

Ситник И.М. и др.
Б2-10-84-715.

Ц 8408

С-412



ОБЪЕДИНЕННЫЙ ИНСТИТУТ ЯДЕРНЫХ ИССЛЕДОВАНИЙ

Ц 8408

955/85

Б2-10-84-715

ДЕПОНИРОВАННАЯ ПУБЛИКАЦИЯ

Дубна 19 84г.

ОБЪЕДИНЁННЫЙ ИНСТИТУТ ЯДЕРНЫХ ИССЛЕДОВАНИЙ

Лаборатория высоких энергий

БЭ-10-84-715

И.М.Ситник , Е.А.Строковский , П.Темников^{х)}

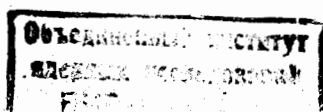
Ц 840.6
С-412

СИСТЕМА ГИСТОГРАММИРОВАНИЯ "НОРА"

х) ИЯИЯЭ БАН, София

Дубна, 1984 г.

6 12 84



АННОТАЦИЯ.

Описывается комплекс программ, позволяющий накапливать, обрабатывать и представлять на АЦПУ одномерные и двумерные распределения.

Обращение к распределениям возможно как по номерам, так и по именам.

Предусмотрено два режима представления распределений: на широкий (128) и узкий (80) лист.

Для одномерных распределений реализовано представление на плоттере, позволяющее изготавливать оригиналы рисунков для публикаций в физических журналах (за основу взят стандарт *Physics Letters*).

Подпрограммы написаны на Фортране. Комплекс программ поставлен на ЭВМ БЭСМ-6 и ЕС-1055.

По количеству стандартных услуг предлагаемая система не уступает известной системе HBOOK, однако потребляет памяти примерно в 4 раза меньше.

Работа выполнена в Лаборатории высоких энергий.

I. ВВЕДЕНИЕ

Первая версия системы "НОРА" (накапливание и обработка распределений) была создана в начале 1976 года, при подготовке первых данных с установки "АЛЬФА" к Международной конференции в Тбилиси^{/1/}. Сообщение о существовании этой системы было сделано на Дубненском международном семинаре 1977 года^{/2/}.

В тот момент авторы решили не делать подробного сообщения о системе "НОРА", т.к. на том же семинаре дубненцы познакомились с системой "НВООК"^{/3/}, которая заметно превосходила систему "НОРА" по количеству услуг.

Новый толчок развитию системы "НОРА" дала обработка последних данных с установки "АЛЬФА"^{/4/}, которая потребовала такого количества стандартных услуг, которых (в нужном качестве) не могла дать ни одна из упомянутых систем.

Сравнивая сегодняшние версии "НОРА" и "НВООК", необходимо отметить важнейшее преимущество первой из систем, которое состоит в меньшем потреблении памяти ЭВМ, что касается объема услуг то (на, возможно, субъективный взгляд авторов) "НОРА", уступая по части представления распределений на АЦПУ, имеет заметное преимущество при выполнении операций над распределениями. Необходимо отметить также в качестве преимущества тот факт, что к распределениям можно обращаться как по номерам, так и по символическим именам, что значительно облегчает труд при написании программ, содержащих много арифметических операций над распределениями. Облегчается также чтение забытых или чужих программ. Не последнее место уделено вопросу экономии бумаги (однако не в ущерб наглядности). Так,

- а) полностью пустые гистограммы не печатаются;
- б) левые и правые нулевые каналы гистограмм не печатаются;
- в) если гистограмма полностью умещается на остаток листа, перелистывание не происходит (по желанию можно всегда начать печать распределений с нового листа).

2. ОБЩИЕ СВЕДЕНИЯ О СИСТЕМЕ

Система накапливает, представляет и обрабатывает одномерные и двумерные распределения.

Авторы стремились создать систему, позволяющую максимально оптимизировать обработку электронных экспериментов.

1. Будем называть гистограммой область памяти ЭВМ, отведенную для хранения того или иного распределения. Накопление распределений производится по следующему правилу: номер канала гистограммы, к которому добавляется величина W (вес), вычисляется по формуле

$$I = (X - B) / S + 1 \quad (I)$$

где X - гистограммируемая величина, величины B , S , а также количество каналов L в гистограмме будем называть параметрами гистограммы. Для двумерных распределений параметрами являются BX , SX , BY , SY , LX , LY . Упоминание этих величин в дальнейшем будет делаться без пояснений.

2. Имена всех подпрограмм начинаются с буквы H (для одномерных распределений) или с буквы S (для двумерных). Для подпрограмм-функций это не всегда так.

3. Одномерные распределения хранятся в
`COMMON/HIST/HWORDS(1000)`

двумерные - в `COMMON/SCAT/SWORDS(3000)`

эти размеры `COMMON` - блоков декларированы внутри системы; если они достаточны для распределений пользователя, нет необходимости упоминать эти блоки в программе пользователя. В противном случае требуемую длину коммон-блоков нужно декларировать в головной программе.

4. При описании подпрограмм для обозначения гистограмм используются идентификаторы:

1) NX , NY , NZ ;

2) NS , NE .

При написании программ они могут иметь следующую форму:

1) целое больше 0 (номер под которым гистограмма хранится в системе);

2) холерическое выражение ("образ"), по которому нужная гистограмма отыскивается путем сравнения "образа" с первыми шестью

символами заголовка гистограммы (с четырьмя для ЕС I055). "Образ" может иметь от одного до шести (четырёх) символов. Сравнение происходит только в ~~тех~~^{тех} позициях, в которых символы "образа" отличны от пробела. На БЭСМ-6 этот поиск осуществляется с помощью системной подпрограммы JSEART А.Сапожникова (см. *LIBRA:23). Идентификаторы NX, NY, NZ имеют следующий смысл: NZ - гистограмма в которую помещается результат какого-либо преобразования над гистограммами NX, NY. Все идентификаторы группы I) могут иметь значение 0 или отрицательного целого. Это понимается системой как шаг по номеру гистограммы (положительный) относительно номеров гистограмм, упоминавшихся последними. Параметры с обозначениями 2) упоминаются в подпрограммах массовой печати, "очистки" и исключения гистограмм. Их надо понимать как "от NS до NE" . Эти параметры могут иметь вид целого положительного числа или холерического выражения. Если параметр NE описан холерическим выражением 'EQ' или 'NE', обслуживаются все гистограммы узнанные по образу NS (или все, кроме узнанных). Если нужно обслужить гистограммы с I-той по последнюю, в качестве параметра NE можно использовать заведомо большое число или холерическое выражение, которое не даст совпадения ни с одним из заголовков.

5. Написание параметров, которые обозначают числа, начинаются в нижеследующем тексте с буквы I, если подразумевается целое, и с буквы R, если - реальное. Если первые два символа IR, то параметр может быть как целым, так и реальным числом. Если в описании какой-либо подпрограммы параметр, начинающийся с символов IR, комментируется как целый, это означает, что в случае передачи подпрограмме реального числа из неё получается целое по формуле (I).

6. Написание параметров, определяющих тот или иной вид работы подпрограммы, начинается с буквы J. Это всегда холерические переменные, имеющие длины от одного до четырёх символов. На БЭСМ-6 можно использовать при этом тот минимум левых символов, которого достаточно для того, чтобы отличить нужную модификацию работы от всех других (за редким исключением достаточно бывает одного символа). На ЭВМ типа ЕС как адреса гистограмм

(если они задаются холерикой) так и указатели режима работы подпрограммы должны всегда иметь четыре символа. Если можно (а иногда и необходимо) обойтись меньшим числом символов, после последнего значащего символа нужно написать столько пробелов, чтобы в сумме получилось четыре символа.

7. Если система обнаруживает ошибки в программе пользователя, то а) при заказе и накоплении происходит выброс задачи и диагностика;

б) в остальных случаях диагностика и отказ от выполнения ошибочной инструкции.

3. ОПИСАНИЕ ПОДПРОГРАММ

3.1 Подпрограммы заказа

CALL HOPEN(LWORD, NMAX)

CALL SOPEN(LWORD, NMAX)

Читают заказ гистограмм, сформированный на картах в той части пакета задачи, которая читается операторами READ. Чтение осуществляется в свободном формате. Разделителем считается запятая.

