD2.

/GIST01/ NUSL, KUSL(5,NUSL) - условия на параметры

NUSL - количество условий

KUSL(1,..) - имя условия

KUSL(2,..) - номер параметра в блоке /GIST02/

KUSL(3,..) - первая граница

KUSL(4,..) - вторая граница

KUSL(5,..) - результат проверки условия для очередного события
(0 или 1)

/GIST02/ NPAR, NAMPAR(NPAR) - имена параметров

/GIST03/ FORM(.) - тексты логических формул. Следуют непрерывно друг за другом

CHARACTER *1 FORM

На шаге ввода тексты формул не меняются, а на шаге исполнения имена условий заменены на номера условий в /GIST01/, имена формул заменены на <10000 + номер формулы> в /GIST04/.

/GIST04/ NFORM, NFORML(4,NFORM)

NFORM - количество формул

NFORML(1,..) - имя формулы

NFORML(2,..) - индекс первого символа формулы в FORM

NFORML(3,..) - количество байтов в формуле в FORM

NFORML(4,..) - результат вычисления формулы (0 или 1)

/GIST09/ IGPARG(.) - список номеров параметров, которые используются для занесения в гистограммы.

/GIST10/ NGISO, NGISTO(6,NGISO)

NGISO - количество нулевых гистограмм

NGISTO(1-2,..) - имя гистограммы или пробелы

NGISTO(3-4,..) - не используется (занесено FEDCBA9876543210)

NGISTO(5,..) - номер условия отбора событий. Является номером условия на параметр, если не превышает 10000, в противном случае является <номером формулы + 10000>. Если 0, то условие не задано.

NGISTO(6,..) - характеристика списка параметров гистограммы. В первом полуслове - начальный индекс списка номеров в /GIST09/, второе полуслово - количество номеров в

списке.

/GIST11/ NGIS1, NGIST1(8,NGIS1)

Первые 6 описателей гистограммы на 30 каналов
такие же, как в предыдущем блоке.

NGIST1(7,..) - центр гистограммы

NGIST1(8,..) - шаг гистограммы

/GIST12/ NGIS2, NGIST2(8,NGIS2) - описание одномерных гистограмм
с числом каналов 80. Структура описания такая же, как
для 30 - канальных.

/GIST13/ NGIS3,NGIST3(11,..)-описание двумерных гистограмм 24*24.

Первые 8 слов описания имеют такой же смысл, как в
описании одномерных гистограмм, включая
"параметр-центр-шаг" для первой оси. NGIST(9-11,..)
описывают "параметр-центр-шаг" по второй оси.

/GIST14/ NBUNGO(6,..) - бухгалтерия по нулевым гистограммам

DIMENSION QBUNGO(3,..)

EQUIVALENCE (NBUNGO(1,1),QBUNGO(1,1))

NBUNGO(1,..) - количество событий, попавших в гистограмму

NBUNGO(2,..) - количество забракованных событий

QBUNGO(2,..) - сумма значений параметров

QBUNGO(3,..) - сумма квадратов значений параметров

/GIST15/ NBUNG1(36,..) - бухгалтерия по 30-канальным
гистограммам + 30 накопителей

/GIST16/ NBUNG2(86,..) - бухгалтерия по 80-канальным
гистограммам + 80 накопителей

/GIST17/ NBUNG3(586,..) - бухгалтерия по двумерным гистограммам +
24*24= 576 накопителей

/GIST18/ PAR(.) - значения параметров

/UMB045/ NPGIST, VNAMPR(7,NPGIST)

NPGIST - количество программ вычисления параметров

VNAMPR(1-3,..) - название программы

VNAMPR(4,..) - количество входных параметров

VNAMPR(5,..) - количество выходных параметров

VNAMPR(6,..) - индекс первого номера входного параметра в

ULPIN в /UMB047/

VNAMPR(7,..) - индекс первого выходного параметра в /GIST18/

/UMB048/ UGEOM, UNAMG(4,UGEOM)

UGEOM - количество геометрических программ

UNAMG(1-3,..) - название программы

UNAMG(4,..) - количество параметров, описывающих геометрическую структуру, или число меньше нуля, если количество параметров может быть переменным и на заводе его контролировать не надо.

/UMB049/ NCHPR, UNCHPR(4,NCHPR)

NCHPR - количество программ моделирования проволочных камер

UNCHPR(1-3,..)- название программы

UNCHPR(4,..) - количество параметров, характеризующих этот тип камер, или число меньше нуля, если это количество может быть переменным.

/UMB050/ USION, USMUL, UNION(3), UNMUL(3)

USION =0 нет ионизационных потерь
 =1 ионизационные потери моделируются

USMUL =0 нет многократного рассеяния
 =1 многократное рассеяние моделируется

UNION - название программы ионизационных потерь

UNMUL - название программы многократного рассеяния

/UMB055/ NLENT(120) - размеры общих блоков в байтах.

/UMB057/ UINTAB(6,..)

