

З. П. ВОСТРИКОВА, С. Т. МИТИН

КУРС ОПЕРАТОРНОГО ОБСЛУЖИВАНИЯ ОС ЕС ЭВМ

*Допущено Министерством
высшего и среднего специального образования СССР
в качестве учебного пособия
для студентов высших учебных заведений*



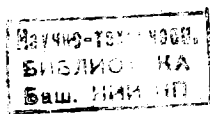
МОСКВА «НАУКА»
ГЛАВНАЯ РЕДАКЦИЯ
ФИЗИКО-МАТЕМАТИЧЕСКОЙ ЛИТЕРАТУРЫ
1983

22.18
В 78
УДК 519.6

Курс операторного обслуживания ОС ЕС ЭВМ. Востриков А. З. П.,
Митин С. Т. — М.: Наука. Главная редакция физико-математической литературы, 1983. — 248 с.

В книге содержатся общие сведения о ЕС ЭВМ, принципы работы основных устройств, организация данных на разных носителях. Приводятся сведения об операционной системе. Подробно рассматривается работа оператора с ОС, порядок подготовки ОС к работе, команды оператора, управление порядком прохождения заданий в системе MVT, работа с некоторыми программами обслуживания. Рассматриваются команды дисплей-консоли и управления режимами консолей, системы ДУВЗ и КРОС.

Для студентов вузов, а также для учащихся техникумов и курсов по подготовке операторов.



57557

В 1702070000—137 КБ-21-54-83
053(02)-83

© Издательство «Наука».
Главная редакция
физико-математической
литературы, 1983

ПРЕДИСЛОВИЕ

Электронные вычислительные машины Елиной системы (ЭВМ ЕС) широко используются практически во всех областях народного хозяйства. Их эффективная эксплуатация во многом зависит от хорошей подготовки операторов ЭВМ.

Основными обязанностями оператора являются: запуск, останов и повторный запуск (рестарт) операционной системы (ОС); управление прохождением потока заданий на ЭВМ; подготовка томов прямого доступа к работе и т. д. Поэтому он должен: уметь работать со всеми устройствами ввода-вывода, знать операторский пульт и правила работы с ним, свободно пользоваться командами оператора; разбираться в сообщениях системы и отвечать на них, быть знакомым с основными компонентами ОС и с языком управления заданиями.

Для начинающего оператора можно рекомендовать главы 1—7. В главе 1 читатель знакомится с основными характеристиками ЕС ЭВМ, комплектацией ее моделей, логической структурой ЭВМ, системой прерываний. В Главе 2 дается описание работы устройства ввода-вывода. В ней основное внимание уделяется работе дисплей-консоли, которая все более прочно входит в практику работы операторов. При изложении этих глав авторы считали, что читатели знакомы с основными техническими и логическими особенностями построения ЭВМ.

В главе 3 излагаются сведения об операционной системе (ОС) и языке управления заданиями. Здесь приводится общая характеристика команд оператора. Этим подчеркивается связь команд оператора с языком управления заданиями. Сам язык управления заданиями рассматривается не полностью. В главе приводятся только те сведения об языке управления заданиями, которые особенно необходимы оператору при работе с заданиями программистов на ЭВМ. Остальные сведения, углубляющие знание языка управления заданиями, изучаются в главе 9.

В главе 4 дано описание работы информационных команд. Информационные команды позволяют получить информацию о состоянии системы и отвечать на ее запросы.

В главе 5 описана группа команд, предназначенная для управления состояниями устройств ввода-вывода.

В главе 6 рассматриваются команды управления системными программами, которые предназначены для обработки заданий программистов на ЭВМ. Правильное управление системными программами является очень важным с точки зрения наилучшей эксплуатации ЭВМ.

Глава 7 посвящена работе оператора с ОС, а именно, ее загрузке. Материал главы 7 рекомендуется изучать после того, как читатель ознакомился со всеми командами, рассмотренными в главах 4—6.

Оператору средней квалификации, кроме глав 1—7, следует ознакомиться с главами 8 и 9. В главе 8 рассматривается вопрос управления порядком прохождения заданий на ЭВМ. В главе 9 изучается язык управления заданиями в объеме, необходимом для работы оператора-программиста.

Главы 12 и 13 посвящены изучению работы оператора с системами ДУВЗ и КРОС, облегчающими процесс управления прохождением заданий на ЭВМ.

В основу учебного пособия положены версии ОС 4.1 и 6.1.

Главы 1, 3—13 написаны Востриковой З. П., глава 2 написана Митиным С. Т. и Востриковой З. П.

Авторы выражают искреннюю признательность Милюкову А. Г., Мызиной С. А., Кузьмину А. В., Бутягиной Н., Богомазовой В. Т., Яшиной Л. В., Янкелевичу А. П. за помощь, полезные советы и замечания, которые были сделаны ими при подготовке настоящего пособия к изданию.

З. П. Вострикова, С. Т. Митин

1.1. Характеристики ЭВМ

К настоящему времени в семейство ЕС ЭВМ входят два подмножества: машины первой очереди «Ряд 1» и машины второй очереди «Ряд 2». В «Ряд 2» входят модели ЕС-1015, ЕС-1025, ЕС-1035, ЕС-1045, ЕС-1053, ЕС-1060 и ЕС-1065. Эти модели разработаны на основе опыта работы с моделями «Ряд 1», который образуют ЭВМ ЕС-1010, ЕС-1020, ЕС-1021, ЕС-1030, ЕС-1040, ЕС-1050, ЕС-1012, ЕС-1022, ЕС-1033 и другие. В основу разработки ЭВМ ЕС положен модульный принцип, который позволяет наращивать и видоизменять технические средства и его программное обеспечение. Программное обеспечение помогает более эффективно использовать ЭВМ для решения различных задач с минимальным участием оператора в работе машины. Программное обеспечение как бы расширяет технические возможности машины.

Основными характеристиками ЭВМ являются быстродействие и емкость оперативной памяти (табл. 1.1). *Быстродействие* определяется количеством

Таблица 1.1. Основные характеристики ЭВМ ЕС

Модели системы	Быстродействие (тыс. оп./с)	Емкость оперативной памяти (К байт)
ЕС-1020	10–20	64–256
ЕС-1022	80–90	128–512
ЕС-1030	40–60	128–512
ЕС-1032	200	128–512
ЕС-1033	120–200	128–512
ЕС-1035	140–160	256–1024
ЕС-1040	380–400	128–1024
ЕС-1045	540–880	до 4 М байт
ЕС-1050	480–520	128–1024
ЕС-1055	450–600	1024–2048
ЕС-1060	1 300	до 8192
ЕС-1065	4 500	до 16 324

операций в секунду, выполняемых процессором. Из таблицы видно, что быстродействие ЭВМ находится в пределах от 20 тыс. оп./с для ЕС-1020 и до 450 тыс. оп./с для ЕС-1065. *Емкость памяти* — это наибольшее количество единиц информации, которое может одновременно храниться в памяти. Емкости памяти находятся в пределах от 64 К байт до 16 324 К байт. Машины Единой системы отвечают современному уровню вычислительной техники, рассчитаны на серийное производство, удобны в эксплуатации и позволяют обеспечить различные потребности народного хозяйства.

1.2. Структура ЭВМ

Общую логическую структуру любой модели ЕС ЭВМ можно представить в виде обобщенной схемы, приведенной на рис. 1.1. Основными блоками являются оперативная память (ОП), процессор, каналы ввода-вывода (СК1 — СК6 — селекторные каналы, МК — мультиплексный канал), устройства ввода-вывода (УВВ) и блок управления.

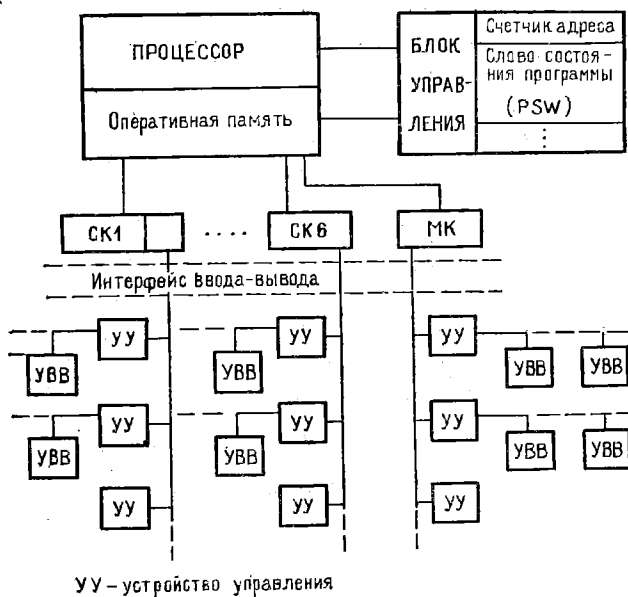


Рис. 1.1.

Оперативная память является центральным блоком машины. Она служит для записи в нее информации, хранения и выдачи ее разным блокам машины. ОП обменивается информацией с процессором и через каналы с УВВ. Минимально адресуемой единицей информации в памяти является *байт*, который состоит из восьми двоичных разрядов (*битов*) для представления данных. Каждый байт, расположенный в памяти, имеет адрес, который определяет его местонахождение в памяти. Адреса в памяти начинаются с 0 для первого байта и последовательно возрастают на единицу для каждого последующего байта. Максимальное значение адреса равно 16 777 215. Основной характеристикой памяти является емкость. Единица изменения емкости 1 К байт равен 1024 байт; 1 М байт — 1024 К байт.

Процессором называется основная часть ЭВМ, непосредственно осуществляющая процесс обработки информации и управляющая им. В процессоре сосредоточены все средства для выполнения арифметических и логических операций, средства обращения к памяти, средства управления выполнением заданной последовательности команд, организации начала обмена данными между оперативной памятью и устройствами ввода-вывода. В зависимости от модели процессор может выполнять от 143 до 183 машинных команд. С помощью машинных команд процессор выполняет все арифметические и логические операции.

Каналы представляют собой малую ЭВМ со своей системой управления, своей системой команд, которая обеспечивает обмен информацией между памятью и устройствами ввода-вывода. Имеется два типа каналов — селекторные (от 2 до 6) и один мультиплексный.

Селекторный канал работает в каждый момент времени только с одним устройством ввода-вывода, т.е. в *монопольном режиме*. К селекторному каналу подсоединяются устройства с достаточно высоким быстродействием (до 1,3 Мбайт/с) такие, как накопители на магнитных дисках (НМД) и накопители на магнитных лентах (НМЛ). К каждому каналу может быть подсоединено от 1 до 8 подканалов, а к каждому подканалу — до 8 НМД или до 16 НМЛ. Устройства ввода-вывода подключаются к подканалам с помощью устройств управления (УУ).

Мультиплексный канал рассчитан на работу в двух режимах: в *мультиплексном*, когда одновременно работают несколько устройств ввода-вывода; в *монопольном*, когда канал работает только с одним устройством. Мультиплексный канал может иметь до 224 мультиплексных подканалов и до двух селекторных подканалов. К мультиплексному каналу подсоединяются медленно действующие устройства. К таким устройствам относятся: устройства ввода с перфокарт и перфолент, пишущие машинки, АЦПУ и др. Максимальное число устройств, которое может быть подсоединено к мультиплексному каналу, равно 256. В старших моделях каналы выделены в отдельное устройство. В младших моделях канал конструктивно расположен в процессоре, так как логически тесно с ним связан.

Интерфейс ввода-вывода — это блок, который соединяет два устройства, например канал с УУ УВВ. Он представляет собой систему стандартного соединения, которая позволяет подключать к каналам разнотипные устройства ввода-вывода, а следовательно, произвольно устанавливать конфигурацию ЭВМ в соответствии с запросами потребителей. Интерфейс обеспечивает полную взаимозаменяемость однотипных устройств ввода-вывода.

Устройство ввода служит для передачи данных в ОП. В качестве устройства ввода используются устройства ввода с перфокарт и перфолент, магнитные ленты, диски, пишущая машинка, дисплей и графопостроители.

Устройство вывода обеспечивает передачу данных из ОП на перфокарты, перфоленты, магнитные ленты, диски, АЦПУ, графические устройства вывода.

Блок управления ЭВМ обеспечивает все режимы ее работы, в том числе и выборку команд с помощью счетчика адреса, имеющего 24 двоичных разряда. В блоке управления находится слово состояния программы (PSW), в котором собирается вся информация, отражающая состояние работы машины в текущий момент времени.

Кроме перечисленных устройств, в состав моделей входят устройства, которые непосредственно не связаны с ЭВМ. К ним относятся *устройства подготовки данных к вводу в машину, контрольно-считывающие устройства, перфораторы* и т. д.

1.3. Комплектация моделей

В состав каждой модели ЭВМ входят комплекты устройств и система программного обеспечения. Для каждой модели существует минимальный состав технических средств, обязательный при комплектации (Приложение 1).

Модульная структура позволяет приспособить ЭВМ наилучшим образом к решению определенного класса задач. На рис. 1.2 приведена в качестве

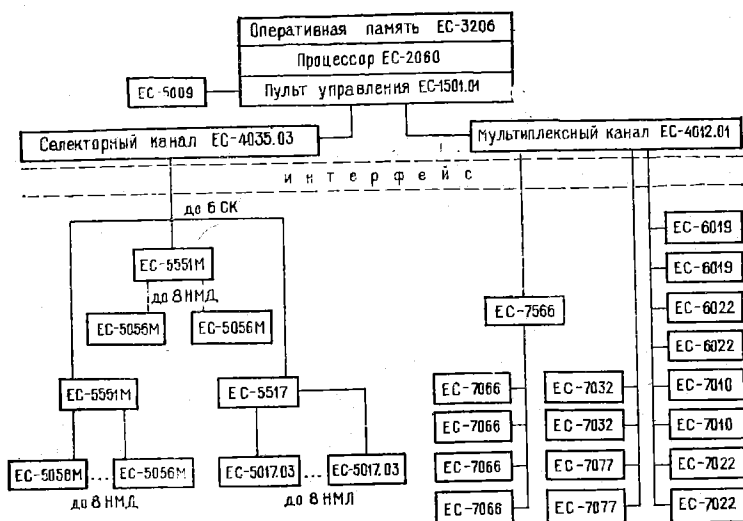


Рис. 1.2.

примера структурная схема ЭВМ ЕС-1060. В зависимости от решаемых задач конфигурация ЭВМ может быть изменена за счет изменения количества блоков ОП и устройств ввода-вывода.

1.4. Системы счисления

Конструкция любой ЭВМ тесно связана с используемой системой счисления. *Системой счисления* называется позиционное представление чисел, в котором последовательные цифровые разряды являются последовательными целыми степенями целого числа, называемого основанием. *Основание системы счисления* — это отношение соседних разрядов. Например, в десятичной системе счисления, основанием которой является число 10, каждый следующий старший разряд в десять раз больше предыдущего.

Целое число M в любой системе счисления с основанием n записывается в виде

$$M = a_k a_{k-1} \dots a_1 a_0,$$

где $a_k a_{k-1} \dots a_1 a_0$ — цифры от 0 до $n-1$. Например, в десятичном числе 19: $a_1 = 1$ и $a_0 = 9$. Развернутая форма записи целого числа имеет вид

$$M = a_k n^k + a_{k-1} n^{k-1} + \dots + a_1 n + a_0. \quad (1)$$

Десятичное число 19 будет иметь следующее изображение:

$$1 \cdot 10 + 9 = 19.$$

В ЕС ЭВМ используется двоичная система счисления (2 с.с.), поскольку элементы, на которых построена ЭВМ, имеют два устойчивых состояния.

Кроме того, широко применяется шестнадцатеричная система счисления (16 с.с.).

Двоичная система счисления. Для изображения чисел в двоичной системе счисления требуются только две цифры: 0 и 1. Сложение выполняется по тому же принципу, что и в десятичной системе счисления: если в данном разряде образуется сумма, которая не уместается в нем, то превышение переносится в следующий старший разряд. Таблица сложения двух однозначных чисел в двоичной системе счисления имеет следующий вид:

$$\begin{array}{l} 0 + 0 = 0 \\ 0 + 1 = 1 \\ 1 + 0 = 1 \\ 1 + 1 = 10 \end{array}$$

Из последней строки таблицы видно, что десятичное число 2 в двоичной системе счисления имеет вид 10_2 и является основанием системы счисления. Для того чтобы различать, в какой системе счисления записано число, в дальнейшем основание системы счисления будет указываться в виде индекса. Индекс записывается десятичными цифрами. При записи цифр в десятичной системе счисления индекс не указывается.

Пользуясь таблицей сложения, приведенной выше, получим число 3 путем прибавления к числу 2 единицы:

$$\begin{array}{r} + 10 \\ 1 \\ \hline 11_2 \end{array}$$

Представление первых 20 чисел натурального ряда в двоичной и шестнадцатеричной системах показано ниже.

Изображение чисел в системах счисления

Десятичной	Двоичной	Шестнадцатеричной	Десятичной	Двоичной	Шестнадцатеричной
0	0	0			
1	1	1	11	1 011	B
2	10	2	12	1 100	C
3	11	3	13	1 101	D
4	100	4	14	1 110	E
5	101	5	15	1 111	F
6	110	6	16	10 000	10
7	111	7	17	10 001	11
8	1000	8	18	10 010	12
9	1001	9	19	10 011	13
10	1010	A	20	10 100	14

Перевод целых чисел из двоичной системы счисления в десятичную производится методом подстановки. Для этого необходимо представить число в развернутой форме (1), записав все его цифры $a_k, a_{k-1}, \dots, a_1, a_0$ и основание 2 в десятичной системе счисления. Далее вычисляем, например,

$$1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2 + 1 = 19.$$

Перевод целых чисел из десятичной системы счисления в двоичную выполняется последовательным делением нацело десятичного числа на основание

двоичной системы счисления. Разделим число M на 2. В результате получим некоторое частное, которое обозначим b_0 , и целый остаток a_0 :

$$M = b_0 \cdot 2 + a_0. \quad (2)$$

Полученное частное b_0 снова разделим на 2; получим новое частное b_1 и остаток a_1 :

$$b_0 = b_1 \cdot 2 + a_1.$$

Подставим полученное выражение в (2):

$$M = (b_1 \cdot 2 + a_1) 2 + a_0 = b_1 \cdot 2^2 + a_1 \cdot 2^1 + a_0 \cdot 2^0.$$

Частное от деления b_1 снова следует разделить на 2. Последовательное деление надо продолжать до тех пор, пока последнее частное от деления не окажется меньше основания 2. На этом процесс деления заканчивается, и число оказывается представленным в виде выражения (1). Первый остаток a_0 есть младший разряд числа в новой системе счисления. Переведем, например, число 19 в двоичную систему счисления. Для перевода следует число 19 последовательно делить на число 2, как показано ниже:

$$\begin{array}{rcl} \begin{array}{r} - 19 \overline{) 2} \\ \underline{-18} \\ a_0 = 1 \end{array} & b_0 = \begin{array}{r} - 9 \overline{) 2} \\ \underline{-8} \\ a_1 = 1 \end{array} & b_1 = \begin{array}{r} - 4 \overline{) 2} \\ \underline{-4} \\ a_2 = 0 \end{array} & b_2 = \begin{array}{r} - 2 \overline{) 2} \\ \underline{-2} \\ a_3 = 0 \end{array} & b_3 = 1 \end{array}$$

В результате получим число 19 в двоичной системе счисления, равное 10011. Обратный перевод этого числа в десятичную систему счисления показан выше.

На практике часто при переводе из десятичной системы счисления в двоичную и обратно пользуются готовыми таблицами степеней 2^n (Приложение 2). Для перевода следует подобрать ближайшее меньшее или равное десятичное число. Например, для числа 18 таким числом будет 16 (см. колонка 1 таблицы, которая приводится ниже). В колонке 3, которая используется для изображения числа в двоичной системе счисления, против числа 16 поставить 1. Из числа 18 вычесть 16, получим 2. В колонке 3 против числа 2 поставим 1. В результате получим число 18 в двоичной системе счисления: 10010.

2^n	n	Число в двоичной системе счисления
1	2	3
1	0	0
2	1	1
4	2	0
8	3	0
16	4	1

Шестнадцатеричная система счисления имеет основание 16. Каждый ее разряд может принимать значения от 0 до 15. Для изображения чисел ис-

пользуются десятичные цифры от 0 до 9 и буквы латинского алфавита от A до F.

Перевод из шестнадцатеричной системы счисления в десятичную осуществляется методом подстановки, например:

$$13_{16} = 1 \cdot 16 + 3 = 19, \quad 2C_{16} = 2 \cdot 16 + 12 = 44.$$

Перевод целого десятичного числа в шестнадцатеричную систему счисления осуществляется методом последовательного деления нацело на 16. Переведем число 19 в шестнадцатеричную систему счисления:

$$\begin{array}{r|l} 19 & 16 \\ \hline 16 & \\ \hline a_0 = 3 & b_1 = 1 \end{array}$$

Получим число 13_{16} .

Перевод целых чисел из шестнадцатеричной системы счисления в двоичную и обратно достаточно прост, так как основание шестнадцатеричной системы счисления $16 = 2^4$ относится к основанию двоичной системы счисления как 4:1. Это значит, что каждые четыре разряда числа в двоичной системе составляют один шестнадцатеричный разряд. Например, для того чтобы перевести двоичное число 10011 в шестнадцатеричную систему счисления разобьем его на тетрады (четыре двоичных разряда) справа налево:

0001		0011		двоичные цифры
1		3		шестнадцатеричные цифры

Если левая, последняя тетрада будет неполной, то нужно дополнить ее нулями. Затем каждую тетраду записать с помощью шестнадцатеричных цифр. Перевод из шестнадцатеричной системы счисления в двоичную выполняется обратным действием, например:

7		C		3		шестнадцатеричные цифры
0111		1100		0011		двоичные цифры

1.5. Система прерываний

Прерывание — это временное прекращение выполнения ЭВМ последовательности команд одной программы с целью выполнения последовательности команд другой программы. Прерывания могут возникнуть по ряду причин, и в некоторых случаях при прерывании ОС выдает информацию о причине прерывания, состоянии системы и задания. Эта информация содержится в системном коде аварийного завершения, коде состояния ожидания и сообщениях ОС. *Системный код завершения* — это трехзначное шестнадцатеричное число, которое вырабатывается управляющей программой и соответствует определенной аварийной ситуации завершения задания или пункта задания. *Код состояния ожидания* представляет собой число, которое устанавливается в *слове состояния программы (PSW)*.

Коды состояния ожидания вырабатываются управляющей программой и высвечиваются на пульте процессора.

В PSW, высвечиваемом на пульте процессора, выдается информация, по которой оператор может определить причину перехода процессора в состояние

ожидания, пользуясь технической документацией, в которой приведены все коды ожидания и дана их интерпретация. Слово состояния программы имеет 64 двоичных разряда (или 16 шестнадцатеричных разрядов).

Слово состояния программы

SM		K		AMWP		IC		xxxxx		zzz	
0	7	8	11	12	15	16	31	32	51	52	6

SM — маска системы, которая управляет внешними прерываниями и прерываниями от ввода-вывода с помощью своих двоичных разрядов, которые называются: 0-й разряд — маска мультиплексного канала; 1—6-й разряды — маски селекторных каналов с 1-го по 6-й; 7-й разряд — маска от внешних прерываний. Указанные выше разряды масок PSW могут находиться в одном из двух состояний: 0 или 1. Если соответствующий разряд маски содержит 0, то считается, что прерывания во время выполнения программы не происходят. В этом случае прерывания замаскированы, т.е. запрещены. Позднее замаскированные прерывания, которые запоминаются ЭВМ, могут быть обработаны процессором либо сброшены. Если один из вышеуказанных разрядов масок содержит 1, то при появлении внешнего прерывания или сигнала о неправильной работе ввода-вывода процессор прерывает свою работу.

K — ключ защиты памяти исключает возможность ошибочного обращения из одной программы в область памяти, где находится другая программа или программы операционной системы. Каждой программе отводится определенная область памяти (*раздел*) и присваивается ключ защиты: с одной стороны, памяти и, с другой стороны, программе. Ключ защиты при работе данной программы находится в слове состояния программы. Использование области памяти разрешается только в случае, если ключи защиты памяти и выполняемой команды совпадают. Если ключ защиты памяти программы не совпадает с ключом защиты, имеющимся в слове состояния программы, то происходит нарушение защиты памяти и прерывание программы.

AMWP занимают четыре двоичных разряда. Назначение разрядов следующее:

A — режим работы машины с разными кодами. Если ЭВМ работает с кодом ДКОИ, то 12-й разряд равен 0; если КОИ-8, то 12-й разряд равен 1.

M — маска прерывания от схем. контроля машины. Если $M = 1$, то происходят прерывания при сбоях машины; если $M = 0$, то машинные сбоя игнорируются.

W — состояние «ожидание». Если $W = 1$, то процессор находится в состоянии «ожидание»; если $W = 0$, то процессор находится в состоянии «счет», при котором процессор работает.

P — состояние «задача». Если $P = 1$, то процессор находится в состоянии «задача»; если $P = 0$, то процессор — в состоянии «супервизор». Разряды 14 и 15 взаимосвязаны: они обеспечивают три режима работы процессора. В состоянии «ожидание» процессор ожидает задание, а в состоянии «счет» процессор выполняет вычисления, причем, если он находится в состоянии «задача», то выполняется программа пользователя, а в состоянии «супервизор» выполняется ввод-вывод данных.

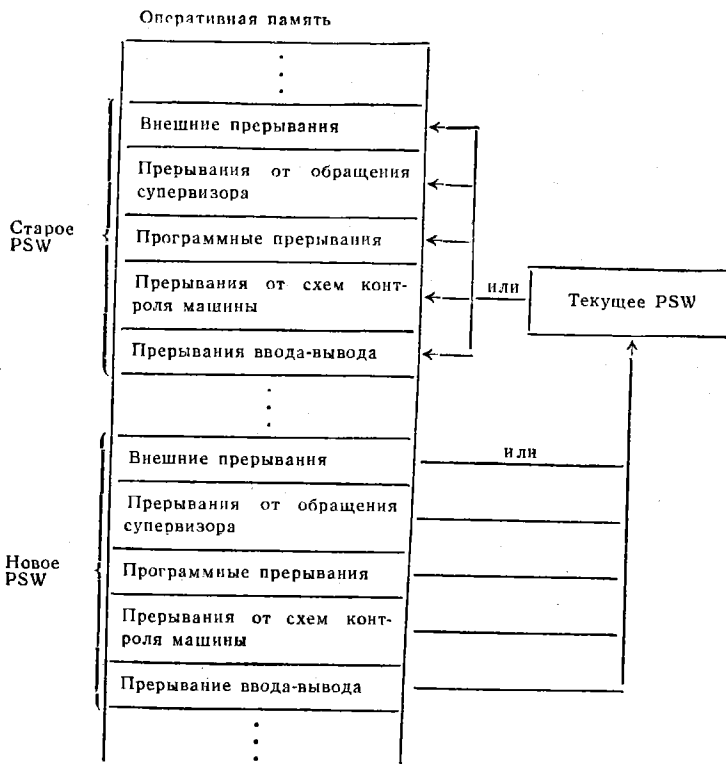


Рис. 13. Механизм прерывания.

Таблица 1.2. Адреса постоянно распределенной области памяти

Адрес в десятичной системе счисления	Назначение
0	PSW программы начальной загрузки
8	CCW ₁ > > >
16	CCW ₂ > > >
24	Старое PSW для внешних прерываний
32	Старое PSW для обращения к супервизору
40	Старое PSW для программных прерываний
48	Старое PSW для прерываний от схем контроля машины
56	Старое PSW для прерываний от ввода-вывода
64	Слово состояния канала CSW
72	Адресное слово канала CAW
76	Не используется
80	Таймер
84	Не используется
88	Новое PSW для внешних прерываний
96	Новое PSW для обращения к супервизору
104	Новое PSW для программных прерываний
112	Новое PSW для прерываний от схем контроля машины
120	Новое PSW для прерываний от ввода-вывода
128	Область памяти для информации о состоянии машины при диагностике

IC — код прерывания позволяет ОС определить причину прерывания. Информация, которая помещается в *IC*, включает адрес устройства, от которого пришел сигнал о его неправильной работе, номер канала, номер устройства и т. д. В случае появления сигнала о прерывании устройство управления ЭВМ либо прерывает выполнение программы, либо запоминает это прерывание, а затем его обрабатывает или совсем его сбрасывает. При прерывании в определенных ячейках оперативной памяти запоминается текущее PSW (старое PSW) и выбирается новое PSW из постоянно распределенной области ОП (табл. 1.2 и рис. 1.3). Старое PSW содержит адрес следующей команды, которая была бы выполнена, если бы не было прерывания, код длины последней выполнявшейся команды и т. д. Новое PSW содержит адрес программы ОС, которая обрабатывает данный тип прерывания. После окончания обработки прерывания, в зависимости от результата анализа причины прерывания, управление может быть возвращено к прерванной программе или передано другой программе.