Группа карт заказа должна начинаться с карты, содержащий произвольный комментарий и отделённое от него запятой число, указывающее количество гистограмм, которые будут заказаны с карт и (или) образованы в процессе счёта. Это число может превосходить фактическое количество гистограмм, которое будет использовано. Пользователю полезно знать, что служебная часть массивов HWORDS, SWORDS после заказа будет иметь длину

$$LSERV = 10 + 6 * N \quad (2)$$

где N - число гистограмм, заказанное в первой карте, так что использование разумной величины этого числа позволяет экономить память ЭВМ.

Номера, которые присваиваются гистограммам, совпадают с порядковыми номерами карт заказа.

Карта заказа одной гистограммы имеет следующий вид:

заголовок, B, S, L

для одномерных распределений;

заголовок, VX, SX, VY, SY, LX, LY

для двумерных. Система запомнит первые тридцать символов указанного в карте заголовка (первые 20 для ЭВМ типа ЕС). Первые шесть символов заголовка (четыре на ЕС-1055) используются при поиске по "образу", остальные полезны только при выводе распределения на АЦПУ. Заголовок не должен начинаться с последовательности символов 'READ', 'SCRE', 'MOFF'

и более, чем с трёх пробелов подряд. LY можно задать положительным или отрицательным числом. Если $LY > 0$, для двумерной гистограммы отводится в памяти два байта на один канал. Такую гистограмму будем называть целой. При $LY < 0$ одному каналу отводится одно слово памяти (заказывается при этом $IABS(LY)$ каналов по Y). Такую гистограмму будем называть реальной.

В целях экономии памяти предпочтительнее пользоваться вариантом $LY > 0$; однако при этом следует помнить, что содержимое канала не должно превосходить 32767 (при накоплении распределений переполнение каналов не контролируется).

Y - каналов можно заказать не более 54. Если заказывается целая гистограмма, то LY округляется в сторону увеличения до кратного 3-м (2-м для ЕС).

Предусмотрена возможность для гистограмм, параметры которых будут определены в результате операций над другими гистограммами, заказывать только заголовки. Для такой гистограммы после заказа будет отведен только номер. Место в памяти будет отведено при засылке результата в эту гистограмму.

Карты заказа заголовков должны следовать после последней карты заказа гистограммы с параметрами.

Статус гистограмм, для которых параметры не заказаны в программах `НОРЕН`, `СОРЕН`, отличен от статуса полностью заказанных гистограмм (фиксированных). Подробно об этом в главе "Операции над распределениями". После последней карты заказа должна следовать карта - признак конца заказа '****'.

При первом обращении к подпрограммам `НОРЕН`, `СОРЕН` параметры `NMAX`, должны иметь значение 0. Параметры `LWORD` должны указывать длины соответствующих коммон-блоков (`/HIST/` или `/SCAT/`). Если эти коммон-блоки не упоминались в программе пользователя, то `LWORD` - любое положительное целое число, не превосходящее 1000 для `НОРЕН` и 3000 для `СОРЕН`.

В конце своей работы подпрограммы `НОРЕН`, `СОРЕН` печатают сводку о принятом заказе, включая таблицу распределения памяти. Поскольку возможно образование гистограмм, минуя подпрограммы `НОРЕН`, `СОРЕН`, предусмотрена печать сводки без осуществления заказа гистограмм. Это происходит, если `LWORD=0`.

Дальнейшее описание работы подпрограмм `НОРЕН`, `СОРЕН` при первом чтении можно пропустить.

Предусмотрена возможность редактирования набора карт заказа, позволяющая держать в одном наборе карты заказа для всех вариантов счёта, а фактически заказывать только какое-то подмножество набора, необходимое для выбранного варианта счёта. Редактирование осуществляется расставлением между картами заказа карт, имеющих вид `READ,I`.

Если $I=0$, то все последующие карты заказа, вплоть до следующей карты READ, I или признака конца заказа игнорируются. Если $I \leq N$, где N - число уже заказанных гистограмм, то сделанные до этой карты заказы с номерами $J \geq I$ аннулируются, а заказы следующие после этой карты, начинают заполнять номера, начиная с I . Если $I > N$, то следующие после управляющей карты заказы заполняют номера, начиная с $(N+1)$, т.е. система следит за тем, чтобы разрыва в номерах при заказе не образовалось.

При повторном обращении к подпрограммам HOPEN, SOPEN заказ будет производиться начиная с номеров, оставшихся свободными после предыдущего заказа. Такую ситуацию легко изменить, пользуясь управляющими картами READ, I . Естественно, что для повторных обращений к этим подпрограммам необходимо предусмотреть новые пакеты заказа.

Повторными заказами можно оперировать при переопределении области памяти, доступной для системы, и максимального количества номеров. При этом выход на чтение карт не происходит, если второй параметр ($NMAX$) отличен от нуля. Параметр $NMAX$ в этом случае определяет максимальное количество номеров, которые может использовать система. Таким переопределением удобно, например, воспользоваться, когда вслед за коммон-блоком / HIST / (или / SCAT /) располагается часть памяти, которая занята во время накопления распределений и освобождается, когда необходимо проделать операции над распределениями с образованием новых гистограмм (такой областью памяти может быть, например, буфер для чтения ленты).

Часто возникает ситуация, когда способом обмена информацией между разными задачами оказывается листинг, который напечатала одна задача, и эта информация нужна в качестве входных данных для другой задачи. Таблицу данных можно ввести в гистограммы внутри пакета заказа гистограмм, если сразу после карты заказа какой-либо гистограммы поместить карты с числовым материалом. Признаком карты с числовым материалом является отсутствие перед первым разделителем (запятой) каких-либо

символов, кроме пробелов. Числа задаются в свободном формате, разделителем является запятая. Предполагается, что карта с числовым материалом в одномерном случае содержит информацию о десяти каналах распределения, а в двумерном - для всех Y-каналов, соответствующих I-тому X-каналу, где I - порядковый номер карты с числовым материалом. Видно, что возможности такого способа ввода в двумерном случае не беспредельны, т.к. трудно на одной карте разместить, например, 54 числа. Несколько облегчить эту ситуацию помогают следующие средства:

- 1) Правые нули можно не печатать на карте;
- 2) Если два разделителя следуют подряд, то считается, что между ними было число 0;
- 3) Несколько нулей подряд можно задать конструкцией $N'Z'$ где N - число нулей;
- 4) Несколько одинаковых чисел подряд можно задать конструкцией $N*'R$, где R - число, а N - количество повторений;
- 5) Если карта оканчивается символом $\$$, то следующая карта считается продолжением предыдущей.

Приведем пример двух аналогичных заказов с числовым материалом

а) , I, ,2,5Z ,II,I5.5,2 * 3.I4

б) , I, 0, 2, 0, 0, 0, 0, II, I5.5, 5, 3.I4 \$
3.I4, 0, 0

В подпрограммах печати реализуется вывод на лист шириной I28 позиций (по умолчанию). Предусмотрено ограничение до 80 позиций (им удобно пользоваться при просмотре распределений на терминале). Для этого надо в пакет карт заказа в качестве второй карты вставить карту 'SCRE'.

Если в задаче имеется более одного пакета заказов, такую карту надо поместить в первом из них. Для двумерных гистограмм количество каналов по Y автоматически обрежется до 30.

Подпрограммы операций над распределениями печатают одну строку сообщения о своей работе (MESSAGE). Запретить эту печать можно, вставив карту 'MOFF' в любом месте после первой карты и раньше признака конца заказа.

3.2 Подпрограммы "очистки"

CALL HCLEAR(NS,NE)

CALL SCLEAR(NS,NE)

Производят обнуление гистограмм.

Примеры:

а) CALL HCLEAR(5,10) -

очистить гистограммы с 5-ой по 10-тую

б) CALL HCLEAR(7,7) -

очистить 7-ую гистограмму;

в) CALL HCLEAR('IMP',10000) -

очистить гистограммы, начиная с той, которая найдена по "образу" 'IMP' и кончая последней;

г) CALL HCLEAR('P','EQ') -

очистить все гистограммы, заголовки которых начинаются с буквы P .

