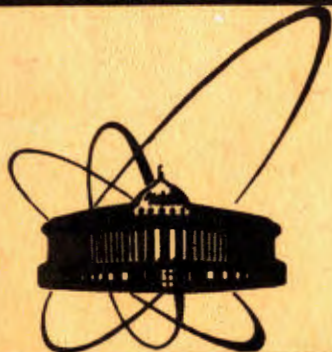


85-956



**СООБЩЕНИЯ  
ОБЪЕДИНЕННОГО  
ИНСТИТУТА  
ЯДЕРНЫХ  
ИССЛЕДОВАНИЙ  
ДУБНА**

P10-85-956

**Н.В.Горбунов, А.Г.Карев, Е.А.Ладыгин,  
Э.И.Мальцев, Б.А.Морозов, Ю.П.Петухов,  
А.Ю.Суханов**

**СИСТЕМА MES ДЛЯ ПРИЕМА И ОБРАБОТКИ  
ЭКСПЕРИМЕНТАЛЬНЫХ ДАННЫХ**

**Взаимодействие процессов, генерация системы**

**1985**

## ВВЕДЕНИЕ

Мы продолжаем описание системы MES (Multi Executive System) приема и экспресс-обработки данных с установок физики высоких энергий. В двух предыдущих работах/1,2/ были изложены общие положения, аппаратное обеспечение, язык управления приемом данных, возможности системы на стадии анализа информации, команды языка обработки данных. В этой, третьей статье будут рассмотрены особенности TS-монитора при работе MES, взаимодействие процессов приема и анализа данных, способы адаптации системы к экспериментам и методы генерации MES. Напоминаем, что система ориентирована на пользователя-физика, имеющего некоторый /но не профессиональный/ опыт работы с вычислительным оборудованием для приема и обработки данных с экспериментальных установок.

### 1. СТРУКТУРА СИСТЕМЫ MES В РЕЖИМЕ РАЗДЕЛЕНИЯ ВРЕМЕНИ

Система MES работает под управлением монитора разделения времени в рамках операционной системы РАФОС/3/ для СМ ЭВМ. Используются следующие средства, предоставляемые TS-монитором.

#### 1.1. Режим разделения времени

Все функции системы MES по приему и обработке данных выполняются двумя задачами в режиме разделения времени. Автономная задача MESBAK осуществляет прием экспериментальной информации и запись ее на магнитную ленту. MESBAK является резидентной частью системы MES, она загружается во время запуска TS-монитора и не имеет средств диалога с оператором. Как уже было сказано в/1/ /см. раздел 1.1/, это необходимо для обеспечения максимальных скоростных характеристик аппаратуры связи КАМАК-ЭВМ. /Запись идет по схеме КАМАК - память ЭВМ - магнитная лента, без промежуточной записи на магнитный диск/.

Задача MESFOR обрабатывает информацию, полученную от задачи MESBAK, или с магнитофонных лент, выводит обработанную информацию на графические устройства, управляет работой задачи MESBAK и обеспечивает диалог с оператором.

#### 1.2. Разделяемые ресурсы

Совместно используемым ресурсом для обеих задач, MESBAK и MESFOR, является разделяемая резидентная библиотека BUFFER.



MES, в которой хранится информация, принятая задачей MESBAK, и используемая задачей MESFOR.

### 1.3. Почтовые ящики

Аппарат почтовых ящиков используется для синхронизации обмена данными между MESBAK и MESFOR, а также для управления задачей MESBAK со стороны MESFOR. Использование почтовых ящиков снимает проблему критической секции, возникающую при одновременной работе нескольких процессов с общими ресурсами.

Почтовый ящик представляет собой структуру типа очереди "первый пришел, первый вышел" (FIFO), в которой хранятся сообщения. Каждому вновь образуемому почтовому ящику присваивается имя в момент отправки первого сообщения. Средства работы с почтовыми ящиками представляет операционная система, а их параметры /размер, максимальное число сообщений/ задаются при генерации TS-монитора.

### 1.4. Спулинг

Процедура спулинга, поддерживаемая TS-монитором, используется при выводе информации на печатающие устройства и графопостроители. Информация, выводимая через эти устройства, предварительно записывается в файл на диске, а затем уже с этого файла передается на устройства.

### 1.5. Командные файлы с параметрами

Аппарат командных файлов, организованный в TS-мониторе, обеспечивает возможность обработки параметров командного файла. Используется при генерации и запуске системы MES.