DIMENSION ALFASK(3,..)

EQUIVALENCE (UINTAB(1,1),ALFASK(1,1))

UINTAB(1,..) - индекс в ALFI (/UMB035/) начала массива парциальных коэффициентов поглощения

UINTAB(2,..) - количество компонент в UMB035 для этого вещества

UINTAB(3,..) - номер вещества, которому они соответствуют

UINTAB(4,..) - номер первой компоненты вещества в /UMB003/

ALFASK(3,..) - суммарный коэффициент поглощения по компонентам и процессам

/UMB058/ NGATE, GATCH(3,..)

NGATE - количество временных ворот

GATCH(1,..) - имя ворот

GATCH(2-3,..) - временные ворота в нсек (интервал)

/UMB059/ NDPMAT, DORMAT(.)

NDPMAT - количество вспомогательных величин на каждое вещество (расширение /UMB003/).

Если обозначить за $NSH=(UMAT-1) \cdot NDPMAT$, то

$$DOPMAT(NSH+1)=AMAT(4,UMAT) \cdot AMAT(2,UMAT)/AMAT(3,UMAT)$$

$$DOPMAT(NSH+2)=0.3066 \cdot DOPMAT(NSH+1)$$

$$DOPMAT(NSH+3)=(11.9185+16.7 \cdot AMAT(5,UMAT)) \cdot AMAT(5,UMAT) \cdot \cdot 2$$

$$DOPMAT(NSH+4)=0.978467E-3 \cdot AMAT(5,UMAT) \cdot EXP(0.666667+DOPMAT(NSH+3)/AMAT(2,UMAT))$$

$$DOPMAT(NSH+5)=0.666667/DOPMAT(NSH+4) \cdot \cdot 1.5$$

$$DOPMAT(NSH+6)=AMAT(2,UMAT) \cdot \cdot 2 \cdot AMAT(4,UMAT) \cdot (5. +$$

$$AMAT(2,UMAT) \cdot \cdot 2 / AMAT(3,UMAT) / (1. +$$

$$AMAT(2,UMAT) \cdot \cdot 2)$$

$$DOPMAT(NSH+7)=0.978467E-3 \cdot AMAT(5,UMAT)$$

/UMB060/ CUTKIN(2,.) - списки пороговых энергий
 DIMENSION NPCUT(2,.)
 EQUIVALENCE (CUTKIN(1,1),NPCUT(1,1))
 NPCUT(1,.) - номер типа частицы в /UMB023/
 CUTKIN(2,.) - пороговая кинетическая энергия (МэВ).
 /UMB061/ ALLCUT(NTYPAR,NCUTS)
 ALLCUT(i,j) - пороговая кинетическая энергия (МэВ) для i-ой частицы в j-ом комплекте управляющих чисел каскада (см. UMB033)
 /UMB062/ NDECPR,UDECDS(5,NDECPR) - список программ распадов
 NDECPR - количество программ во всём списке
 UDECDS(1-3,.) - имя программы
 UDECDS(4 ,.) - номер типа нестабильной частицы
 UDECDS(5 ,.) - номер её канала распада

3. ПРАВИЛА НАПИСАНИЯ ТИПОВЫХ ПРОГРАММ

3.1. Общие замечания

Программы следует писать на языках программирования Фортран-77 или Ассемблер.

При написании программ на Фортране-77 и при использовании общих блоков в дополнение к фортрановским правилам описания типов переменных по умолчанию следует пользоваться следующим определением:

```
IMPLICIT INTEGER *2 (U-W), REAL *8 (Q-S)
```

Это рекомендуется делать для уменьшения вероятности неправильного описания типа переменной из общих блоков программы

UNIMOD2, так как во всех инструкциях подразумевается именно такое неявное описание типа. С другой стороны, следует помнить об этом описании и не называть, например, вектор скорости частицы переменной VECTOR - она получится переменной целого типа.

Общие блоки, необходимые для работы программ, не выписываются в этой главе. Описание общих блоков надо смотреть в главе 4. Здесь же названия переменных из этих общих блоков используются без пояснений.

Любая диагностика о ненормальном ходе моделирования должна начинаться с имени подпрограммы, которая её выдаёт. Если требуется после диагностики аварийно завершить работу, то в фортрановских программах следует пользоваться оператором

STOP 8

Программам, предназначенным для включения в программу UNIMOD2 для общего пользования должны назначаться имена в соответствии со следующим правилом: имя начинается с букв UM, затем следует номер версии (для начального комплекта - 1) и число, указывающее на принадлежность тому или другому автору:

000 - 099	Букин А.Д.
100 - 199	Эйдельман С.И.
200 - 299	Таюрский В.А.
300 - 399	Иванченко В.Н.
400 - 499	Грозина Н.А.
600 - 699	Грозина Н.А.
700 - 799	Дубровин М.С.
800 - 899	Кац И.Л.