д) CALL HCLEAR('S','NE') -

очистить все гистограммы, вторая буква заголовка которых не S .

Аналогично указанным примерам можно поступать при обращении ко всем подпрограммам, параметры которых в описании обозначены через NS , NE .

3.3 Подпрограммы накопления

CALL HACCUM(NX, X, W)

CALL SACCUM(NX, X, Y, W)

используются для накопления гистограмм (см. формулу I). В программе SACCUM величина w округляется до ближайшего целого, если гистограмма NX целая.

Если значение параметра B (BX или BY для двумерного случая) превосходит $1. E+8$, считается, что этот параметр необходимо настроить. Параметр B находится из анализа величины X при первых K

обращениях к подпрограмме накопления, где $K = L$ для одномерных гистограмм, $K = LX * LY$ для двумерных реальных, $K = LX * LY / J$ для двумерных целых гистограмм (где $J = 3$ для БЭСМ-6 и $J = 2$ для ЕС-1055), но во всяком случае не превосходит 100. Вычисленное значение B округляется до величины $B = N * S$, где N - целое.

Если нужно настроить параметр s , необходимо задать его начальное значение в интервале $0. > s > -1.$. Среднее распределения при этом совмещается с каналом гистограммы $L * (-s)$. Найденная величина S округляется до двух знаков, если старший знак 1; до 0 или 5 во втором знаке, если старший знак 2 - 4; до одного знака в остальных случаях.

ПРЕДУПРЕЖДЕНИЕ:

- а) Настройка параметра S допустима только в том случае, когда величины X заведомо превышают величину B (как следствие, нельзя одновременно объявлять ненастроенными обе величины B и s).
- б) В режиме настройки первые K событий гистограммируются с весом 1., независимо от значения параметра w .

С помощью подпрограммы

```
CALL HVEST(NX, X, L, NSH)
```

можно провести гистограммирование L - компонентного вектора X , где NSH - шаг по номеру гистограммы. Величина W (см. формулу (I)) полагается равной 1. Если X превосходит $1.E+8$, гистограммирование не производится. Последним обстоятельством удобно воспользоваться, когда заранее неизвестно, будет ли определена величина, подлежащая гистограммированию при обработке очередного события. Например, если перед началом обработки события весь массив X заполнить числами $1.E+8$, то часть массива, которая в процессе обработки события будет переопределена, при обращении `CALL HVEST...` пройдет процедуру гистограммирования, а остальная нет.

Пример : `DIMENSION X(50)`

```
CALL HVEST(1, X, 50, 2)
```

С помощью этой процедуры будет проведено гистограммирование пятидесяти величин из массива X в гистограммы: $X(1)$ - в 1-ю, $X(2)$ - в 3-ю, $X(3)$ - в 5-ю и т.д.

Отметим, что `HVEST` дает выигрыш в скорости по сравнению с `HASSUM`

К скорости работы подпрограммы накопления предъявляются высокие требования, поэтому, если параметр NX задан в колерическом виде, то поиск номера гистограммы по таблице осуществляется только при первом обращении к подпрограмме, после чего параметр NX "портится", в него заносится номер гистограммы.

3.4 Подпрограммы представления распределений на АЦПУ

Подпрограмма CALL HPRIN(NS,NE,JP)

осуществляет печать одномерных гистограмм. Параметр JP определяет способ печати:

JP=4HTABL - печать таблицей;

JP=4HGRAF - печать графиком (оггибающая гистограммы). Для любых других значений JP - печать "столбиком".

Подпрограмма

CALL HBIPRI(NX,NY,JP,L)

печатает пару гистограмм NX, NY способом, который определяется заданием JP:

а) JP=4HBILI - печать гистограмм одна под другой;

б) JP=4HOVER - гистограммы печатаются столбиком в одной рамке, каждая своим символом, область перекрытия печатается третьим символом;

в) JP=4HGRAF - то же что б), но печать графиком (оггибающей);

г) JP=3HERR - предполагается, что гистограмма NX содержит значения экспериментальных точек, а гистограмма NY - соответствующих им ошибок. Печать осуществляется в виде точек с "усами".

Параметром L определяется масштаб печати:

L = 0 - линейный масштаб,

L = 1 - лог. масштаб, 10 позиций на порядок,

L = 2 - лог. масштаб, 20 позиций на порядок, и т.д.

Пример:

отпечатать гистограммы, имеющие образы 'DCS', 'ERR' в виде точек с ошибками в лог. масштабе:

```
CALL HBIPRI('DCS','ERR','ERR',1)
```

или

```
CALL HBIPRI('DCS','ERR','E',1)
```

При работе подпрограмм HPRIN, HBIPRI вычисляются и печатаются параметры распределений среднее, сигма, мода (полезна при распределении типа рэлеевского) в двух приближениях:

первое - по всем каналам гистограммы;

второе - в окрестности максимума распределения; при этом считается, что окрестность максимума закончилась, если имеется

два подряд канала, содержащих величины, не превосходящие 0.135 от максимума. Для гауссова распределения этот уровень отсечки соответствует взятию в учет каналов, содержащихся в области +/- два сигма от максимума. После вычисления по стандартным формулам, параметры сигма, мода корректируются таким образом, чтобы для точного гауссова (рэлеевского) распределения они имели правильное значение.

Извлечь значения параметров распределений (вычисленных в первом или втором приближении) в программу пользователя можно с помощью подпрограмм-функций

A1 = HMEAN1(NX),

A2 = HMEAN2(NX),

A3 = HSIGM1(NX),

A4 = HSIGM2(NX),

A5 = HMODA1(NX),

A6 = HMODA2(NX),

A7 = HEVEN1(NX)

- число событий в распределении,

A8 = HEVEN2(NX)

- число событий в окрестности

максимума,

A9 = HMAXIM(NX)

- максимум распределения.

Подпрограмма

CALL SPRIN(NS,NE,JP)

печатает двумерные распределения таблицей чисел, если

JP=1HT,

символами при любых других значениях JP. Последовательность символов: 1-9, далее латинский алфавит в котором пропущены буквы I, J, всего 20 символов.

Количество знаков в числе внутри таблицы зависит от числа заказанных Y-каналов (чем больше каналов, тем меньше знаков), множитель IO**N печатается вне таблицы.

Если число каналов превосходит 36, то печать идет символами, независимо от задания JP.

При обращении к подпрограмме SPRIN вычисляется и печатается корреляция между двумя переменными в линейном приближении

$$Y=A+B*X$$

$$X=C+D*Y$$

Параметры корреляции можно извлечь в программу пользователя с помощью подпрограммы

$$A=SYFXO(NX),$$

$$B=SYFX1(NX),$$

$$C=SXFYO(NX),$$

$$D=SXFY1(NX).$$

Содержимое и максимум распределения извлекается с помощью

$$R=SEVEN1(NX),$$

$$R=SMAXIM(NX).$$

Подпрограммы HPRIN, SPRIN используют для печати 128 позиций листа (по умолчанию). Можно заказать печать в пределах 80 позиций (см. описание HOREN, SOREN). Подпрограмма HBIPRI для печати в пределах 80 позиций не приспособлена.

3.5 Управление перелистыванием.

Если не проявлять никакой заботы о перелистывании, то устанавливается следующий режим: печать первого из распределений, подлежащих выводу, делается с начала листа, все последующие печатаются на остаток очередного листа, если они умещаются на нём целиком, в противном случае с нового листа.

После обращения

CALL HISLIS(0)

очередная печать делается с начала листа и далее в том режиме, который описан выше. Отметим, что чередование печати распределений и какой-либо другой печати сбивает этот режим. Поэтому после "своей" печати пользователю рекомендуется делать обращение

CALL HISLIS(0) ,

или перед "своей" печатью делать обращение

CALL HISLIS(L) ,

где L - количество строк выводимой информации.

После обращения

CALL HISLIS(L) ,

где L - любое отрицательное число, распределения печатаются без всяких перелистываний вплоть до обращения к HISLIS , меняющего этот режим.

3.6 Представление распределений на плоттере .