Все прерывания в ЕС ЭВМ разбиты на пять классов:

- 1) прерывания от схем контроля машины (приоритет наивысший 1);
- 2) программные прерывания (приоритет 2);
- 3) прерывания от обращения к супервизору (приоритет 2);
- 4) внешние прерывания (приоритет 3);
- 5) прерывания ввода-вывода (приоритет 4).

Для каждого класса прерываний выделена постоянная область памяти, в которой помещается значение старого PSW, и область памяти, из которой выбирается новое слово состояния программы (рис. 1.3). Адреса постоянно распределенной области памяти для размещения значений PSW приведены в табл. 1.2. Все прерывания независимо от класса прерывания обрабатываются машиной одинаково, а именно, запоминается состояние PSW, которое было перед прерыванием программы; управление передается программе, обслуживающей данный класс прерывания; управление возвращается в зависимости от причины прерывания либо прерванной программе, либо другой программе.

Вся левая половина PSW (с 0-го по 31-й разряд) может иметь одну из двух форм: $FF0S0000_{16}$ означает, что в системе нет готовых задач для выполнения, а $000S0000_{16}$ говорит о том, что состояние ожидания вызвано ошибкой. В обоих случаях буква *S* обозначает одну из возможных комбинаций разрядов 12—15 (*AMWP*), отражающих режим работы процессора.

$xxxx_{16}$ — в этих разрядах находится информация, которая отражает состояние выполняемой программы.

zzz_{16} — содержит код состояния ожидания. Коды ожиданий и возможные действия оператора в случае их появления приведены в [2].

2.1. Пультовая пишущая машинка

Пультовая пишущая машинка относится к устройствам непосредственной связи оператора с ЭВМ. Она позволяет вводить команды в ЭВМ с клавиатуры пишущей машинки с одновременной их печатью на бумаге, а также получать на бумаге сообщения из ЭВМ о ходе решения заданий. Некоторые модели пультовых машинок могут быть дополнены средствами дистанционного управления, дублирующими некоторые операции, которые обычно выполняются с пульта управления процессора.

Основными составными частями пультовых пишущих машинок являются: *печатающий механизм* (пишущая машинка Консул-260); *пульт управления*, позволяющий следить или задавать режим работы устройства; *устройство управления*, которое обеспечивает связь устройства с каналами через стандартный интерфейс ввода-вывода и управляет работой самого устройства.

Технические характеристики пишущей машинки Консул-260 (ЕС-7070) следующие: скорость печати до 10 зн./с; число знаков в строке 106; ширина бумажного рулона 280—320 мм; расстояние между строками 4,5 мм; красящая лента двухцветная; количество копий 5; имеется блокировка при одновременном нажатии двух клавиш. На клавиатуре машинки размещено 92 печатных знака (46 литерных рычагов). Для знака «пробел» используется центральная клавиша. В набор знаков входят 26 прописных латинских буквы от А до Z, 10 цифр от 0 до 9, прописные русские буквы и 26 специальных знаков. Расположение знаков на клавиатуре показано на рис. 2.1. Управляющие клавиши: → — возврат на один шаг вправо; ← — возврат на один шаг влево; ⏏ — возврат каретки с переводом на новую строку; ↓ — перевод на новую строку; ВР — верхний регистр; НР — нижний регистр; ЗБ — забой.

На пульте управления располагаются ручки, кнопки управления и лампочки индикации (рис. 2.2). Чтобы подготовить пишущую машинку к работе, следует поставить специальный ключ с микровыключателем в положение ВКЛ. В результате на пишущую машинку подается напряжение питания и загорается лампочка ПИТАНИЕ ВКЛ. Далее следует нажать кнопку ГОТОВ. В ответ загораются лампочки ГОТОВ и ВВОДИТЕ. Это значит, что устройство готово к работе. Если устройство не готово, например находится в автономном состоянии, то загорается лампочка НЕ ГОТОВ. Это значит, что на другом, техническом пульте следует переключить устройство в состояние КОМПЛЕКС, т.е. логически подсоединить его к ЭВМ. В случае сбоя оборудования загорается лампочка ОБОРУД. В этом случае следует обратиться к техническим работникам, обслуживающим ЭВМ, и после ликвидации неполадок нажать кнопку СБРОС.

Ввод команды с пульта пишущей машинки начинается с нажатия кнопки ВН (внимание). Для устройства это означает, что оператор собирается набирать на клавиатуре команду. После нажатия кнопки ВН загораются лампы ВВОДИТЕ, ВН и ЧТ (чтение). После этого оператор может начать

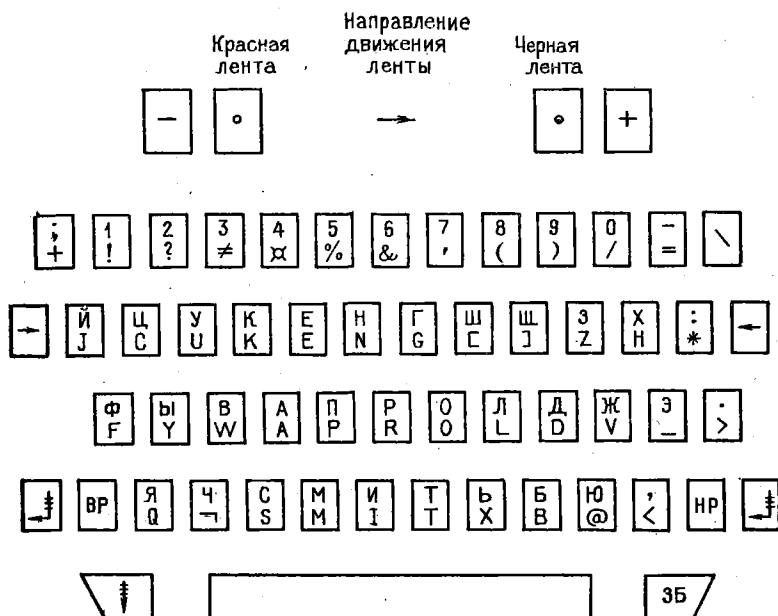


Рис. 2.1.

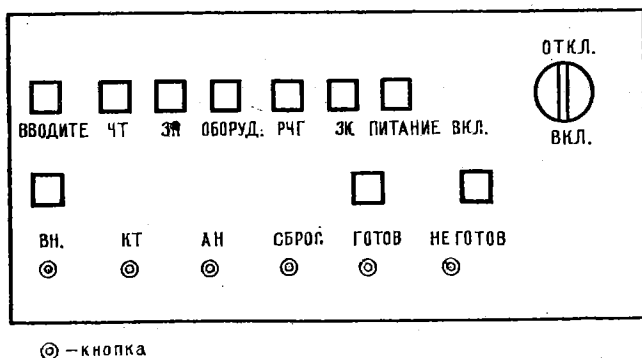


Рис. 2.2.

набирать команду на клавиатуре. Если оператор во время набора команды сделал какую-либо ошибку, то следует нажать кнопку АН (аннулировать), после чего каретка машинки устанавливается на новую строку и оператор снова может набирать на клавиатуре команду. После того как оператор

закончил набирать команду, он должен нажать кнопку КТ (конец текста). Лампочки ВВОДИТЕ и ЧТ выключаются, но через небольшое время лампочка ЧТ снова загорается. Включенная лампочка ЧТ говорит о том, что команда считывается в буферную память устройства. Как только чтение закончится, гаснет лампочка ЧТ и загорается лампочка ЗП (запись), которая говорит о том, что команда записывается в ОП. Если при работе с клавиатурой окажется, что рычаг с литерой не вернулся в исходное положение, то загорается лампочка РЧГ (рычаг). Если каретка пишущей машинки не перемещается, загорается лампочка ЗК (заклинивание). В обоих этих случаях следует ликвидировать технические неполадки и нажать кнопку СБРОС.

В режиме выдачи сообщений для печати клавиатура пишущей машинки блокируется и загорается лампочка ЗП.

2.2. Дисплей-консоль оператора

В настоящее время дисплейные комплексы получили широкое распространение. Они используются в качестве пультов операторов, абонентских пунктов подготовки данных, для отладки программ и т. д. Для операторов в первую очередь следует знать, как дисплей используется в качестве консоли вместо пультушной пишущей машинки или совместно с пишущей машинкой. Наиболее распространены дисплейные комплексы ЕС-7906 и ЕС-7920.

Технические характеристики дисплейного комплекса ЕС-7920. В состав комплекса могут входить до 32 дисплеев и печатающих устройств (АЦПУ). Разработаны два вида комплексов: *локальные* и *дистанционные*. Локальные комплексы подключаются к каналу ЭВМ на удалении не более 1200 м, а дистанционные — на расстоянии свыше 1200 м. Скорость обмена информацией между локальными групповыми комплексами находится в пределах 180—200 К байт/с, а дистанционных — 600, 1200, 2400 и 4800 бит/с. Комплекс ЕС-7920 может использоваться в справочно-информационных системах, системах сбора и подготовки данных и как пульты операторов.

К локальным групповым комплексам относятся модели ЕС-7920-00 (01, 02 и 03). Они подключаются к ЭВМ через мультиплексные или селекторные каналы. Дистанционные групповые комплексы ЕС-7920-10 (11 и 12) подключаются к мультиплексному или селекторному каналам через линии связи с помощью *модемов* или *мультиплексоров передачи данных*, т. е. устройств, обеспечивающих подготовку и передачу данных на расстоянии.

В состав дисплейного комплекса ЕС-7920 входит дисплей ЕС-7927, с которым непосредственно работает оператор. Дисплей имеет экран и пульт управления. Экран дисплея имеет 24 строки по 80 колонок в строке (рис. 2.3). Он делится на четыре области: информации состояния (строки 1—19), сообщений в ответ на ввод команд оператора (строки 20 и 21), ввода команд оператора (строки 22 и 23) и выдачи предостережений (строка 24).

В каждой строке области информации состояния информация располагается определенным образом: колонки 1 и 2 отводятся под номер строки или номер сообщения; в колонку 3 помещается вертикальная черта (|) для удаляемых сообщений и для сообщений, которые не требуют никаких действий со стороны оператора, или горизонтальная черта (—), отмечающая информационные сообщения, и пробел для сообщений, которые требуют

принятия решения оператора; в колонке 4 звездочка * или @ отмечают сообщения, которые требуют действий со стороны оператора (пробел в случае, если сообщения не требуют никаких действий со стороны оператора и если в колонке 3 находится вертикальная или горизонтальная черта); в колонках с 5 по 74—79 размещается текст системных сообщений и команд оператора после их ввода в ЭВМ. В строке 20 появляются номера функциональных клавиш (от 1 до 12), которые выдаются в ответ на ввод команды оператора K D,PFK

← 80 колонок →			
Колонки 1—2	Колонка 3	Колонка 4	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19			Область информации состояния
			Область сообщений оператору в ответ на ввод команды
			Две строки области ввода команд оператора
			Строка предостережений
Номера строк	Пробел, или —	Пробел, * или @	← Текст сообщений 76 символов →

Рис. 2.3. Расположение областей на экране дисплей-консоли ЕС-7927.

(см. гл. 11). В строке 21 выдаются сообщения об ошибках оператора. В области ввода команд появляется изображение текста команды, которая набирается оператором на клавиатуре дисплея. Команды могут занимать не более 126 колонок области ввода. Строка 24 предназначена для сообщений оператору, относящихся непосредственно к работе дисплея.

Описанный выше экран называется *форматизированным*, т.е. разделенным на области. Если экран не разделен на области, то он называется *неформатизированным*. Каждая область может быть защищена от воздействия оператора или не защищена. Это зависит от назначения области. Все области данного экрана защищены, кроме области ввода команд оператора. Если оператор случайно начнет вводить команду в защищенную область, то на

пульте оператора включается лампочка ВВОД ЗАПРЕЩЕН и передача текста команды не выполняется (рис. 2.4).

Пульт управления дисплеем показан на рис. 2.4. Он состоит из панели, на которой находятся индикаторные лампочки, ручки управления и клавиатура.

Подключение дисплея к системе. При включении питания на панели дисплея загорается лампочка СЕТЬ. Чтобы подключить дисплей к системе, следует вставить специальный ключ в положение ОТКЛ, повернуть его

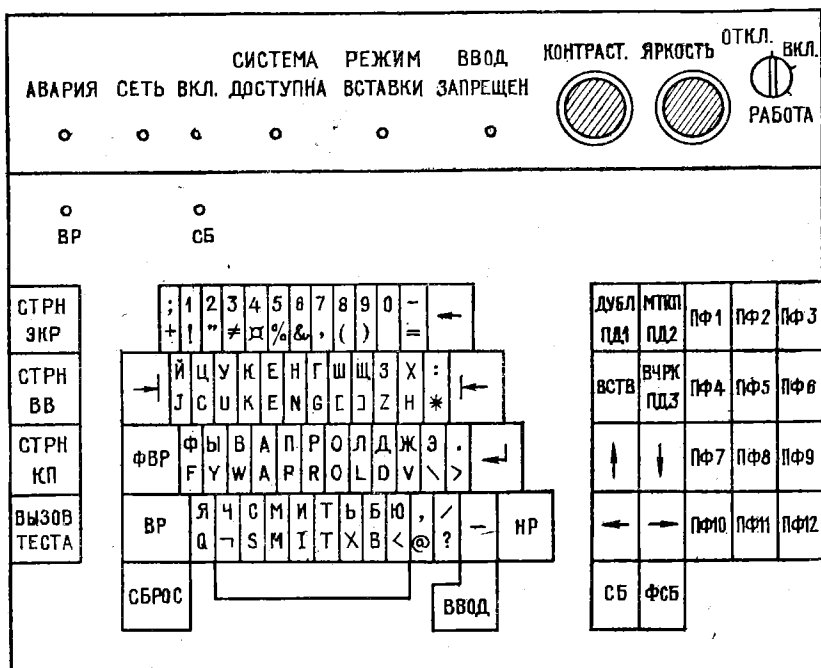


Рис. 2.4.

в положение ВКЛ и подождать пока загорятся лампочки ВКЛ и ВВОД ЗАПРЕЩЕН. После этого надо повернуть ключ в положение РАБОТА. При этом лампочка ВВОД ЗАПРЕЩЕН гаснет и загорается лампочка СИСТЕМА ДОСТУПНА, которая указывает на то, что устройство готово к работе, а на экране в верхнем левом углу появляется курсор. Курсор — это знак подчеркивания, который находится несколько ниже строчек, предназначенных для текстов сообщений и команд. Он постоянно высвечивается на экране. После загрузки ОС курсор устанавливается системой в первую колонку строки 22. Ручки КОНТРАСТ и ЯРКОСТЬ предназначены для установления нужной контрастности и яркости изображения текстов на экране. При повышении или понижении питания устройства загорается лампочка АВАРИЯ. Чтобы по окончании работы выключить дисплей, следует ключ вернуть в положение ОТКЛ.

Клавиатура. На клавиатуре расположены следующие типы клавиш: 49 алфавитно-цифровых и специальных знаков, 8 для управления курсором, 8 функциональных, 14 программного функционирования и 5 регистровых.

Клавиши управления курсором. Курсор устанавливается оператором каждый раз на то место, куда он в дальнейшем будет заносить алфавитно-цифровой или специальный знак с клавиатуры. Четыре клавиши для управления курсором сгруппированы в правой части клавиатуры. Они позволяют перемещать курсор по экрану дисплея. Клавиши управления курсором можно разделить на две группы: перемещения курсора по позициям строки и по областям (рис. 2.3). Клавиши перемещения никак не влияют на содержимое экрана. Однократное нажатие клавиши ВВЕРХ ↑ приводит к перемещению курсора вверх на одну строку без изменения колонки. Если курсор находился на первой строке экрана, то он появляется на последней строке той же колонки. Однократное нажатие клавиши ВНИЗ ↓ перемещает курсор вниз на одну строку в той же колонке. Если курсор находился на последней строке, то он перемещается на первую строчку. Однократное нажатие клавиши ВПРАВО → перемещает курсор вправо на одну колонку в той же строке. Если курсор находился в конце строки, то он появляется в первой колонке следующей строки или с конца последней строки экрана перемещается в первую колонку первой строки. Однократное нажатие клавиши ВЛЕВО ← перемещает курсор влево на одну колонку в той же строке. Если курсор находился в начале строки, то он перемещается в конец предыдущей строки или с начала первой строки в конец последней строки экрана. В правом верхнем углу центральной части клавиатуры имеется клавиша ←, которая называется ВОЗВРАТ. Маркировка клавиш ВЛЕВО и ВОЗВРАТ одинакова, так же, как и их функции.

Клавиши →| (ТАБУЛЯЦИЯ ВПРАВО), |← (ТАБУЛЯЦИЯ ВЛЕВО) и ←| (НОВАЯ СТРОКА) устанавливает курсор в определенные позиции в пределах областей. Рассмотрим действие этих клавиш при работе с форматизованным экраном в области ввода команд оператора, которая является незащищенной. Нажатие клавиши →| приводит к тому, что курсор автоматически движется вправо и пробегает (невидимо для глаз) все колонки и строки до конца области ввода команд оператора, а затем устанавливается в его первую колонку. При нажатии клавиши |← курсор автоматически движется влево, пробегает все колонки и строки и устанавливается в первой колонке области ввода команд оператора. Выбор клавиши зависит от того, как близко от первой колонки области ввода команд находится курсор. При нажатии клавиши ←| курсор перемещается в первую колонку следующей строки области ввода команд.

Регистровые клавиши. Прежде чем ввести на экран дисплея какой-либо знак с клавиатуры, следует установить в нужное положение клавиши регистров НР, ВР или ФВР. Нажатие клавиши НР (нижний регистр) переводит клавиатуру в состояние нижнего регистра. Если нажать клавишу ВР (верхний регистр), то клавиатура перейдет в состояние верхнего регистра. Состояние этой клавиши фиксируется клавишей ФВР (фиксация верхнего регистра). Если на пульте горит лампочка ВР, то фиксируется состояние ВР, если не горит, то состояние НР. Переход от фиксации одного состояния в другое производится повторным нажатием клавиши ФВР. Клавиши СБ (строчная

буква) и ФСБ (фиксация строчной буквы) являются резервными на случай усовершенствования клавиатуры и в данной конструкции не используются.

Центральную часть клавиатуры занимают *клавиши с латинскими, русскими и специальными знаками*. После того как положение клавиатуры и курсора установлено, можно нажимать нужную клавишу с алфавитно-цифровым или специальным знаком. Текст команды оператор должен набирать, начиная с 3-й колонки строки 22. В этой же колонке должен находиться курсор. Если команда не уместилась на строке 22, то курсор следует перевести в позицию 3 строки 23 и продолжить текст команды. Всего на обеих строчках может быть введено не более 126 знаков, включая пробелы. После ввода очередного знака курсор переходит вправо на следующую колонку строки. Если нужно сделать пробел, то нажимается центральная нижняя клавиша, которая называется ПРОБЕЛ и маркировки не имеет.

Клавиши программного функционирования: ВВОД, ПФ1 — ПФ12 (программное функционирование) и ВЫЗОВ ТЕСТА. При нажатии на любую из этих клавиш клавиатура блокируется, о чем свидетельствует лампочка ВВОД ЗАПРЕЩЕН, которая загорается. Клавиша ВВОД нажимается после того, как оператор набрал команду без ошибок и должен ввести ее в систему. Команда ВВОД не выполняется, если устройство занято выполнением операции, начатой устройством управления накопителей или фотоселектором (световым пером).

Фотоселектор — это малогабаритный фоточувствительный приемник. Он подсоединяется к пульту управления электрическим путем с помощью кабеля. На корпусе дисплея, справа от экрана имеется специальный держатель для фотоселектора. Фотоселектор может воспринимать сигналы от изображаемого на экране текста. Для этого надо поднести его к поверхности экрана. На своем конце фотоселектор имеет микровыключатель, который срабатывает при нажатии фотоселектора о поверхность экрана. Считанная информация запоминается в памяти устройства. Чтобы передать информацию в ЭВМ, надо повторно поднести фотоселектор к специальной области на экране, которая имитирует работу клавиши ВВОД, иногда для этой цели используется строка 24 экрана. Фотоселектором пользуются в информационно-справочных системах, когда на экране имеется несколько сообщений и надо выбрать какие-то из них. Могут быть и другие применения. Фотоселектором можно пользоваться, если устройство свободно от всех других видов работ с клавиатурой или передачей данных.

За клавишами ПФ1 — ПФ12 закрепляются команды, которые достаточно часто используются оператором. При нажатии любой из этих клавиш в строках 22, 23 области ввода команд появляется текст команды, закрепленной за этой клавишей. Чтобы ввести команду в ЭВМ, следует нажать клавишу ВВОД. Такой способ набора часто используемых команд упрощает работу оператора.

Клавиша ВЫЗОВ ТЕСТА используется техническими работниками при наладке дисплея с целью диагностики его работы.

К группе *функциональных клавиш* относятся клавиши СТРН ЭКР (стирание экрана), СТРН ВВ (стирание ввода), СТРН КП (стирание до конца поля), ДУБЛ (дублирование), МТП (метка конца поля), ВСТВ (вставка), ВЧРК (вычеркивание) и СБРОС. Эти клавиши предназначены для полного

или частичного стирания текстов, разрешают вставить пропущенные знаки в тексте или их вычеркнуть и т. д.

Исправление текста команды. Если при наборе текста была сделана ошибка, то следует подвести курсор под неправильный знак, нажать нужную клавишу на клавиатуре и знак будет исправлен. Чтобы курсор вернуть в нужное положение, следует использовать клавиши управления курсором. Нажатие этих клавиш не влияет на введенный на экран текст, т. е. он полностью сохраняется. Если нужно полностью удалить текст, начиная с места, где установлен курсор, до конца экрана, то следует нажать функциональную клавишу стирания до конца поля СТРН КП или многократно нажимать клавишу ПРОБЕЛ. Чтобы удалить всю информацию из области ввода, надо нажать функциональную клавишу стирания ввода СТРН ВВ, и курсор устанавливается в колонку 1. Если надо удалить какой-нибудь лишний знак из текста, то следует подвести курсор под этот знак и нажать функциональную клавишу вычеркивания ВЧРК. При этом весь текст, который находился справа от курсора, перемещается влево на одну позицию. С помощью функциональной клавиши ВСТВ можно вставить в текст пропущенную букву (или буквы). Для этого нужно установить курсор на то место, куда должен быть вставлен знак, нажать клавишу ВСТВ (на пульте загорится лампочка РЕЖИМ ВСТАВКИ) и нужную алфавитно-цифровую клавишу или клавишу со специальным знаком. В результате весь текст, начиная с установленной позиции курсора, сдвигается на одну позицию вправо. На том месте, где был установлен курсор, появится пропущенный знак. Клавиша СБРОС нажимается для того, чтобы выключить лампочки ВВОД ЗАПРЕЩЕН и РЕЖИМ ВСТАВКИ. Если в системе имеется ошибка, то команда сброса не выполняется. Клавиша стирания экрана СТРН ЭКР предназначена для стирания всей информации с экрана. При этом курсор устанавливается в первую колонку строки 1 экрана.

Дублирование алфавитно-цифрового или специального знака. Если оператору надо ввести поле, которое, начиная с некоторого места и до конца, состоит из одного и того же знака, то в этом случае можно этот знак не набирать на клавиатуре. Можно нажать клавишу ДУБЛ после того, как знак введен на экран один раз. На следующей колонке рядом с этим знаком появится символ <, а курсор установится в первую колонку строки 22. Ввод символа < означает, что специальная программа, которая имеется в устройстве, заполнит поле до его конца символом, который предшествует знаку <. В общем случае, когда экран состоит из нескольких незащищенных полей, курсор возвращается в первую колонку того незащищенного поля, в котором он находился перед нажатием клавиши ДУБЛ, а дублирование остатка поля выполняется, как было описано выше.

Ограничение длины поля. Для того чтобы в ЭВМ ввести не все поле, а только часть его, следует нажать клавишу МТКП в том месте, где оператор хочет ограничить поле. В колонке, определяемой курсором, при нажатии клавиши МТКП запишется символ >. Это значит, что в ЭВМ будут переданы данные, начиная с начала поля и до этого символа. Таким образом, сокращается объем информации передаваемой в ЭВМ, если она оказывается лишней или ненужной.

2.3. Накопители на сменных магнитных дисках

Накопители на сменных магнитных дисках в качестве внешней памяти используются для хранения достаточно больших массивов информации. Для выборки информации из НСМД требуется относительно небольшое время. Поэтому они используются в качестве запоминающих устройств, в которых

Таблица 2.1. Технические характеристики на сменных магнитных дисках

Технические характеристики	Шифр	
	ЕС-5031	ЕС-5066
Максимальный объем памяти, байт	29·10 ⁶	100·10 ⁶
Скорость вращения пакета дисков, об/мин	2400	3600
Число магнитных головок	20	20
Скорость передачи данных, К байт/с	156	83,3
Максимальное время доступа, мс	80	80
Минимальное время доступа (время перехода от цилиндра к цилиндру), мс	20	10
Среднее время доступа, мс	50	58,3
Максимальная плотность информации на дорожке, бит/мм	90	160
Расстояние между серединами соседних дорожек, мм	0,257	0,125
Габаритные размеры, мм — высота	975	1150
— ширина	772	670
— глубина	610	1135

информация собирается, хранится, обновляется и выдается с достаточно высоким быстродействием. В настоящее время в основном используются НСМД

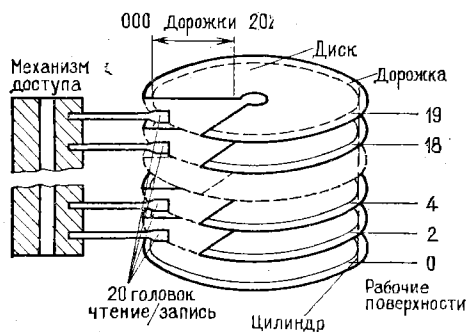


Рис. 2.5.

емкостью 29 М байт (ЕС-5061) и 100 М байт (ЕС-5066). Технические характеристики пакетов дисков емкостью 29 М байт и 100 М байт приведены в табл. 2.1.

В качестве примера рассмотрим конструкцию пакета на сменных магнитных дисках ЕС-5061 (рис. 2.5). Носителем информации является магнитный диск с ферромагнитным покрытием. Сменный пакет — это набор из 11 дисков, насаженных на одну ось. Информация записывается на рабочих поверхностях

дисков. К ним относятся все диски, кроме внешних (нижнего и верхнего в пакете), на которых информация не записывается. На каждой рабочей поверхности располагается 203 concentric дорожки (200 рабочих и 3 резервных). Нумерация дорожек идет от внешнего диаметра диска (дорожка № 000) к внутреннему (дорожка № 202). Плотность записи меняется от 60 бит/мм на внешней дорожке до 90 бит/мм на внутренней дорожке.

Механизм поиска нужной дорожки состоит из подвижной каретки и системы радиального перемещения. На каретке монтируется 20 магнитных головок записи-воспроизведения, которые предназначены для записи информации на диск и считывания информации. Нумерация магнитных головок от 0 до 19 производится снизу вверх. Каретка перемещается в радиальном направлении по отношению к пакету дисков с помощью электромагнитного привода. Система перемещения двигает магнитные головки внутри пакета, при этом магнитные головки не касаются поверхности дисков, а плавают на воздушной подушке, которая образуется при вращении дисков. Каждая рабочая

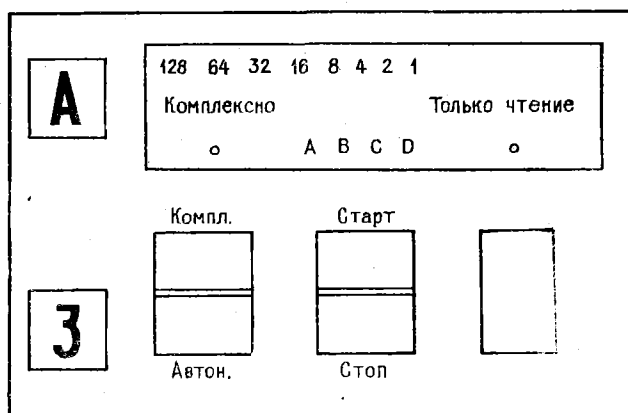


Рис. 2.6.