Пример имени: UM1024

Для таких программ не рекомендуется вводить без всяких оснований общие блоки - лучше пользоваться параметрами подпрограмм. Если же необходимо ввести общий блок, то название для него следует выбирать UMBnnn, где nnn трехзначный номер в соответствующих пределах для каждого автора (см. выше).

Для моделирования случайных процессов следует использовать генератор случайных чисел DRNDM и другие генераторы, написанные на его основе: DDCOSI, DDGAUS и т.д.

Для поворота единичного вектора на угол TETA со случайным азимутальным углом можно пользоваться программой UM1026:

```
DOUBLE PRECISION RUNIT(3),RCT,RST
CALL UM1026(RUNIT,RCT,RST)
```

Здесь перед обращением к программе должен быть определён вектор RUNIT, который после вызова программы UM1026 окажется повернутым. Нормированность вектора существенным образом используется при преобразовании, поэтому проводится проверка неравенства $ABS(RUNIT**2-1.) < 1.E-10$, и если оно не выполняется, то делается аварийный останов. Угол поворота задаётся своими косинусом RCT и синусом RST.

Однако, для экономии времени при поворотах одного и того же вектора начальной частицы для получения векторов направления скорости вторичных частиц можно пользоваться следующей возможностью в программе UM1026. Если первая компонента вектора RUNIT, на входе в программу UM1026 окажется больше 1.5, то это не рассматривается как нарушение нормировки, но меняет алгоритм вычисления компонент повернутого вектора. Вектор RUNIT, дополнительные к нему орты и азимутальный угол поворота берётся от предыдущего обращения, из параметров данного обращения к программе используется только косинус и синус полярного угла (переменные RCT и RST).

Для чтения записей из файлов на диске можно пользоваться программой UM1077.

Обращение на Фортране:

```
CALL UM1077(A,LMAX,file,typ,mode,ILINE,IFLAG,LENGTH)
```

где

A - адрес буфера для ввода записи,
LMAX - максимальная длина записи (длина буфера в байтах),
file - имя файла (8 символов),
typ - тип файла (8 символов),
mode - имя диска (2 символа),
ILINE- номер записи файла,
IFLAG- код завершения:
=0 запись успешно прочитана,
=1 файл отсутствует,
=2 строка отсутствует,
=3 строка превышает длину буфера,
=4 сбой при чтении,

LENGTH-количество переданных байт в буфер A (меньше или

равно LMAX).

Примечания:

1. Переменные file, typ, mode должны иметь указанную длину и при необходимости должны быть дополнены справа пробелами. Если file, typ или mode начинаются с символа "*", то это означает "любой" и следующие за ним символы не имеют значения. При этом будет прочитан первый из файлов с заданными характеристиками.
2. В случае IFLAG=3 часть записи, которая помещается в буфер, прочитана успешно.
3. Записи можно читать из разных файлов в любом порядке, открытие и закрытие файлов производится при каждом обращении к UM1077.

Для чтения больших файлов к меньшим затратам процессорного и астрономического времени ЭВМ приводит использование программы UM1078 - последовательное чтение файла с диска.

Два режима обращения к программе:

- 1) CALL UM1078(1, FILE, TYPE, MODE, IFLAG) - открыть файл

где FILE - имя файла (8 байтов),

TYPE - тип файла (8 байтов),

MODE - режим файла (2 байта),

IFLAG - код возврата:

=0 нормальное открытие,

=20 ошибка в имени, типе или режиме файла,

=28 файл не найден

Примечание: если в первой позиции поля FILE, TYPE или MODE есть символ "*", то это означает "любой".

- 2) CALL UM1078(2, A, LMAX, IFLAG, LEN) - читать очередную запись

где A - начало буфера ввода длиной LMAX байт,

IFLAG - код возврата

= 0 запись прочитана,

= 1 файл не открыт,

= 2 адрес буфера недействительный,

= 3 постоянная ошибка ввода,

= 8 не вся запись поместилась в буфер,

= 12 конец файла.

LEN - длина прочитанной записи в байтах.

Для записи информации о событиях на диск можно использовать

программу UM1079 - последовательная запись файла на диск с переменной длиной записи.

Три режима обращения к программе:

1) CALL UM1079(1,FILE,TYPE,MODE,IFLAG) - открыть файл

где FILE - имя файла (8 байтов),

TYPE - тип файла (8 байтов),

MODE - режим файла (2 байта),

IFLAG - код возврата:

=0 нормальное открытие,

=20 ошибка в имени, типе или режиме файла.

Если файл с таким именем уже существует, то возможны варианты, зависящие от значения IFLAG при входе в программу:

IFLAG=28 файл затирается,

=32 файл продолжается (запись в конец файла),

в остальных случаях происходит аварийный останов.