Набор описываемых ниже подпрограмм использует в качестве подпрограмм нижнего уровня систему "ГРАФОР"/5/, в частности PAGE , MOVE , SYMBOL , MARKER и подпрограммы, которые они вызывают. Этот набор позволяет представлять на плоттере только одномерные распределения. Стандарт построения графиков соответствует стандарту, принятому в физических журналах.

При описании подпрограмм будем подразумевать , что параметры, описанные которых содержит последовательность символов "FIG" , нужно задавать в см (сантиметрах).

Работа с плоттером начинается с обращения

```
CALL HPLINI(XMIN,XMAX,XFIG,YMIN,YMAX,YFIG,LOG)
```

здесь

XMIN,XMAX,YMIN,YMAX определяют область графика по X, Y осям.

XFIG , YFIG определяют размер картинки; эти величины не могут превосходить (34.5, 26.).

LOG=3NLGX,3NLGY,4NLGXU , если графики нужно строить в лог. масштабе по оси X, Y, или по обоим осям. При любом другом значении LOG график строится линейным по обоим осям.

Работу с плоттером можно начать также с обращения

```
CALL HPLIN1(NX,XFIG,YMIN,YMAX,YFIG,LOG) ;
```

в этом случае XMIN , YMIN будут выбраны в соответствии с параметрами гистограммы NX . Остальные параметры имеют тот же смысл, что и в HPLINI .

При обращении к HPLINI или к HPLIN1 левый нижний и правый верхний угол будущего графика определяются таким образом, чтобы можно было сделать цифровую разметку осей и буквенные надписи слева и снизу от графика. При желании границы графика можно переопределить обращением

```
CALL HPLZER(XFIG1,XFIG2,YFIG1,YFIG2) ;
```

здесь

XFIG1 , YFIG1 определяют левый нижний угол графика

XFIG2 , YFIG2 - правый верхний. Если значение како-

го-либо из параметров задается отрицательным, то оно остается старым. Вычерчивание осей графика осуществляется обращением

```
CALL HPLAXE(XS,DX,LX,YS,DY,LY) ,
```

где:

XS - любое из значений X внутри ранее заданной через HPLINI области (XMIN , XMAX) которое нужно пометить крупным штрихом, остальные крупные штрихи наносятся вправо и (или) влево от заданного в соответствии со значением параметра

DX - расстояние между крупными штрихами;

LX - соотношение частот мелких и крупных штрихов.

если LX = 1, мелкие штрихи не наносятся.

YS , DY , LY имеют тот же смысл для оси Y.

Обращением

CALL HPLNUX(IP, IS, SCILE)

делается цифровая разметка шкалы X в F - формате. Надписи делаются возле крупных штрихов. Здесь

IP - число позиций для числа (не более 4-х),

IS - число знаков после запятой.

SCILE - масштабный коэффициент (должен быть задан числом, являющимся степенью 10.).

SCILE, отличный от единицы, используется, когда диапазон значений X таков, что возникают трудности при разметке осей в указанном формате.

Обращением

CALL HPLNUY(IP, IS, SCILE)

делается цифровая разметка шкалы Y, смысл параметров такой же, как при обращении к HPLNUX. Если для оси Y выбран лог. масштаб, значение параметров IP, IS несущественно.

Раздельное обращение к подпрограммам цифровых надписей на осях X, Y оставляет возможность строить несколько графиков один под другим, при этом надписывать ось X только нижнего графика.

К подпрограммам HPLNUX, HPLNUY нельзя обращаться раньше, чем к подпрограмме HPLAXE.

Нанесение экспериментальных точек на график осуществляется обращением

```
CALL HPLERR(NX, NY, ISYMB)
```

где

`NX` , `NY` - гистограммы, в которых содержатся значения экспериментальных точек и соответствующих им ошибок.
`ISYMB` - символ, которым изображается точка на графике (смотри описание подпрограммы `MARKER` системы "графор").
Если `ISYMB < 0` , экспериментальная точка представляется пересечением вертикальной линии, характеризующей ошибку, и горизонтальной линии длиной 2 мм.

Представление экспериментальных точек в переменных, отличных от тех, при которых накапливалась гистограмма, осуществляется обращением

```
CALL HPLERT(NX, NY, ISYMB, TRANSX, TRANSY)
```

где:

`TRANSX` , `TRANSY` - подпрограммы-функции, делающие необходимые преобразования, они должны быть описаны в операторе `EXTERNAL` . Для случая, когда необходимо преобразование только по одной из переменных, запасена подпрограмма-функция тождественного преобразования `TRANS1` .

Представить гистограмму "столбиком" можно обращением

```
CALL HPLHIS(NX) .
```

Нанесение функции на график осуществляется обращением
CALL HPLFUN(FUN,X1,X2, IDASH)

Здесь

FUN -подпрограмма-функция, которая должна быть описана в операторе EXTERNAL .

X1, X2 - нижняя и верхняя области аргумента для этой функции. Если функция строится от левой границы графика и (или) до его правой границы, X1 можно задать заведомо малым и (или) X2 - заведомо большим.

IDASH=1 для случая, когда график нужно нанести пунктиром. При любых других значениях IDASH график рисуется сплошной кривой.

Функцию, заданную таблично в гистограмме NX , можно построить обращением

CALL HPLGRF(NX, IDASH) .

При построении используется подпрограмма PARINT из ОБСП-1. Следует соблюдать осторожность при одновременном использовании ОБСП-1 и ОБСП-5, в которой находятся подпрограммы системы "графор". ОБСП-5 следует заказать раньше, чем ОБСП-1.

Сделать надписи (снизу для оси X и слева для оси Y) можно, обращаясь к подпрограммам

```
CALL HPLETX(IALF,ITEXT,INDEX,N) ,  
CALL HPLETY(IALF,ITEXT,INDEX,N) .
```

Здесь:

ITEXT - основная текстовая строка, которая задается латинско-русскими символами,

N - количество символов.

IALF - текстовая строка - указатель шрифта, которым нужно вывести основную строку. Если, например, символы с 1-го по N-й, нужно написать одним шрифтом, а далее другим, то в строке нужно поставить разметочные символы в позициях 1-й и (N + 1)-й. Используются следующие разметочные символы:

- L - латинские и русские заглавные буквы
- A - латинские и русские малые буквы (мнемоническое правило: LATINIC),
- G - греческие заглавные буквы,
- R - греческие малые буквы.

INDEX - текстовая строка - указатель индекса. Если в N-й позиции этой строки стоит символ U (up), или D (down), то соответствующий символ основной строки поднимается или опускается. Допускается выражение с наличием у основного символа как верхнего, так и нижнего индекса. Порядок, в котором указываются верхний и нижний индексы, несущественен; в любом случае при написании индексов отчет ведется от символа, расположенного на основной строке.

Для надписей в произвольном месте и произвольного размера с теми же правилами разметки следует обратиться к подпрограмме

F1PLOT

которую используют вышеназванные подпрограммы.

Эта подпрограмма имеется в *LIBRA:23 .

На одном листе можно построить любое число графиков функций и наборов экспериментальных точек. Цвет пера меняется с помощью подпрограммы из * LIBRA:23 .

CALL GETPEN(J)

где:

- J = 1 - черн.
- = 2 - зел.
- = 3 - фиол.
- = 4 - красн.
- = 5 - голуб.
- = 6 - корич.

При отсутствии обращения к этой подпрограмме, цвет пера всегда чёрный.

Заканчивать рисунок следует обращением

CALL ENDPG(I) .

Причём, если I отлично от 0, работа заканчивается надписью (см. "графор").

Работа с плоттером на БЭСМ-6 должна быть оговорена управляющей картой

* CALL PLOTTER:WX4675 .

3.7 Операции над распределениями

Общие сведения.

Любая из подпрограмм этого класса имеет как минимум, параметры

NZ - гистограмма результата операции,

NX - первый операнд, и, может быть,

NY - второй операнд.

Тип операции может быть указан как параметр или содержаться в названии подпрограммы. Остальные параметры - числа.

Каждая подпрограмма печатает строку сообщения о своей работе.