### 1.6. Некоторые особенности работы с РАФОС-TS

Для операционной системы РАФОС-TS характерным недостатком является медленная реакция на внешние прерывания /~ 2 мс/. Однако, если система используется при работе экспериментальных установок физики высоких энергий с циклическим характером поступления информации от ускорителей, то этот недостаток несуществен.

### 1.7. Операционная система РАФОС-TS

Эта система не позволяет, в общем случае, производить операции ввода/вывода в области резидентных библиотек. Это могло бы вызвать снижение скорости записи информации на магнитные ленты. Для решения этой проблемы пришлось изменить драйвер магнитофона. В измененном драйвере при указании нечетного адре-

са памяти для операции ввода/вывода драйвер воспринимает усеченный до четного адрес, как адрес двойного слова, в котором хранится истинный физический адрес /18 бит/.

## 2. ОБМЕН ДАННЫМИ И СИНХРОНИЗАЦИЯ МЕЖДУ ЗАДАЧАМИ

Принимаемые с экспериментальной установки данные хранятся в разделяемой резидентной библиотеке BUFFER.MES. Размер этой библиотеки определяется размером файла BUFFER.MES, который в момент запуска TS-монитора находится на системном устройстве. Резидентная библиотека представляет собой набор фиксированных по размеру и местоположению буферов. Эти параметры определяются и могут быть изменены во время инициализации системы MES. Каждый буфер имеет свой дескриптор-массив из трех слов, содержащий необходимую информацию о буфере.

Каждый процесс обработки входных данных получает из входного почтового ящика сообщения-дескрипторы буферов и посылает в выходной почтовый ящик сообщения-дескрипторы отработанного буфера.

В системе MES имеются три основных почтовых ящика:

- MYQ - содержит дескрипторы свободных буферов,
- TPQ - содержит дескрипторы буферов, находящихся в очереди записи на магнитофонную ленту,
- ANQ - содержит дескрипторы буферов, находящихся в очереди на обработку данных.

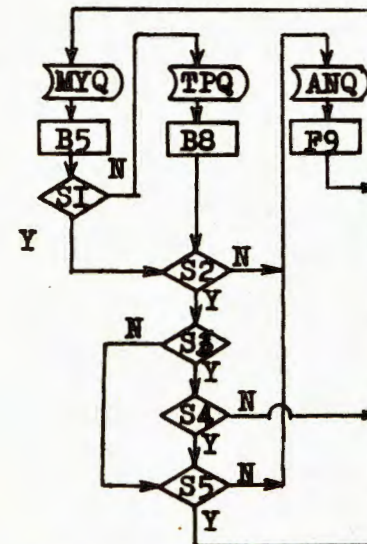


Рис.1. Синхронизация задач MESBAK и MESFOR.

На рис.1 показан алгоритм переключения буферов, обеспечивающий синхронизацию процессов B5, B8 и F9, при которой каждый из них выполняется независимо от остальных. Распределением буферов можно управлять с помощью двух системных переменных SCHEME и LIMIT.

Переменная SCHEME определяет три схемы обмена между MESBAK и MESFOR:  
SCHEME = 0 - обеспечивается обработка всех принятых буферов. В этом случае скорость приема данных будет определяться скоростью обработки.  
SCHEME = 1 - обеспечивает наличие в очереди на обработку по крайней мере "LIMIT" штук буферов.  
SCHEME = -1 - обеспечивается наличие хотя бы одного свободного буфера. При наличии нескольких сво-



бодных буферов действия такие же, как и при SCHEME = 1. Эта схема обеспечивает максимальную скорость приема данных, но необходимо наличие не менее трех буферов, чтобы все три процесса могли работать одновременно. В этом случае два буфера подключаются к процессам B5 и F9, а третий будет находиться в очереди MYQ.

Этапы процесса /см.рис.1/ имеют следующие значения:

- MYQ - очередь свободных буферов,
- TPQ - очередь буферов на запись на НМЛ,
- ANQ - очередь буферов на обработку,
- B5 - процесс приема текущего события,
- B8 - процесс записи очередного буфера,
- F9 - процесс обработки очередного события,
- S1 - запись на НМЛ. запрещена?
- S2 - SCHEME ≠ 0?
- S3 - SCHEME < 0?
- S4 - в MYQ есть еще буферы?
- S5 - в ANQ уже "LIMIT" штук буферов?