поверхность диска обслуживается одной магнитной головкой. При вращении дисков дорожки и система магнитных головок образуют поверхность цилиндра. Число таких concentric расположенных цилиндров в диске равно числу дорожек, т. е. 203. Переход с дорожки на дорожку в пределах одного цилиндра производится электронным путем. Информация, которая записывается в накопитель или считывается с него, поступает по одному каналу, т. е. в каждый момент времени работает только одна магнитная головка. Среднее время доступа 50 мс, время перехода на соседний цилиндр 20 мс и максимальное время доступа 80 мс.

Рассмотрим пульт управления накопителя на сменных магнитных дисках ЕС-5061. Пульт используется для включения, установки режимов работы и индикации состояний накопителя. Работа с НСМД начинается с установки фишки с последней цифрой адреса устройства, например 3, как показано на рис. 2.6. В распоряжении оператора имеются фишки с номерами от 0 до 7. Установка фишки с номером 3 означает, что цифра 3 будет использована в качестве последней цифры физического номера накопителя, например 133.

После установки фишки оператор должен поставить пакет дисков на устройство, закрыть крышку над пакетом и установить клавишу КОМПЛ/АВТОН (комплекс/автоном) в положение КОМПЛ. В этом положении накопитель подсоединяется к устройству управления накопителем и загорается лампочка КОМПЛЕКСНО. Клавиша КОМПЛ/АВТОН устанавливается в положение АВТОН, при автономной наладке накопителя. После того как загорится лампочка «Комплексно», следует нажать клавишу СТАРТ. При нажатии этой клавиши включается приводной двигатель, и пакет дисков начинает набирать скорость. При достижении 70% номинальной частоты вращения пакета и очистки пакета от пыли, которая выполняется автоматически через 45 с после пуска, подвижная каретка с магнитными головками совершает операцию первичного поиска цилиндра 000 и включается лампочка индикации с латинской буквой, например А. Каждый накопитель имеет свою индикаторную латинскую букву. Подсвет этой буквы означает, что накопитель готов к работе. Когда оператор заканчивает работу с накопителем, то он должен сначала нажать клавишу КОМПЛ/АВТОН, переведя ее в положение АВТОН, а затем клавишу СТОП. После нажатия клавиши СТОП напряжение питания отключается, каретка с магнитными головками выдвигается из пакета дисков и динамический тормоз за 25 с останавливает вращение пакета дисков. Справа на пульте находится клавиша без маркировки. Подсвет этой клавиши появляется в случае технической неисправности накопителя. После устранения неисправности следует нажать эту клавишу, и лампочка подсвета клавиши погаснет. Постоянно подсвечиваются различные комбинации цифр 128, 64, 32, 16, 8, 4, 2, 1, расположенных в верхней части пульта. Сумма подсвеченных цифр определяет адрес цилиндра, который в данный момент обрабатывается. Подсвеченные латинские буквы А, В, С и D отражают режим работы накопителя и предназначены для технических работников, так же как и индикация лампочки ТОЛЬКО ЧТЕНИЕ.

Работа накопителя. Накопитель выполняет следующие основные операции: поиск, запись, воспроизведение. Операция поиска связана с установлением магнитных головок по адресу, указанному устройством управления. Операция записи начинается со сравнения адреса, указанного устройством управления, с адресом, на который установлены магнитные головки. Если адреса совпадают, то данные записываются в накопитель. Операция воспроизведения выполняется в основном так же, как и операция записи, но запись данных не производится. Разрешается только считывать данные с накопителя.

2.4. Накопители на магнитных лентах

Накопители на магнитных лентах (НМЛ) используются в качестве внешней памяти. Они предназначены для записи информации на магнитную ленту и для ее считывания. В накопителях на магнитных лентах можно хранить большие массивы информации. Часто НМЛ используются для обмена информацией между разными моделями ЭВМ. По сравнению с накопителями на сменных магнитных дисках время выборки информации с магнитных лент больше. В настоящее время в основном используются НМЛ емкостью 20—40 М байт. Технические характеристики накопителей на магнитных лентах приведены в табл. 2.2.

**Таблица 2.2. Технические характеристики накопителей
на магнитных лентах**

Технические характеристики	Шифр		
	ЕС-5010	ЕС-5017	ЕС-5014
Емкость накопителя, М байт	20	20	40
Плотность записи, бит/мм	8 и 32		63
Скорость считывания-записи, К байт/с	64		126
Способ считывания информации	При прямом и обратном движении ленты		
Число каналов (дорожек) записи:	9		
1 контрольный; 8 информационных			
Номинальная скорость движения, м/с	2		
Время разгона ленты, мс	5,5	5	
Время останова ленты, мс	5,5	5	
Время перемотки катушки не более минуты	2,5		
Время смены катушки, с	30		
Среднее время информации, с	75		
Габаритные размеры:			
внешний диаметр катушки, мм;	267		
внешний диаметр контейнера, мм	296		

Конструкции накопителей ЕС-5017 и ЕС-5014 идентичны. Эти накопители обладают лучшими стартовыми характеристиками, чем накопитель ЕС-5010.

В накопителях используется магнитная лента на пластиковой основе, покрытой ферромагнитным слоем. Магнитная лента имеет следующие параметры: длина ленты 750 м, толщина 48 микрон, ширина ленты 12,7 мм

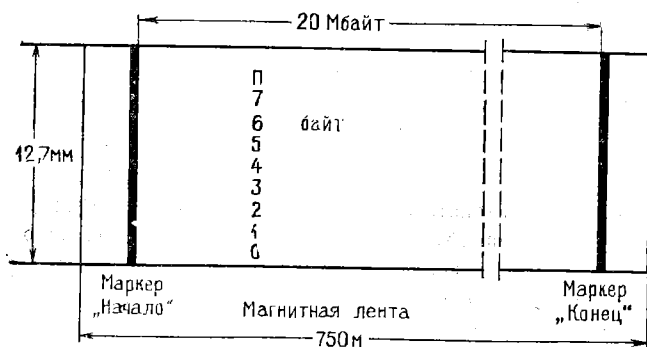


Рис. 2.7

(рис. 2.7). Примерно в трех метрах от физического начала ленты наклеивается маркер «Начало» — это кусочек блестящей металлической полоски. Маркер «Конец» наклеивается на расстоянии 4—5 м от физического конца ленты. Маркерами ограничивается область магнитной ленты, которая используется при работе с ней.

В работе участвуют два типа магнитных головок: записывающая и считывающая. Считывающая магнитная головка предназначена для проверки

записанной информации. Если информация совпадает с записанной, то запись произведена правильно; если не совпадает, то выдается сигнал об ошибке. Считывание и запись осуществляются одновременно девятью головками. Девять одновременно записанных битов представляют один байт информации с контрольным битом четности, который на рис. 2.7 обозначен буквой П. При движении лента последовательно проходит под магнитными головками.

Рассмотрим работу пульта накопителя ЕС-5010-01. Пульт предназначен для включения, выключения и управления работой НМЛ. С левой стороны на пульте находится приспособление барабанного типа с номерами (рис. 2.8).

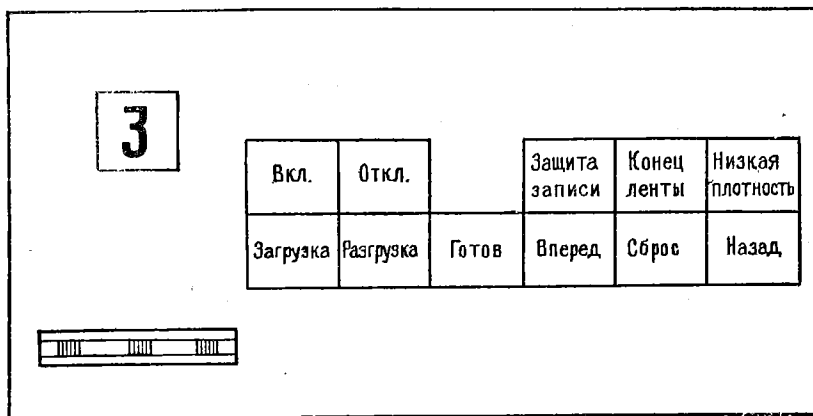


Рис. 2.8.

На барабане имеется некоторый постоянный набор цифр. Эти номера являются последними цифрами физического адреса накопителя, например цифра 3 означает, что физический адрес данного накопителя 283. Цифры устанавливаются с помощью кольцевого переключателя, который можно поворачивать вручную.

Чтобы включить накопитель, следует нажать клавишу ВКЛ с подсветом. После нажатия клавиши ВКЛ включается мотор. Далее следует открыть защитное стекло, поставить подающую бобину с магнитной лентой, заправить конец магнитной ленты на приемной бобине и закрыть защитное стекло. На подающей бобине должно быть установлено специальное внутреннее защитное кольцо. Если кольцо защиты не поставлено, то загорается лампочка ЗАЩИТА ЗАПИСИ. Далее оператор должен нажать клавишу ЗАГРУЗКА, после чего начинается вращение бобины, чтобы установить магнитную ленту в начальное положение, т.е. система должна прочесть ленточный маркер «Начало» (рис. 2.7). После того как магнитная лента установится в начальное положение, загорается подсвет клавиши ЗАГРУЗКА. Далее следует нажать клавишу ГОТОВ, т.е. устройство готово к работе в режиме записи-считывания. Подсвет клавиши ГОТОВ появляется, если устройство находится в состоянии «Комплекс», т.е. логически подсоединено к ЭВМ. При плотности записи информации 8 бит/мм включается лампочка НИЗКАЯ ПЛОТНОСТЬ, которая информирует оператора об этом факте.

Кроме режима записи-считывания, с накопителем можно работать в режимах **ПЕРЕМОТАТЬ** и **ПЕРЕМОТАТЬ и РАЗГРУЗИТЬ**. Работа по перемотке выполняется в любое время, если устройство не работает в режиме записи-считывания, т. е. по окончании работы с данной бобиной. В этом случае нажимается клавиша **РАЗГРУЗКА**. В режиме разгрузки магнитная лента перематывается обратно на подающую бобину. Клавиша **НАЗАД** обеспечивает движение магнитной ленты в направлении от маркера «Конец» к маркеру «Начало». Когда магнитная лента будет перемотана, включится лампочка **КОНЕЦ ЛЕНТЫ**. Далее следует открыть защитное стекло и снять бобину. Накопитель выключается нажатием клавиши **ОТКЛ**. Клавиша **СБРОС** предназначена для прекращения всех режимов работы накопителя, кроме режима загрузки.

Глава 3. КРАТКИЕ СВЕДЕНИЯ ОБ ОПЕРАЦИОННОЙ СИСТЕМЕ И ЯЗЫКЕ УПРАВЛЕНИЯ ЗАДАНИЯМИ

3.1. Основные понятия операционной системы

3.1.1. Структура и функции системы программного обеспечения (СПО). Совокупность программ, процедур и правил, позволяющих использовать ЭВМ для решения различных задач, называется *системой программного обеспечения*. Она берет на себя основную работу по ведению процесса вычисления: управляет работой процессора, памяти, устройств ввода-вывода и т. д.

В систему программного обеспечения входит *операционная система* (общее программное обеспечение) и *специальное программное обеспечение* (рис. 3.1). Операционная система предназначена для планирования и организации безостановочного процесса обработки программ, ввода-вывода

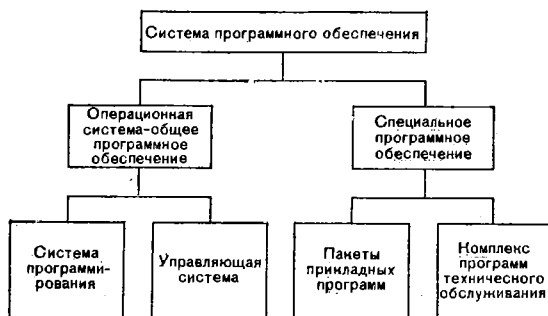


Рис. 3.1. Состав системы программного обеспечения.

управления данными, распределения ресурсов (устройств ввода-вывода), отладки программ и других вспомогательных операций обслуживания. Специальное программное обеспечение представляет собой совокупность программ, используемых для решения определенного класса задач. В него входят пакеты прикладных программ (ППП), а также комплекс программ технического обслуживания (КПТО). *Пакеты прикладных программ* — это комплекс программных средств вместе с документацией, необходимой для его установки и эксплуатации, ориентированной на решение определенных классов задач. *Комплекс программ технического обслуживания* — это типовые программы, предназначенные для наладки, планирования и диагностики функционирования аппаратуры ЭВМ.

Операционная система состоит из системы программирования и управляющей системы (рис. 3.2). *Системой программирования* называется совокупность средств, которые используются для написания, трансляции и отладки текстов программ. Систему программирования образуют языки программирования

и обрабатывающие программы. *Язык программирования* — это специальный язык, применяемый для написания программ. В ЕС ЭВМ для написания программ используются следующие языки программирования: ассемблер, алгол-60, фортран-IV, РПГ, кобол и ПЛ/1. *Обрабатывающие программы* — это общее название трансляторов, проблемных, сервисных и обслуживающих программ. *Трансляторы* переводят программы, написанные на одном из вышеперечисленных языков, на язык машинных кодов, с которыми непосредственно работает машина. В ЕС ЭВМ имеются следующие трансляторы:



Рис. 3.2. Состав операционной системы.

ассемблер (F), алгол (F), кобол (E) или (F), фортран (H) или (G), РПГ (E) и ПЛ/1 (F), где буквы в скобках обозначают величину объема памяти, которая необходима для транслятора: E = 32K, F = 64K, G = 128K, H = 256K, где K = 1024 байт.

Проблемная программа (программа пользователя) — это любая программа, которая выполняется на ЭВМ под управлением ОС в состоянии «задача». Программа может быть написана по частям, которые называются *модулями*. Программа (или ее часть), подготовленная для однократной обработки транслятором, называется *исходным модулем*. В распоряжении программиста имеются средства, которые позволяют в модулях указать связи между ними, чтобы в дальнейшем иметь возможность объединить их в одну программу. Исходные модули транслируются независимо друг от друга. В результате получаются *объектные модули*, в которых программа оказывается представленной на машинном языке, т. е. после трансляции полностью утрачивается специфика языка программирования. На рис. 3.3 показана упрощенная схема обработки модуля в ЭВМ.

Сервисные программы (редактор связей, загрузчик, сортировка-объединение) и *программы обслуживания* помогают программисту в его работе с вычислительной системой.

Редактор связей устанавливает связь между отдельно оттранслированными объектными модулями, если программа состоит из нескольких модулей. В результате получается *загрузочный модуль*. После редактирования программа загрузки отправляет загрузочный модуль в оперативную память для

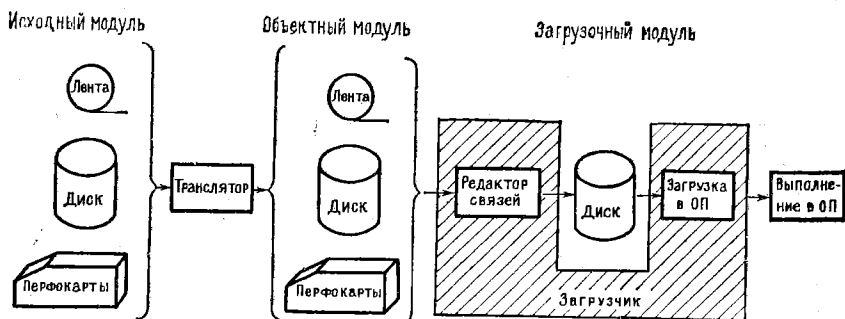


Рис. 3.3.

выполнения. *Загрузчик* — это редактор связей и программа загрузки. *Программа сортировки-объединения* сортирует, объединяет и упорядочивает данные на магнитной ленте или диске в требуемом порядке.

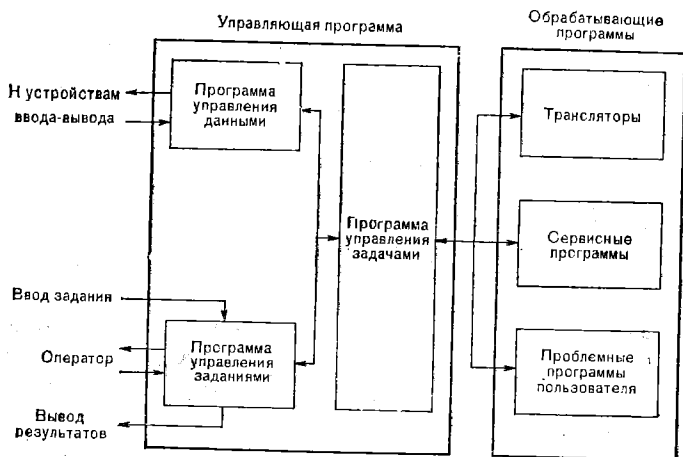


Рис. 3.4. Взаимосвязь между управляющей и обрабатывающими программами.

Программы обслуживания обеспечивают разнообразные виды работ с наборами данных, которые находятся на магнитных дисках и магнитных лентах, начиная с их первоначальной подготовки к работе с ОС.

Все вышеперечисленные обрабатывающие программы выполняются под управлением управляющей программы, которая входит в управляющую систему. *Управляющая программа* предназначена для планирования и контроля процесса обработки данных и имеет три основных компонента: программу

управления заданиями, программу управления задачами и программу управления данными, как показано на рис. 3.4.

Программа управления заданиями. Оператор получает от программиста исходный модуль, оформленный в виде задания. *Задание* — это текст программы, написанный на одном из языков программирования, к которому добавляются операторы программы управления заданиями, записанные на языке *управления заданиями* (рис. 3.2). Задание представляет собой основную, независимую единицу работы, выполняемую вычислительной системой. Характерной особенностью заданий является их независимость друг от друга, т.е. выполнение каждого задания не зависит от факта завершения любого другого задания или результата его решения. Независимость заданий друг от друга позволяет вычислительной системе с помощью ОС выполнять их одновременно, т.е. осуществлять *мультипрограммную* (многопрограммную) обработку заданий. Оператор с помощью команд может изменять некоторые параметры программы управления заданиями, чтобы обеспечить более быстрое выполнение задания на ЭВМ.

Программа управления задачами. Задание состоит из одного или более пунктов. *Пунктом (шагом) задания* называется единица работы, связанная с одной обрабатывающей программой и соответствующими данными. Для каждого пункта задания операционная система организует одну или несколько задач. *Задача* — наименьшая независимая единица работы, выполняемая процессором, которая может претендовать на ресурсы ЭВМ. С точки зрения ОС все задачи являются независимыми друг от друга. Задачи выполняются под управлением программы управления задачами, которая выделяет в распоряжение задачи ОП, процессор и прочие системные ресурсы, за исключением устройств ввода-вывода. Система образует задачи на основании данных, указанных в программе управления заданиями, а также макрокоманд супервизора, заданных в обрабатывающих программах.

Программа управления данными предназначена для организации ввода-вывода данных, т.е. для обмена информацией между ОП и внешней памятью (или устройствами ввода-вывода), а также для сохранения информации, ее организации и т. д. Программист-пользователь обращается к программе управления данными с помощью *макрокоманд ввода-вывода*.

3.2. Система управления заданиями

3.2.1. Общие сведения. В систему управления заданиями входят язык управления заданиями (ЯУЗ) и программа управления заданиями. ЯУЗ можно рассматривать как язык программирования высокого уровня по отношению к алгоритмическим языкам программирования. К программе управления заданиями обращаются с помощью девяти управляющих операторов ЯУЗ. Эти операторы сообщают управляющей программе о необходимости выделения заданию обрабатывающих программ, используемых данных и об общих требованиях к периферийным устройствам.

Программа управления заданиями (рис. 3.4) обеспечивает подготовку каждого задания (или пункта задания) к работе. Она имеет два основных набора программ: *главного планировщика* (MASTER SCHEDULER) и *планировщика заданий*. Эти программы обеспечивают прохождение каждого зада-

ния на ЭВМ от начала до конца, т.е. от ввода до вывода результатов вычислений.

Набор программ главного планировщика предназначен для выполнения команд оператора ЭВМ и обеспечения его связи с ЭВМ путем диалога: главный планировщик выдает оператору запросы и сообщения, отвечает на вопросы оператора относительно состояния задания и вычислительной системы в целом.

В набор программ планировщика заданий входят программы инициатора INIT, программы системного ввода RDR, программы системного вывода WTR и программы прямого системного вывода DSO, с которыми постоянно работает оператор. Программа RDR организует обработку заданий: считывает задания с устройства ввода, проверяет правильность записи управляющих операторов ЯУЗ, планирует выполнение заданий. Программа INIT инициирует (начинает) выполнение заданий. Программа WTR (или DSO) обеспечивает выдачу выходных данных и диагностических сообщений.

Режим работы с заданиями зависит от варианта управляющей программы, используемой при работе. Имеется четыре варианта управляющих программ: PCP, MFT, MVT и SVS.

Управляющая программа PCP обеспечивает *однозадачный режим*. В ней используется последовательный планировщик, который в каждый момент времени обрабатывает только один пункт задания, причем пункты задания обрабатываются в порядке их расположения в задании. Минимальный объем оперативной памяти, необходимый для управляющей программы PCP, равен 64 К. Режим PCP целесообразно использовать для моделей ЭВМ с объемом памяти до 128 К*).

Управляющая программа MFT позволяет организовать *мультипрограммный режим с фиксированным числом задач*, при котором оперативная память до начала работы с заданиями делится на фиксированное число разделов. В каждом разделе выполняется одно задание. Внутри каждого раздела система может организовать мультипрограммную обработку задач (вариант MFT с подзадачами). Общее число подзадач не должно превышать 255. Управляющая программа MFT имеет приоритетный планировщик и занимает в оперативной памяти минимальный объем памяти 128 К.

Управляющая программа MVT служит для обработки *переменного числа задач в мультипрограммном режиме*. Память выделяется заданиям по мере необходимости. После выполнения каждого задания выделенная ему память освобождается. В управляющей программе MVT используется приоритетный планировщик, который строит для заданий очереди, чтобы повысить производительность системы. Управляющая программа MVT требует не менее 256 К оперативной памяти.

Управляющая программа SVS используется только в моделях «Ряд 2». Она работает в мультипрограммном режиме с переменным числом задач. Этот режим позволяет обеспечить одновременную работу до 63 программ пользователя. Для SVS требуется минимальный объем памяти 512 К.

Первоначально для управляющих программ MFT и MVT использовалась версия ОС 2.1, затем 4.1. Последняя версия 6.1 имеет в своем

*) Однозадачный режим PCP в настоящее время не используется, поэтому далее не рассматривается.

распоряжении управляющую программу SVS и является дальнейшим развитием ОС.

3.2.2. Операторы языка управления заданиями. В табл. 3.1 приведен сокращенный перечень управляющих операторов ЯУЗ *) и структура их форматов.

Таблица 3.1. Форматы операторов ЯУЗ

№ п/п	Название	Операция	Операнды	Комментарии
1	//имя	JOB	[параметры]	[комментарии]
2	///имя]	EXEC	параметры	[комментарии]
3	///имя]	DD	параметры	[комментарии]
4	//	команда	параметры	[комментарии]
5	/*			[комментарии]
6	//*			[комментарии]
	***			[комментарии]
7	//			[комментарии]

Назначение операторов: JOB — оператор начала задания; EXEC — оператор начала пункта задания; DD — оператор описания данных; «команда» — команда оператора ЭВМ; /* — оператор конца набора данных: во входном потоке отделяет данные от управляющих операторов ЯУЗ; // — оператор конца задания; /* или *** — операторы-комментарии, которые используются для включения между управляющими операторами информации, полезной для оператора ЭВМ или программиста при чтении листинга.

3.2.3. Общие правила записи управляющих операторов на бланке и их представление на перфокартах. Управляющие операторы записываются на бланке кодирования с помощью алфавита, который включает следующий набор символов:

— латинские буквы от A до Z, а также денежный знак \$, коммерческое И — @ и номер #, которые воспринимаются системой, как буквы;

— цифры от 0 до 9;

— специальные знаки:

+ плюс , запятая

— минус . точка

* звездочка = знак равенства

/ дробная черта ' одиночный апостроф

(открывающая скобка & знак амперсанда

) закрывающая скобка пробел

Алфавит закодирован определенными наборами двоичных цифр. В ЕС ЭВМ имеется несколько систем кодирования алфавитов. Наиболее часто используются коды ДКОИ (двоичный код для обмена информацией) и КОИ-8 (8-битный код для обмена информацией), которые приведены в Приложении 3. Код каждого символа образуется комбинацией двух шестнадцатеричных цифр, которые называются зонной и цифровой, например символ A имеет код C1₁₆.

*) Всего в ЯУЗ 9 операторов. Операторы PROC и PEND опущены, см. [2, 3, 8].

[illegible]

задания и т. д. Бланк кодирования состоит из двух полей: колонки 1—71 относятся к полю оператора (или предложения), а колонки 73—80 используются для присвоения названия программе и нумерации исходных перфокарт по порядку, т. е. для идентификации последовательности. Поле оператора отделено от поля идентификации последовательности колонкой 72, которая называется *колонкой продолжения* предложения на следующую строку бланка. В поле оператора записывается текст исходных программ на языке ассемблера, языках высокого уровня или языке управления заданиями.

В подполе названия записывается идентификатор управляющих операторов: // или /* — в первых двух колонках бланка, и //* или *** — в первых

трех колонках бланка. Непосредственно за идентификатором // без пробелов записывается «имя», начиная с третьей колонки бланка, например SUB.

В подполе операции записывается тип управляющего оператора, например JOB (рис. 3.5, строка 1), или «команда», например STOP или P (рис. 3.5, строки 7, 8).

В подполе операндов указываются параметры, разделенные запятыми. Параметры бывают позиционные и ключевые. *Позиционные параметры* находятся на определенном месте (позиции) в формате и представляют собой выражение, например ИВАНОВА. *Ключевые параметры* состоят из ключевого слова со значениями, которые выбираются для этого ключевого слова из определенного известного набора параметров, например CLASS=B, где CLASS ключевое слово, значение которого равно B. Между ключевым словом и значением (или значениями) ставится знак равенства.

В подполе операндов всегда сначала записываются все позиционные параметры, например 000125 и ИВАНОВА, в определенном порядке, заданном в формате, а затем ключевые, например: MSGLEVEL, CLASS и PRTY, в произвольном порядке (рис. 3.5, строки 1—3). Если в формате опускается первый позиционный параметр, например 000125, то запятая, разделяющая первый и второй позиционные параметры, сохраняется (рис. 3.5, строка 4). Если опускаются позиционные параметры справа, то запятые справа также опускаются (рис. 3.5, строка 6). Если опускаются ключевые параметры, то запятые, разделяющие их, не сохраняются. Если параметры не помещаются на одной строке бланка, то запись их может быть продолжена на следующих строках. Каждая следующая строка должна начинаться с идентификатора // в первых двух колонках (рис. 3.4, строки 2, 3). Запись параметров продолжается, начиная с колонки 4, но не позднее 16-й колонки, иначе система рассмотрит продолжение записи параметров как комментарий. Подполе комментариев может содержать любую информацию (рис. 3.4, строка 8), улучшающую наглядность распечатки, выдаваемой на АЦПУ, которая называется *аистингом*.

Бланк кодирования используется как документ для перфорации. Каждая заполненная строка бланка перфорируется на одной перфокарте. Перфокарта имеет 12 строк по 80 колонок. Колонки пронумерованы слева направо от 1 до 80. Номера колонок на бланке соответствуют номерам колонок на перфокарте. Строки пронумерованы сверху вниз от 0 до 9. Под строки 12 и 11 отведено место в верхней части бланка без указания номера строки. Строки 12, 11 и 0 называются *зонными*, а все остальные строки — *цифровыми*. В Приложении 4 показана пробивка цифр, букв и специальных знаков ДКОИ в виде прямоугольных отверстий (код КПК-12). Каждый символ ДКОИ занимает одну колонку перфокарты.