подавить эту печать можно, обратившись

CALL MESOFF

или предусмотрев такой запрет в пакете заказа (смотри описание HOPEN, SOPEN). Вновь разрешить такую печать можно, обратившись

CALL MESON

Параметры гистограммы NZ полностью определяются параметрами NX (или, в одном случае, NX и NY). Поэтому при обращении к подпрограммам HOPEN, SOPEN их можно заказывать произвольными (с некоторыми ограничениями, которые станут ясными из дальнейшего текста), либо не заказывать их вообще.

Гистограммы параметры которых определены полностью с помощью подпрограмм HOPEN, SOPEN, назовём фиксированными. Они располагаются в коммон-блоках, сразу после служебных массивов, длина которых определяется в (2). Остающаяся часть памяти коммон-блока назовём рабочим полем.

Гистограммы, для которых определены только заголовки, имеют номера, соответствующие порядковому номеру карты заказа. Гистограммы, которые не были заказаны вообще, в момент образования получают наименьший из свободных номеров. Использование явных номеров при образовании таких гистограмм недопустимо. Нефиксированные гистограммы располагаются на рабочем поле в том порядке, в каком они образуются. Так продолжается до тех пор, пока для очередной гистограммы отведенного места в памяти не хватает. Эта гистограмма разместится снова в начале рабочего поля, то есть, самая "старая" нефиксированная гистограмма (или группа таких гистограмм) затрётся. О случившемся делается сообщение, которое можно запретить той же инструкцией,

что и остальную печать этих подпрограмм (см. несколько абзацев выше).

Таким образом, нефиксированные гистограммы отличаются от фиксированных тем, что они могут потерять место в памяти.

При выполнении большого количества операций над гистограммами в условиях незначительного объема памяти избежать стихийной порчи гистограмм можно с помощью обращений

```
CALL HDELETE(NS,NE) ,
```

```
CALL SDELETE(NS,NE) ,
```

уничтожая те гистограммы, которые больше не понадобятся и тем самым отдавая порчу нужных гистограмм. Подпрограммы HDELETE , SDELETE уничтожают не только содержимое гистограмм, но и их номера. Номера гистограмм, которые превосходили номера уничтоженных гистограмм, уменьшаются на количество уничтоженных гистограмм. Гистограммы, которые занимали в памяти место позже уничтоженных гистограмм, переместятся на их место.

Если результат операции помещается в уже существующую гистограмму, а требуемое количество каналов меньше, чем в существующей гистограмме, то образовавшаяся "дырка" в памяти уничтожается за счёт сдвигки полей гистограмм, расположенных в памяти позже.

Если требуемое количество каналов больше, то нефиксированная гистограмма уничтожается на старом месте и образуется в конце рабочего поля (то есть "молодеет"). Попытка поместить в фиксированную гистограмму результат, превышающий её длину, считается ошибкой (выдается диагностика). Таким образом, ясно, что если уж гистограмма для результата заказывается в подпрограммах

HOPEN , SOPEN , то её длина не должна быть меньше длины результата.

Если для размещения результата не хватает всего рабочего поля , выдается диагностика "не хватает памяти". В этом случае в задаче нужно расширить коммон-блок.

Все вычисления ведутся с реальными числами, т.е. если операнд целый, то он превращается сначала в реальный; если результат должен быть целым, то после вычисления реальной величины происходит её округление до целой. Поясняя работу подпрограмм будем для содержимого каналов гистограмм-операндов использовать обозначения V, VX , VY ; для содержимого канала гистограммы-результата-W.

Подпрограммы

```
CALL HARI1 (NZ,NX,JP,IR)
CALL SARI1I(NZ,NX,JP,IR)
CALL SARI1R(NZ,NX,JP,IR)
```

осуществляют трансформацию гистограммы NX . Коды операций и соответствующие им действия приведены в таблице I, индексы в тривиальных случаях не упоминаются.

Таблица I

JP	ТИП	IR	действие
+	R		$W=V+IR$
-	R		$W=V-IR$
*	R		$W=V*IR$
/	R		$W=V/IR$
**	I, R		$W=V**IR$, если IR реальн. то IR>0.
SQRT	R		$W=SQRT(V)*IR$
LN	R		$W=ALOG(V)*IR$
LG	R		$W=ALOG10(V)*IR$
GT	R		$W=0$, если $V < VMAX*IR$ $W=V$ в остальных случаях
NORM	R		$W=V*R/VMAX$
INTG	R		$W(I)=(V(1)+V(2)+...+V(I)) * IR$
PRES	I		$W(1)=V(1)+...+V(IR)$ $W(2)=V(IR+1)+...+V(IR+IR)$, $S(NZ) = S(NX)*IR$

Подпрограммы SARI1I , SARI1R отличаются типом результата. Если Вы обращаетесь к SARI1I это означает, что результат (после округления до целого) будет помещен в гистограмму целого типа.

Последние две операции (интегрирование и сжатие) для двумерных распределений выполняются вдоль оси X.

Для более сложной трансформации гистограммы NX следует воспользоваться подпрограммами

```
CALL HZFXU (NZ,NX,FUN1)
```

```
CALL SUFXYZ(NZ,NX,FUN2)
```

где: $FUN1$, $FUN2$ составляются пользователем и должны иметь вид

```
FUNCTION FUN1(X,DX,Y)
```

где Y

X передаётся середина I -того канала гистограммы NX , B
 $DX - S/2$, B

Y - содержимое I -того канала.

Значение функции заносится в I -тый канал результата;

```
FUNCTION FUN2(X,DX,Y,DY,Z)
```

где

X - середина I -того канала по X ,

Y - середина J -того канала по Y ,

$DX - SX/2$,

$DY - SY/2$,

Z - содержимое канала (I, J) .

Значение функции заносится в канал (I, J) результата. Параметрами DX , DY , удобно воспользоваться когда какую-то функцию нужно кусочно проинтегрировать или задать её значение в точке, отличной от середины канала.

Параметры гистограммы NZ совпадают с параметрами NX . В двумерном случае совпадают также и типы гистограмм NX , NZ (целый, реальный).

Пример 1.

Задача: заслать в гистограмму 'GAUS' нормальное распределение.

Решение:

```
      SUBROUTINE START
      EXTERNAL FGAUS
      CALL HZFX('GAUS','GAUS',FGAUS)
      RETURN
      END
      FUNCTION FGAUS(X,DX,Y)
COMMON/A/ C
      FGAUS=C*EXP(-X**2/2.)
      RETURN
      END
```

Пример 2.

Задача: В гистограмме 'EXP' - сечение, полученное в эксперименте. Получить из него инвариантное сечение.

Решение:

```
      SUBROUTINE PHYSIC
      EXTERNAL FINV
      CALL HZFX('INV','EXP',FINV)
      RETURN
      END
      FUNCTION FINV(P,DP,Y)
      E = SORT(P**2+ RM**2)
      FINV = Y*E/P**2
      RETURN
      END
```

Пример 3.

Задача: Определить N - ый момент распределения, находящегося в гистограмме IMP...

Решение:

```
      SUBROUTINE FIND
COMMON/A/ SUM,SUM1,N
      EXTERNAL FUN
      SUM=0.      * SUM1=0.
      CALL HZFX('IMP','IMP',FUN)
      RMEAN=SUM1/SUM
      RETURN * END
      FUNCTION FUN(X,DX,Y)
COMMON/A/ SUM,SUM1,N
      FUN = Y
      SUM = SUM+Y
      SUM1= SUM1+Y*X**N
      RETURN * END
```

Пояснение к примеру 3.

Как видно из условия задачи, гистограмма-результат нам в данном случае не нужна, но подпрограмма HZFX требует её наличия. Чтобы не заводить лишней гистограммы, мы переписываем гистограмму 'IMP' в себя с помощью оператора
FUN = Y .

Подпрограммы

CALL HARI2 (NZ, JZ, A, NX, JP, B, NY)

CALL SARI2I (NZ, JZ, A, NX, JP, B, NY)

CALL SARI2R (NZ, JZ, A, NX, JP, B, NY)

осуществляют поканальные операции между гистограммами NX, NY.

Если

JZ=1H=, результат помещается в NZ, если

JZ=1H+, результат добавляется к NZ.

Коды операций JP и действия приведены в таблице 2.