Для правильного выполнения согласованных действий задачи MESBAK и MESFOR должны иметь возможность обмениваться сообщениями. Для этой цели предназначены два почтовых ящика с именами:

FOR - сообщения для MESFOR и BAK - сообщения для MESBAK.

Формат сообщений имеет следующий вид:

контрольная сумма, сообщение, ИСТ /источник/, код.

Здесь ИСТ - байт, определяющий источник сообщения /ИСТ=1 - MESFOR, ИСТ=2 - MESBAK, ИСТ=0 - инициализирующая программа/.

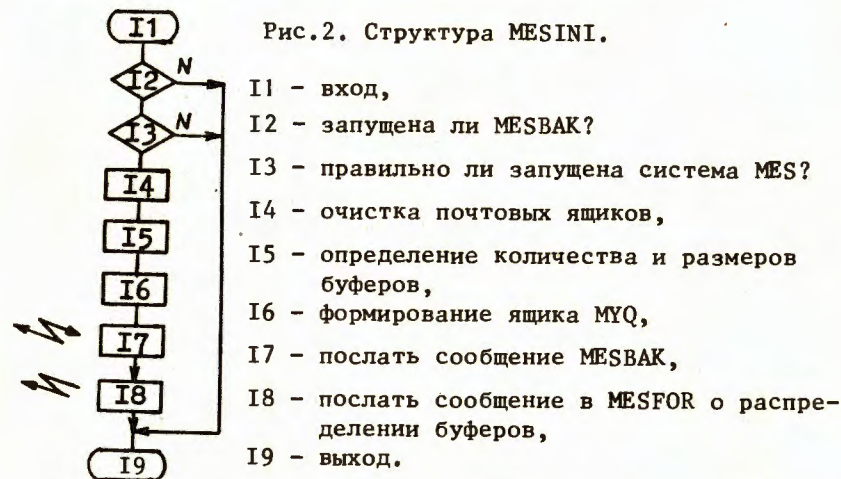
Код - байт, указывающий тип сообщения, ответным сообщениям присваивается отрицательный код. Значения байта "Код" определяют следующие действия источника сообщения:

- код = 0 - требует инициализации приемника,
- = 1 - передает информацию о распределении ресурсов,
- = 2 - сообщает о начале нового эксперимента, передает значения переменных RUNNUM, SCHEME, LIMIT, LOG, FLG, MESPTY.
- код = 3 - сообщает о конце эксперимента,
- = 4 - сообщает об останове процесса приема (SUSPEND),
- = 5 - сообщает о продолжении процесса приема (RSUMERUM),
- = 6 - сообщает об окончании работы системы (STOP),
- = 7 - требует выполнения команды системы /обычно в случае серьезного сбоя вызов команды SUSPEND/.

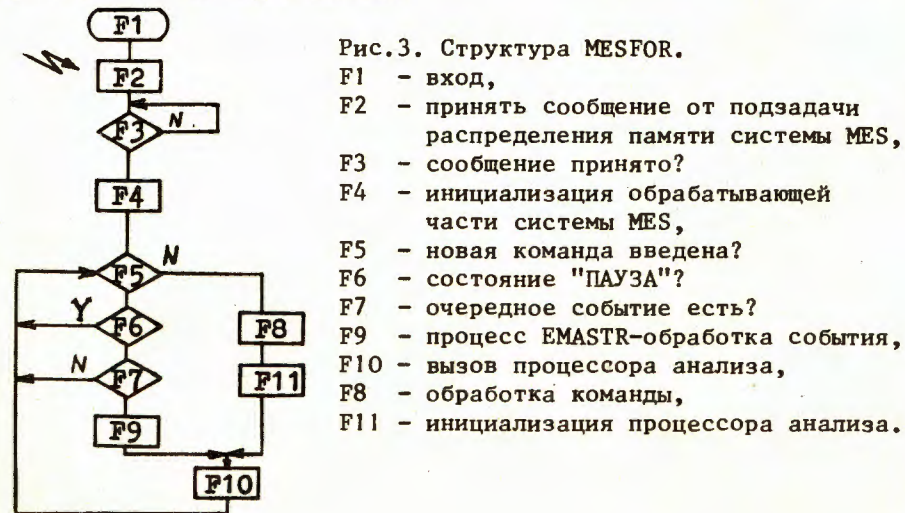
Процедура обмена сообщением происходит следующим образом. Источник посылает сообщение, приемник его принимает, проверяет правильность приема, посылает подтверждение, а источник его принимает.