3.2.4. Основные обозначения, используемые при записи управляющих операторов ЯУЗ. Информация, записанная в форматах прописными латинскими буквами, остается без изменения, например JOB или EXEC. Слова, указанные строчными русскими буквами, заменяются на то или иное значение, которое они принимают в каждом конкретном случае. Например, в операторе JOB заменим «имя» на символическое имя PRIM (см. табл. 3.1):

//PRIM JOB

Символическое имя составляется по следующему правилу: оно не должно иметь более 8 знаков, не иметь внутри себя пробелов и специальных знаков и начинаться обязательно с буквы, например: PR, FUNG, T1, OLGA, F125. Запишем еще в качестве примера команду оператора START (сокращенно S):

```
// S WTR,00F
```

Информация, указанная внутри квадратных скобок в любых операторах, является необязательной, т. е. при необходимости она может быть опущена. Например, в операторе JOB могут быть опущены [параметры] и [комментарии] (см. табл. 3.1). В форматах операндов часто используются фигурные скобки, которые означают, что из нескольких предлагаемых значений следует выбрать одно необходимое. Многоточия указывают на продолжающуюся последовательность параметров. Многоточия, фигурные и квадратные скобки при записи опускаются. Числа и знаки пунктуации такие, как круглые скобки, точки, запятые, сохраняются в форматах параметров.

3.3. Управляющий оператор JOB

В управляющем операторе JOB программист указывает параметры, которые задают условия выполнения задания. После того как задание будет помещено в ЭВМ для выполнения, оператор ЭВМ может с целью эффективного использования вычислительной машины изменить некоторые параметры с помощью команд.

Общая структура формата оператора JOB приведена в табл. 3.1. Он обязательно должен содержать имя в подполе названия. Параметры в подполе операндов не обязательны. Оператор JOB сообщает системе о начале

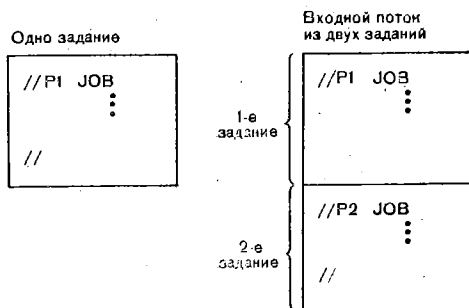


Рис. 3.6. Виды заданий.

одного задания или о начале входного потока заданий. Входной поток представляет собой последовательность заданий, вводимых с устройства ввода с помощью команд оператора. В частном случае во входном потоке может быть только одно задание. Если задания следуют во входном потоке, то каждый встретившийся оператор JOB указывает на конец предыдущего задания и на начало следующего задания, как показано на рис. 3.6. Последнее задание, в том числе и единственное задание, заканчивается идентификатором //.

Подполе названия. Имя задания используется планировщиком заданий. Если во входном потоке имеется несколько заданий, то желательно, чтобы все задания имели разные имена. Кроме того, не рекомендуется пользоваться именами JOBNAME, Q, N, DSNAME, T, R, SPACE, A, U, STATUS, CONSOLE, PFK, SQA, так как эти имена используются в качестве ключевых слов в командах оператора ЭВМ. Если какое-либо из вышеперечисленных имен все же используется в операторе JOB, то оператор в команде

Таблица 3.2. Параметры оператора JOB *)

№ п/п	Параметры	Назначение	Варианты управляющих программ		
			MFT	MVT	SVS
1	[учет. информация]	Задаёт учетный номер и дополнительную учетную информацию	+	+	+
2	[имя программиста]	Содержит имя программиста	+	+	+
3	[ADDRSPC] **)	Определяет вид памяти	-	-	+
4	[CLASS]	Задаёт класс задания	+	+	+
5	[COND]	Задаёт условия окончания задания	+	+	+
6	[MSCLASS]	Определяет класс выдачи системных сообщений	+	+	+
7	[MSGLEVEL]	Определяет уровень системных сообщений	+	+	+
8	[NOTIFY] **)	Обеспечивает передачу пользователю сообщения о завершении задания в режиме разделения времени (TSO)	-	+	+
9	[PRTY]	Задаёт приоритет заданию	+	+	+
10	[RD]	Запрашивает рестарт задания в случае аварийного окончания	+	+	+
11	[REGION]	Запрашивает объем ОП	-	+	+
12	[RESTART]	Запрашивает отсроченный рестарт задания	+	+	+
13	[ROLL]	Управляет сверткой-разверткой ОП	-	+	-
14	[TIME]	Задаёт процессорное время работы для задания	+	+	+
15	[TYPRUN]	Откладывает выполнение задания	+	+	+

*) Здесь и далее в таблицах операторов ЯУЗ приняты следующие обозначения: знак «-» — параметр не используется; знак «+» — параметр используется.
**) Версия 6.1.

DISPLAY при запросе информации об этих заданиях должен заключить его в скобки.

В подполе операндов оператора JOB может быть указано до 15 параметров, приведенных в табл. 3.2. Если в подполе операндов ничего не указывается, то это значит, что используются значения, которые были установлены во время генерации ОС в программе системного ввода RDR. Если параметры указываются, то они сообщают системе о некоторых дополнительных условиях выполнения задания или изменяют те значения, которые были

установлены при генерации системы. Как видно из табл. 3.2 для каждого вида управляющей программы (MFT, MVT или SVS) используется определенный набор параметров. В этой главе рассматриваются параметры «учетная информация», «имя программиста», CLASS, PRTY, MSGLEVEL, MSGCLASS, REGION, TIME и TYPRUN.

Остальные параметры рассматриваются в гл. 9.

3.3.1. Идентификация задания. Учетная информация является первым позиционным параметром. Форма записи учетной информации устанавливается в соответствии с требованиями, принятыми в данном учреждении. Формат параметра:

[*учетный номер*] [*доп.учет.информация*]

Если указывается только учетный номер, то общие скобки опускаются, например:

```
//PRIM1 JOB 5427
```

Если учетный номер содержит специальный знак, то параметр заключается в одиночные апострофы, например:

```
//PRIM2 JOB '5427'
```

Учетная информация, состоящая из учетного номера и дополнительной учетной информации, разделяется запятыми и заключается в круглые скобки или одиночные апострофы, например:

```
//PRIM3 JOB (5427,A)
```

или

```
//PRIM4 JOB '5427,A'
```

Круглые скобки и одиночные апострофы не входят в учетную информацию. Если дополнительная учетная информация состоит из нескольких подпараметров и какой-либо из них содержит специальные знаки (за исключением дефиса), то можно либо весь параметр учетной информации заключить в апострофы, либо только один подпараметр дополнительной учетной информации, но в последнем случае следует заключить в скобки весь параметр, например:

```
//PRIM5 JOB '5427,23/03/81'
```

или

```
//PRIM6 JOB (5427,'23/03/81')
```

Знак амперсанда или апостроф при записи удваиваются, например '54&&27,A'. Если учетная информация содержит только дополнительную учетную информацию, то перед дополнительной учетной информацией следует оставить запятую, например:

```
//PRIM7 JOB ',OTD136,ТЕЛ.21-45'
```

Учетная информация может содержать до 142 подпараметров, включая запятые. Если учетная информация не помещается на одной строке, то она разрывается на запятой, разделяющей подпараметры, и весь параметр учетной информации должен быть заключен в скобки, как показано ниже:

```
//PRIM8 JOB (240,000005,B,'26/01/81',  
// ОТДЕЛ 15)
```

Имя программиста является вторым позиционным параметром. Оно может содержать до 20 знаков и записывается без скобок и апострофов, если используются буквы латинского алфавита, например PETROVA. Имя программиста заключается в апострофы, если используется русский алфавит, например, 'ГУСЕВА', а также если имя программиста содержит специальные знаки, например, 'О.Р.IVANOVA' или 'Е.А.КУРИКОВА'. Ниже приведены примеры записи оператора JOB, содержащего два параметра: учетную информацию и имя программиста

```
//PRIM9 JOB (498,2335,0001,0400,'6C=09844'),'ЕФАНОВА'
```

```
//PRIM10 JOB ',О.И.ЕФАНОВА'
```

В примере PRIM9 указаны два параметра, которые разделены запятыми. В примере PRIM10 первый параметр опущен, поэтому перед вторым параметром поставлена запятая.

3.3.2. Класс и приоритет задания. *Ключевой параметр CLASS* присваивает класс заданию. *Класс задания* — это признак, который присваивается заданию и опознается программой системного ввода RDR при вводе его с устройства ввода. Формат параметра

CLASS=класс задания

в котором *класс задания* заменяется одним из символов от А до О, например:

```
//PRIM11 JOB CLASS=C
```

В примере PRIM11 задан класс задания С. Если параметр CLASS опущен, то заданию присваивается класс, установленный в каталогизированной процедуре системного ввода RDR. В соответствии со значениями, указанными в параметрах CLASS, программа RDR составляет список, который называется *входной очередью* JOB. Общее количество классов в очереди равно 15 по числу букв от А до О. Следует назначать одинаковые классы заданиям исходя из одинакового времени их выполнения, требований к объему памяти, количеству выводимой информации и т. д.

Ключевой параметр PRTY присваивает приоритет заданию. В соответствии со значением приоритета задания размещаются внутри определенного класса во входной очереди заданий. *Приоритетом* называется ранг задачи, определяющий ее право на получение ресурсов системы. Если параметр PRTY опущен, то заданию присваивается приоритет, установленный в программе системного ввода RDR. Если заданию необходимо присвоить приоритет, отличный от установленного, то в формате

PRTY=nn

буквы *nn* заменяются цифрами от 0 до 13 (13 — наивысший приоритет). Следует избегать пользоваться приоритетом, равным 13, так как он используется в основном системными программистами. В исключительных случаях можно назначать приоритет 13 в программах, если в них устранены все ошибки и время выполнения задания мало. Приведем пример записи параметра PRTY:

```
//PRIM12 JOB CLASS=C,PRTY=5
```


Значение приоритета выбирается исходя из того, насколько быстро (по сравнению с другими заданиями) требуется пропустить задание на ЭВМ.

3.3.3. Уровень и выходной класс системных сообщений. Системные сообщения являются частью выходной информации, выдаваемой системой, которая помогает определить ее состояние во время выполнения задания (рис. 3.7).

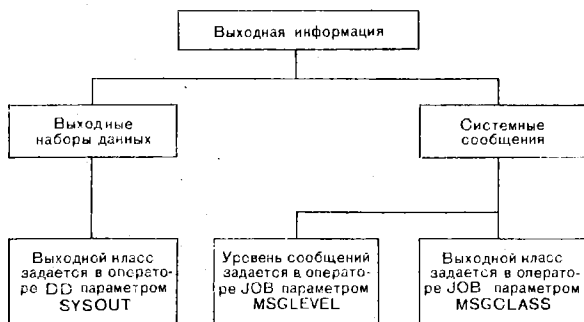


Рис. 3.7. Состав выходной информации.

Параметр *MSGLEVEL* предназначен для того, чтобы сообщить планировщику о том, какую информацию о ходе выполнения задания желательно получить в листинге, т.е. задает уровень системных сообщений. Формат параметра

$$\text{MSGLEVEL} = \left(\left\{ \begin{matrix} 0 \\ 1 \\ 2 \end{matrix} \right\}, \left\{ \begin{matrix} 0 \\ 1 \end{matrix} \right\} \right)$$

где значения первого подпараметра обозначают следующее: 0 — на печать выводится только оператор JOB; 1 — на печать выводятся все входные управляющие операторы JY3 и операторы каталогизированных процедур (каталогизированная процедура — это набор предложений JY3, помещенный в библиотеку процедур SYS1.PROCLIB); 2 — на печать выводятся только управляющие операторы JY3. Во втором подпараметре значения цифр обозначают: 0 — на печать не выводятся сообщения о выделении устройств ввода-вывода пунктам заданий и о состоянии этих устройств после окончания выполнения пункта задания, за исключением случая, когда задание заканчивается аварийно (при аварийном окончании задания сообщения об устройствах выдаются всегда); 1 — на печать выводятся все сообщения о распределении устройств ввода-вывода. Если опущен параметр *MSGLEVEL* или один из его подпараметров, то используются значения, установленные при генерации системы. С помощью параметра *MSGLEVEL* уровень сообщений можно изменить, например:

```
//PRIM13 JOB MSGLEVEL=(2,0)
```

Ниже приведены примеры записи параметра *MSGLEVEL*, если один из подпараметров опущен:

```
//PRIM14 JOB MSGLEVEL=2
```

```
//PRIM15 JOB MSGLEVEL=(,1)
```

В примерах PRIM14 и PRIM15 оба позиционных параметра (учетная информация и имя программиста) опущены. Вместо опущенных подоперандов в параметре MSGLEVEL будут использоваться значения, установленные в процедуре системного ввода RDR, например (0,1).

Ключевой параметр MSGCLASS задает выходной класс системных сообщений (рис. 3.7). В формате параметра

MSGCLASS=выходной класс

слово *выходной класс* заменяется буквенным символом от A до Z или цифрой от 0 до 9. Ниже приводится пример записи параметра MSGCLASS:

//PRIM16 JOB MSGCLASS=C

где задается выходной класс сообщений C. Если параметр MSGCLASS опущен, то выходные сообщения выдаются в стандартном классе A. В соответствии со значением, указанным в параметре MSGCLASS, планировщик ставит задание в выходную очередь SOUT. Всего в SOUT может быть создано до 36 выходных классов (по числу букв алфавита и цифр). Чтобы системные сообщения и выходные наборы данных были выведены на одно устройство вывода, необходимо, чтобы классы, указанные в параметре MSGCLASS и в параметре SYSOUT оператора DD, были одинаковыми (рис. 3.7 и п. 3.5.2).

3.3.4. Изменение объема памяти для задания. Параметр REGION используется только в MVT и позволяет запросить объем памяти для раздела, в котором будет выполняться задание. Если параметр REGION опущен, то заданию будет выделен минимальный объем памяти, установленный в процедуре системного ввода RDR. Чтобы запросить объем памяти, отличный от этого размера в формате параметра

REGION=nnnnnK

следует заменить nnnnn числом 1024-байтных областей, необходимых для задания. Значение nnnnn не должно превосходить 16 383 и должно быть обязательно четным. Если программист укажет нечетное число, то система увеличит его до ближайшего четного числа. Значение nnnnn следует выбирать исходя из величины объема памяти, необходимой для задания. Ниже приведен пример записи параметра REGION:

//PRIM16 JOB 'ПЕТРОВ', REGION=100K

3.3.5. Ключевой параметр TIME позволяет задать время, которое процессор может затратить на выполнение задания. Если параметр TIME опущен, то время выполнения задания процессором не может превосходить значения, установленного в процедуре RDR обычно равного 30 минутам. Параметр TIME позволяет ограничить время выполнения задания, особенно при отладке, и полезен как средство борьбы с заикливанием программ. В формате параметра TIME:

TIME={ (минуты, секунды) }
1440

слово *минуты* заменяется числом минут (максимальное значение равно 1439), слово *секунды* заменяется числом секунд, значение которого меньше 60.

Если время работы процессора превосходит время, установленное в RDR или программистом с помощью параметра TIME, то выполнение задания пре-

крашается. Если в параметре TIME указать число 1440, то время выполнения задания становится неограниченным. Ниже приведены примеры записи параметра TIME:

//PRIM17 JOB CLASS=A, TIME=5 — время 5 минут (если секунды не указываются, то скобки опускаются);

//PRIM18 JOB TIME=(5,30) — время 5 минут 30 секунд;

//PRIM19 JOB TIME=(,30) — время 30 секунд (при отсутствии минут запятая в скобках остается).

3.3.6. Задержка выполнения задания. Ключевой параметр TYPRUN предназначен для того, чтобы задержать выполнение задания до особого распоряжения программиста. В этом случае задание помещается программой RDR в очередь задержанных заданий HOLD. Ниже приведен пример записи параметра:

//PRIM20 JOB CLASS=B, PRTY=3,

// TYPRUN=HOLD

Если параметр TYPRUN=HOLD опущен, то задание помещается во входную очередь в соответствии с его классом и приоритетом.

3.4. Управляющий оператор EXEC

Управляющий оператор EXEC предназначен для описания программы (PGM) или процедуры (PROC), вызываемой в данном пункте задания. *Процедурой* называется набор управляющих операторов задания, который может

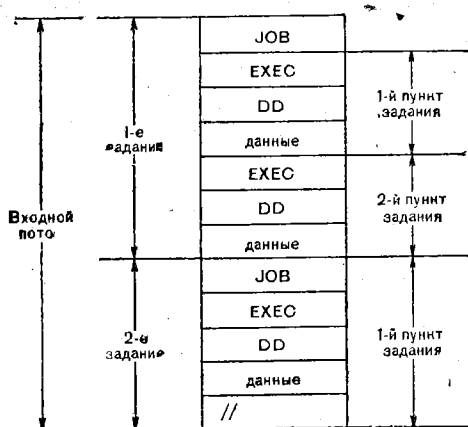


Рис. 3.8. Структура входного потока заданий, состоящего из двух заданий.

находиться либо во входном потоке, либо в специальной библиотеке системы. Оператор EXEC сообщает системе о начале пункта задания (или процедуры). В задании, если оно состоит из нескольких пунктов, EXEC отмечает конец предыдущего и начало следующего пункта задания (рис. 3.8). Последний пункт задания заканчивается пустым оператором //. Как видно из рисунка, оператор EXEC является первым оператором в каждом пункте задания,

За ним следуют операторы DD и данные, которые относятся к соответствующему пункту задания.

3.4.1. Формат оператора EXEC приведен в табл. 3.1. Имя в подполе названия указывать не обязательно, но тем не менее рекомендуется каждому пункту задания присваивать имя. Имя используется во многих системных сообщениях, которые выдаются оператору ЭВМ и программисту и помогают установить, какой пункт задания вызвал то или иное сообщение. Если имя в операторе EXEC отсутствует, то в сообщении на том месте, где должно указываться имя пункта задания, будет пробел, что затрудняет анализ сообщений. Желательно, чтобы в пределах одного задания не было одинаковых имен для разных пунктов заданий.

В подполе операндов может быть указано от 1 до 10 параметров, перечень которых приведен в табл. 3.3. Из таблицы видно, что набор параметров

Таблица 3.3. Параметры оператора EXEC

№ п/п	Параметр	Назначение	Варианты управляющих программ		
			MFT	MVT	SVS
1	{PGM}	Вызывает программу по имени	+	+	+
2	{PROC}	Вызывает процедуру по имени	+	+	+
3	{ACCT}	Сообщает данные учетной информации	+	+	+
3	{ADDRSPC} *)	Задаст вид основной памяти (виртуальной или реальной)	-	-	+
4	{COND}	Задаст условие перехода к следующему пункту задания	+	+	+
5	{DPRTY}	Сообщает данные для определения диспетчерского приоритета	+	+	+
6	{PARM}	Передаст параметры программе, выполняемой в пункте задания	+	+	+
7	{RD}	Запрашивает рестарт пункта задания в случае аварийного окончания	+	+	+
8	{REGION}	Запрашивает объем ОП для пункта задания	-	+	+
9	{ROLL}	Управляет сверткой-разверткой ОП	-	+	-
10	{TIME}	Задаст процессорное время работы для пункта задания	+	+	+

*) Версия 6.1.
 **) В режиме MFT без подзадач DPRTY не используется.

зависит от варианта управляющей программы. Обязательным параметром является PGM или PROC. Остальные ключевые параметры являются необязательными. По своим функциям многие параметры EXEC аналогичны параметрам оператора JOB. При первом знакомстве с оператором EXEC наибольший интерес представляют параметры DPRTY, REGION и TIME.

3.4.2. Параметр DPRTY используется для вычисления диспетчерского приоритета, который определяет порядок, в котором процессор будет обрабатывать задачи в условиях мультипрограммирования. В простейшем случае DPRTY имеет следующий формат:

$$DPRTY=(nn_1, nn_2)$$

где nn_1 и nn_2 заменяются числами от 0 до 15. Диспетчерский приоритет может быть определен тремя способами:

1. Непосредственно из DPRTY по формуле

$$\text{диспетчерский приоритет} = (nn_1 \times 16) + nn_2$$

Значения всех диспетчерских приоритетов находятся в пределах от 0 до 255. Чем больше значения nn_1 и nn_2 , тем быстрее выполняется пункт задания. Например, если $nn_1 = 8$ и $nn_2 = 5$, то значение диспетчерского приоритета равно 133. Следует избегать значений, равных 15, так как они обычно используются системными программистами. Если в параметре DPRTY указывается только значение nn_1 , то по умолчанию принимается $nn_2 = 11$. Например, пусть DPRTY = 8. Тогда значение диспетчерского приоритета равно 139. Если в DPRTY указан только nn_2 , то по умолчанию принимается $nn_1 = 0$. Например, пусть $nn_2 = 12$, тогда параметр DPRTY запишется следующим образом: DPRTY = (,12). Значение диспетчерского приоритета равно 12.

2. Косвенно по параметру PRTY оператора JOB, если параметр DPRTY опущен

$$\text{диспетчерский приоритет} = (nn \times 16) + 11$$

3. По умолчанию, если параметры PRTY и DPRTY опущены, диспетчерский приоритет равен 11.

В системе с подзадачами диспетчерские приоритеты подзадач такие же, как и у задачи, образовавшей подзадачу.

3.4.3. Изменение объема памяти для пункта задания. Ключевой параметр REGION позволяет запросить объем памяти для пункта задания, отличный от того, который был установлен в процедуре системного ввода RDR. В простейшем случае параметр REGION имеет формат

$$\text{REGION} = nnnnnK$$

где число $nnnnn$ заменяется числом 1024-байтных областей, которые следует отвести для пункта задания. Это число может изменяться в пределах от 1 до 16383 и должно быть обязательно четным. Если указано нечетное число, то система устанавливает следующее четное. Ниже дан пример записи параметра REGION:

```
//SHAG1 EXEC PROC=RECT,REGION=100K
```

Если в операторе JOB указан параметр REGION, то указание о выделении памяти для пункта задания игнорируется.

3.4.4. Параметр TIME позволяет задать время работы процессора для данного пункта задания. Формат параметра TIME полностью совпадает с форматом TIME для оператора JOB. Если время, указанное в операторе JOB, меньше, чем в операторе EXEC, то для пункта задания будет выделено время, указанное в операторе JOB, например:

```
//PRIM21 JOB TIME=5 — время 5 минут;
```

```
//SHAG2 EXEC TIME=10 — время 10 минут.
```

Пункту задания SHAG2 будет выделено не более 5 минут времени работы процессора.

3.5. Управляющий оператор DD

В управляющем операторе DD дается описание наборов данных, связанных с выполнением определенного пункта задания.

3.5.1. Общие сведения о наборах данных. Набор данных представляет собой самую крупную единицу информации при организации запоминания и поиска данных в ОС. Представление информации в виде наборов данных позволяет ОС автоматизировать процессы ее передачи, обработки и т. д. Набор данных ограничивается некоторыми либо физическими, либо формализованными признаками, которые зависят от конкретного типа УВВ, на котором он находится. Граница набора данных отслеживается и защищается ОС. Так, например, для обработки набора данных с определенного перфокарточного считывающего устройства или лентопротяжного механизма эти устройства выделяются до конца выполнения пункта задания, и невозможно обратиться в это время к этим устройствам из другого задания. На магнитном диске граница набора данных защищается управляющими таблицами и ОС всегда проверяет ее при обращении к набору данных. Если по ошибке задание обращается к чужому набору данных, то происходит прерывание, и система сообщает об ошибке. Такое задание оператор должен вывести из ЭВМ.

3.5.2. Типы наборов данных. Все наборы данных можно разделить на три типа: входные, промежуточные (или временные) и выходные. *Входной набор данных* SYSIN — это набор данных, который подлежит обработке с помощью обрабатывающей программы и передается пункту задания во входном потоке с перфокарт либо с магнитной ленты, магнитного диска и т. д. Наборы данных, находящиеся во входном потоке, вводятся с помощью программы системного ввода RDR. Входными наборами данных, например, на шаге трансляции являются исходные модули, написанные программистами. Входные наборы данных в конце выполнения пункта задания могут либо уничтожаться (DELETED), либо сохраняться (KEPT).

Временный набор данных — это набор данных, который создается и уничтожается в одном и том же пункте задания. Временные наборы данных могут располагаться на магнитных лентах или магнитных дисках.

Выходной набор данных содержит результат решения задания и системные сообщения. В качестве примера можно привести некоторые выходные наборы данных, DD-операторы которых имеют определенные имена: SYSPRINT — выдается выходной набор данных на печать; SYSABEND — выводится на печать содержимое памяти (ядро и раздел, в котором решалось задание), если пункт задания закончился аварийно; SYSUDUMP — выводится на печать содержимое памяти (раздел, в котором выполнялось задание), если пункт задания закончился аварийно. Выходные наборы данных в конце пункта задания либо передаются следующему пункту задания (PASSED), либо выводятся (SYSOUT), либо сохраняются (KEPT).

Сообщения о типах наборов данных и об их состоянии после выполнения пункта задания выдаются в листинге. Уровень (полнота) этого сообщения задается программистом в операторе JOB параметром MSGLEVEL, а класс сообщений — параметром MSGCLASS.

3.5.3. Формат оператора DD. В каждом задании операторы EXEC отделяют один пункт задания от другого. Все операторы DD, которые следуют за

оператором EXEC, описывают наборы данных, принадлежащие этому пункту задания, вплоть до следующего оператора EXEC или оператора // (рис. 3.8). Формат оператора DD приведен в табл. 3.1.

Подполе названия. Имя в подполе названия служит для обращения к оператору DD по имени. Имя в подполе названия указывать не обязательно. Если имя оператора DD опущено, то это значит, что группа операторов DD описывает сцепленные (соединенные в один) наборы данных, например:

```
//PTOM DD ...
```

```
//      DD ...
```

Во втором операторе DD опущено имя, т. е. он описывает набор данных, присоединяемый к набору данных, описанному в операторе DD с именем PTOM.

Если в пункте задания встречаются несколько операторов DD с одинаковыми именами, то распределение ресурсов, выделение памяти и обработка диспозиции выполняется для всех операторов DD, но работа производится только с набором данных, который был определен первым.

Имеются специальные имена операторов DD, которые используются управляющей программой ОС ЕС.

Имя JOBLIB используется для определения библиотеки, содержащей загрузочные модули (программы), которые выполняются в данном задании, т. е. задаются в параметре PGM операторов EXEC. Имя STEPLIB определяет аналогичную библиотеку, используемую только для одного данного пункта задания. Если в задании нет DD операторов с такими именами, то выполняемые программы должны находиться в системной библиотеке SYS1.LINKLIB.

Имя SYSCHK используется для описания набора данных, который используется при повторном выполнении задания, начиная с контрольной точки (см. гл. 9).

Имя SYSABEND описывает набор данных, в который может быть записан дамп, если пункт задания завершился аварийно. Выполнить дамп — это значит скопировать содержимое всей или части памяти. Имя SYSUDUMP используется для описания набора данных, в который может быть записан дамп только области памяти проблемной программы, если пункт задания завершился аварийно.

В подполе operandов оператора DD указываются параметры, перечень которых приводится в табл. 3.4. Параметры в таблице разделены на четыре группы в зависимости от их назначения. Версия 6.1 предлагает более широкий набор параметров, чем версия 4.1, в основном за счет параметров, расширяющих возможности управляющей программы. Оператору в первую очередь необходимо знать параметры VOLUME, SYSOUT, UNIT и DSNAMES, с которыми он встречается в системных сообщениях и при работе с командами. Ниже рассматриваются наиболее простые варианты форматов параметров оператора DD.