Таблица 2

JP	I	Действие
+	I	$W = A * VX + B * VY$
-	I	$W = A * VX - B * VY$
*	I	$W = A * VX * B * VY$
/	I	$W = A * VX / B * VY$, $X/0=0$
SQ	I	$W = A * VX ** 2 + B * VY ** 2$
AND	I	$W = A * VX$, если $VY > 0$; I $W = 0$. в остальных случаях ,
NE	I	$W = A * VX$, если $VY < 0$; I $W = 0$. в остальных случаях .

Параметры гистограмм NX, NY не обязательно должны быть идентичны. Требуется только совпадение параметров $S(NX)$, $S(NY)$. Параметры $B(NX)$, $B(NY)$ могут отличаться на величину, кратную S . Выполнение этих условий контролируется системой. Параметры $L(NX)$, $L(NY)$ могут быть любыми. После окончания работы подпрограммы параметры совпадают с параметрами NX. Заказанная операция выполняется для каждого канала NX и канала NY, имеющих совпадающие значения аргумента. Если соответствующего канала гистограмма NY не имеет, то считается, что его содержимое равно нулю.

Подпрограммы SARI2I, SARI2R отличаются типом результата (целый, реальный). Операции в обоих случаях могут быть целыми и реальными в любой комбинации.

Пример 2 (из жизни)

Пусть имеем две гистограммы с одинаковыми параметрами, в которых хранятся распределения по одной и той же переменной, накопленные при нескольких разных условиях. Каждая из гистограмм имеет 90 каналов, причём, известно, что в гистограмме 'TRIG1' распределение достоверно в первых 60-ти каналах, а в гистограмме 'TRIG2' - в каналах с 31-го по последний. Если распределение в 'TRIG1' разделить на число $RM1$, а в 'TRIG2' - на $RM2$, то получим сечения.

Очевидно, что для получения результата, достоверного во всех каналах, для каналов 1-30 нужно взять статистику только из 'TRIG1' для каналов 31-60 статистику из 'TRIG1' и 'TRIG2' надо объединить. А для каналов 61-90 взять статистику только из 'TRIG2'.

Приводим решение этой задачи.

```
SUBROUTINE DIFCROS
CALL HARI1('DCS1','TRIG1','/',RM1)
CALL HARI1('DCS2','TRIG2','/',RM2)
CALL HARI2('R','=',1.,'TRIG1','+',1.,'TRIG2')
RM=RM1+RM2
CALL HARI1('DCS12','R','/',RM)
CALL HGLUE('DIFCRO','DCS1','DCS12',30)
CALL HGLUE('DIFCRO','DIFCRO','DCS2',60)
RETURN
END
```

В гистограмме 'DIFCRO' мы имеем сечение, достоверное во всех каналах. Описанная процедура дала бы нам тот же результат, если бы каждая из гистограмм 'TRIG1', 'TRIG2' содержала только достоверные каналы, то есть

$$L('TRIG2')=L('TRIG1')=60,$$
$$(B('TRIG2')-B('TRIG1'))/S=30.$$

Подпрограмма

CALL HREJEC(NZ,NX,IR1,IR2)

Образует гистограмму, первый канал которой совпадает с каналом IR1 гистограммы NX, а последний - с каналом IR2. Параметры NZ определяются следующими соотношениями

$$S(NZ) = S(NX)$$

$$B(NZ) = B(NX) + S(NX) * (IR1 - 1)$$

$$L(NZ) = IR2 - IR1 + 1$$

Если $IR1 < 1$, или $IR2 > L(NX)$, то в NZ появляется $(1 - IR1)$ левых или $(IR2 - L(NX))$ правых нулей по сравнению с гистограммой NX.

Двумерный аналог отсутствует.

Подпрограммы

CALL SLISEX(NZ,NX,JP,IR1,IR2)

CALL SLISEY(NZ,NX,JP,IR1,IR2)

Образуют одномерные распределения путем свертки двумерных.

Это единственные подпрограммы, в которых NZ и NX принадлежат разным коммон-блокам. Подпрограмма SLISEY, в зависимости от вида JP, вычисляет для I-того канала сумму, среднее или сигму I-той строки двумерного распределения начиная с элемента (I, IR1) и кончая элементом (I, IR2), где IR1, IR2 - индексы по Y. Для второго и третьего случаев

JP = 4HMEAN

JP = 4HSIGM

При любых других значениях JP вычисляется сумма.

Подпрограмма SLISEX работает аналогично.

Существование подпрограммы HBNDR , SBNDR связано с необходимостью в некоторых случаях учитывать тот факт, что мы имеем дело с дискретными распределениями. Пусть, например, мы накапливаем распределение по переменной X_1 в гистограмму 'DISTR' , причем вес W извлекаем из гистограммы 'WEIGH' , содержащей распределение по переменной X , которая функционально связана с переменной X_1 . В этом случае область X , где W отлично от нуля, можно отобразить на область X_1 , причем скорее всего границы этой области не совпадут с границами каналов гистограммы, в которую ведется накопление. В каналах, которые пересекла граница, статистика будет подавлена по причинам отнюдь не физического характера.

Обращением

```
CALL HBNDR('MASK', 'WEIGH', FUN, R)
```

мы создаем в гистограмме 'MASK' (она должна иметь такие же параметры как и 'DISTR') распределение-маску. В каналы, которые целиком находятся в области, где $W > 0$, заносится число R , в остальные - 0. Функция FUN должна выразить зависимость $X_1 = F(X)$. (Просьба не забыть описать её в операторе EXTERNAL !). Уничтожить неполноценные каналы в гистограмме 'DISTR' можно с помощью подпрограммы HARI2 и её операции 'EQ' , где первым операндом будет 'DISTR' , а вторым - 'MASK' .

В двумерном случае аналогично работает

```
CALL SBNDR('MASK', 'WEIGH', FUNX, FUNY, R) ,
```

где

FUNX должна выразить зависимость $X_1 = F_X(X, Y)$, а

FUNY - зависимость $Y_1 = F_Y(X, Y)$.

3.8 Обмен данными между программой пользователя и системой.

Параметр W , который используется при обращении к подпрограммам $HACCUM$, $SACCUM$, как правило, извлекается из уже существующих распределений. Это можно сделать с помощью

$W = HBIN(NX, IRX)$, $W = HWEIGH(NX, RX)$
 $W = SBIN(NX, IRX, IRY)$, $W = SWEIGH(NX, RX, RY)$.

Подпрограммы $HBIN$, $SBIN$ пересылают в W содержимое канала IRX (канала (IRX , IRY) для двумерных распределений). По поводу параметров IRX , IRY нелишне вспомнить пункт 5 общих сведений.

Подпрограммы $HWEIGH$, $SWEIGH$ вычисляют W с помощью линейной интерполяции. Так, если $RX = (RX1+RX2)/2$, где

$RX1$ - середина I -того канала, а $RX2$ - середина $(I+1)$ -ого канала, то $W = (V1+V2)/2$, где $V1$ - содержимое I -того канала, а $V2$ - содержимое $(I+1)$ -ого канала.

Если

$$B < RX < B+S/2$$

то проводится экстраполяция. Аналогичная процедура проводится на другом краю гистограммы. Отметим, что результатом экстраполяции может оказаться отрицательное число, даже если все каналы гистограммы содержат положительные числа.

Перечисленные подпрограммы относятся к разряду быстродействующих, и потому параметр NX , если он задан в холерическом виде, "портится".

При накоплении распределений не всегда удобно задание параметра NX в подпрограммах $HACCUM$ и др. в холерическом виде, например, когда номер гистограммы, к которой обращаешься, зависит от переменной цикла. В этом случае бывает полезно перед входом в цикл, в котором проводится накопление, найти номер некоторой гистограммы по её "образу" с помощью подпрограмм

$$N = NUMHIS(NX),$$

для одномерного случая, или

$$N = NUMSCA(NX),$$

для двумерного.