### 3. ВЗАИМОДЕЙСТВИЕ ЗАДАЧ

Запуск системы MES производится командным файлом MES. Сначала запускается задача MESINI /рис.2/, которая проверяет правильность запуска системы и осуществляет распределение буферов. После обработки MESINI все буфера будут сложены в почтовый ящик MYQ, а в почтовом ящике FOR будет находиться сообщение о распределении буферов и динамической памяти.



После этого произойдет запуск исполняющей задачи MESFOR /рис.3/, которая прочитает сообщения из MYQ, произведет инициализацию в соответствии с принятым сообщением и войдет в фазу обработки команд и данных:









#### 4. СТРУКТУРА ЗАДАЧ MESBAK И MESFOR

##### 4.1. Структура MESBAK

Задача MESBAK выполняет следующие функции:

а/ осуществляет прием данных с установки в текущий открытый буфер в соответствии с алгоритмом, описанным в файле CAMAC.DEF; б/ производит проверку правильности протокола приема. В случае ошибок посылает диагностические сообщения на первую физическую линию;

в/ записывает заполненные буферы на магнитную ленту /устройство МТФ/;

г/ выполняет запросы задачи MESFOR, поступающие в почтовый ящик ВАК /чтение нового списка команд, установка внутренних переменных/.

Задача может находиться в одном из следующих состояний:

- пассивное состояние - задача не выполняет никаких действий;
- состояние обработки сообщения - в него задача входит при появлении сообщения в почтовом ящике ВОХВАК;
- состояние приема - в него задача входит при появлении сигнала прерывания от системы сбора. В этом состоянии в задаче могут быть активны одновременно два процесса: процесс чтения данных с аппаратуры КАМАК и процесс записи данных на магнитную ленту.

Алгоритм процесса приема данных показан на рис.9.

При возникновении прерывания начинается выполнение списка команд, полученного при трансляции файла CAMAC.DEF и переданного задаче во время сообщения BEGINRUN. В процессе выполнения списка команд заполняется очередное событие в текущем открытом буфере. Если событие заведомо не поместится в буфере, то текущий буфер будет передан во вспомогательный ящик HIQ, а из ящика MYQ будет взят свободный. Алгоритм распределения заполненного буфера показан на рис.10. Из ящика HIQ буферы поступают в очередь записи на ленту - в ящик MTQ. В одно и то же время один буфер может заполняться данными, а другой - записываться на магнитную ленту. После приема каждого события генерируется сигнал "Конец опроса", разрешающий прохождение очередного запуска системы считывания. С момента поступления сигнала прерывания и до сигнала "Конец опроса" система MES будет игнорировать поступающие сигналы запуска.

##### 4.2. Структура MESFOR

Задача MESFOR выполняет следующие функции:

а/ обрабатывает информацию, находящуюся в буферах в почтовом ящике ANQ. Ящик ANQ может заполняться буферами, поступающими от задачи MESBAK /состояние переменной DATA=EX/, либо читаемыми с магнитной ленты (DATA=MT);

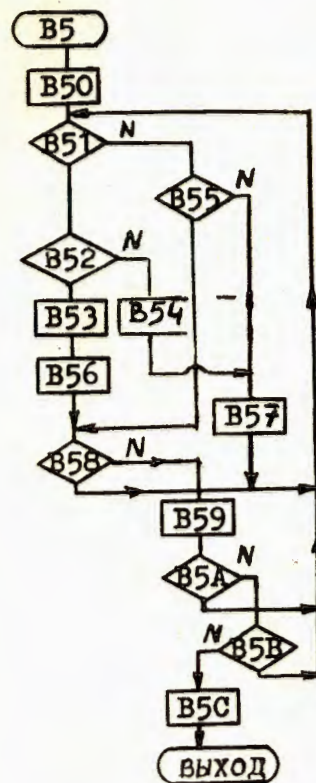


Рис.9. Алгоритм обработки прерывания.

- B5 - появление сигнала прерывания,
- B50 - запретить прерывание,
- B51 - есть текущий буфер,
- B52 - есть свободное место в буфере?
- B53 - прием данных согласно списку команд, переданных в сообщении BEGINRUN,
- B54 - передать текущий буфер в HIQ,
- B55 - состояние "без буферов" (MYQ пуста)?
- B56 - выдать импульс на GIGO - ЭВМ готова,
- B57 - взять из MYQ буфер, сделать его текущим,
- B58 - есть необработанное прерывание при наличии свободного буфера,
- B59 - распределить заполненный буфер/рис.5/,
- B5A - остались ли необработанные прерывания?
- B5B - появилось ли новое требование прерывания?
- B5C - разрешить прерывание.