2) CALL UM1079(2,A,LEN,IFLAG) - записать очередную запись

где A - начало буфера вывода длиной LEN байт,

IFLAG - код возврата

= 0 запись прошла успешно,

= 1 файл не открыт,

= 2 адрес буфера недействительный,

= 3 постоянная ошибка вывода,

= 4 буква режима файла недействительна,

= 5 цифра режима файла недействительна,

= 12 попытка записать на диск, открытый только для чтения,

= 13 диск переполнен,

= 17 длина записи больше 65К байт,

= 20 имя файла содержит запрещенный символ,

= 21 тип файла содержит запрещенный символ.

3) CALL UM1079(3) - закрыть файл (обязательная процедура).

3.2. Генерация параметров первичных частиц

Используемые общие блоки:

/UMB013/, /UMB037/, /UMB038/.

Программа первичного моделирования должна поочерёдно моделировать характеристики частиц, рождённых в столкновении электрона и позитрона, заносить их в общий блок /UMB013/ (UTYRF - тип частицы, целое число, ENERGF - полная энергия (МэВ), TIMEF - начальное время события (нсек), QRF(3) - вектор положения частицы в общей системе координат (см), QVF(3) - единичный вектор направления движения частицы, QSPINF(3) - вектор направления поляризации) и записывать в буфер частиц путем обращения к программе UM1024 (программа без параметров).

Перед обращением к программам первичного моделирования в /UMB013/ определены величины TIMEF = 0 - стартовое время события в нсек, QRF - радиус-вектор точки взаимодействия в соответствии с положением центра области взаимодействия встречных пучков и средне-квадратичными размерами по трём осям (см), QSPINF(1) - первая компонента вектора поляризации положена равной 2, что соответствует отсутствию поляризации. Вообще говоря, программы первичного моделирования могут менять эти величины по своему усмотрению, однако, это должно быть оговорено в описании этой программы для общего пользования.

Возможным источником информации для программ первичного моделирования являются переменные: RINTRG(3) - координаты центра области взаимодействия, SINTRG(3) - среднеквадратичные размеры области взаимодействия по каждой оси, ENERGY - энергия одного пучка и E1SPR разброс энергий в одном пучке, МэВ. Эти переменные определяются данными во входном потоке и их не следует менять.

При первом обращении к программе первичного моделирования она должна выдать сообщение, содержащее название этой программы, имя автора, дату фиксации версии и, желательно, краткую характеристику процесса. Это необходимо для повышения надёжности моделирования, т.к. головная программа никак не контролирует работу программ первичного моделирования.

Не накладывается ограничений на вызов нескольких программ первичного моделирования, этим можно разумно пользоваться.

В общем блоке /UMB037/ имеются переменные QE2TOT и QPULST,

которые формируются головной программой на основе данных о разбросе энергий в каждом пучке и имеют смысл, соответственно, полной энергии и вектора полного импульса (МЭВ) системы начального электрона и позитрона (в предположении, что пучки движутся в направлении оси X). Эти величины могут использоваться или игнорироваться программами первичного моделирования, а также модифицироваться для использования последующими программами первичного моделирования. Например, под видом программы первичного моделирования можно включать программы, модифицирующие энергию и импульс системы для других программ первичного моделирования. Так можно моделировать радиационные поправки в некоторых случаях.

На программы первичного моделирования возлагается работа по формированию сечения регистрации. В переменной SLUM надо накапливать такую сумму, чтобы отношение числа зарегистрированных событий к SLUM давало сечение регистрации в нанобарнах этих событий. Если известен интеграл сечения по фазовому объёму SIGMA, то на каждом событии программа первичного моделирования должна в SLUM добавлять $1./SIGMA$. Если программа первичного моделирования не производит такого суммирования, то сечение регистрации придётся определять вне рамок программы UNIMOD.

3.3. Программы моделирования процессов взаимодействия частиц с веществом

В программе используются общие блоки:

/UMB029/, /UMB003/, /UMB012/, /UMB013/, /UMB014/.

UMATIN в /UMB029/ - номер вещества, в котором движется частица. Вакууму соответствует UMATIN=0. Программа моделирования взаимодействия не должна анализировать, простое это вещество или сложное. Для неё все вещества - простые. Многокомпонентность веществ учитывается головной программой.

Необходимо реализовать два режима работы программы в зависимости от значения переменной UREGPR:

UREGPR=1 программа моделирования процесса должна вычислить коэффициент поглощения частицы в указанном веществе в единицах $1/см$ и занести его в переменную PINTR(UNMBR). Программа может использовать метод выравнивания сечения

и "сообщать" головной программе вместо истинного коэффициента поглощения завышенный коэффициент, соответствующий сумме истинного и "ложного" взаимодействия. Затем при генерации продуктов взаимодействия иногда (с соответствующей вероятностью) надо отказываться от моделирования взаимодействия (см. ниже). Такой приём рекомендуется использовать, когда сечение взаимодействия известно в дифференциальном виде и не интегрируется в элементарных функциях или сечение зависит от меняющейся энергии (для заряженной частицы). В последнем случае можно пользоваться информацией из /UMB012/ и /UMB014/: ENERГ - полная энергия частицы в начале очередного перемещения (МэВ), DENRS максимально возможное уменьшение энергии (МэВ) на очередном перемещении.