3.5.4. Ключевой параметр VOLUME дает описание томов, на которых находятся или будут находиться наборы данных. Том — это носитель данных такой, как магнитная лента, пакет дисков и т. д. Для обращения к набору

Таблица 3.4. Параметры оператора DD

№ п/п	Параметры	Назначение	Версия ОС	
			4.1	6.1
1. Связь с управляющей программой				
1	[DATA]	Описывает набор данных во входном потоке	+	+
2	[COPIES]	Запрашивает несколько копий	-	+
3	[DUMMY]	Дает возможность обойти операции ввода-вывода	+	+
4	[DYNAM]	Позволяет отложить установку тома	-	+
5	[FCB]	Определяет формат страниц для печати выходного набора данных	-	+
6	[TERM]	Осуществляет ввод-вывод через абонентский пункт	-	+
7	[OUTLIM]	Устанавливает предел для числа логических записей выходного набора данных	-	+
8	[UCS]	Определяет вид набора знаков, используемых для выходного набора данных на АЦПУ	-	+
9	[DLM]	Вводит новый ограничитель вместо /*	-	+
2. Использование каналов				
10	[AFF]	Запрашивает каналы	+	+
11	[QNAME]	Дает возможность использовать общий телекоммуникационный метод доступа	-	+
12	[SEP]	Запрашивает отдельные каналы	+	+
3. Организация набора данных				
13	[DCB]	Задаёт информацию, необходимую для завершения блока управления данными	+	+
14	[DDNAME]	Откладывает описание набора данных	+	+
15	[DISP]	Описывает состояние набора данных	+	+
16	[DSNAME] [DSN]	Присваивает имя набору данных	+	+
17	[LABEL]	Сообщает информацию о метке набора данных	+	+
18	[SPACE]	Описывает место, выделяемое набору данных на томе прямого доступа	+	+
19	[SPLIT]	Запрашивает дополнительный объем памяти на томе прямого доступа	+	+
20	[SUBALLOC]	Позволяет подраспределить память на томе прямого доступа	+	+
21	[SYSOUT]	Задаёт имя выходного класса для набора данных	+	+
22	[VOLUME] [VOL]	Описывает том	+	+
4. Запрос на устройства ввода-вывода				
23	[UNIT]	Определяет тип и количество устройств для набора данных	+	+

данных в параметре VOLUME указывается серийный номер тома:

$$\left\{ \begin{array}{l} \text{VOLUME} \\ \text{VOL} \end{array} \right\} = \text{SER} = (\text{список серийных номеров})$$

Например:

$$\text{VOL} = \text{SER} = 111114$$

где 111114 — серийный номер тома. Если указывается список серийных номеров, то они заключаются в круглые скобки и разделяются запятыми, например:

$$\text{VOL} = \text{SER} = (100101, 100102, 100103)$$

3.5.5. Ключевой параметр SYSOUT определяет выходной класс. Выходной класс является признаком, который присваивается выходным данным пункта задания и опознается программой системного вывода WTR. В формате параметра

$$\text{SYSOUT} = \text{класс}$$

класс заменяется именем выходного класса от A до Z или от 0 до 9, например:

$$\text{SYSOUT} = \text{A}$$

Чтобы системные сообщения и выходные наборы данных выдавались на одно устройство вывода, следует указывать один и тот же класс в параметрах MSGCLASS оператора JOB и SYSOUT оператора DD.

3.5.6. Ключевой параметр UNIT выделяет устройство (устройства) для ввода-вывода наборов данных. В формате параметра

$$\text{UNIT} = \left[\begin{array}{l} \text{адрес} \\ \text{тип} \\ \text{группа} \end{array} \right]$$

адрес заменяется физическим адресом устройства; тип — типовым именем (шифром); группа — общим названием класса однотипных устройств, которые задаются при генерации. В Приложении 5 приведены физические, типовые и групповые имена устройств ввода-вывода. Например, при выводе набора данных на АЦПУ параметр UNIT можно записать одним из следующих способов: UNIT=00E, или UNIT=7030, или UNIT=SYSPRINT. При вводе набора данных со считывающего устройства: UNIT=00C, или UNIT=6010, или UNIT=SYSIN. Вывод набора данных на перфокарты: UNIT=0FF, или UNIT=7010, или UNIT=SYSCP. При работе с магнитной лентой: UNIT=5010, или UNIT=280, или UNIT=SYSSQ. Устройства прямого доступа кодируются: UNIT=130, или UNIT=5050, или UNIT=SYSDA.

3.5.7. Ключевой параметр DSNAME позволяет присвоить имя набору данных. В простейшем случае формат содержит имя набора данных

$$\left\{ \begin{array}{l} \text{DSNAME} \\ \text{DSN} \end{array} \right\} = \text{имя}$$

Имя может содержать от 1 до 44 символов и должно начинаться с буквы, например: DSN=PTOM1 или DSN=SYS1.LINKLIB.

Если параметр DSN опущен, то набор данных считается временным, а ОС присваивает ему имя.

3.6. Общая характеристика команд оператора

Команды оператора предназначены для обращения с пультового устройства или управляющего терминала к управляющей программе, изменения хода операций, выполняемых ЭВМ, инициирования новых или прекращения текущих операций. Перечень общих команд оператора приводится в табл. 3.5, а при работе с системой ДУВЗ добавляются команды, приведенные в табл. 12.1 и 12.2. Ниже показан формат команды, если она вводится оператором с пульта пишущей машинки или дисплея:

Команда	Параметры
Мнемокод команды	Параметры, разделенные запятыми

В подполе операции задается мнемокод команды в полном или сокращенном виде (табл. 3.5). Подполе операции отделяется от подполя операндов хотя бы одним пробелом (в некоторых случаях устанавливается только один пробел). Пробелы в самом мнемокоде команды и в параметрах не допускаются. Если команда вводится во входном потоке, то ее формат такой, как приведен в табл. 3.1.

3.6.1. Функциональное назначение команд. Для изучения за основу взят набор команд, который используется при работе с управляющей программой MVT. По ходу изложения отмечаются особенности команд при работе с MFT и SVS.

Команды оператора по функциям можно разделить на следующие группы:
1) DISPLAY, MONITOR, STOP, STOPMN, REPLY, LOG и WRITELOG обеспечивают информационную связь с системой;

2) MOUNT, CANCEL, UNLOAD, VARY и SWAP позволяют управлять состоянием устройств ввода-вывода;

3) SET и HALT обеспечивают начало и конец работы с ОС; DEFINE позволяет перейти к изменению размеров разделов ОП, заданных при генерации системы (режим MFT); DUMP — получить дамп (содержимое) ОП; MODE — получить сведения о режиме работы технических средств контроля и внести изменения в этот режим;

4) START, MODIFY, CANCEL и STOP управляют работой системных программ;

5) START, MODIFY, RESET, CANCEL, STOP, HOLD и RELEASE управляют процессом прохождения заданий в системе;

6) CONTROL, DISPLAY, MONITOR, MSGRT и VARY обеспечивают работу с дисплей-консолями.

При введении в ЭВМ почти любой команды система выдает в ответ сообщение (сообщения) о том, что она восприняла команду, или выдает затребованные оператором данные о состоянии системы. Команды и сообщения системы появляются на консоли не обязательно друг за другом, а в произвольном порядке. Поэтому желательно, чтобы оператор научился видеть команды вместе с сопутствующими им вариантами сообщений системы, наиболее часто встречающиеся на практике.

3.6.2. Формат системных сообщений. В общей сложности система выдает для оператора порядка 500 сообщений на английском языке. Это вызывает

Таблица 3.5. Команды оператора ЭВМ

№ п/п	Мнемокод команды	Сокращение	Назначение	Вид управляющей программы		
				MFT	MVT	SVS
1	CANCEL	C	Прекращает процесс	+	+	+
2	CONTROL	K	Обеспечивает информационную работу с дисплеем	+	+	+
3	DEFINE	N	Запрашивает переопределение разделов	•	—	—
4	DISPLAY	D	Позволяет получить одноразовую, текущую информацию	+	+	+
5	DUMP *)		Позволяет получить дампы ОП	•	•	•
6	HALT	Z	Подготавливает систему к выключению питания	•	•	•
7	HOLD	H	Задерживает выборку заданий	+	+	+
8	LOG	L	Записывает информацию в системный журнал	+	+	+
9	MODE *)		Позволяет получить сведения о режиме работы средств контроля	•	•	•
10	MODIFY	F	Изменяет характеристики процесса	+	+	+
11	MONITOR	MN	Позволяет получить непрерывную текущую информацию	+	+	+
12	MOUNT	M	Резервирует устройство	+	+	+
13	MSGRT *)	MR	Устанавливает маршрутные коды для сообщений ОС	+	+	+
14	RELEASE	A	Разрешает выборку заданий	+	+	+
15	REPLY	R	Используется при ответах оператора на запросы системы	+	+	+
16	RESET	E	Изменяет классы и приоритет	+	+	+
17	SET	T	Сообщает параметры системы планировщику заданий	+	+	+
18	START	S	Начинает процесс	+	+	+
19	STOP	P	Останавливает процесс	+	+	+
20	SWITCH *)		Позволяет заменить набор данных при работе системной мониторной программы (СМП)	+	+	+
21	STOPMN *)	PM	Отменяет выдачу непрерывной информации на консоль	+	+	+
22	SWAP	G	Разрешает обмен томов в динамическом режиме	•	•	•
23	UNLOAD	U	Готовит том для снятия	+	+	+
24	VARY	V	Изменяет состояние устройств	+	+	+
25	WRITELOG	W	Распечатывает системный журнал	+	+	+

Примечание. Знак «•» означает, что команда не может быть введена во входном потоке.

*) Версия 6.1.

определенные трудности при работе, несмотря на то, что имеется интерпретация сообщений на русском языке в документации к системе и в литературе [2, 6, 9].

Ниже приводится формат системных сообщений:

[*][номер]хххууу текст сообщения Рпп

Символ * означает, что требуется обязательное ответное действие со стороны оператора ЭВМ; такое сообщение называется *запросом системы*. Номер представляет собой двузначное десятичное число, которое идентифицирует запрос системы. При ответах на запросы оператор в команде REPLY должен указать тот же номер, что и в запросе. Система постоянно следит за всеми ответами оператора. Номер позволяет системе определить, на какой запрос оператор дал ответ. Если сообщение не требует ответа от оператора, то номер в нем отсутствует. Порядок формирования номеров сообщений устанавливается системой при ее генерации. Номер может иметь одно и то же значение, например 00, или изменяться с наращиванием от 00 до 99, после чего нумерация повторяется сначала. В шестизначном коде хххууу: ххх представляет собой префикс системного сообщения и ууу — порядковый номер сообщения, например 1ЕА001. Наиболее часто встречающиеся префиксы системных сообщений приведены в Приложении 6. Показатель действий оператора к представляет собой букву, означающую: А — *действие* — оператор должен выполнить действие, указанное в тексте сообщения; D — *решение* — оператор должен принять решение, исходя из ряда предложенных системой вариантов; Е — *отложенное действие* — оператор должен выполнить некоторые действия, а система будет продолжать обработку заданий; I — *информация* — информационное сообщение, которое со стороны оператора не требует никаких действий; W — *ожидание* — необходимо выполнить некоторые действия, прежде чем будет продолжена работа системы, которая на это время устанавливается в состояние ожидания; S — *ошибка* — произошла ошибка и система не может продолжать обработку задания, т.е. испорчены какие-то системные области и необходима перезагрузка ОС. В «тексте сообщения» помещается текст, выдаваемый системой на английском языке. При работе с управляющей программой MFT в сообщение включается Рпп, где пп — номер раздела, в котором выполняется задание.

Знакомство с командами оператора ЭВМ следует начать с команд, обеспечивающих информационную связь с системой и позволяющих управлять работой устройствами ввода-вывода, чтобы далее свободно пользоваться ими при изучении процесса загрузки ОС и команд управления выполнением заданиями.

Информационные команды имеют достаточно широкий круг применений *). Они используются без каких-либо ограничений. Команды DISPLAY и MONITOR позволяют получить информацию о состоянии системы и заданий. Команды STOP и STOPMN (ОС версия 6.1) отменяют выдачу информации, начатую по команде MONITOR. Команда REPLY предназначена для ответов системе на ее запросы. Команда LOG служит для записи информации в системный журнал, а команда WRITELOG — для распечатки содержимого системного журнала для анализа работы ЭВМ. Информационные команды могут быть выданы оператором в любой момент времени после загрузки ОС и не влияют на ее работу.

4.1. Получение информации о состоянии системы и заданий

4.1.1. Команда DISPLAY позволяет получить на консоли одноразовую, текущую информацию о времени и дате, состоянии активных заданий в системе, запросах ОС, на которые оператор еще не ответил, состоянии устройств ввода-вывода, состоянии очередей, именах заданий в очередях и др. Сообщения команды DISPLAY позволяют более эффективно следить за прохождением заданий на ЭВМ. Формат команды DISPLAY приводится ниже:

Команда	Параметры	Примечание
{DISPLAY} {D}	T A U [TP [GRAPHIC [TAPE [DASD [UR] [.ONLINE [.OFFLINE] [,адрес] [,nnn] R Q [=список] N [=список] имя задания SQA	В версии 4.1 не используется

Текущее время дня и дату можно получить по команде

D T

*) Информационные команды рассматриваются также в гл. 7 и 11.

в виде сообщения *), например:

IEE136I TIME=10.10.17 DATE=80.058

где время (TIME) равно 10 часов 10 минут и 17 секунд, дата (DATE) — 1980 год, 58 день (27 февраля). Если таймер не работает, то указывается время, которое было установлено при загрузке ОС.

Информация об активных заданиях, которые выполняются в данный момент времени, выдается по команде

D A

Пример сообщения: **)

IEE102I	10.10.32	ACTIVE		DISPLAY		
STRADDR	ENDADDR	SQA	R	SUBT	NAME1	NAME2 NAME3
00482K	00512K	03248		03	MASTER	SCHEDULER
00462K	00482K	00384		00	WTR	00F
00000K	00000K	00600		00	A1	
00000K	00000K	00600		00	A2	
00398K	00462K	00896		00	RDR	00C
00314K	00398K	00896		00	SVODKA20	SVODKA20

В сообщении в колонке NAME1 указаны имена системных задач: WTR — программы системного вывода, RDR — программы системного ввода, MASTER SCHEDULER — программы главного планировщика; A1 и A2 — идентификаторы двух инициаторов INIT и имя нестандартной каталогизированной процедуры — SVODKA20. Для них приведены значения начальных (STRADDR) и конечных (ENDADDR) адресов разделов ОП, в которых они находятся, например, для WTR — 462K и 482K. Программа MASTER SCHEDULER состоит из трех подзадач, о чем указано в колонке SUBT. В колонке NAME2 указаны идентификаторы программ WTR и RDR соответственно; SVODKA20 — идентификатор, присвоенный каталогизированной процедуре SVODKA20. В колонке NAME3 выдается дополнительная информация об активном задании. В колонке SQA указывается величина ОП (в байтах), отведенная под управляющие блоки системной очереди. В колонке R пробел, если раздел, занятый задачами, принадлежит только ей.

Если активных заданий нет, то выдается сообщение:

IEE800I NO AKTIVE TASKS — нет активных задач.

Информация о состоянии всех устройств ввода-вывода (UNIT STATUS) выдается по команде

D U

в виде таблицы, например:

IEE450I	10.20.01	UNIT	STATUS		
UNIT	TYPE	STATUS	VOLSER	VOLSTATE	
008	6020	OFFLINE			
00F	7030	A			
132	5061	S	SD001	PUB/RSDNT	

*) Здесь и далее сообщения, приводимые в примерах, относятся к версии ОС 6.1, режим MVT.

**) В версии 4.1 выдается сообщение IEE821I, которое отличается расположением колонок и отсутствием названий колонок.

Названия колонок: UNIT — физический адрес; TYPE — типовое имя; STATUS — состояние; VOLSER — серийный номер тома; VOLSTATE — характеристика тома. В сообщении приведены данные только для трех устройств: устройства ввода с перфолент типа 6020 с физическим адресом 008, которое находится в недоступном (автономном) состоянии OFFLINE; АЦПУ типа 7030 (физический адрес 00F) — в активном (распределенном) состоянии (A); магнитный диск типа 5061 (физический адрес 132) является системным резидентным устройством (S), имеет серийный номер SD001, том является общим резидентным (PUB/RSDNT).

Данные, полученные по команде D U, полезны для определения состояния устройств ввода-вывода или для определения причины перехода процессора в состояние ожидания и т. п. Надо заметить, что команда D U приводит к выдаче большого количества данных, так как устройств ввода-вывода в системе достаточно много. Параметры, следующие в формате команды DISPLAY за параметром U, позволяют запросить выборочную информацию об устройствах: TP — телекоммуникационных (графических и алфавитно-цифровых) дисплеях, GRAPHIC — графических устройствах, TAPE — накопителях на магнитных лентах, DASD — устройствах прямого доступа, UR — устройствах ввода-вывода символьной информации (пишущих машинках, устройств ввода-вывода с перфолент и т. д.), ONLINE — находящихся только в доступном (оперативном) состоянии, OFFLINE — находящихся только в автономном состоянии. Адрес заменяется физическим адресом устройства, начиная с которого выдается информация для *nnn* опрашиваемых устройств. Если адрес опущен, то начальный физический адрес равен 000, а число опрашиваемых устройств равно 100. Если указано *nnn* > 100, то выдается сообщение об ошибке IEE4521. При неправильном задании параметров выдаются сообщения IEE453I, IEE454I и IEE455I.

Пример 4.1. Запросить сообщение о состоянии двух накопителей на магнитных лентах, начиная с адреса 280:

D U,,280,2

Пусть, например, система выдала сообщение IEE450I со следующими данными:

IEE450I 10.27.09		UNIT STATUS	
UNIT	TAPE	STATUS	VOLSER VOLSTATE
280	5010	OFFLINE	
281	5010	OFFLINE	

Из сообщения видно, что имеются два НМЛ типа 5010 с адресами 280 и 281, которые находятся в автономном состоянии OFFLINE.

Пример 4.2. Запросить сообщение о состоянии двух первых устройств прямого доступа:

D U,DASD,,2

Предположим, что в ответ оператор получил следующее сообщение:

IEE450I 10.28.50		UNIT STATUS	
UNIT	TYPE	STATUS	VOLSER VOLSTATE
130	5061	A=BSY	WD001 PUB/RSDNT
131	5061	O	PD103 PRIV

На устройстве прямого доступа 130, типа которого 5061, находится общий резидентный (PUB/RSDNT) том с серийным номером WD001 в активном состоянии, но он занят (A=BSY). На устройстве 131, тип которого 5061, находится том с серийным номером PD103 в оперативном состоянии (O) и является личным (PRIV) томом. Запятые в примерах 4.1 и 4.2 сохраняются в формате в соответствии с правилами, описанными в гл. 3, для позиционных параметров.

Пример 4.3. Запросить сообщение о состоянии всех дисплеев:

D U,TP

Предположим, что система выдала сообщение IEE450I со следующими данными:

```
IEE450I 10.34.01 UNIT STATUS
UNIT    TYPE    STATUS VOLSER VOLSTATE
050     CHX1    OFFLINE
051     CHX1    OFFLINE
052     CHX2    OFFLINE
053     CHX2    OFFLINE
```

Из сообщения видно, что все четыре дисплея с адресами 050—053 находятся в автономном состоянии OFFLINE; CHX1 — CHX2 — типовые имена дисплеев.

Пример 4.4. Запросить сообщение о состоянии всех графических устройств:

D U,GRAPHIC

В ответ оператор получает сообщение IEE450I, например:

```
IEE450I 10.35.32 UNIT STATUS
UNIT TYPE    STATUS VOLSER VOLSTATE
030  7J66    OFFLINE
.
.
.
0D3   7066    OFFLINE
```

В сообщении указывается, что алфавитно-цифровые дисплеи типа 7066 с адресами 060 и 0D3 находятся в автономном состоянии OFFLINE.

Пример 4.5. Запросить сообщение о состоянии устройств прямого доступа, находящихся только в автономном состоянии:

D U,DASD,OFFLINE

В ответ система выдает сообщение IEE450I, в котором перечисляются адреса устройств (UNIT) и их типовые имена (TYPE) до 7 пар на строке, например:

```
IEE450I 12.35.39 UNIT STATUS
UNIT TYPE UNIT TYPE UNIT TYPE UNIT TYPE
133  5061 134  5061 135  5061 233  5061
```

Пример 4.6. Запросить сообщение о состоянии устройств прямого доступа, находящихся только в оперативном состоянии:

D U,DASD,ONLINE

Система в ответ выдает сообщение IEE450I, например, со следующими данными:

IEE450I 12.36.51			UNIT STATUS	
UNIT TYPE STATUS			VOLSER	VOLSTATUS
130	5061	O	WD001	PUB/REMOV
131	5061	O	PD101	PRIV/RSERV
132	5061	S	SD001	PUB/RSDNT

В сообщении указывается, что на устройстве 130 (типовое имя 5061), которое находится в оперативном свободном (не занятом) состоянии (O), размещается общий перемещаемый (PUB/REMOV) том с серийным номером WD001. На устройстве 131 (типовое имя 5061), которое находится в оперативном (O) состоянии, помещается личный резервный (PRIV/RSERV) том с серийным номером PD101. На устройстве 132 (типовое имя 5061), который является системным (S), находится общий резидентный (PUB/RSDNT) том с серийным номером SD001.

По команде

D R

выдаются все идентификаторы запросов ОС, на которые оператор не ответил с помощью команды REPLY; адреса всех устройств, для которых было выдано сообщение об установке томов, но оператор не установил их; информация об установочных сообщениях IEE504A, которые не были выполнены. Если оператор ответил на все запросы, то система выдает сообщение: IEE111I NO OUT STANDING REQUESTS — нет необслуженных запросов.

Если имеются запросы, на которые оператор не дал ответа, то выдается сообщение IEE110I, например:

IEE110I REPLY IDS : 04
READY UNIT : 142

где 04 — номер (IDS) сообщения, ждущего ответа (REPLY) оператора; устройство 142 ожидает установки (READY UNIT).

Информация о количестве заданий в очередях, если они там имеются, выдается по команде

D Q

Пример сообщения:

IEF869I 10.15.33 QUEUE DISPLAY
QUEUE JOBS STAT QUEUE JOBS STAT QUEUE JOBS STAT
JOB A 0002 JOB B 0001 SOUT L 0001

В сообщении три группы колонок: QUEUE — тип очереди и класс; JOBS — количество заданий в очереди; STAT — состояние очереди. В данном сообщении указано, что в очереди JOB класса A находятся два задания; в очереди JOB класса B имеется одно задание; в выходной очереди SOUT класса L находится одно задание. В колонке STAT появляется значение HELD, если очередь задержана, в противном случае в ней ничего не указывается.

Можно запросить информацию о количестве заданий, которое находится в определенных очередях. В этом случае указывается:

D Q=список

«Список» составляется из имен классов входной очереди от A до O, названий очередей SOUT и HOLD. Всего в списке должно быть не более трех имен и названий очередей. Если список содержит более одного элемента, то они должны быть разделены запятыми и заключены в круглые скобки, например:

D Q = (A,B,C) — запрос сообщения о количестве заданий во входной очереди в классах A,B,C;

D Q=SOUT — запрос сообщения о количестве заданий в выходной очереди SOUT;

D Q = (A,HOLD) — запрос сообщения о количестве заданий во входной очереди JOB класса A и очереди задержанных заданий HOLD.

Если в подполе операндов ничего не указано, то считается, что система должна выдать информацию о всех 15 входных очередях, очереди задержанных заданий и выходной очереди. В ответ на ввод таких команд выдаются сообщения типа IEF869I. Если в очередях заданий нет, то выдается сообщение:

IEE318I QUEUE EMPTY — заданий в очередях нет.

Список имен заданий, находящихся во всех очередях, выдается по команде

D N

Если в очередях имеются задания, то система в ответ на команду DISPLAY в сообщении IEE870I выдает их имена (NAME), например:

```
IEE870I 10.13.47 NAME DISPLAY
QUEUE JOBNAME STAT JOBNAME STAT JOBNAME STAT
JOB A LIST SMFP
SOUT R FI
```

Названия колонок таблицы: QUEUE — вид очереди, JOBNAME — имя задания, STAT — состояние очереди: пробел для JOB и SOUT; HOLD, если задание задержано. В данном сообщении во входной очереди JOB класса A находятся задания LIST и SMFP, в очереди SOUT класса R — задание FI.

Список имен заданий в определенных очередях получается по команде

D N=список

Список составляется так же, как было описано выше. Обе команды **D Q=список** и **D N=список** позволяют оператору получить информацию, которая помогает ему следить за прохождением заданий и вовремя перестроить очереди для более быстрого их выполнения. Если в очередях нет заданий, то выдается сообщение: IEE318I.

Информацию о состоянии определенного задания в очереди можно получить, если в команде DISPLAY указать имя этого задания, например:

D VIK

В этом случае система выдает сообщение только для этого задания, например:

IEF409I VIK CLASS=R, PRTY=00, POS=002, SOUT

где VIK — имя задания, CLASS (класс) — R, PRTY (приоритет) — 0, задание находится в выходной очереди (SOUT), во второй по порядку позиции (POS).

Имена заданий T, A, U, R, Q, N, CONSOLES, SQA, JOB NAMES, DSNNAME, SPACE, STATUS, PFK заключаются в скобки.

Если в команде указано имя задания, которого нет в системе, то выдается сообщение, например:

IEE316I VIK JOB NOT FOUND — задание VIK не обнаружено.

Если же задание уже выбрано из очереди системой, то выдается сообщение

IEE317I VIK JOB SELECTED — задание VIK выполняется.

Верхняя и нижняя границы областей, занимаемых системными очередями, и размер свободной области памяти получаются по команде

D SQA

В ответ система выдает сообщение, например:

IEE051I SQA 00120K 00162K 021.512 — область системных очередей начинается с адреса 00120K (нижняя граница) и заканчивается на адресе 00162K (верхняя граница); размер свободной области памяти равен 21 512 байтов.

4.1.2. Команда MONITOR предназначена для обеспечения непрерывной выдачи на консоль текущей информации о работе планировщика с заданием, об установке и снятии томов, о количестве свободного места на томах, о диспозициях наборов данных. Желательно эту команду выдавать сразу же после загрузки ОС. Формат команды MONITOR:

Команда	Параметры
{MONITOR} {MN}	{JOBNAMES{T}} {DSNAME SPACE STATUS}

Параметр JOBNAMES обеспечивает режим непрерывной выдачи информации о работе планировщика с заданием. В ответ на ввод команды

MN JOBNAMES, T

система выдает сообщения, которые появляются каждый раз, когда планировщик планирует работу очередного задания. Например, выдаются сообщения:

IEF403I RDR STARTED TIME=10.09.11 — программа системного ввода RDR стартовала в 10.09.11;

IEF236I ALLOC. FOR RDR 00C — программе RDR выделено устройство 00C;

IEF237I 00C ALLOCATED TO IEFRDER — устройство 00C выделено в распоряжение планировщика заданий.

Параметр DSNAME обеспечивает режим непрерывной выдачи на консоль сообщений IEF280E и IEF280I о снятии томов с устройств. В этих сообщениях указываются имена постоянных наборов данных, которые используются пунктом задания. Постоянные наборы данных сохраняются после завершения задания, сформировавшего этот набор. Выдача команды

MN DSNAME

приводит к тому, что в сообщениях об установке и снятии томов будет указано имя первого постоянного набора данных, находящегося на этом томе, например, сообщение:

IEF280E K 130,DP6009,ASDK6009,SR,NABOR

означает, что том с серийным номером DP6009 должен быть снят с устройства 130, так как пункт SR задания ASDK6009, работающий с этим томом, закончил с ним работу. NABOR — имя первого постоянного набора данных на этом томе.

При указании параметра SPACE на консоль выводятся данные о количестве свободного места, которым располагает память прямого доступа. По команде

MN SPACE

выдается сообщение, например:

IEF234E R 130,WD001,SPACE=196,0047,0003/0193,0019,RDR

Символ R указывает на то, что том должен быть снят с устройства 130, но оставлен недалеко от ЭВМ для возможного использования в ближайшем будущем. Том используется программой системного ввода RDR. Серийный номер тома WD001 имеет стандартные метки. В параметре SPACE указывается: 196 — общее число всех свободных цилиндров на томе, 0047 — общее число дорожек в дополнение к свободным цилиндрам, 0003 — области (или экстенды), разделяющие цилиндр и дорожки, 0193 — максимальное число непрерывных свободных цилиндров самого большого экстенда во всем остающемся пространстве, 0019 — максимальное число дорожек в дополнение к свободным цилиндрам самого большого экстенда во всем оставшемся пространстве. *Экстенд* — это непрерывная область памяти в устройстве прямого доступа, выделенная для конкретного набора данных.