Часто случается, что на каком-то этапе задачи возникает необходимость переопределить параметры B , S некоторых гистограмм. Например, какая-то группа гистограмм в начале задачи соответствующим заданием параметров S или B объявлена ненастроенной. После обработки некоторой части информации нужно снова объявить их ненастроенными, т.к. условия эксперимента поменялись. Параметры B , S переопределяются с помощью подпрограмм

```
CALL HPARIN(NS,NE,JP,R) ,  
CALL SPARIN(NS,NE,JP,R) ,
```

где JP указывает, какой параметр нужно переопределить,
 R - новое значение параметра. Для одномерного случая
 $JP = 1NB$ или $1NS$ или $1NC$;

для двумерного

$JP = 2NBX$, $2NBY$, $2NSX$, $2NSY$, $2NCX$, $2NCY$. Задание JP в виде $1NC$, $2NCX$, $2NCY$ (COUNT B) означает, что в качестве R задано среднее распределения, и параметр B надо вычислить таким образом, чтобы

$$R = B + S * L * 0.5$$

(совмещаем среднее распределения с серединой гистограммы).

Вычисленное таким образом B округляется до величины

$$B = N * S$$

Подпрограммы

```
CALL HAREX(NX,JP,ARRAY)  
CALL SAREX(NX,JP,ARRAY)
```

пересылают в массив $ARRAY$ содержимое NX , если
 $JP = 3NOUT$, и делают обратную операцию, если
 $JP = 2NIN$, при любых других значениях JP в $ARRAY$
пересылаются параметры B , S
(или BX, SX, BY, SY).

Подпрограммы

```
CALL HFUNEX(NX,FUN1)  
CALL SFUNEX(NX,FUN2)
```

засылают в NX значения функций $FUN1(X)$, $FUN2(X)$,
вычисленных в середине каждого канала. Эти функции должны быть описаны в операторе $EXTERNAL$. Значения функций, вычисленных в других точках или кусочно проинтегрированных по каждому каналу удобнее заслать с помощью подпрограмм $HZFXY$, $SUFXYZ$.

3.9 Подготовка данных для фита

Подпрограммы

```
CALL HFUMIL(NX,NY,IST,IE,IBAD,EXDA,NED)  
CALL SFUMIL(NX,NY,EXDA,NED,FUN)
```

формируют данные в коммон-блоках /EXDA / , /NED / , после чего надо обратиться к подпрограмме

```
FUMILI .
```

Через EXDA , NED мы обозначили имена массивов, содержащихся в коммон-блоках с теми же именами. Предполагается, что NX содержит значения экспериментальных точек, а NY - соответствующих им ошибок. Подпрограмма HFUMIL передаёт в указанные коммон-блоки данные с канала

IST по канал

IE .

IBAD - имя массива, содержащего список номеров "плохих" точек (или целый ноль, если в таком списке нет нужды). Признаком конца списка является целый 0. Иногда может оказаться удобнее воспользоваться не списком "плохих" точек, а продолжить выборку точек с другими значениями IST , IE . Такой прием можно осуществлять много раз, задавая для всех обращений, кроме первого NX = 0 (целый). Кроме того, экспериментальная точка автоматически считается плохой, если её ошибка не превосходит 0. Для двумерного случая процедура выброса плохих точек, описанная выше, перестает быть удобной. Взамен её пользователю предлагается самому написать

```
FUNCTION FUN(IX,IY) , которая должна быть описана  
в операторе EXTERNAL . Считается , что точка ( IX , IY )  
"плохая", если значение функции для неё не превосходит 0. Мелкое  
неудобство состоит в том, что такую функцию надо написать даже тогда,  
когда никаких точек выбрасывать не надо.
```


3.10 Генерация событий по заданным распределениям.

К такого рода подпрограммам предъявляются высокие требования на скорость их работы. Обеспечение этих условий достигается за счёт того, что исходное распределение NX в начале задачи преобразуется к специальному виду обращениями

CALL HRNDMO(NZ,NX) ,

и (или)

CALL SRNDMO(NZ,NX) ,

а при генерации событий используются подпрограммы

X = HRNDM1(NZ) ,

CALL SRNDM1(NZ,X,Y) , в которых

используется подпрограмма RNDM .

При генерации в учёт берутся только такие каналы исходной гистограммы, содержание которых превосходит $1/65000$ от суммы по всем каналам. В пределах одного канала X (или X,Y для двумерных распределений) разыгрываются равномерно.

Примечание.

Если гистограмма NZ заказывается через подпрограмму HOPEN (или SOPEN), она должна иметь то же число каналов, что и NX при этом двумерная гистограмма должна быть заказана как реальная.

3. II Обмен гистограммами между задачами через внешнюю память

Запись группы гистограмм во внешнюю память осуществляется с помощью подпрограмм

```
CALL HSAVE(IS,IE,SUB)
```

и (или)

```
CALL SSAVE(IS,IE,SUB) ,
```

где способ записи задается пользователем в

```
SUBROUTINE SUB(ARR, LONG) ,
```

которая должна быть описана в операторе EXTERNAL . Параметры IS , IE определяют номера первой и последней гистограмм множества, которые надо записать. Они могут иметь холерический вид. Подпрограмме SUB сообщаются адрес первого элемента массива и длина массива, который надо записать.

Например, если запись ведётся фортранными операторами, подпрограмма записи имеет вид

```
SUBROUTINE SUB(ARR, LONG)
```

```
DIMENSION ARR(LONG)
```

```
WRITE (I) ARR
```

```
RETURN
```

```
END
```

Параметр IS может иметь значение 0. В этом случае, кроме самих гистограмм (с 1-ой по IE), во внешнюю память запишется служебная часть коммон-блока, где содержится информация о его структуре.

Если запись ведется с $IS > 0$, предполагается, что при чтении будет предварительно осуществлен заказ гистограмм через подпрограммы HOPEN , SOPEN . При этом гистограммы, в которые будет записываться информация из внешней памяти, должны

быть заказаны в том же порядке и иметь те же длины, что и перед записью. Длина коммон-блока / HIST / или / SCAT / и, соответственно первый параметр в HOPEN , SOPEN в читающей и записывающих программах могут отличаться. Чтение осуществляется обращением

```
CALL HSAVE(IS,IE,SUB) ,
```

```
CALL HSAVE(IS,IE,SUB) ,
```

где IS , IE те же, что были при записи , а в SUB

(из примера) нужно оператор WRITE заменить оператором READ

Если запись ведётся с $IS = 0$, предполагается, что перед чтением обращение к `NOPEH`, `SOPEN` не делается. В этом случае для чтения подпрограммой `HSAVE` воспользоваться невозможно. Если запись велась фортранными операторами, прочитать информацию можно с помощью подпрограмм

```
CALL HFREAD(I) ,  
CALL SFREAD(I) ,
```

где I - номер устройства.

Если запись велась нефортранными операторами, то в читающей программе необходимо:

- а) Описать `COMMON/HIST/ARN(L1)` и (или) `COMMON/SCAT/ARS(L2)`, где $L1 \geq 1000$, $L2 \geq 3000$;
 - б) Читать запись в первый элемент массива `ARN` (и, или `ARS`)
- Если необходимо знать длину записи, то она будет содержаться в четвёртом слове массива `ARN` (или `ARS`), после того как информация в него будет считана.

После чтения записи, которая содержала служебную часть блоков `/ HIST /`, `/ SCAT /` нет необходимости обращаться к `NOPEH`, `SOPEN`. Если такое обращение будет сделано, оно будет восприниматься как дозаказ.

4. КАК В РАМКАХ СИСТЕМЫ НАПИСАТЬ ПОДПРОГРАММУ, КОТОРУЮ ЗАБЫЛИ ПРЕДУСМОТРЕТЬ АВТОРЫ.