UREGPR=2 программа должна моделировать продукты взаимодействия. Со времени последнего обращения к программе с UREGPR=1 могут измениться такие характеристики, как полная энергия ENERГ, номер вещества UMATIN, коэффициент поглощения PINTR(UNMBR).

Если программа приняла решение, что произошло "ложное" взаимодействие, то она должна "выставить флаг" UFALS=1 и вернуть управление головной программе. Для правильного принятия решения необходимо использовать также значение PINTR(UNMBR).

Если же принято решение, что взаимодействие действительно произошло, то моделировать его необходимо для текущих параметров частицы и вещества из /UMB012/. Все вышеизложенные рассуждения относительно UFALS - "ложного" взаимодействия, - касаются только случая WSTOP=7 в /UMB014/. Во всех остальных случаях надо обязательно генерировать продукты взаимодействия.

Если в результате взаимодействия возникает энергия возбуждения среды, которая может регистрироваться в виде сцинтилляционного света, то её следует добавить в переменную EXCIT. Характеристики продуктов реакции: UTYPF - номер типа частицы, ENERGF-полная энергия в МэВ, QVF(3) - единичный вектор направления движения,

QSPINF(3) - вектор поляризации частицы, формируются по очереди для всех продуктов реакции в общем блоке /UMB013/ с записью характеристик каждой частицы в буфер ожидания путем обращения к программе UM1024 (программа без параметров). В настоящее время движение частиц со спином не моделируется, поэтому следует полагать QSPINF(1)=2.

3.4. Распады нестабильных частиц

На каждый канал распада каждой нестабильной частицы регистрируется имя программы, моделирующей этот распад. Программа должна иметь столько параметров, сколько зарегистрировано частиц - продуктов распада. Обращение из головной программы производится в следующей форме:

```
CALL XXXXXX(IT1,IT2,IT3,...)
```

где IT1, IT2, ... - номера типов частиц - продуктов распада в том порядке, в каком они зарегистрированы в описании частиц.

При работе программ следует использовать следующие блоки:

```
/UMB012/, /UMB013/, /UMB023/.
```

Для каждой частицы, получающейся при распаде, надо в общем блоке /UMB013/ сформировать переменные UTYPF (тип частицы), ENERGF (полная энергия в МэВ), QVF - единичный вектор направления движения. Поляризация не моделируется, поэтому QSPINF(1) = 2. После этого обращением к программе UM1024 (без параметров) записать характеристики новой частицы в буфер ожидания и перейти к моделированию следующей частицы.

Преобразование Лоренца можно осуществлять программой UM1064(QV, GAM, QVF, ENERGF, EMAS), где QV - единичный вектор направления движения распадающейся частицы, GAM - её Лоренц-фактор, EMAS - масса новой частицы (МэВ). Параметры QVF и ENERGF перед обращением должны содержать единичный вектор направления движения и полную энергию в МэВ новой частицы в системе покоя распадающейся. После обращения в этих параметрах будут аналогичные величины в лабораторной системе.

3.5. Программы моделирования геометрических блоков

Программа должна иметь следующий список параметров: (UGPAR, QPAR, QRO, QVO, RMAX, RMIN, BFIELD, BOVO, QFAZE, RBUNIT, RVOB, JR, R), где массивы имеют следующую размерность:

DIMENSION QPAR(.), QRO(3), QVO(3), RBUNIT(3), RVOB(3)

Все параметры являются входными, кроме JR и R.

Транслятор не принимает запись QPAR(UGPAR), поэтому в программе надо писать реальную размерность массива, а UGPARG можно использовать для контроля, что массив описания сформирован правильно.

Такие параметры частицы, как тип, энергия, заряд, масса, следует выбирать из общих блоков /UMB012/ и /UMB023/ (см. главу 2).

Входные параметры:

- QPAR(.) - массив геометрических параметров блока (в программе в операторе DIMENSION надо подставлять реальную размерность,
- QRO(3) -- вектор положения частицы в локальной системе координат,
- QVO(3) - единичный вектор направления движения частицы в локальной системе координат,
- RMAX - максимальное расстояние вдоль траектории, на которое следует анализировать пересечение с поверхностью блока,
- RMIN - минимальное расстояние вдоль траектории, которое считается отличным от нуля. Геометрическая программа никогда не должна выдавать в качестве результата расстояние, меньшее RMIN. Если расстояние до какой-то грани меньше RMIN, надо считать, что эту грань уже прошли.
- BFIELD - абсолютное значение индукции магнитного поля (кГс) в точке, где находится частица. Если BFIELD=0 или заряд=0, то остальные входные параметры не определены.
- BOVO - косинус угла между напряженностью поля и скоростью частицы,
- QFAZE - множитель перед пройденным расстоянием для получения приращения фазы винтовой линии,
- RBUNIT(3) - единичный вектор направления магнитного поля в

локальной системе координат,
 $RVOB(3)$ - вектор, равный векторному произведению вектора направления скорости на единичный вектор вдоль вектора магнитной индукции (всё в локальной системе координат).