Параметр STATUS устанавливает режим выдачи на консоль в момент окончания выполнения задания или пункта задания имен наборов данных с указанием диспозиций PASSED, KEPT, DELETED, CATALOGED, UNCATALOGED, SYSOUT, SYSIN и серийных номеров томов, на которых они расположены. Например, в ответ на команду

MN STATUS

может быть получено следующее сообщение:

IEF285I GIS.ATS KEPT — набор данных GIS.ATS сохраняется;

IEF285I VOL SER NOS=LD001 — серийный номер тома LD001.

Часто команды MONITOR с параметрами JOBNAMES,T и SPACE включаются в систему при ее генерации, как автоматические, подобно командам START (п. 7.4).

4.2. Отмена непрерывной выдачи информации

Отменить выдачу информации, начатой по команде MONITOR, можно командами STOP и STOPMN (версия 6.1). Формат команд:

Команда	Параметры
{STOP} P	{JOBNAMES DSNAME SPACE STATUS}
{STOPMN} PM	

В формате команды STOP указывается тот же параметр, который был указан в команде MONITOR. В зависимости от того, какой параметр указан в команде STOP, отменяются те или иные сообщения, которые выдавались по команде MONITOR.

4.3. Выдача ответов на запросы системы

Команда REPLY предназначена для ответов на запросы системы, требующих от оператора введения определенной информации для ЭВМ. Оператор обязательно должен дать ответ на каждый такой запрос. Команда REPLY может быть выдана сразу же или позднее. Формат команды:

Команда	Параметры
{REPLY} R	номер, 'текст'

В формате команды REPLY указывается *номер запроса*, на который оператор дает ответ. Номер в запросе состоит из двух десятичных цифр от 00 до 99. Значение номера оператор берет из запроса, выдаваемого системой (см. п. 3.6). Часто текст ответа находится в запросе системы или дается указание о том, какой ответ хочет получить система. Апострофы, указанные в формате команды, в текст не входят. Апострофы обязательно указываются в версии 4.1 и в некоторых случаях в версии 6.1. Если ответ принят системой, то выдается сообщение IEE600I, в котором появляются только первые 80 символов текста команды REPLY.

Пример 4.7. Ответить на запрос системы:

*15 IGF509I REPLY DEVC OR 'NO'

где номер запроса 15; система запрашивает ответ: оператор должен сообщить номер устройства (DEVC) или ответить, что устройство не нужно (NO). Если устройство не нужно, то оператор отвечает:

R 15,'NO'

Если система приняла ответ, то выдается сообщение:

IEE600I REPLY TO 15 IS; NO — ответ оператора NO принят (версия ОС 6.1);

Пример 4.8. Пусть оператор обратился к системе за информацией: имеются ли запросы, которые остались без ответов, по команде

D R

и получил ответ в виде следующего сообщения:

IEE1101 REPLY IDS:12

По протоколу оператор находит запрос 12 и сопутствующие ему сообщения, например:

IEF8611 FOLLOWING RESERVED DATA SET NAMES UNAVAILABLE TO PRIM1 — заданию PRIM1 требуется набор данных, который зарезервирован в постоянное время системой:

IEF8631 DSNAMES=EC.LINKLIB2 — набор данных EC.LINKLIB2 недоступен заданию.

Запрос следует за этим сообщением:

*12 IEF864D REPLY 'RETRY' OR 'CANCEL' OR 'WAIT' — предлагаются следующие ответы (REPLY): 'RETRY' — повторить обращение к набору данных или 'CANCEL' — прекратить выполнение задания, или 'WAIT' — приостановить его обработку до тех пор, пока набор данных EC.LINKLIB2 не станет доступным заданию. Для того чтобы дать правильный ответ, необходимо знать состояние системы и запрашиваемого набора данных.

В данном запросе ответ заложен в виде набора слов, заключенных в одиночные апострофы. Для ответа следует выбрать одно из предлагаемых слов. Пусть, например, оператор ответит:

R 12,'WAIT'

Такой ответ можно выдать в ОС с переменным числом задач, чтобы приостановить обработку задания до освобождения требуемого набора данных. Если система приняла ответ оператора, то на консоль выдается сообщение (версия 4.1):

IEE6001 ACCEPTED REPLY TO MSG 12 IS:'WAIT' — ответ на запрос 12 принят системой.

Сообщение IEE7011 выдается, если выдан ответ, а запроса не было; IEE7021 — если сделана синтаксическая ошибка в команде REPLY; IEE7041 — ответ на запрос уже выдан или номер запроса неверный.

4.4. Работа с системным журналом

Системный журнал SYSLOG ведется в двух наборах данных: SYS1.SYSVLOGX и SYS1.SYSVLOGY. Они находятся на устройствах прямого доступа и включаются в ОС при генерации системы. Системный журнал является копией информации на консоли оператора. С помощью команды LOG оператор может ввести в системный журнал информацию, связанную с описанием необычных, сбойных ситуаций и т. д. Системный журнал открывается при начальной загрузке системы. Запись в журнал осуществляется попеременно: сначала в один набор данных до полного его заполнения, а потом в другой. Заполненный журнал периодически закрывается системой, и содержимое его выдается на АЦПУ. Системный журнал закрывается в конце работы ОС после ввода команды HALT, но при желании можно закрыть журнал в любое время по команде WRITELOG. Система постоянно информирует опе-

ратора о состоянии журнальных наборов данных с помощью сообщений для наборов данных SYS1.SYSVLOGX или SYS1.SYSVLOGY, например:

IEE041I LOG NOW RECORDING ON SYS1.SYSVLOG {X^{*}Y} ON 131

В данном сообщении указывается, что журнальный набор данных SYS1.SYSVLOGX (или SYS1.SYSVLOGY) находится на устройстве 131 и в него производится запись. После выдачи этого сообщения можно пользоваться командой LOG. Как только набор данных, например X, заполнится, ОС его закрывает, открывает набор данных Y, и наоборот. При закрытии набора данных выдается сообщение:

IEE042I LOG DATA SET SYS1.SYSVLOG {X^{*}Y} ON 131 CLOSE

После того как набор данных закрывается, он ставится ОС в системную выходную очередь SOUT, например класса C, о чем выдается сообщение:

IEE043I LOG DATA SET {X^{*}Y} QUEUED TO SYSOUT CLASS C

а затем выводится на АЦПУ оператором с помощью команды START (гл. 6).

4.4.1. Команда LOG предназначена для записи оператором информации в системный журнал. Формат команды LOG приводится ниже.

Команда	Параметры
{LOG} {L}	'текст'

где *текст* — это информация, которую оператор хочет включить в системный журнал (до 128 символов). Одиночные апострофы являются обязательными синтаксическими знаками и не входят в содержание текста. Текст, указанный в команде LOG, например:

L 'РАБОТАЕТ ОПЕРАТОР ПЕТРОВА'

помещается в системный журнал и будет иметь вид:

201958 0000 L 'РАБОТАЕТ ОПЕРАТОР ПЕТРОВА'

где 201958 — время записи в журнале (час, минута, секунда); 0000 — информация выдается на главную консоль без маршрутного кода. Далее идет повторение команды LOG. Если в ЭВМ таймер не включен, то информация о времени будет отсутствовать. Если запись в журнале не обеспечивается системой, то может быть выдано сообщение (в зависимости от причины):

IEE026I LOG NO SUPPORTED — набор данных для LOG недоступен;

IEE037I LOG NOT SUPPORTED — набор данных для LOG недоступен;

IEE038I LOG NOT SUPPORTED. PTOM1 NOT MOUNTED — где PTOM1 — номер тома, который не смонтирован;

*) В сообщениях буква X или Y присоединяется к SYSVLOG, т.е. получается SYS1.SYSVLOGX или SYS1.SYSVLOGY, в зависимости от того, о каком наборе данных идет речь.

IEE039I LOG SYS1. SYSVLOG {X} NOT
FOUND ON 131— {Y}

набор данных не обнаружен на устройстве 131;

IEE040I LOG I/O ERROR ON 131 — ошибка ввода-вывода в момент записи текста;

IEE044I LOG ABEND COMPLETION CODE=hhh — аварийное завершение с кодом завершения hhh;

IEE045I LOG INACTIVE — содержимое обоих журналов выводится или ждет вывода.

После того как закончится вывод набора данных, выдается сообщение:

IEE046I LOG NOW ACTIVE — системный журнал активен.

4.4.2. Команда **WRITELOG** используется для печати системного журнала или для его закрытия. Формат команды **WRITELOG**:

Команды	Параметры
{WRITELOG} W	[класс] [CLOSE]

Если в команде **WRITELOG** параметр опущен, то это значит, что для печати содержимого системного журнала используется класс, установленный при генерации ОС. При вводе команды

W

наиболее часто выдаются сообщения, например:

IEE042I LOG DATA SET — SYS1.SYSVLOGY ON 250 CLOSED — набор данных SYS1.SYSVLOGY на устройстве 250 закрыт;

IEE041I LOG NOW RECORDING ON SYS1.SYSVLOGX ON 250 — запись производится в набор данных SYS1.SYSVLOGX, который находится на устройстве 250;

IEE043I LOG DATA SET Y QUEUED TO SYSOUT CLASS A — набор данных SYS1.SYSVLOGY выводится в классе A.

Возможна выдача сообщений IEE043I, IEE037I, IEE039I, IEE045I, IEE046I (см. выше). Кроме того, в случае, если команда W выдана повторно, а предыдущая не выполнена, то выдается одно из сообщений IEE032I или IEE033I. Если системный журнал занят, то выдается сообщение IEE034I.

В формате команды W можно указать класс, заменяя его именем выходного класса от A до Z или от 0 до 9, например:

W D

По этой команде системный журнал будет распечатан в выходном классе D. Если этот класс отличается от того, который был установлен при генерации, то оператор обязан обеспечить работу программы системного вывода WTR в этом классе, если это не было сделано ранее (см. гл. 6).

Параметр CLOSE предписывает закрыть системный журнал:

W CLOSE

Вновь системный журнал можно открыть только при начальной загрузке системы.

Г л а в а 5. УПРАВЛЕНИЕ УСТРОЙСТВАМИ ВВОДА-ВЫВОДА

В распоряжении оператора имеются пять команд: VARY, MOUNT, CANCEL, UNLOAD и SWAP, пользуясь которыми он может управлять устройствами ввода-вывода. Команда VARY позволяет перевести устройства ввода-вывода из автономного состояния в оперативное, и наоборот. *Оперативным состоянием (ONLINE)* называется такое состояние устройства, при котором оно может быть выделено обрабатывающей программе; *автономным состоянием (OFFLINE)* — когда устройство не может быть выделено обрабатывающей программе. Команда MOUNT предназначена для резервирования (монтирования) тома для работы, а команда CANCEL — для отмены команды MOUNT. Команда UNLOAD подготавливает устройство для снятия с него тома. Команда SWAP позволяет динамически перемещать тома с устройства на устройство.

5.1. Изменение состояния устройств ввода-вывода

Команда VARY предназначена для перевода устройств ввода-вывода в оперативное или в автономное состояние. Формат команды VARY:

Команда	Параметры
{VARY} {V}	(адрес 1, адрес 2, ...) { .ONLINE .OFFLINE .PATH,адрес{.ONLINE .OFFLINE}}

В формате команды VARY указывается адрес (адреса) устройства ввода-вывода, состояние которого следует изменить. Одной командой можно изменить состояние нескольких устройств. Если указывается адрес одного устройства, то круглые скобки опускаются. Параметр ONLINE переводит устройство (устройства) ввода-вывода в оперативное состояние, и оно становится доступным планировщику заданий для выделения его обрабатывающей программе. Когда устройство находится в оперативном состоянии, то в сообщении IEE450I, которое получается по команде DISPLAY, это состояние конкретизируется: занято (BSY), консоль (C), ждет установки (MTR), оперативное свободное (O) и является системным (S).

Пример 5.1. Перевести устройства с физическими адресами 00C и 00F в оперативное состояние:

V (00F,00C),ONLINE

После перевода устройств в состояние ONLINE система выдает для каждого из них сообщение. В данном случае будут выданы два сообщения:

IEE302I 00F ONLINE — устройство 00F переведено в состояние ONLINE;

IEE302I 00C ONLINE — устройство 00C переведено в состояние ONLINE.

Параметр OFFLINE выводит из подчинения планировщика заданий устройство (устройства) ввода-вывода и переводит его в автономное состояние, т.е. оно не может быть выделено обрабатывающей программе. Переводить устройства ввода-вывода следует либо до начала выполнения задания, либо после завершения пункта задания. Если указанные в команде VARY устройства уже распределены планировщиком заданий другим заданиям, то их состояния изменятся только после того, как все текущие задачи закончат работу с этими устройствами.

Пример 5.2. Перевести устройство с физическим адресом 135 в автономное состояние:

V 135,OFFLINE

Сообщение о переводе устройства в состояние OFFLINE оператор получит только после того, как система обратится к планировщику заданий, т.е. начнет выполнять какое-либо задание. Чтобы получить сообщение после выдачи команды VARY, очень часто выдается команда START для заведомо несуществующей процедуры, например:

S S — вызывается каталогизированная процедура с именем S, хотя известно, что такой процедуры нет в библиотеке SYS1.PROCLIB.

Система в ответ выдает сообщения:

IEE122I START COMMAND JCL ERR — ошибочно введена команда START;

IEF452I S JOB NOT RUN — JCL ERROR TIME = 08.52.59 — из-за ошибки задание S не выполняется.

Вслед за этими сообщениями появляется сообщение о переводе устройства в состояние OFFLINE:

IEF281I 135 NOW OFFLINE — устройство 135 переведено в состояние OFFLINE.

Если устройство, состояние которого следует изменить по команде VARY, находится в процессе изменения состояния, то выдается сообщение, например:

IEE339I 135 CHANGING STATUS — устройство 135 находится в процессе изменения состояния. После получения такого сообщения следует повторить команду VARY.

В системе все устройства ввода-вывода подсоединяются к процессору и ОП через каналы (рис. 1.1). При работе возможна ситуация, когда путь, ведущий к устройству окажется неисправным или долго занятым, о чем система уведомляет оператора с помощью сообщений. В этом случае можно назначить новый путь к устройству, указав в команде VARY параметр PATH (система при этом должна иметь средство переключения каналов ввода-вывода — APR). Кроме параметра PATH, указывается адрес устройства. Каждое устройство ввода-вывода может иметь до четырех ведущих к нему путей. Параметр ONLINE разрешает использовать новый путь, а OFFLINE запрещает его использовать. Если к устройству ведет один-единственный путь, то такой формой команды пользоваться нельзя. Нельзя менять путь к устройствам прямого доступа, если они используются несколькими ЭВМ в многомашинной вычислительной системе, и к телекоммуникационным устройствам.

Пример 5.3. Определить состояние устройств 130 и 132 и перевести их в автономное состояние. Чтобы определить в каком состоянии находятся устройства, следует сначала дать команду DISPLAY:

D U,DASD,,130,3

Пусть окажется, что устройства 130 и 132 находятся в оперативном состоянии O (STATUS) согласно сообщению:

```
IEE450I 10.20.01 UNIT STATUS
UNIT TYPE STATUS VOLSER VOLSTATE
130 5061 O WD001 PUB/REMOV
131 5061 O PD101 PRIV/RSERV
132 5061 O UD001 PUB/REMOV
```

Для того чтобы перевести устройства 130 и 132 в автономное состояние, оператор должен выдать команду:

V (130,132),OFFLINE

Как только устройства освободятся и не будут нужны заданиям, система переведет их в автономное состояние и выдаст об этом сообщения:

```
IEF281I 130 NOW OFFLINE
IEF281I 132 NOW OFFLINE
```

5.2. Резервирование устройства ввода-вывода

Команда MOUNT резервирует устройство ввода-вывода для определенного тома, который может потребоваться заданию или нескольким пунктам задания. Обычно в качестве резервируемого устройства выбирается накопитель на сменных магнитных дисках или накопитель на магнитной ленте, на которых том должен быть сразу же установлен после выдачи команды MOUNT. Зарезервированный том распределяется заданию или пунктам задания планировщиком. Формат команды MOUNT *):

Команда	Параметры
{MOUNT} {M}	адрес {VOL=(NL,номер)} [USE={STORAGE PUBLIC PRIVATE}]

В формате команды MOUNT указывается физический адрес устройства ввода-вывода, которое резервируется. В MFT устройство ввода-вывода предварительно должно быть разгружено с помощью команды UNLOAD.

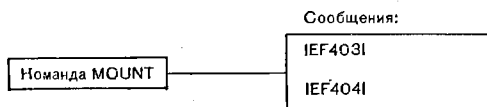
В версии 4.1 параметр VOL используется только при резервировании накопителей на магнитных лентах и не используется для резервирования томов прямого доступа. В версии 6.1 параметр VOL можно использовать для резервирования томов прямого доступа. Символы NL указываются, если том не имеет стандартных меток. Символы SL показывают, что том имеет стан-

*) Здесь и далее подчеркнутое значение параметра используется системой по умолчанию параметра.

дартные метки. Вместо *номер* указывается серийный регистрационный номер тома.

Сообщения, выдаваемые при монтировании тома, зависят от того, в каком состоянии находится устройство: оперативном ONLINE или автономном OFFLINE. Порядок выдачи сообщений приведен на рис. 5.1.

а) Устройство 231 в исходном состоянии находится в ONLINE.



б) Устройство 231 в исходном состоянии находится в OFFLINE.

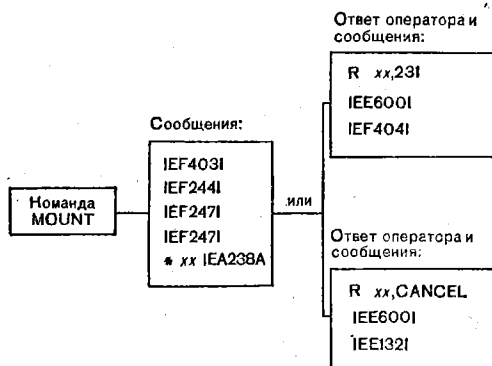


Рис. 5.1. Порядок выдачи сообщений при монтировании тома, например, 231.

Пример 5.4. Перевести устройство 231 в оперативное состояние и смонтировать на нем личный том MVT670. Для перевода устройства в оперативное состояние следует выдать команду

V 231,ONLINE

После перевода устройства в оперативное состояние получим сообщение:

IEE302I 231 ONLINE

Далее выдаем команду для монтирования на нем тома со стандартными метками:

M 231,VOL=(SL,MVT670)

В ответ система выдает сообщения:

IEF403I MOUNT STARTED TIME=13.48.58 — начался монтаж тома;

IEF404I MOUNT ENDED TIME=13.49.08 — монтаж тома закончен,

Проверим состояние тома по команде DISPLAY:

D U,,,231,1

IEE450I 13.49.21 UNIT STATUS

UNIT TYPE STATUS VOLSER VOLSTATUS

231 5061 O MVT670 PRIV/RSERV

Из сообщения видно, что устройство 231 находится в оперативном состоянии и свободно. Том MVT670 личный резервный (PRIV/RSERV).

Пример 5.5. Резервировать устройство 131 для тома с серийным номером OM0023, в котором используются стандартные метки. Исходное состояние устройства 131 автономное:

M 131,VOL=(SL,OM0023)

В ответ система выдает серию сообщений, например:

IEF403I MOUNT STARTED TIME=12.00.00 — начался монтаж тома;

IEF244I MOUNT. 131. IEFRDEN UNABLE TO ALLOCATE — система не может распределить устройство набору данных IEFRDER;

IEF247I MOUNT ALLOCATION RECOVERY — система не может распределить устройство для задания и пытается повторить распределение;

IEF247I MOUNT 131 OFFLINE — устройство 131 резервировано и находится в автономном состоянии.

Далее система обращается к оператору с запросом:

*04 IEF238A REPLY DEVICE NAME, OR 'CANCEL' — система просит подтвердить номер устройства или уничтожить задание.

Предположим, что оператор подтверждает номер устройства с помощью команды REPLY:

R 4,131 — номер запроса 4, номер устройства 131. Система отвечает оператору:

IEE600I REPLY TO 04 IS; 131 — ответ на запрос 4 принят системой; адрес устройства — 131;

IEF404I MOUNT ENDED TIME=12.00.24 — монтирование тома окончено.

Параметр USE предназначен для резервирования томов прямого доступа. Выбирается один из трех параметров: STORAGE, PUBLIC, PRIVATE. Параметр STORAGE сообщает системе о том, что том прямого доступа будет использоваться как *рабочий том*. На рабочем томе можно размещать постоянные и временные наборы данных без указания серийного регистрационного номера тома. Параметр PUBLIC указывается, если том общий. *Общий том* используется для хранения на нем временных и постоянных наборов данных. В последнем случае указывается серийный регистрационный номер тома. Временные тома монтируются только на время, и регистрационный номер не указывается. Параметр PRIVATE является признаком личного тома. *Личный том* служит для записи набора данных пользователя. Если резервируется личный том, то обязательно надо указать серийный регистрационный номер тома. По умолчанию параметра USE том считается личным. Ниже приведены примеры записи команды MOUNT:

M 196,USE=PUBLIC — резервировать общий том на устройстве прямого доступа 196;

M 197,VOL=(SL,111555),USE=PUBLIC — резервировать общий том с серийным номером 111555 на устройстве прямого доступа 197;

M 195,USE=STORAGE — резервировать устройство прямого доступа 195 для хранения на нем рабочего набора данных;

M 284,VOL=(ML0058) — резервировать устройство 284 для тома с серийным номером ML0058 со стандартными метками;

М 193,VOL=(PTOM),USE=PRIVATE — зарезервировать устройство прямого доступа 193 для хранения на нем личного тома со стандартными метками; серийный номер PTOM.

5.3. Отмена зарезервированного устройства

Если команда MOUNT введена ошибочно и зарезервированный том еще не выделен системе, то она может быть отменена по команде CANCEL. Формат команды CANCEL приводится ниже.

Команда	Параметры
{CANCEL} C	адрес

В формате команды CANCEL указывается физический адрес устройства, так как команда MOUNT работает как каталогизированная процедура, для которой *адрес* является идентификатором.

Пример 5.6. Предположим, что на устройстве 146 установлен том DL3015. Во время выдачи команды MOUNT была сделана ошибка при указании номера тома, а именно, указан том DL3005:

М 146,VOL=(SL,DL3005),USE=PRIVATE — смонтировать личный том DL3005 на устройстве 146.

Система выдает сообщения:

IEF403I MOUNT STARTED TIME=08.59.26 — монтаж тома начался;

IEF234E R 146,DL3015,MOUNT — том DL3015 должен быть снят с устройства 146;

IEF533A M 146,DL3005,,MOUNT,146 — смонтировать том DL3005 на устройстве 146.

Поскольку при выдаче команды произошла ошибка в указании номера тома, то оператору следует выдать команду CANCEL для уничтожения команды MOUNT:

C 146

В ответ система выдает сообщения:

IEE301I 146 CANCEL COMMAND ACCEPTED — задание с идентификатором 146 отменено в ответ на ввод команды CANCEL;

IEE134I MOUNT COMMAND DEV ALLOC ERR — во время обработки команды MOUNT была обнаружена ошибка;

IEF251I MOUNT JOB CANCELED — система закончила выполнение команды MOUNT в соответствии с командой CANCEL;

IEF450I MOUNT. 146. ABEND=S222 TIME=08.59.49 — задание MOUNT аварийно прекращено с системным кодом 222.

После получения этих сообщений оператору следует повторно выдать команду MOUNT с правильным номером тома.

5.4. Разгрузка устройства ввода-вывода

Команда UNLOAD используется для подготовки устройства к снятию (разгрузке) с него любого съемного тома. Чаше она используется для томов, установленных по команде MOUNT. Команда UNLOAD сообщает си-

стеме о том, что с устройства будет снята магнитная лента или пакет дисков. Подготовка к снятию заключается в том, что система следит за окончанием выполнения задания, связанного с этим устройством. Формат команды:

Команда	Параметры
{UNLOAD} U	адрес

В формате команды UNLOAD указывается физический адрес устройства, с которого должен быть снят том, например:

U 193

В ответ система выдает сообщение:

IEF282I 193 NOW UNLOAD — том с устройства 193 можно снять.

Чтобы при переводе устройства в состояние OFFLINE быстрее получить сообщение IEF282I, можно вслед за UNLOAD выдать команду START, как было показано в п. 5.1.

После того как выдана команда UNLOAD, следует выдать команду VARY для перевода устройства в состояние OFFLINE. Если этого не сделать,

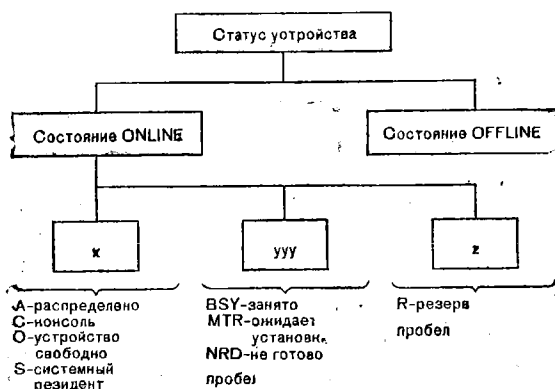


Рис. 5.2. Состояние устройств ввода-вывода.

то может возникнуть ситуация, когда система обратится к этому устройству и выделит его обрабатывающей программе. В результате процессор может перейти в состояние ожидания.

Пример 5.7. Размонтировать том DL3014 на устройстве 130 и том DL3003 на устройстве 131. Проверим состояние этих устройств до разгрузки (рис. 5.2):

D U,,,130,2

IEE450I 08.28.15 UNIT STATUS

UNIT TYPE STATUS VOLSER VOLSTATE

130 5061 O DL3014 PUB/REMOV

131 5061 A DL3003 PUB/REMOV

Дадим команды для подготовки к размонтированию:

```
U 130
U 131
```

В ответ получим сообщения:

```
IEF282I 130 NOW UNLOADED
IEF282I 131 NOW UNLOADED
```

Проверим состояние устройств:

```
D U,,,130,2
IEE450I 08.31.33 UNIT STATUS
UNIT TYPE STATUS VOLSER VOLSTATE
130 5061 O=NRD /REMOV
131 5061 O=NRD /REMOV
```

Выдадим команды для перевода устройств в состояние OFFLINE:

```
V 130,OFFLINE
V 131,OFFLINE
```

В ответ получим сообщения:

```
IEF281I 130 NOW OFFLINE
IEF281I 131 NOW OFFLINE
```

Проверим состояние устройств:

```
D U,,,130,2
IEF450I 08.34.44 UNIT STATUS
UNIT TYPE STATUS VOLSER VOLSTATE
130 5061 OFFLINE /REMOV
131 5061 OFFLINE /REMOV
```

5.5. Динамическое перемещение тома

Команда SWAP служит для запроса динамического перемещения тома или для ответа на системный запрос о динамическом перемещении тома. *Динамическое перемещение* — это средство, позволяющее демонтировать и переставлять (в случае необходимости) съемные тома с устройства на устройство, не вызывая при этом аварийного окончания задания и не повторяя загрузки ОС. Это бывает нужно, когда какое-либо устройство работает не очень хорошо. Формат команды SWAP:

Команда	Параметры
{SWAP} {G}	{ON OFF xxx,yyy}

В формате команды SWAP параметр ON означает, что активизируются средства динамического перемещения томов. Это позволяет переставлять тома с одного устройства на другое. Параметр OFF говорит о том, что тома

переставлять нельзя, так как они находятся в неактивном состоянии. Формат команды SWAP позволяет также задать конкретный физический адрес устройства (по основному пути): *xxx* — адрес устройства, с которого должен быть снят том, и *yyy* — адрес устройства, на которое следует поставить том. Задания, использующие эти устройства, не снимаются. Устройства, указанные в команде SWAP, должны быть одного типа.

Пример 5.8. Заменить устройство 191 на 195:

G 191,195

Если система разрешит произвести такую замену, то выдается сообщение:
IGF502E PROCEED WITH SWAP 191 TO195 — система разрешает произвести замену устройства 191 на 195.