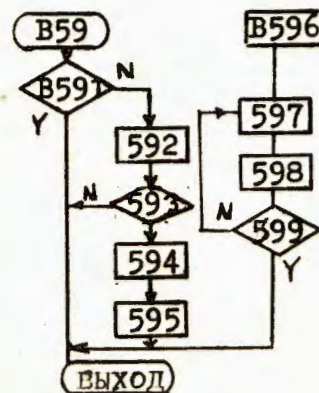


Рис.10. Алгоритм распределения заполненного буфера.

- B591 - нет заполненного буфера?
- B592 - взять заполненный буфер из MYQ
- 593 - запись разрешена?
- 594 - передать буфер в MTQ
- 595 - активизировать запись на магнитофон
- 596 - магнитофон записал буфер
- 597 - взять записанный буфер из MTQ
- 598 - переслать буфер в MYQ или в ANQ в зависимости от значения SCHEME и LIMIT /рис.4/
- 599 - MTQ пуста?

б/ представляет результаты обработки в удобном для восприятия виде;

в/ выполняет команды, поступающие от оператора. Управляет работой MESBAK.



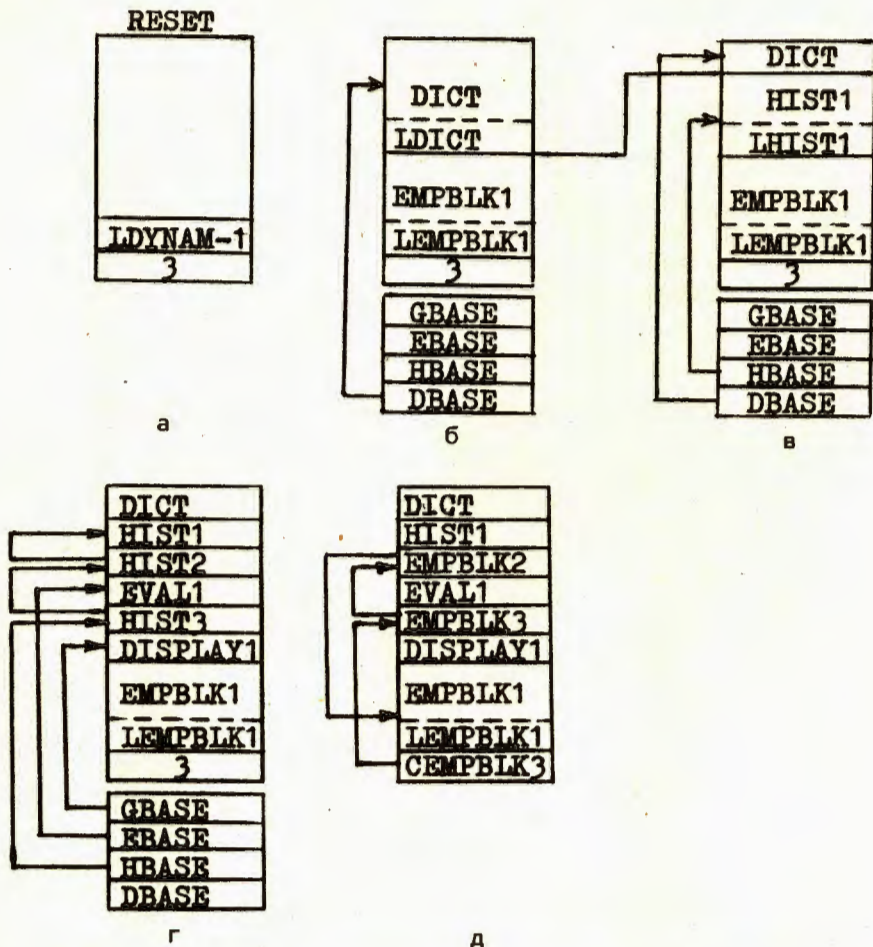


Рис.11. Структура динамической памяти: а/ начальное распределение памяти; б/ после команды RESET; в/ после команд HIST, EVAL, HIST, DISPLAY; г/ после команд HIST, EVAL, HIST, DISPLAY; д/ после команд ERASE HIST2, HIST3.