Параметр JR является и входным, и выходным. Параметр R - расстояние вдоль траектории до ближайшей грани блока, - является выходным.

Режим работы программы зависит от значения JR:

	JR	
	Вход	Выход
1) Определить, частица внутри или снаружи блока. R вычислять не надо, параметры магнитного поля не использовать.	0	0 - снаружи 1 - внутри
2) Вычислить расстояние до ближайшей грани (частица внутри блока)	1	0 - частица не внутри 1 - есть R 2 - $R > RMAX$
3) Вычислить расстояние до ближайшей грани (частица снаружи блока)	2	0 - частица не снаружи а внутри 1 - есть R 2 - $R > RMAX$ или нет пересечения

В тех случаях, когда расстояние до грани превышает RMAX, то его можно не вычислять точно и R не определять.

Если программа обнаружит, что статус частицы ошибочный, т.е. головная программа "думает", что частица внутри, а она - снаружи блока, или наоборот, то следует выдавать сообщение с подробной информацией для анализа ситуации.

Предполагается, что все геометрические программы могут работать с магнитным полем. По крайней мере, все должны проверять BFIELD и если есть поле, а программа не рассчитана на магнитный вариант, то выдавать сообщение и аварийно останавливаться (STOP 8).

3.6. Моделирование ионизационных потерь

Эта программа должна учитывать любые непрерывные потери энергии, то есть те потери, при которых или не рождаются вторичные частицы, или их "личной жизнью" можно пренебречь.

Программа, генерирующая ионизационные потери энергии на участке пути с учётом флуктуаций, используется для всех заряженных частиц. Вызов:

CALL XXXXXX(E,PM,CH,EP,D,RX,MAT,*)

E - начальная энергия летящей частицы в МэВ,

PM - масса частицы в МэВ,

CH - заряд частицы,

EP - устанавливаемый порог рождения дельта-электронов в МэВ,

D - значение потерянной энергии [МэВ] на перемещении RX [см], в случае выхода по метке в D должно быть расстояние до точки остановки,

RX - перемещение,

MAT- номер вещества, в котором находится частица,

* - выход по метке, если частица не пройдет заданное перемещение RX, в этом случае выдаётся расстояние до точки остановки.

3.7. Генерация угла многократного рассеяния

Вызов:

CALL XXXXXX(E,PM,CH,RX,MAT,TETA)

где E - энергия летящей частицы в МэВ,

PM - масса частицы в МэВ,

CH - заряд частицы,

RX - перемещение в см,

MAT - номер вещества, в котором находится частица,

TETA- угол поворота вектора скорости от первоначального направления (радиан).

Первые пять аргументов - входные, последний аргумент - выходной.

3.8. Вычисление магнитного поля

Эти программы необходимо писать, если поле не однородное и не представляется кубическим сплайном.

Обращение к программе:

```
CALL XXXXXX(BFIL,FSP,NP)
```

где

BFIL(3) - компоненты поля (кГс), которые программа должна вычислить без учёта общего множителя ко всем полям в /UMB010/ FIELD0,

FSP(NP) - параметры поля (задаются в исходных данных при запуске программы UNIMOD2).

Координаты точки, в которой надо вычислить магнитное поле (радиус-вектор частицы), находятся в общем блоке /UMB012/..., QR(3).

3.9. Моделирование проволочной камеры

Вызов программы:

```
CALL XXXXXX(IFL,ICH,QCH,UCH,QR,QV,QDL,BFIELD,BOVO,QFAZE,RFIELD,  
RVOXB,NWIRE)
```

где

IFL - управляющая переменная. Перед обращением к программе моделирования камеры эта переменная содержит нуль. Если пересечения траектории с камерой нет, то эту переменную следует оставить без изменения. В зависимости от ситуации можно выдать в этой переменной другие два значения: 1 - есть номер проволочки, и на этом перемещении другие проволочки не сработали, 2 - есть номер проволочки, но требуется ещё обратиться к этой же программе с теми же параметрами и она выдаст ещё номер сработавшей проволочки. В последнем случае повторное обращение производится с IFL=2 и не предполагается, что сработавших проволочек всего две - цикл обращений может продолжиться под управлением IFL по тем же правилам:

0 - нет проволочки

1 - есть одна проволочка и больше не будет

2 - есть проволочка и надо ещё обратиться к программе за номером следующей сработавшей проволочки на этом

же элементарном перемещении.

ICH - номер камеры (использовать необязательно)

QCH(UCH) - описание камеры (в подпрограмме следует писать реальную размерность массива, например, QCH(10), если камера описывается 10 параметрами),

Далее все вектора заданы в локальной системе координат.