Приведем пример одной из подпрограмм с комментарием.

```
      SUBROUTINE HZFXU(NZ,NX,FUN)
      COMMON/HIST/AR(1000)
      */HNUMBS/N1,N2,N3,K1,K2,K3,KOP3,M1,M3
      DIMENSION B(1),S(1),BIN(1),LONG(1)
      EQUIVALENCE (LONG,AR(1)),(B,AR(2)),(S,AR(3))
      *,(LT,AR(4)),(LGT,AR(5)),(BIN,AR(6))
      DATA(NY=0)
      IF(IHSCA1('HZFXU',NX,NY).NE.0) RETURN
      M3=M1
      IF(IHSCA3('HZFXU',NZ).NE.0) RETURN
      DXF=S(K1+1)
      DX=DXF*0.5
      X=B(K1+1)+DX
      DO 1 I=1,M1
      BIN(K3+I)=FUN(X,DX,BIN(K1+I))
      X=X+DXF
      CONTINUE
      M=IHSCA4(1,1)
      RETURN
      END
```

1

С помощью IHSCA1 отыскиваем операнды NX, NY. Если операнд только один, то параметр NY = 0. Если операнды найдены, то значение функции - 0, в противном случае IHSCA1 напечатает диагностику с сообщением имени подпрограммы, при обращении к которой это случилось (передаем его в первом параметре), а нам остаётся сделать только RETURN. После работы IHSCA1 в COMMON/HNUMBS/ имеем N1 - номер первого операнда, N2 - номер второго операнда, K1 - адрес в памяти первого операнда, K2 - адрес в памяти второго операнда, M1 - количество каналов в первом операнде. При желании можем определить количество каналов во втором операнде оператором M2 = LONG(K1+1)-5. Теперь определим длину результата (M3 = M1), После чего обратимся к IHSCA3, которая или отыщет гистограмму -

-результат, или её образует, при этом её параметры В, S станут совпадать с параметрами первого операнда. Если параметры В, S гистограммы-результата должны отличаться от соответствующих параметров первого операнда, пользователь должен переопределить их после обращения к INSCA3 . После работы INSCA3

N3 - номер гистограммы - результата,

K3 - её адрес в памяти. Если в Вашей подпрограмме было обращение к INSCA3 , то работу подпрограммы нужно заканчивать оператором M= INSCA4(1,1) .

5. НЕКОТОРЫЕ ХАРАКТЕРИСТИКИ СИСТЕМЫ.

Хронометраж работы "быстрых" подпрограмм представлен в таблице 3.

п/пр	I	БЭСМ-6, СОБ./СЕК	I	ЕС-1055, СОБ./СЕК
НАССUM	I	4000	I	5000
НВЕСТ	I	6600	I	
НВІN	I	5000	I	5400
НВЕIGH	I	3800	I	3600
САССUM (ТИП I)	I	1800	I	2000
САССUM (ТИП R)	I	2100	I	2000
SBIN	I	2100	I	
SWEIGH	I	650	I	
HRNDM1	I	2800	I	2600
SRNDM1	I	1700	I	

Для хранения гистограмм и структуры данных в системе используются коммон-блоки /HIST/1000 , /SCAT/3000 . Эти блоки могут быть расширены в головной программе, о чём нужно сообщить в системе при обращении к подпрограмме HOREN и(или) SOREN .

Для хранения результатов промежуточных вычислений используется блок /NARWOR/(200) . В этих же целях его можно использовать в подпрограммах пользователя.

Кроме того, используются блоки /HFLAG/(2) , /HNUMBS/(9) , /SNUMBS/(21) , /NZER/(6) , /PARHIS/(16) , /HPLOTT/(19) .

Все подпрограммы системы "НОРА" занимают память на ЕС-1055 65 Кбайт, на БЭСМ-6 (без учёта подпрограмм для плоттера) - 8200 слов.

6. ЗАКЛЮЧЕНИЕ.

Авторы считают своим приятным долгом поблагодарить В.Г.Аблеева, С.А.Запорожца, А.П.Чеплакова, общение с которыми в процессе эксплуатации системы существенно повлияло на её окончательный вид.

Авторы приносят глубокую благодарность А.П.Сапожникову, предложившему для нужд системы свою подпрограмму JSEART , а также за разработку по просьбе авторов подпрограммы F1PLOT .

Авторы благодарны также Е.Ю.Мазепе, М.Ю.Попову, Е.Д.Федюнькину в дискуссии с которыми определился стандарт представления данных для подпрограмм надписей на графопостроителе.

Авторы признательны З.П.Мотиной и Б.Науманн за помощь в оформлении рукописи.

ЛИТЕРАТУРА.

1. Аблеев В.Г. и др. доклад 48/А6-5, Кайдалов А.Б., рапортёрский доклад А1-1 на ЖУШ Междунар. Конф. по физике высоких энергий, Тбилиси, 1976 год. Труды ЖУШ Междунар. конф. по физике высоких энергий. ОИИИ, Д1,2-10400, Дубна, 1977.
2. Мазный Г.Д., Ситник И.М., Строковский Е.А. в сб. "Труды совещания по программированию и математическим методам решения физических задач", ОИИИ, Д10, II-II264, С.504, Дубна, 1978.
3. HBOOK USER'S GUIDE . CERN computer centre program library , long write-up , Y-250 .
4. V.G.Ableev et al. Nucl. Phys. A393 (1983) 491; JINR preprint E1-82-777 , Dubna, 1982.
5. Баяковский Ю.М., Михайлова Т.Н., Мишакова С.Т. , препринт ИИИ, 1972, 341.

Алфавитный список подпрограмм

HACCUM.....12
HAREX37
HARI127

HARI230

HBIN36
HBIPRI.....14
HBNDR35
HCLEAR.....11
HDELET.....26
HEVEN1.....15
HFREAD.....41
HFUMIL.....38
HFUNEX.....37
HGLUE32
HISLIS.....17

HMAXIM.....15
HMEAN1.....15
HMODA1.....15
HOPEN..... 7
HPLAXE.....19
HPLERR.....21
HPLERT.....21
HPLETX.....23
HPLETY.....23
HPLFUN.....22
HPLGRF.....22
HPLHIS.....21
HPLINI.....18
HPLIN1.....18
HPLNUX.....20
HPLNUY.....20
HPLZER.....19
HPRIN14
HREJEC.....34
HRNDMO.....39
HRNDM1.....39
HSAVE40
HSIGM1.....15
HVECT13
HWEIGH.....36
HZFX Y28

MESOFF.....25
MESON25
NUMHIS.....36
HRARIN.....37

SACCUM.....12
SAREX37
SARI1I.....27
SARI1R.....27
SARI2I.....30
SARI2R.....30
SBIN36

SBNDR35
SCLEAR.....11
SDELET.....26
SEVEN1.....16
SFREAD.....41
SFUMIL.....38
SFUNEX.....37

SLISEX.....34
SLISEY.....34
SMAXIM.....16

SOPEN 7

SPRIN16

SRNDMO.....39
SRNDM1.....39
SSAVE40

SWEIGH.....36
SUFXYZ.....28
SXFYO16
SYFXO16

NUMSCA.....36
SPARIN.....37

Список подпрограмм нижнего уровня

AREQAR
AREQNU
ARPLNU
ARPLAR
EQNUAR
ARSUM
ARMAX
ARMOM
HNUMRE
HVECTO
HROUND
OPEN
SHIFOP
OPNPR2
OPNPR3
CARDRE
CARDR1

HSHIFT
IHSCA1
ISCAN1
MESSAG
MESSER
MESFAT
IHCONT
IHSCA3
IHSCA4
ISCAN3
ISERV
HSTAR
HISPAR
HISPR
DELET
DATAPR
TIMEPR

HPLTMX
HPLTMY
HPLTM
SNUMRE
SVECTO
NUMSCA
ISSCA1
ISERSC
ISSCA3
ISSCA4
SPRBIN
SCAPAR
SCAPR
MESER1
ISX
ISY
HMDDL

ОГЛАВЛЕНИЕ

I.	Введение.....	3
2.	Общие сведения о системе.....	4
3.	Описание подпрограмм.....	7
3.1	Подпрограммы заказа.....	7
3.2	Подпрограммы "очистки".....	II
3.3	Подпрограммы накопления	I2
3.4	Подпрограммы представления распределений на АЦПУ:.....	I4
3.5	Управление перелистыванием.....	I7
3.6	Представление распределений на плоттере.....	I8
3.7	Операции над распределениями.....	25
3.8	Обмен данными между программой пользователя и системой.....	26
3.9	Подготовка данных для фита	38
3.10	Генерация событий по заданным распределениям.....	39
3.II	Обмен гистограммами между задачами через внешнюю память.....	40
4.	Как в рамках системы написать подпрограмму, которую забыли предусмотреть авторы	42
5.	Некоторые характеристики системы.....	44
6.	Заключение	45
	Литература.....	46
	Алфавитный список подпрограмм.....	47
	Список подпрограмм нижнего уровня.....	48