Задача MESFOR должна быть загружена с системного физического устройства SY: с первой физической линии /т.к. диагностические сообщения MESBAK будут поступать именно на эту линию/. Структура MESFOR показана на рис.3.

Задача MESFOR взаимодействует со следующими структурами:  
 а/ Динамический блок: массив данных, с которым работает один из процессоров. Динамический блок занимает фиксированное положение в динамической памяти. Он может быть удален при выполнении некоторых команд.



Рис.12. Структура словаря: а/ структура словаря; б/ структура элемента; в/ структура пустого элемента.

б/ Динамическая память. Область физической памяти, в которой задача хранит динамические блоки. Динамическая память является разделяемой резидентной библиотекой, размер которой определяется в момент запуска TS-монитора и равен размеру файла DYNAMI.MES, находящемуся на системном устройстве в момент запуска. Структура динамической памяти показана на рис.11.

в/ Словарь: динамический блок, в котором хранятся имена и характеристики переменных. Длина словаря определяется переменной LDICT в момент выполнения команды RESET. Структура словаря показана на рис.12.

г/ Вектор: переменная, доступ к которой осуществляется по имени /см. определение, данное в работе/1/, описания системы MES/.

Вектор - это единственный объект, с которым система может работать. Для получения значения вектора используется подпрограмма-функция GETVAL (VCODE, END) - возвращающая реальное значение вектора, код которого равен VCODE. Здесь END - целочисленная переменная, имеющая следующий смысл:

- END = -1 - значение вектора получено, вектор исчерпан;
- END = 0 - значение вектора не получено, вектор исчерпан;
- END = 1 - очередное значение вектора получено, вектор не исчерпан, т.е. необходимо обратиться к функции GETVAL еще раз за следующим значением.

VCODE - код вектора, соответствующий его имени, VCODE образуется следующим образом:

$$VCODE = AREA * 2048 * VDISPL(AREA) + VINCR(AREA) * INDEX,$$

здесь AREA - область определения вектора, она однозначно определяется по первым двум буквам названия вектора;

INDEX - индекс в области, определяемый числом, входящим в название вектора;

VDISPL, VINCR - массивы, определяющие вектор.



В табл.1 описаны все области определения векторов.

Таблица 1

Название вектора	AREA	Название области	Описание
нет	1	SYSVAR	Системная переменная, которая может принимать числовое значение.
нет	2	SYSNAM	Системная переменная, которая может принимать системное символьное значение.
нет	3	SYSVAL	Системное символьное значение.
нет	4	HIST/WB	Объект, ссылающийся на динамический блок.
RV	5	RVAR	Элемент массива реальных переменных.
IV	6	IVAR	Элемент массива целых переменных.
DA	7	DATA	Элемент события.
	.		
	14		
	15	USER	Вектор, определяемый пользователем.

## 5. АДАПТАЦИЯ СИСТЕМЫ

Система MES предоставляет пользователю возможность введения в систему собственных подпрограмм. Обеспечивается это следующими средствами:

- в процессе выполнения каждой команды происходит обращение к пользовательским подпрограммам. Например, при обработке команды DISPLAY EVENT система не сможет распознать ключевое слово EVENT и вызовет пользовательскую подпрограмму DIUSER. Пользовательская подпрограмма должна сама распознать ключевое слово и обработать остальные аргументы. Наиболее широко используемые подпрограммы: REUSER - во время RESET процессора произвести, например, вычисление калибровочных параметров. DIUSER - графическое представление события, распечатка важной информации;

- в процессе обработки события происходит обращение к пользовательским подпрограммам EUSERA и EUSERB /см. работу/2//. В программе EUSERA обычно вызывается пользовательская подпрограмма проверки формата события;

- если происходит обращение к вектору из пятнадцатой области определения, то подпрограмма GETVAL(VCODE, END) вызовет пользовательскую подпрограмму GETUSR(INDEX, END).

Пользователь может работать со следующими COMMON-блоками и подпрограммами /табл.2 и 3/.