QR - вектор положения частицы

QV - вектор направления скорости частицы (единичный)

QDL - величина перемещения частицы (см)

BFIELD - абсолютное значение индукции магнитного поля (кГс) в точке, где находится частица. Если BFIELD=0 или заряд частицы равен нулю, то остальные входные параметры не определены.

BOVO - косинус угла между напряженностью поля и скоростью частицы,

QFAZE - множитель перед пройденным расстоянием для получения приращения фазы винтовой линии,

RFIELD(3) - единичный вектор направления магнитного поля,

RVOXB(3) - вектор, равный векторному произведению вектора направления скорости на единичный вектор вдоль вектора магнитной индукции,

NWIRE - порядковый номер в камере (1,2,...) сработавшей проволочки.

Информация о положении частицы в /UMB012/UTYP,..., QR(3), QV(3),..., BET,GAM; о точке, в которую перемещается частица, в /UMB013/..., QRF(3), QVF(3),..., BETF,GAMF и о величине перемещения в /UMB014/QDL.

3.10. Генерация амплитуды с проволочки

Программа моделирования амплитуды с проволочки может генерировать сразу несколько аналоговых характеристик срабатывания проволочки, причём эти характеристики не обязательно описывают величину сигнала, а например, время срабатывания, координату вдоль проволочки и т.д. Весь этот комплект аналоговых сигналов называется здесь "амплитудой". Перед началом события все амплитуды зануляются и далее программы генерации амплитуд должны модифицировать эти величины при каждом срабатывании данной проволочки от последовательно пролетающих около неё частиц.

Хронологический порядок моделирования частиц не гарантируется основной программой, но каждая частица в блоке /UMB012/ с переменной TIME характеризуется временем (нсек) от начала события, что можно использовать при моделировании амплитуды.

Как правило, алгоритм генерации амплитуды тесно связан с алгоритмом определения пересечения траектории с камерой (программа моделирования камеры). Поэтому во многих случаях представляется удобным писать единую программу для моделирования пересечения траектории с камерой и амплитуды, обеспечивая второй вход в программу оператором ENTRY, или вводить дополнительные общие блоки для связи между этими двумя подпрограммами.

Обращение к программе:

```
CALL XXXXXX(AMP,NA,ICN,NWIRE)
```

где

AMP,NA - комплект аналоговых величин, входящих в одну амплитуду,

ICN - номер камеры (использовать не обязательно);

NWIRE - номер сработавшей проволоки в ней, выданной программой моделирования камеры.

Характеристики частицы в начале и конце перемещения следует брать из общих блоков /UMB012/ и /UMB013/. Описание камеры, если требуется, находится в /UMB004/.

3.11. Генерация амплитуды в счётчике

Вызов программы:

```
CALL XXXXXX(AMP,UNAM)
```

где AMP(UNAM) - массив аналоговых величин, составляющих одну амплитудный комплект.

Перед обращением элементы массива определены: в начале события весь массив зануляется, затем при каждом обращении к программе генерации амплитуды она должна модифицировать элементы массива, используя их прежние значения. При модификации можно использовать характеристики очередного перемещения частицы в /UMB012/, /UMB013/, /UMB014/.

3.12. Программы вычисления параметров события

Все параметры программы представляют собой целые числа - индексы в массиве PAR в общем блоке

/GIST18/ PAR(2)

Индексы формируются головной программой и менять их не следует - испортятся константы в головной программе.

Пример оформления программы OMEG, описанной во входном потоке следующим образом:

PARAMETR: OMEG(DTET,DFI;OM,AM)

Возможный текст программы:

```
SUBROUTINE OMEG(IN1,IN2,IOUT)
COMMON/GIST18/PAR(2)
PAR(IOUT)=SQRT(PAR(IN1)**2 + PAR(IN2)**2)
PAR(IOUT+1)=ABS(PAR(IN1))
IF(PAR(IOUT+1).LT.ABS(PAR(IN2))) PAR(IOUT+1)=ABS(PAR(IN2))
RETURN
END
```

3.13. Программы, анализирующие класс события по информации из выделенного блока

Эти программы вызываются в точке NEXTSURV, если во входном потоке указан выделенный блок. Тогда при моделировании события в первую очередь анализируются частицы, рождённые внутри этого блока, затем вызываются программы в точке NEXTSURV, которые могут изменить порядок работы головной программы (завершить анализ данного события), затем, если принято решение не прерывать событие, продолжается анализ остальных частиц. Нормальный порядок работы головной программы изменяется с помощью меток 200 и 600 в головной программе UM1000 следующим образом.

Во входном потоке данных программу, анализирующую класс события (например, название программы SUBTST) следует описать следующим образом:

••ПРОГРАММЫ

NEXTSURV: SUBTST(*200,*600)

Сама программа SUBTST на языке FORTRAN-77 должна выглядеть

следующим образом:

```
SUBROUTINE SUBTST(*,*)
.....
RETURN
.....
RETURN 1
.....
RETURN 2
END
```

Оператор RETURN соответствует нормальному развитию события, RETURN 1 - переход на начало следующего события, RETURN 2 - переход на обработку конца текущего события. Более детально использование меток 200 и 600 можно изучить по тексту головной программы UM1000, оставшейся на диске D по окончании счёта.