Пример 5.9. Пример назначения оператором заменяющего устройства. Пусть система выдала сообщения:

IGF509I SWAP 131 — взамен устройства 131 необходимо выделить другое устройство;

*14 IGF509A REPLY DEVC, OR 'NO' — запрос системы о том, чтобы оператор указал номер устройства или решил не выделять заменяющего устройства.

Пусть, например, оператор решил выделить устройство 130:

R 14,130

После ввода команды система выдает сообщения:

IEE600I REPLY TO 14 IS; 130 — ответ оператора принят системой;

IGF502E PROCEED WITH SWAP 131 TO 130 — выполнить перестановку тома со 131 на 130 устройство.

Г л а в а 6. УПРАВЛЕНИЕ СИСТЕМНЫМИ ПРОГРАММАМИ (РЕЖИМ MVT)

К системным программам относятся программа системного ввода (RDR, RDR400, RDR3200), программа инициатора (UNIT), программа системного вывода (WTR, DSO)*). Они являются частью планировщика заданий и предназначены для обработки заданий в системе.

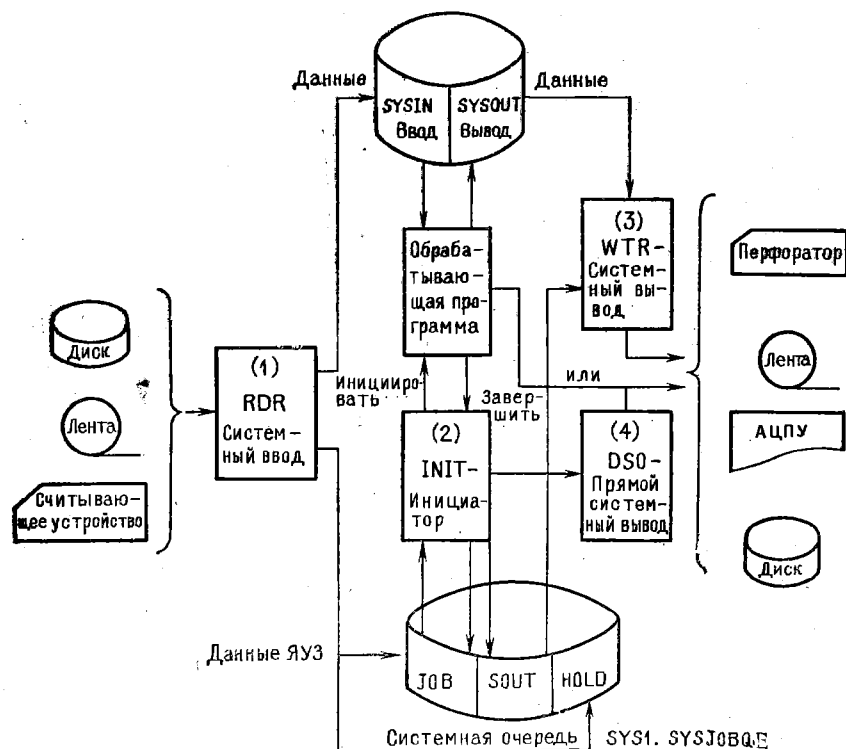


Рис. 6.1.

На рис. 6.1 показан порядок обработки заданий системными программами, если задания поступают во входном потоке. Программа системного ввода RDR(1) обеспечивает ввод заданий. Она проверяет правильность записи

*) Здесь и далее используются стандартные имена системных программ, хотя при генерации разных модификаций ОС могут использоваться и другие имена.

управляющих операторов языка управления заданиями и, если они правильны, помещает задания на устройство прямого доступа. Одновременно программа RDR организует для заданий входную очередь JOB и очередь задержанных заданий HOLD. Задания во входной очереди упорядочиваются в соответствии с параметрами CLASS, указанными в операторах JOB (п. 3.3).

Таблица 6.1. Имена стандартных каталогизированных процедур системных программ

Имя	Назначение
DSO	Программа прямого системного вывода
UNIT	Инициатор
RDR	Программа системного ввода (неблокированные наборы данных SYSIN)
RDR400	Программа системного ввода (наборы данных SYSIN заблокированы по 400 байтов)
RDR3200	Программа системного ввода (наборы данных SYSIN заблокированы по 3200 байт)
WTR	Программа системного вывода

Всего входная очередь может иметь 15 классов (от А до О). Внутри каждого класса задания устанавливаются в очередь согласно приоритетам, указанным в параметрах PRTY операторов JOB. Если в управляющих операторах ЯУЗ имеются ошибки, то задание снимается. Инициатор заданий UNIT (2) выполняет задание под управлением супервизора. Кроме того, он передает выходные данные на устройство прямого доступа для вывода информации с помощью системного вывода (3), организует выходную системную

Таблица 6.2. Команды, управляющие системными программами

Имя системной программы	START	MODIFY	STOP	CANCEL
RDR (RDR400, RDR3200)	+	-	+	+
INIT	+	+	+	-
WTR	+	+	+	+
DSO	+	+	+	-

очередь SOUT в соответствии с классами, указанными в параметре SYSOUT оператора DD. Максимальное количество выходных классов равно 36 (от А до Z и от 0 до 9). Имеется другой способ вывода информации, а именно, непосредственный вывод (прямой) из ОП на устройство вывода, минуя устройство прямого доступа, с помощью программ прямого системного вывода DSO (4). Если выбирается каталогизированная процедура из библиотеки, то она обрабатывается только программами UNIT и WTR (или DSO).

В мультипрограммном режиме система может обеспечить одновременную работу трех программ системного ввода RDR, 15 инициаторов, 36 программ системного вывода WTR и 36 программ прямого системного вывода DSO.

Все системные программы запускаются с помощью стандартных каталогизированных процедур, которые находятся в системной библиотеке SYS1.PROCLIB. Каждая каталогизированная процедура вызывается по имени,

которое указывается в команде START. В табл. 6.1 приведены имена и назначение каталогизированных процедур, с которыми наиболее часто приходится работать оператору. Все системные программы работают с определенными параметрами, установленными при генерации в каталогизированных процедурах. Эти параметры могут быть изменены командой START и MODIFY. Команда CANCEL может отменить команду START, а команда STOP — остановить работу системных программ. В табл. 6.2 показан набор команд, который используется при работе с каждой конкретной системной программой.

6.1. Управление программой системного ввода (RDR)

6.1.1. Запуск программы системного ввода RDR осуществляется по команде START, формат которой приводится в табл. 6.3. Обязательным параметром является имя системной программы — RDR (или RDR400, или RDR3200). Остальные параметры являются необязательными*). Идентификатор представляет собой имя, которое присваивается системной программе, если оператор запускает их несколько. Он используется в режимах MVT и SVS**). Если идентификатор опущен, то система образует его сама, причем она может создать одинаковые имена для разных системных программ, что оказывается не всегда удобным для работы. В последнем случае в качестве идентификатора система использует адрес устройства или имя самой каталогизированной процедуры. Идентификаторы используются в дальнейшем в командах STOP и CANCEL. Адрес заменяется физическим адресом устройства. Набор сообщений, который выдается системой после ввода команды START, зависит от того, в каком состоянии находится устройство ввода: ONLINE или OFFLINE. Здесь и далее считаем, что в системе действует команда MN JOBNAMES,T. В примерах 6.1 и 6.2 приведены оба случая запуска программы системного ввода (рис. 6.2).

Пример 6.1. Запустить программу системного ввода RDR для считывания заданий во входном потоке с устройства 00C, которое находится в состоянии ONLINE:

S RDR,00C — стартовать программу системного ввода; идентификатором является адрес устройства 00C.

В случае успешного запуска программы RDR система выдает следующие сообщения, например:

IEF403I RDR STARTED TIME=10.09.11 — программа системного ввода стартовала в 10 часов 09 минут 11 секунд;

IEF236I ALLOC. FOR RDR 00C — программе RDR выделено устройство 00C;

IEF237I 00C ALLOCATED TO IEFRDER — устройство 00C выделено в распоряжение планировщика заданий;

*IEA000A 00C,INT REQ,02,0200,4000,,RDR — адрес устройства ввода — 00C; INT REQ — требуется вмешательство; код операции в команде канала

*) Здесь и далее в форматах команд необязательные параметры замещают значения соответствующих параметров в операторах DD каталогизированных процедур.

**) В MFT в качестве идентификатора используется номер раздела.

Таблица 6.3. Форматы команд, управляющих системными программами

Команды	Операнды			
{START} {S}	RDR [идентификатор] адрес [номер тома] [имя задания] [ключевые параметры]	INIT [идентификатор] [...параметры]	WTR [идентификатор] адрес [номер тома] [выходные классы] [ключевые параметры]	DSO [идентификатор] адрес [номер тома] [классы] [ключевые параметры]
{CANCEL} {C}	идентификатор	-	{идентификатор} адрес	-
{STOP} {P}	[RDR.] идентификатор	[INIT.] идентификатор	[WTR.] {идентификатор} адрес	[DSO.] {идентификатор} адрес
{MODIFY} {F}	-	[INIT.] идентификатор CLASS=параметр	[WTR.] идентификатор {CLASS=классы PAUSE={FORMS DATASET}}	[DSO.] идентификатор {JOBCLASS=классы заданий OUTCLASS=выходной класс}

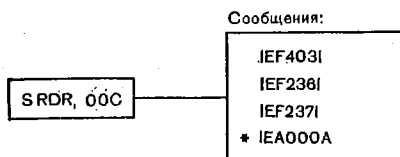
*) Могут быть указаны оба параметра.

*) Могут быть указаны оба параметра.

равен 02; байт состояния в слове состояния канала равен 0200; байт уточненного состояния равен 4000; RDR — имя программы, во время работы которой потребовалось вмешательство.

В ответ на подобного рода сообщения оператор должен выполнить одно из следующих действий: привести устройство в состояние готовности или подать дополнительные карты на устройство 00С, или освободить приемный карман и т. д. Пусть, например, оператор должен ввести одну колоду с перфокартами на устройство считывания. Тогда, после того как он ее поставит,

а) Устройство ввода 00С в исходном состоянии находится в ONLINE



б) Устройство ввода 00С в исходном состоянии находится в OFFLINE

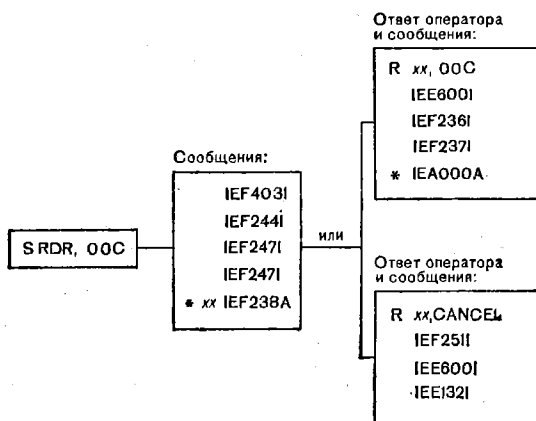


Рис. 6.2. Порядок выдачи сообщений при запуске программы системного ввода RDR.

следует нажать кнопки «Пуск» и «Конец картотеки», и программа RDR начнет считывать задание. После того как RDR закончит работу по вводу задания, он закрывается. Чтобы ввести следующую колоду (новое задание), надо снова вызвать программу RDR. Если клавиша «Конец картотеки» не нажата, то можно вводить следующие задания, так как программа RDR остается в системе. Если задания были установлены и кнопки нажаты до выдачи команды START, то сообщение IEA000A не появляется.

Пример 6.2. Запустить программу системного ввода RDR с идентификатором R для считывания заданий во входном потоке с устройства 00С, которое находится в состоянии OFFLINE, общий рабочий том установлен и находится в ONLINE:

S RDR,R,00C — стартовать программу системного ввода с идентификатором R.

Далее следуют сообщения:

IEF403I RDR STARTED TIME=23.04.10 — программа RDR стартовала в 23.04.10;

IEF244I RDR. R. IEFRDED UNABLE TO ALLOCATE — устройство не может быть выделено RDR;

IEF247I RDR ALLOCATION RECOVERY — сообщение о повторном распределении устройства 00C;

IEF247I RDR 00C OFFLINE — система не может выделить устройство 00C, так как оно находится в состоянии OFFLINE;

*03 IEF238A REPLY DEVICE NAME, OR 'CANCEL' — запрос оператору указать имя устройства, которое требуется для RDR, или уничтожить задание (CANCEL).

Если оператор указывает номер устройства

R 03,00C

то оно выделяется заданию и автоматически переводится в состояние ON-LINE. При этом выдаются следующие сообщения:

IEE600I REPLY TO 03 IS; 00C — ответ на запрос 03 принят системой, адрес устройства — 00C;

IEF236I ALLOC. FOR RDR R — устройство 00C выделено программе RDR с идентификатором R;

IEF237I 00C ALLOCATED TO IEFRDER — устройство 00C выделено в распоряжение планировщика заданий;

*IEA000A 00C,INT REQ,02,0200,4000,,,RDR — устройство готово к вводу задания пользователя во входном потоке.

Если оператор ответит:

R 03,CANCEL — отменить команду START

то система считает, что команда START выдана ошибочно и выдает сообщения:

IEF251I RDR JOB CANCELED — система отменяет запуск RDR в ответ на запрос IEF238A;

IEE600I REPLY TO 03 IS; CANCEL — система приняла ответ на запрос 03, CANCEL — уничтожить задание;

IEE132I START COMMAND DEV ALLOC ERR — команда START снята, как ошибочно выданная.

Номер тома заменяется серийным регистрационным номером магнитной ленты или тома прямого доступа, на котором находится входной поток заданий, например:

S RDR,282,TRM001 — программа RDR будет считывать входной поток с тома с серийным номером TRM001 на устройстве 282.

Если ввод надо начать с определенного задания, то указывается имя этого задания, например:

S RDR,00C,,OLGA

Все задания, которые находились перед заданием OLGA, пропускаются, и ввод начнется с задания OLGA. В примерах 6.1 и 6.2 задания считываются, начиная с первого, установленного на считывающем устройстве ввода.

Ключевые параметры позволяют изменить или добавить параметры, которые установлены в каталогизированной процедуре системного ввода. Для примера рассмотрим изменение параметров в каталогизированной процедуре

системного ввода, приведенной в Приложении 7. Например, изменим уровень выдачи системных сообщений MSGLEVEL со значения (0,1) на (1,1). Для этого достаточно указать в команде START в параметре PARM новое значение (1,1):

```
S RDR.A1,00C,,,PARM='80103005001024905010SYSDA____E00011A'
```

6.1.2. Команда STOP позволяет прекратить работу программы системного ввода RDR. Формат команды STOP приводится в табл. 6.3. Необязательный параметр RDR — это имя процедуры системного ввода. Могут быть использованы также имена RDR400 или RDR3200. Обязательным параметром является идентификатор, который был ранее указан в команде START.

Команда STOP располагает ограниченными возможностями, для того чтобы остановить работу программы RDR. Для того чтобы команда STOP

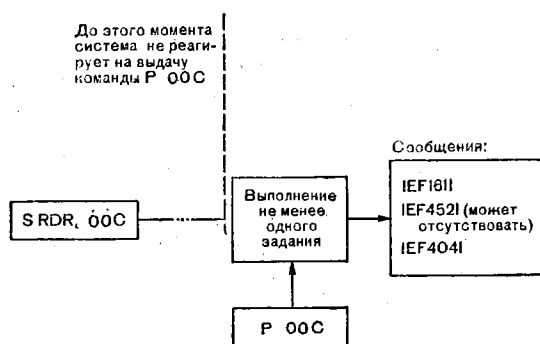


Рис. 6.3. Управляющее воздействие команды STOP.

остановила работу программы RDR, необходимо, чтобы INIT выполнила хотя бы одно задание после выдачи команды STOP. После того как задание будет выполнено, прекращается работа программы RDR, о чем выдаются соответствующие сообщения. При прекращении работы программы RDR память, которую она занимала, освобождается. Команда STOP, выданная до ввода и обработки хотя бы одного задания, никак не влияет на работу программы, как это видно из рис. 6.3.

Пример 6.3. Пусть во входном потоке имеется два задания KPRIM2 и RASTAB. Остановить работу программы системного ввода после начала ввода задания KPRIM2. Пусть программа RDR была запущена по команде

```
S RDR.R,00C —
```

R является идентификатором; 00C — устройство считывания, которое находится в исходном состоянии ONLINE.

Система в этом случае выдает сообщения, как было показано в примере 6.1: IEF403I, IEF236I, IEF237 и *IEA000A. После начала ввода первого задания KPRIM2 получим:

IEF403I KPRIM2 STARTED TIME=15.45.20 — стартовало первое задание KPRIM2 из входного потока. Выдадим команду P R — остановить работу RDR с идентификатором R.

В ответ система выдает сообщения:

IEF161I 00C — READER CLOSED — программа RDR закрыла набор лент на устройстве 00C;

IEF452I RASTAB JOB NOT RUN — JCL ERROR — задание RASTAB снято, так как рассматривается системой, как ошибочно введенное, если команда STOP попала на момент ввода задания RASTAB;

IEF404I RDR ENDED TIME=15.45.58 — программа RDR закончила работу в 15.45.58.

Если команда STOP выдается после того, как задание RASTAB было помещено в очередь JOB, то задание RASTAB стартует, после чего программа RDR закрывается. Из примеров, приведенных выше, видно, что команда STOP не совсем удобна для закрытия программы RDR с консоли оператора. Поэтому наиболее часто используется способ закрытия программы RDR непосредственно с устройства считывания путем ввода пустой перфокарты и нажатия клавиш «Пуск» и «Конец картотеки». В ответ система выдает сообщения IEF161I и IEF404I, которые приводились выше.

6.1.3. Команда CANCEL позволяет отменить команду START, выданную для запуска программы системного ввода RDR. Формат команды CANCEL приводится в табл. 6.3, в котором указывается только идентификатор из команды START. Команда CANCEL позволяет отменить команду START, только если она выдана до момента завершения процесса распределения устройства считывания для программы RDR. Если устройство ввода в исходном состоянии находится в ONLINE, то практически невозможно отменить команду START.

Если устройство ввода находится в исходном состоянии в OFFLINE и команда CANCEL выдана не позднее появления сообщения IEF238A или сразу за ним, то команда START отменяется. До сообщения IEF238A выдаются сообщения IEF403I, IEF244I, IEF247I (см. пример 6.2), далее следует сообщение:

*15 IEF238A REPLY DEVICE NAME, OR 'CANCEL'

Выдадим команду CANCEL, например, с указанием адреса устройства C 00C

Система выдает сообщение:

IEE301I 00C CANCEL COMMAND ACCEPTED — задание с идентификатором 00C отменяется в ответ на ввод команды CANCEL.

Возможен другой вариант отмены команды START, а именно, если оператор подтвердит номер устройства с помощью команды REPLY:

R 15,00C

В ответ система выдает сообщения:

IEF251I RDR JOB CANCEL — система завершила запуск программы RDR;

IEF450I RDR .00C S=222 TIME=00.00.00 — задание RDR с идентификатором 00C завершено аварийно с консоли;

IEE132I START COMMAND DEV ALLOC ERR — во время обработки команды START обнаружена ошибка распределения устройства ввода.

В табл. 6.4 показано управляющее действие команд STOP и CANCEL на работу программы системного ввода RDR. Из таблицы видны способы закрытия программы RDR.

Таблица 6.4. Управляющее действие команд на работу программы системного ввода RDR

Команда	До конца распределения устройства ввода программе RDR	После окончания распределения устройства ввода программе RDR	Работа с заданием
START	RDR не готов	—	—
	То же	RDR готов к работе;	Может быть начата работа с заданиями
CANCEL	Отменяет команду START	Система сообщает, что поздно отменить команду START	
STOP	Не влияет; RDR можно закрыть вводом пустой перфокарты со считывающего устройства		RDR аннулируется после обработки хотя бы одного задания; после выдачи команды STOP

6.2. Управление инициатором (INIT)

Инициатор INIT выбирает задания для выполнения из входной очереди JOB в порядке, установленном в каталогизированной процедуре (Приложение 8). Для каждого пункта задания инициатор выделяет память, устройства ввода-вывода для наборов данных, может потребовать от оператора снятия ненужных и установки недостающих томов и т. д. После выполнения пункта задания инициатор освобождает ненужные устройства ввода-вывода, и если выходные данные не выводятся программой прямого системного вывода DSO, то передает их на устройство прямого доступа и ставит наборы данных в выходную очередь SOUT. Число инициаторов, одновременно работающих в системе, равно числу команд START, выданных оператором (или автоматически системой) для запуска INIT, но не более 15. Если запущено несколько инициаторов, то они должны иметь разные идентификаторы, например A1 и A2. Параметры инициатора INIT изменяются по команде MODIFY. Инициатор прекращает работу по команде STOP (табл. 6.2).

6.2.1. Запуск инициатора INIT выполняется по команде START, формат которой приводится в табл. 6.3. В формате команды указывается обязательный параметр INIT — стандартное имя программы инициатора. Идентификатор интерпретируется, как в команде START для программы RDR (п. 6.1.2).

Пример 6.4. Запустить инициатор INIT с идентификатором A1:
S INIT.A1

На консоли оператор получит, например, сообщение:

IEF403I INIT STARTED TIME = 13.45.56 — инициатор начал свою работу в 13.45.56.

Если заданий нет, то выдается сообщение:

IEF429I INITIATOR INIT 'A' WAITING FOR WORK — инициатор INIT ждет работы; в очереди JOB нет заданий.

Как только задания появятся в очереди JOB или будут вызваны из библиотеки, инициатор INIT начнет их выбирать для выполнения. Сообщения

IEF429I может не появиться, если в системе имеются задания для выполнения.

Параметр позволяет задать порядок обработки классов и назначить приоритет, с которым INIT будет обрабатывать задания. Он имеет следующий формат:

параметр=(*класс* [(*n*)],..., [*LIMIT=k*])

в котором можно указать до восьми классов (от А до О). Ниже приведем пример записи команды START для инициатора, который будет обрабатывать задания классов А, В и С:

S INIT.A1,,,ABC

Возможна другая форма записи:

S INIT.A1,,,(A,B,C)

В обоих случаях первым будет обрабатываться класс, который указан слева, т. е. в данном случае класс А.

Каждому классу заданий можно присвоить принудительный приоритет *n*. Это значит, что INIT будет обрабатывать все задания определенного класса с приоритетом *n*. Значение *n* находится в пределах от 0 до 15. Например, по команде

S INIT.A1,,,(A(5))

инициатор А1 выполняет все задания класса А с приоритетом 5. Предельный приоритет *LIMIT=k* устанавливает наивысший приоритет, с которым будут выполняться задания. Значение *k* находится в пределах от 0 до 15. Например, по команде

S INIT.A2,,,(A,LIMIT=10)

инициатор А2 будет выбирать для выполнения только задания класса А, которые будут выполняться с приоритетом не больше 10.

6.2.2. Команда MODIFY позволяет изменить классы заданий, обрабатываемые инициатором. Первоначально классы заданий, которые будет обрабатывать инициатор, задаются в команде START. В процессе работы эти классы можно изменить по команде MODIFY, формат которой приводится в табл. 6.3.

Идентификатор и *параметр* интерпретируются так же, как в команде START (п. 6.2.1).

Пример 6.5. Пусть была выдана команда

S INIT.A1,,,ABC

— инициатор А1 обслуживает классы А, В и С.

Изменим классы обработки, например:

F INIT.A1,CLASS=D — инициатор А1 будет обрабатывать задания только класса D.

Если во время выдачи команды MODIFY выполняется текущее задание, то новые значения параметров начнут действовать после того, как закончится его выполнение. В ответ на ввод команды MODIFY система не выдает сообщения. Оператор должен помнить классы и приоритеты, с которыми

работают инициаторы, а также новые классы, назначенные им по команде MODIFY.

6.2.3. Команда STOP позволяет остановить работу инициатора. Если команда STOP выдается во время выполнения задания, то обработка задания заканчивается, после чего инициатор закрывается. Чтобы снова запустить инициатор, надо снова выдать команду START, как было показано в п. 6.2.1. Формат команды STOP приведен в табл. 6.3. В формате обязательно указывается только идентификатор.

Пример 6.6. Остановить инициатор с идентификатором A1, если выполняется задание с именем DP:

P A1

Как только закончится выполнение задания DP, система выдаст сообщение:

IEF404I DP ENDED TIME=13.44.44 — выполнение задания DP закончилось в 13.44.44.

Далее последует сообщение:

IEF404I INIT ENDED TIME=13.45.03 — инициатор закончил работу в 13.45.03.

Если команда STOP выдается, когда инициатор свободен, то система его закрывает и выдает только сообщение IEF404I, как показано выше.

6.3. Управление программой системного вывода (WTR)

Наборы данных, предназначенные для передачи на устройства вывода, называются *выходными*. Выходные наборы данных, если они выводятся с помощью программы WTR, помещаются инициатором в выходную очередь SOUT. *Выходной очередью* называется список наборов данных, упорядоченных в соответствии с выходными классами. Первоначально выходные классы задаются в операторах DD параметром SYSOUT. Выходные классы системных сообщений задаются в параметрах MSGCLASS операторов JOB. В очереди SOUT может быть образовано до 36 выходных очередей по числу выходных классов (A—Z, 0—9). Кроме выходных наборов данных, для каждого задания выводятся системные сообщения, полнота (уровень) которых задается в параметрах MSGLEVEL операторов JOB. Желательно, чтобы классы выходных наборов данных и системных сообщений для одного задания были одинаковыми. Тогда общая выходная информация задания направляется на одно устройство вывода. Если указаны неодинаковые выходные классы, то перед выдачей на листинг оператора JOB появляется сообщение, например:

IEF298I FIRSTJOB SYSOUT=C — где FIRSTJOB — имя задания; SY-
STOUT=C — класс, назначенный программистом в операторе DD, отличается от класса системных сообщений, указанного в параметре MSGCLASS оператора JOB.

Программа системного вывода WTR может обслуживать до восьми выходных классов очереди SOUT, которые задаются в каталогизированной процедуре системного вывода WTR (Приложение 9). Команды START, MODIFY, STOP и CANCEL позволяют управлять работой программой WTR (табл. 6.2).

6.3.1. Запуск программы системного вывода WTR осуществляется по команде START. Одна программа WTR может обслуживать несколько выходных классов и один и тот же выходной класс может обслуживаться несколькими программами системного вывода. Формат команды START приводится в табл. 6.3. Параметры, указанные в формате, интерпретируются так же, как в команде START для программы системного ввода RDR (п. 6.1.1), кроме параметра *выходные классы*, который заменяется именами (именем) выходных классов. Всего можно указать до восьми классов (от A до Z и от 0 до 9). Выходные классы позволяют выбрать из выходной очереди SOUT задания определенных классов. Обслуживание производится, начиная с класса, указанного слева в формате команды. Формы записи классов такие же, как были описаны для инициатора INIT в п. 6.2.1, например:

S WTR,,,BA

программа системного вывода WTR будет обслуживать классы A и B, начиная с класса B.

По умолчанию параметра *выходные классы* в команде START информация будет выводиться в классах, указанных в каталогизированной процедуре WTR (Приложение 9). Порядок выдачи сообщений при запуске WTR совпадает со схемой, приведенной на рис. 6.2 для программы системного ввода RDR.

Пример 6.7. Выдать команду START для запуска программы системного вывода WTR с идентификатором D, которая будет работать с устройством 00E, если оно находится в состоянии ONLINE:

S WTR.D,00E

В ответ система выдает сообщения:

IEF403I WTR STARTED TIME=23.21.54 — программа WTR стартовала в 23.21.54;

IEF236I ALLOC.FOR WTR D — программе WTR с идентификатором D выделено устройство, адрес которого приводится в сообщении IEF237I;

IEF237I 00E ALLOCATED TO IEFRDER — устройство 00E выделено в распоряжение планировщика заданий.

Пример 6.8. Запустить программу системного вывода, которая будет выводить данные выходных классов D и F на устройство 280; том имеет серийный регистрационный номер PTOM25; имя набора данных — OUT:

S WTR.A,280,PTOM25,DF,DSN=OUT

Возможна другая форма представления этой же команды:

S WTR.A,,, (D,F),DSN=OUT,VOL=SER=PTOM25,UNIT=280

6.3.2. Команда MODIFY позволяет изменить выходные классы, обрабатываемые программой WTR. Первоначально выходной класс (классы), который будет обрабатывать программа WTR, задается оператором в команде START. В процессе работы можно изменить выходные классы и условия работы с помощью команды MODIFY, формат которой приводится в табл. 6.3. Параметр CLASS позволяет изменить классы, причем самый крайний левый класс выбирается первым для выполнения, например:

F WTR.A,CLASS=CD

Первым обрабатывается выходной класс C. В ответ на ввод команды MODIFY система не выдает никаких сообщений.