Таблица 2

COMMON	Массив/ переменная	Действие
ICOMN	IVAR(i)	представляет вектор IV
RCOMN	RVAR(i)	представляет вектор RV
DTCOMN	VNAME(i)	определяет названия векторов
	VAREA(i)	определяет название области определения
	VDISPL(i)	определяет смещения в области определения вектора
	VINCR(i)	определяет приращение индекса вектора
RCOMN	VLIMIT(i)	определяет верхний предел индекса вектора
	ARG(i)	массив двойной точности, содержит аргументы команды
	ATYPE(i)	содержит типы аргументов /символ, число, по умолчанию/

Пользователь может также ввести новый вектор. Для этого нужно в COMMON - блок DTCOMN, находящийся в файле MESCOM.TXT, вставить название, область определения, смещение и приращение вектора. Кроме того, необходимо изменить подпрограмму GETUSR, для того, чтобы она обеспечила нужный алгоритм вычисления значения вектора, откомпилировать подпрограммы RMASTR и GETUSR и запустить компоновщик.

## 6. ГЕНЕРАЦИЯ И ЗАПУСК СИСТЕМЫ MES

Файлы, необходимые для полной генерации системы, перечислены в табл.4.

Предполагается следующий порядок работы с текстовыми модулями.

Текстовые модули группируются в текстовые библиотеки. В текстовых библиотеках не хранится информация, общая для нескольких модулей /в частности, COMMON-блоки/. Подобная информация содержится в файле определения MESCOM.TXT. Существует набор программ для работы с текстовыми библиотеками:

LIBEXT - извлечь модули из библиотеки,  
LIBLST - распечатать оглавление библиотеки,



LIBEDT - вставить /удалить/ заменить модули,  
 PRE - вставить COMMON-блоки из файла определений в модули,  
 COMLOD - подготовить файл определений для программы PRE.

Таблица 3

Программа	Библиотека	Тип	Вызов, описание
MSGOUT	MESXVO		CALL MSGOUT (PRIORY,LENGTH,TEXT) выводит на терминал TEXT длиной LENGTH, если PRIORY больше MESPTY
PRINTL	MESBAS		CALL PRINTL (LENGTH, TEXT) выводит TEXT длиной LENGTH на печать
TYPE	MESBAS	L*1	L=TYPE (IARG, ICLASS) если аргумент IARG принадлежит классу ICLASS (SYMBOL, NUMBER, DEFAULT), то L=.TRUE.
CMDIN	MESXVO	L*1	L=CMDIN( ) принять команду с терминала в массив CITEXT COMMON блока CCOMN
DBUFF	MESX05/V1	I*2	I=DBUFF (nnn) дает значение вектора DA nnn
DYNAM/ PUTDYN	--	--	I=DYNAM(INDEX)/CALL PUTDIN(INDEX, IVAL) возвращает/устанавливает значение элемента INDEX динамической памяти
GETBLK/ PULBLK/ CLBLK	--	--	CALL GETBLK(INDEX, IARRAY, LENGTH) CALL PUTBLK(INDEX, IARRAY, LENGTH) CALL CRLBLK(INDEX, LENGTH) получить/передать/очистить блок динамической памяти, начинающийся с INDEX длиной LENGTH.
MOVSTR	ULIB		CALL MOVSTR (IFROM, IELEML, ITO, IELEMO, LENGTH) копирует строку байт IFROM длиной LENGTH, начиная с элемента IELEMI в строку ITO, начиная с элемента IELEMO
и т.д.			

LIBPRG.TXT	Библиотека программ - утилит работы с текстовыми библиотеками.
IOLIB.TXT	Библиотека ввода-вывода.
GLIB.TXT	Библиотека графических подпрограмм.
ULIB.TXT	Библиотека вспомогательных подпрограмм.
CAMLIB.TXT	Библиотека подпрограмм процесса приема данных /задача MESBAK/.
MESBAK.TXT	Базовая библиотека фортрановских подпрограмм, независимых от операционной системы.
MESXVO.TXT	Библиотека подпрограмм, зависящих от операционной системы.
MESX05.TXT	Библиотека off-line реализации системы.
MESUSR.TXT	Библиотека реализации конкретного эксперимента.
MESXVI.TXT	Библиотека on-line реализации системы.
LIBEXT.FTN MATCH.FTN GETSTR.FTN	Вспомогательные программы для первичной генерации.
MESCOM.TXT	Файл, содержащий COMMON блоки системы MES.
MESGEN.TXT	Библиотека генераторов системы.
PGN.SAV	Интерактивный макропроцессор ПАГЕН.

Процесс генерации объектных модулей из текстовой библиотеки показан на рис.13.