3.14. Моделирование дискриминаторов амплитуд

Программа моделирования дискриминатора должна иметь следующие параметры:

```
SUBROUTINE XXXXXX(URES,UINTOT,UINDEX,PTHR1,PTHR2)
```

и использовать общие блоки:

```
/UMB020/ AMPVA(2)
```

```
/UMB041/ ULISD(2)
```

UINTOT равно полному количеству входных амплитуд, UINDEX - номер первого индекса входной амплитуды в общем блоке UMB041. Таким образом, входными амплитудами являются:

```
AMPVA(ULISD(UINDEX)), AMPVA(ULISD(UINDEX+1)), ..., AMPVA(ULISD
(UINDEX +UINTOT-1))
```

Входными параметрами также являются два порога PTHR1 и PTHR2, несмотря на то, что программа моделирования может их не использовать или использовать один порог.

Единственным выходным параметром является URES, который может принимать значения 0 или 1. 1 означает, что дискриминатор сработал.

3.15. Моделирование блоков логической электроники

Программа моделирования логического блока электроники должна иметь следующие параметры:

```
SUBROUTINE XXXXXX(URES,UINTOT,UINDEX)
```

и использовать общие блоки:

```
/UMB039/NDISCR, UDISCR(10,2)
```

```
/UMB042/NLOG, ULOGS(6,2)
```

```
/UMB044/ULISBL(2,2)
```

На вход логического блока подаётся UINTOT логических импульсов со значениями 0 или 1. UINDEX - индекс описания первого входа в блоке /UMB044/. Следующие входы описаны сразу же вслед за первым. На примере первого входа покажем, как определяется значение 0 или 1 на входе.

ULISBL(1,UINDEX) = 1. Это означает, что входное значение импульса равно UDISCR(3,ULISBL(2,UINDEX))

ULISBL(1,UINDEX) = 2. Это означает, что входное значение импульса равно ULOGS(4,ULISBL(2,UINDEX))

Выходная переменная URES может принимать значения 0 или 1 в зависимости от значений входных импульсов и функции логического блока.

ЛИТЕРАТУРА

1. *С.И.Долинский, В.П.Дружинин и др.* Обзор e^+e^- - экспериментов с нейтральным детектором на ВЭПП-2М.
Часть 1. Препринт ИЯФ СО АН СССР 89-68, Новосибирск, 1989.
Часть 2. Препринт ИЯФ СО АН СССР 89-104, Новосибирск, 1989.
2. *С.Е.Бару, А.Е.Блинов и др.* Детектор МД-1. Препринт ИЯФ СО АН СССР 83-39, Новосибирск, 1983.
3. *А.Д.Букин и др.* UNIMOD - универсальная программа моделирования экспериментов на встречных e^+e^- -пучках. Препринт ИЯФ СО АН СССР 84-33 (1983).
- 4 *А.Д.Букин и др.* UNIMOD2 - универсальная программа моделирования экспериментов на встречных e^+e^- -пучках.
Часть 1. Общее описание. Препринт ИЯФ СО АН СССР 90-93, Новосибирск, 1990.
- 5 *А.Д.Букин и др.* UNIMOD2 - универсальная программа моделирования экспериментов на встречных e^+e^- -пучках.
Часть 2. Руководство пользователя. Препринт ИЯФ СО АН СССР 90-95, Новосибирск, 1990.
6. *В.И.Поталов, А.В.Денщиков, О.Б.Шкода.* Работа в системе виртуальных машин ЕС. М.: Финансы и статистика, 1988.
7. Фортран 77 ЕС ЭВМ / *З.С.Брич, О.Н. Гулецкая, Д.В. Капилевич и др.* - М.: Финансы и статистика, 1989.- 351 с.

А.Д.Букин, Н.А.Грозина, М.С.Дубровин, И.Л.Кац,
В.Н.Иванченко, В.А.Таюрский, С.И.Эйдельман

UNIMOD2 - УНИВЕРСАЛЬНАЯ ПРОГРАММА МОДЕЛИРОВАНИЯ
ЭКСПЕРИМЕНТОВ НА ВСТРЕЧНЫХ e^+e^- -ПУЧКАХ
Часть 3. Руководство пользователя-программиста

Препринт
90-96

Работа поступила - 16 августа 1990 г.

Ответственный за выпуск - С.Г.Попов

Подписано к печати 16 августа 1990 г.

Формат бумаги 60x90 1/16 Усл.2,7 печ.л., 2,2 учётно-изд.л.

Тираж 290 экз. Бесплатно. Заказ № 96

Ротапринт ИЯФ СО АН СССР, г.Новосибирск, 90