Предусмотрена возможность интерактивной генерации как всей системы, так и отдельных библиотек и модулей. Процедуры генерации находятся в текстовой библиотеке MESGEN, они предназначены для управления интерактивным макропроцессором ПАГЕН.

Во время генерации и исполнения предполагаются следующие назначения логических устройств:

- SY: - на этом устройстве будут созданы исполняющие файлы MESFOR и MESBAK и объектные библиотеки IOLIB, GLIB, ULIB,  
 TXT: - устройство, содержащее текстовые файлы,  
 DEF: - устройство, содержащее файл определения,  
 LST: - устройство, содержащее листинги программ,  
 OBJ: - устройство, содержащее объектные модули,  
 DK: - временное хранение файлов.



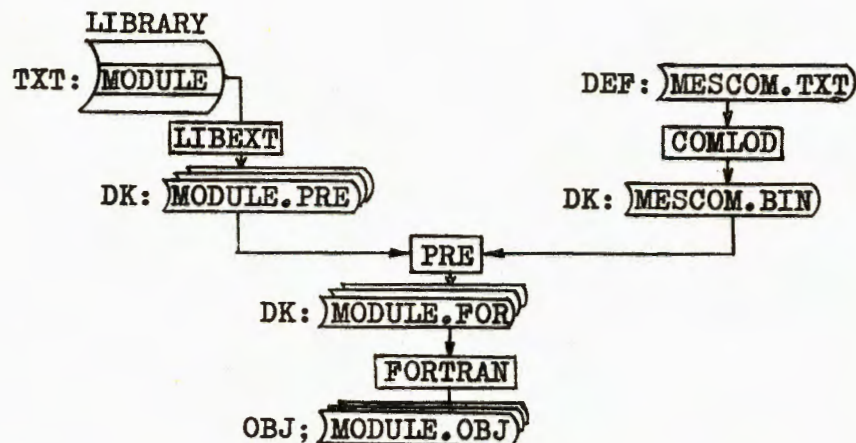


Рис.13. Генерация объектных модулей из текстовой библиотеки.

Запуск системы MES производится командным файлом MES. Для правильного запуска необходимо выполнение следующих условий:

- задача MESBAK должна находиться в памяти,
- оператор должен быть зарегистрирован в системе под PPN=1,1,
- задание должно иметь операторские привилегии;
- файлы MESFOR.SAV. и CAMAC.DEF. должны находиться на устройстве SY:
- при генерации TS-монитора должен быть предусмотрен достаточный размер почтовых ящиков и количество сообщений. Количество автономных линий должно быть не менее 2.

#### ЛИТЕРАТУРА

1. Горбунов Н.В. и др. ОИЯИ, P10-85-954, Дубна, 1985.
2. Горбунов Н.В. и др. ОИЯИ, P10-85-955, Дубна, 1985.
3. Валикова Л.И. и др. Операционная система CM ЭВМ РАФОС. Справочник. "Финансы и статистика", М., 1984.

Рукопись поступила в издательский отдел  
27 декабря 1985 года.

Горбунов Н.В. и др.

P10-85-956

Система MES для приема и обработки экспериментальных данных. Взаимодействие процессов, генерация системы

Предлагается система приема и обработки экспериментальных данных, ориентированная на локальные вычислительные системы, в основе которых используется ЭВМ типа CM-4 и аппаратура в стандарте KAMAK. Система работает под управлением операционной системы РАФОС-TS. В данном описании системы MES рассматривается генерация системы и взаимодействие процессов.

Работа выполнена в Серпуховском научно-экспериментальном отделе ОИЯИ.

Сообщение Объединенного института ядерных исследований. Дубна 1985

Gorbunov N.V. et al.

P10-85-956

Data Acquisition System MES.

System Generation and Interaction of Processes

Here we present data acquisition system, which is based on local computing capacities the core of which being computer series CM-4 (CM-1420) and CAMAC standard equipment. It functions under control of RAFOS-TS operation system and presents the following options for the investigator: 1. interactive dialogue, 2. control and regulation of the data acquisition process, 3. recording of acquired information onto the magnetic tape, 4. fast processing of acquired information or some part of it, as well as information previously recorded on the magnetic tape, 5. presentation of the analysis results in spreadsheet and graphic form, 6. recording of analysis results onto magnetic tape and discs. System generation and interaction of processes of MES system is described.

The investigation has been performed at the Serpukhov Scientific Experimental Department, JINR.  
Communication of the Joint Institute for Nuclear Research. Dubna 1985