Параметр PAUSE устанавливает режим работы программы системного вывода WTR: значение FORMS предписывает программе WTR сделать паузу, для того чтобы, например, изменить формат бумаги на АЦПУ, а значение DATASET — сделать паузу перед обработкой выходного набора данных. В ответ на ввод команды

F 00F,PAUSE=DATASET

система выдает сообщение, например:

*12 IEF382A 00F WTR WAIT DUE TO PAUSE — программа системного вывода ждет указания оператора о продолжении работы.

После выполнения необходимых действий оператор должен выдать команду, например:

R 12,A

где A есть одиночный знак, произвольно выбранный оператором.

После получения ответа

IEE600I REPLY TO 12 IS; A

работа программы системного вывода будет продолжена.

В ответ на ввод команды

F 00F,PAUSE=FORM

система выдает запрос IEF383A. Действия оператора аналогичны предыдущему случаю.

6.3.3. Команда STOP позволяет остановить работу программы системного вывода WTR. Формат команды STOP приведен в табл. 6.3. В нем обязательно указывается идентификатор или физический адрес. Если команда STOP выдана во время вывода набора данных, то она останавливает работу программы WTR после того, как будет закончен вывод набора данных.

Пример 6.9. Остановить программу системного вывода WTR (см. пример 6.7) с помощью команды STOP:

P D

В ответ система выдает сообщения:

IEF301I 00E WTR CLOSED — программа WTR остановлена;

IEF404I WTR ENDED TIME=09.20.25 — программа WTR закончила работу в 09.20.25.

6.3.4. Команда CANCEL может отменить команду START. Формат команды CANCEL приведен в табл. 6.3, в котором указывается идентификатор или физический адрес. Если команда выдана до того, как закончился процесс распределения устройства вывода для программы WTR, система отменяет команду START, например, пусть программа WTR запущена, как в примере 6.7 и сразу же за командой

S WTR.D,00E

выдана команда

C D

Сообщения, выдаваемые в ответ на эту команду, зависят от состояния устройства 00E. Пусть устройство 00E находится в состоянии ONLINE, тогда выдаются сообщения:

IEF251I WTR JOB CANCELED — система завершила задание в соответствии с командой CANCEL;

IEE301I 00F CANCEL COMMAND ACCEPTED — задание отменено в ответ на команду CANCEL, которую выдал оператор;

IEE132I START COMMON DEV ALLOC ERR — обнаружено ошибочное распределение устройства во время выдачи команды START.

Если команда CANCEL выдается после того, как закончилось распределение устройства из любого исходного состояния для программы WTR, то она выдает сообщение:

IEE316I 00E JOB NOT FOUND — команда выдана поздно.

Если в момент выдачи команды CANCEL обрабатывался выходной набор данных, то вывод его прекращается и программа WTR переходит к обработке следующего набора данных из системной очереди SOUT, т.е. команда CANCEL:

C 00E

прекращает обработку только одного набора данных на устройстве 00C.

Если устройство вывода находится в состоянии OFFLINE, то действие команды CANCEL аналогично рассмотренному в п. 6.1.3.

Система в случае появления ошибок, связанных с устройством вывода, или необходимости вмешательства в его работу выдает сообщения типа

*IEA000A 00F,INT REQ,01,0200,4000,,WTR

(аналогичное сообщение было рассмотрено в п. 6.1.1).

6.4. Управление программой прямого системного вывода (DSO)

Выходные наборы данных могут быть выданы не через устройство прямого доступа SYSOUT, а непосредственно из памяти на устройство вывода с помощью программы прямого системного вывода DSO (рис. 6.1). Программа DSO может обслуживать до восьми входных классов и один выходной класс. Классы устанавливаются в каталогизированной процедуре прямого системного вывода (Приложение 10). Программа DSO запускается по команде START и помещается системой в ту же область памяти, в которой находится проблемная программа. Программа DSO может быть остановлена командой STOP только после того, как выполнение задания будет завершено. Программа DSO используется в случае, когда выводимая информация имеет большой объем и важно видеть результаты вычисления во время выполнения задания. Классы заданий, которые обрабатывает программа DSO, могут быть изменены командой MODIFY. Программе DSO выделяется устройство вывода, которое принадлежит заданию до конца его выполнения.

6.4.1. Запуск программы DSO производится по команде START, формат которой приведен в табл. 6.3.

Параметры, указанные в формате команды совпадают с параметрами, приведенными в команде START для программы системного ввода RDR

(п. 6.1.2), кроме параметра *классы*. Параметр *классы* заменяется ключевыми параметрами `JOBCLASS=класс заданий` и `OUTCLASS=выходной класс`. Значения входных классов заданий находятся в пределах от А до О, и количество их в команде не должно превышать 8. Значения выходных классов находятся в пределах от А до Z или от 0 до 9. В DSO указывается только один выходной класс. Если параметр *классы* опущен, то по умолчанию будут использоваться классы, установленные в каталогизированной процедуре DSO. Ниже приведены примеры записи команды START:

```
S DSO,,(JOB CLASS=ABC)
S DSO,,(OUTCLASS=D,UNIT=282)
S DSO,,(JOBCLASS=ABC,OUTCLASS=D)
```

Реакция системы на ввод команды START аналогична той, которая описана выше для команд RDR и WTR. Дополнительно выдается сообщение IEF392I, как показано ниже в примере 6.10.

6.4.2. Команда MODIFY позволяет изменить классы в программе DSO. Формат команды MODIFY приводится в табл. 6.3.

В параметре JOBCLASS может быть указано до восьми входных классов от А до О, а в параметре OUTCLASS указывается только один выходной класс из значений от А до Z или от 0 до 9, например:

```
F DSO.A,JOBCLASS=ABC
F DSO.A,OUTCLASS=F
```

В ответ на ввод команды MODIFY система выдает сообщение IEF392I.

Пример 6.10. Запустить программу DSO с идентификатором DSO, которая будет работать с устройством 00F. Изменить с помощью команды MODIFY сначала класс задания на G, а затем отдельной командой MODIFY выходной класса на класс G.

```
S DSO.DSO,00F
```

Запущена программа DSO с идентификатором DSO, которая будет работать с устройством 00F.

Система выдает сообщения:

```
IEF403I DSO STARTED TIME=00.07.29 — программа DSO стартовала в 00.07.29;
```

```
IEF236I ALLOC. FOR DSO DSO — программе DSO с идентификатором DSO было выделено устройство, адрес которого сообщается в IEF237I;
```

```
IEF237I 00F ALLOCATED TO IEFRDER — устройство 00F выделено в распоряжение планировщика заданий.
```

Изменить входной класс заданий в программе DSO:

```
F DSO,JOBCLASS=G
```

В ответ система сообщает входной (входные) класс и выходной класс, который был до выдачи команды MODIFY:

```
IEF392I 00F DSO JOBCLASS=A OUTCLASS=A — входной класс задания — А и выходной класс при запуске DSO — А.
```

Далее:

```
IEF392I 00F DSO,JOBCLASS=G OUTCLASS=A — входной класс изменился на класс G.
```


Изменим выходной класс на класс G:

F DSO,OUTCLASS=G

В ответ система выдает сообщение:

IEF392I 00F DSO JOBCLASS=G OUTCLASS=G

Выходной класс изменен на класс G.

6.4.3. Команда STOP прекращает работу программы прямого системного вывода после того, как будет завершена обработка текущего задания. Формат команды STOP приведен в табл. 6.3.

Пример 6.11. Запустить программу прямого системного вывода, а затем остановить ее. Программе DSO назначить устройство 00E, которое находится в состоянии ONLINE:

S DSO,00E

Система в ответ выдает сообщения IEF403I, IEF236I, IEF237I (см. пример 6.10).

Остановим работу программы DSO по команде STOP:

P 00E

В команде указан идентификатор 00E.

В ответ система выдает сообщения:

IEF391I 00F DSO CLOSED — программа DSO остановлена;

IEF404I DSO ENDED TIME=00.10.45 — работа программы DSO закончена.

Загрузка ОС состоит из следующих основных этапов (рис. 7.1):

- 1) подготовительный — включение питания ЭВМ и загрузка программы IPL;
- 2) загрузка ядра управляющей программы;
- 3) подготовка планировщика заданий к работе.

Процедура включения питания, как правило, оператором не производится, хотя и относится к его функциям. Логический порядок включения питания

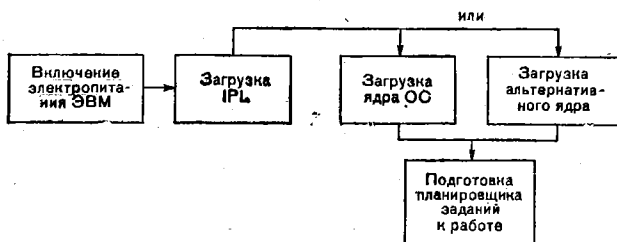


Рис. 7.1. Упрощенная схема подготовки ОС к работе.

является общим для разных моделей ЕС ЭВМ, хотя могут быть незначительные отличия. Второй и третий этапы загрузки зависят от типа управляющей программы (MFT, MVT, SVS), издания управляющей программы, (MVT/2/, M2/3), а также от того, на какой ЭВМ она реализуется. При загрузке ОС применяются специально предназначенные для этого команды. Так, подготовка планировщика заданий начинается с ввода команды SET. Команда DEFINE используется только при работе с управляющей программой MFT. В этой же главе рассматривается команда HALT, которая выдается перед выключением ОС. При работе с ОС может быть использована команда DUMP, которая позволяет получить содержимое памяти, и команда MODE, позволяющая получить информацию о режиме работы технических средств контроля.

7.1. Подготовка ЭВМ к работе

7.1.1. Включение электропитания. Питание ЭВМ включается при смене ОС, после запланированного технического осмотра ЭВМ с целью устранения неполадок, требующих перезагрузки управляющей программы в память и т. д. Включение электропитания производится оператором с пульта процессора путем нажатия кнопки «Вкл. пит.» (рис. 7.2). Электропитание в системе включается в следующем порядке: процессор, селекторные и мультиплексный каналы, ОП, т. е. производится сброс системы в исходное состояние. Все ин-

дикаторы ошибок устанавливаются в нуль, а процессор устанавливается в состояние «ожидания». В результате система устанавливается в такое состояние, из которого ее работа может быть начата без появления сигнала об

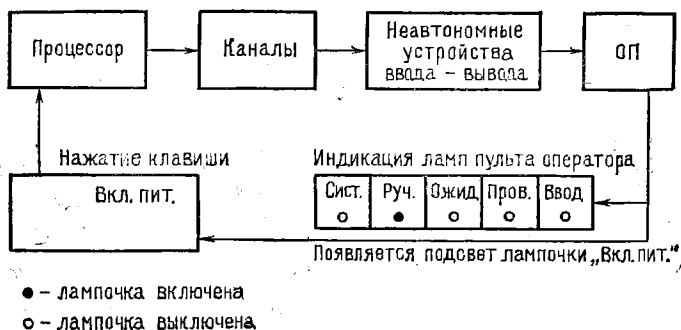
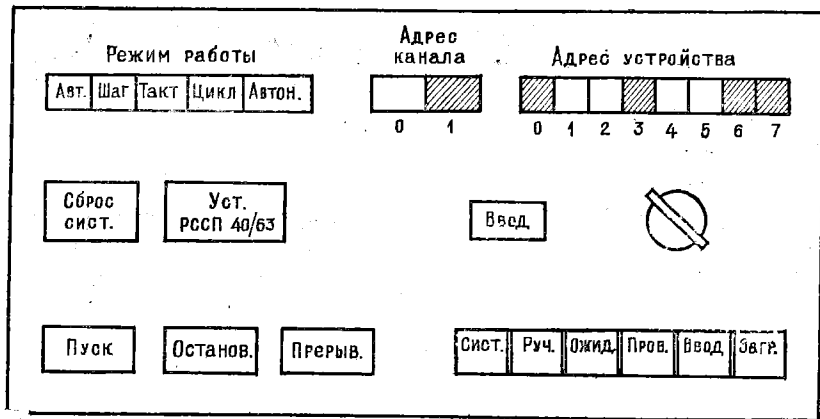


Рис. 7.2.

ошибке, за исключением машинных ошибок. После того как вся последовательность действий по включению питания ЭВМ завершена, появляется внутренняя подсветка клавиши «Вкл. пит.», лампочка «Руч.» на пульте оператора загорается. На этом этап включения электропитания заканчивается.

7.1.2. Процесс загрузки программы IPL (INITIAL PROGRAM LOADINGS) является процедурой загрузки начальной части ОС, после чего она продолжает работу под своим собственным управлением. В этом процессе можно



■ — клавиша включена (состояние 1)

Рис. 7.3.

выделить следующие основные моменты: установление адреса резидентного устройства (SYSPES) и номера канала; загрузка программы IPL.

Адрес резидентного устройства, на котором размещается пакет магнитных дисков с системным резидентным томом системы, на ВЦ известен. Резидентный том содержит ядро ОС (SYS1.NUCLEUS) и главный индекс си-

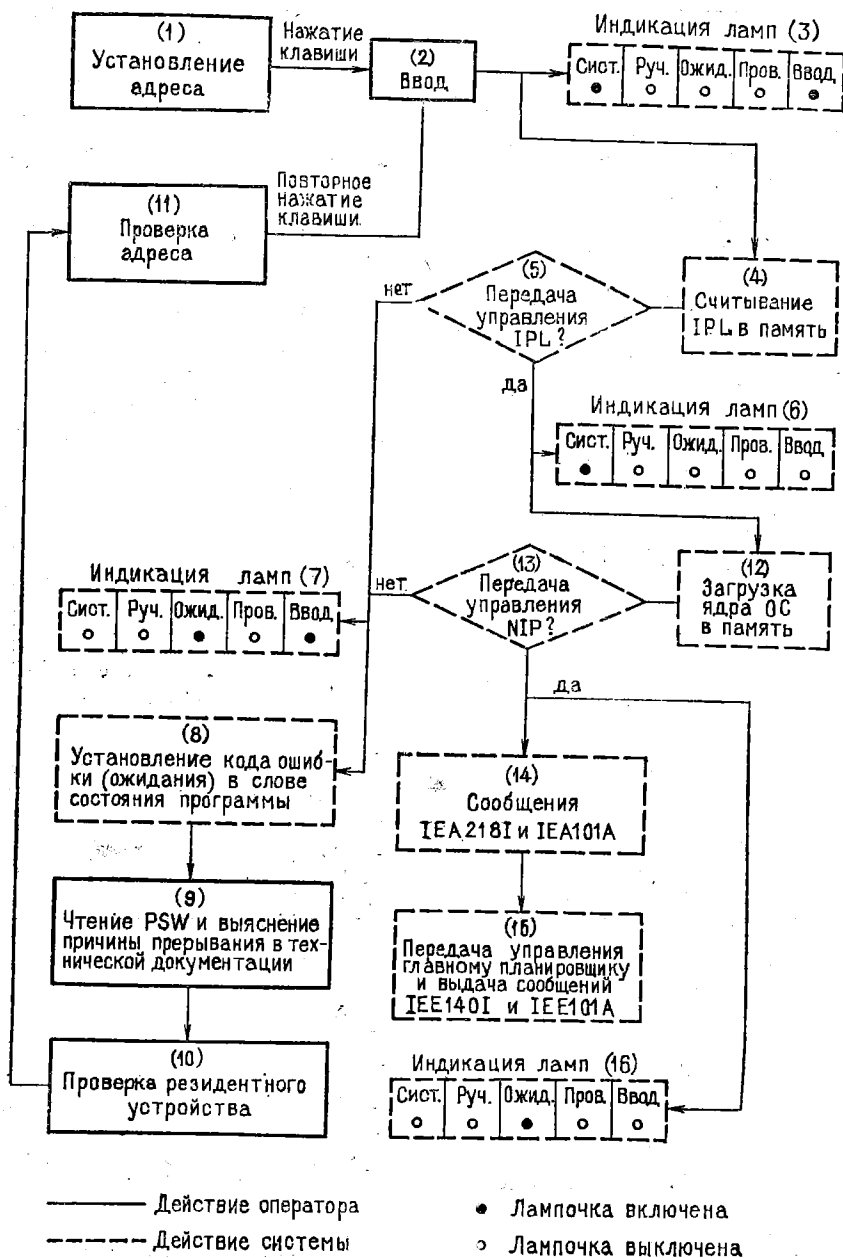


Рис. 7,4.

стемного каталога, по которому управляющая программа находит нужный для работы с заданиями том. *Индекс* — это число или знак, показывающий местонахождение нужного элемента. *Ядром* называется часть управляющей программы, которая постоянно находится в оперативной памяти. Оператор на пульте оператора с помощью клавиши «Адрес канала» и восьми клавиш «Адрес устройства» устанавливает физический адрес системного резидентного устройства (рис. 7.3). Адрес состоит из трех цифр, например, 193, где 1 — адрес канала, 9 — адрес устройства управления (на рис. 7.3 занимает с 0 по 3-й разряды), 3 — адрес самого устройства (с 4-го по 7-й разряды).

Загрузка программы IPL (рис. 7.4) начинается после нажатия клавиши «Ввод» (2). Индикация лампочек пульта оператора (3) изменяется, а именно, включаются лампочки «Ввод» и «Сист.», а лампочка «Руч.» выключается (см. рис. 7.2). Включение лампочки «Сист.» говорит о том, что началось считывание программы IPL с резидентного устройства (4). После окончания процесса считывания управление передается программе IPL (5). Если операция загрузки и передачи управления программе IPL закончилась нормально, то выключается лампочка «Ввод» (6).

Если любая из этих операций не выполнена, то процессор переходит в состояние «Ожидание», о чем свидетельствуют включенные лампочки «Ожид.» и «Ввод» (7). Код ошибки помещается в 52/63 разряды слова состояния программы. Для того чтобы прочесть код ошибки (или код ожидания), необходимо на пульте процессора нажать клавишу «Останов» и нажать клавишу «Уст. Р.С.С.П 40/63» (8). На панели индикации в самом нижнем ряду можно прочесть слово состояния программы (9). Каждая лампочка дает информацию о состоянии соответствующего разряда в двоичном коде. Если код ожидания, т.е. разряды с 52 по 63 равны нулю (лампочки не горят), это значит, что код ожидания равен нулю. В этом случае следует проверить резидентное устройство (10) и убедиться в его работоспособности, далее проверить адрес резидентного устройства (11) на пульте оператора и, если все правильно, то повторно нажать клавишу «Ввод» (2), как показано на рис. 7.4.

7.2. Загрузка ядра управляющей программы (MVT)

Загрузка ядра (12) начинается в том случае, если загрузка программы IPL закончилась нормально. После успешного окончания загрузки ядра управление передается программе инициализации ядра NIP (13), которая позволяет оператору вносить изменения в параметры ОС, установленные при ее генерации. Этот этап заканчивается выдачей сообщений супервизора на главную консоль IEA2181 и IEA101A (14). На рис. 7.5 приведен протокол сообщений системы и команд оператора при запуске управляющей программы MVT версии 6.1 (2). Первые два сообщения:

IEA2181 MOD=1022 ASSUMED — используется программа NIP для модели EC — 1022;

IEA101A SPECIFY SYSTEM PARAMETERS FOR REALESE 6.1/2/MVT — предусматривается изменение системы параметров ОС, версия 6.1/2/, режим MVT.

На основании решения системного программиста оператор либо вводит новые системные параметры с помощью команды REPLY, либо оставляет параметры, которые были заданы при генерации. В последнем случае

Номер
строки *)

Текст сообщений и команд

```

1      IEA218A MOD=1022 ASSUMED
2      IEA101A SPECIFY SYSTEM PARAMETERS FOR RELEASE 6.1/2/MVT
3      IEE140I SYSTEM CONSOLES
4      CONSOLE/ALT COND AUTH ID AREA ROUTCD
5      SYSLOG      H      CMDS      ALL
6      060/0C0      N      ALL      01  Z,A      1-11,16
7      0C0/01F      N      ALL      02  Z,A      1-11,16
8      01F/060      M      ALL      03      1-11,16
9      IEE101A READY
10     IEE103I S      WTR,00F      .
11     IEE103I S      INIT,A1      .
12     IEE103I S      INIT,A2      .
13     IEE103I MN      JOBNAME,T      .
14     T      DATE=80.294,CLOCK=13.25.00,Q=(,F)
15     IEE103I MN      SPACE      .
16     IEE103I V      00C,ONLINE      .
17     *00 IEF423A SPECIFY JOB QUEUE PARAMETERS
18     R      00,'U'
19     IEE600I REPLY TO 00 IS; U
20     IEF249I FOLLOWING P/R AND RSV VOLUMES ARE MOUNTED
21     SD001      ON      132 (P/R-PUB)
22     UD002      ON      230 (RSV-PRV)
23     UD001      ON      231 (RSV-PRV)
24     LD001      ON      232 (RSV/PRV)
25     IEE302I 00C ONLINE
26     IEE041I LOG NOW RECORDING ON SYS1.SYSLOGY ON 132
27     IEE360I SMF NOW RECORDING ON SYS1.MANY ON 132.TIME=13.26.10
28     IEF403I INIT STARTED TIME=13.26.29
29     IEF403I WTR STARTED TIME=13.26.33
30     IEF236I ALLOC.FOR WTR 00F
31     IEF237I 00F ALLOCATED TO IEFRDER
32     IEF403I INIT STARTED TIME=13.26.36
33     IEF429I INITIATOR'A1'WAITING FOR WORK
34     IEF429I INITIATOR'A2'WAITING FOR WORK
35     S      RDR,00C
36     IEF403I RDR STARTED TIME=13.27.25
37     IFF236I ALLOC. FOR RDR 00C
38     IEF237I 00C ALLOCATED TO IEFRDER
39     *IEA000C 00C,INT REQ,02,0200,4000,,,RDR

```

*) Номер строки в протоколе отсутствует.

Рис. 7.5. Один из вариантов загрузки управляющей программы MVT.

управление передается главному планировщику. Чаще эта передача осуществляется путем нажатия клавиши «КТ» с пультной машинки, или с помощью команды R 00,'U'.

Далее управление передается главному планировщику (15, рис. 7.4), который начинает работу с выдачи сообщений IEE1401. На рис. 7.5 это сообщение о состоянии системных консолей (SYSTEM CONSOLES) занимает строки 3—8. Назначение колонок: CONSOLE — адрес консоли, ALT — адрес запасной (альтернативной) консоли, COND — состояние консоли, AUTH — информация о том, какие команды и ответы можно вводить с данной консоли, ID — порядковый номер. AREA — области сообщений дисплея,

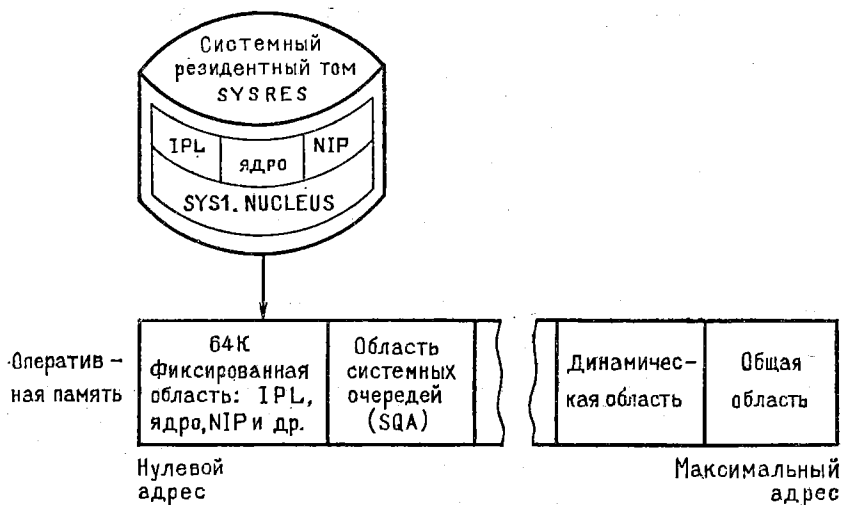


Рис. 7.6.

ROUTD — маршрутные коды, приписанные системной данной консоли. В строке 5 указывается, что все системные команды и команды оператора (CMDS) будут записываться в системный журнал (SYSLOG); в системе организован протокол системы (H), в котором непрерывно фиксируются записи о деятельности системы. Главной консолью (M) является консоль 01F/060; неактивными консолями (N) — 060/0C0 и 0C0/01F. На всех консолях разрешается пользоваться всеми командами, если указано ALL. В колонке ID указываются порядковые номера консолей. В колонке ROUTD сообщаются разрешенные маршрутные коды 1—11, 16. *Маршрутный код* — это путь для передачи сообщений. В колонке AREA указываются: А — самая нижняя граница области информации состояния на дисплее; Z — всегда назначается для основной области сообщений (см. гл. 11). Последним на этом этапе появляется сообщение главного планировщика IEE101A (строка 9) о том, что система готова к выполнению дальнейших операций. На пульте оператора после появления этого сообщения оказывается включенной только одна лампочка «Ожид.», так как процессор переходит в состояние ожидания ввода текущего задания для выполнения (16, рис. 7.4). Если во время загрузки и передачи управления NIP произошла ошибка, то действия оператора такие же, как были описаны при загрузке IPL (рис. 7.4) (8) — (11), (2)).

Этот этап заканчивается тем, что ОП делится на части: фиксированную, SQA — область системных очередей, динамическую и др. (рис. 7.6). В фиксированной области располагаются IPL, ядро, NIP и некоторые другие управляющие программы и таблицы. Фиксированная область занимает примерно 64К и размещается в ОП, начиная с младших адресов. В области SQA размещаются очереди заданий: входная JOB, выходная SOUT и очередь задержанных заданий HOLD. Динамическая область памяти предназначена для размещения в ней в дальнейшем системных задач и заданий программиста-пользователя (П/П). В общей области хранятся некоторые наиболее часто используемые программы из системной библиотеки, которыми могут пользоваться разные задания.

7.3. Команда SET

Подготовка планировщика заданий к работе требует знания команды SET. Команда SET является первой командой, которая выдается оператором в начале работы с ОС (рис. 7.7). Она сообщает системе дату, текущее время

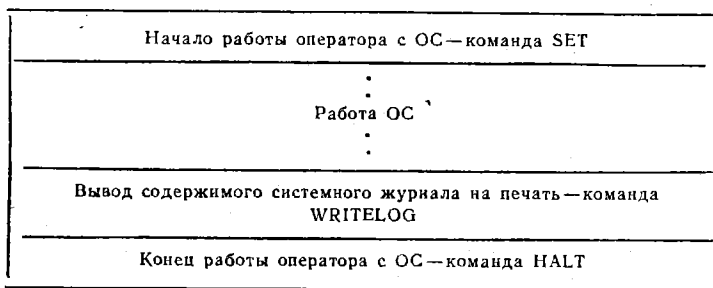


Рис. 7.7. Начало и конец работы оператора с ОС.

суток, адреса двух системных наборов данных SYS1.SYSJOBQE и SYS1.PROCLIB и имеет возможность отменить команды START для запуска системных программ (гл. 6), которые включены в ОС при ее генерации и автоматически выдаются при ее запуске. Формат команды SET:

Команда	Параметры
{SET T}	DATE=yy.ddd[,CLOCK=hh.mm.ss] [,Q=([адрес][.F])][,PROC=адрес][,AUTO=aaa]

Параметр DATE является обязательным в команде SET, которая выдается оператором при начальной загрузке ОС. Он сообщает системе дату: yy заменяется двумя последними цифрами года, значение которых находится в пределах от 00 до 99; ddd — порядковым днем года, значение которого лежит в пределах от 001 до 366, например DATE=81.005, где 81 — это 1981 года, 005 — день года (5 января). Сообщение о дате появляется в системных сообщениях и сообщениях, выдаваемых в листинге программистам. При повторном использовании команды SET указывать параметр DATE не обязательно.

Необязательный, но желательный параметр CLOCK сообщает системе текущее время суток: *hh* заменяется часом, значение которого находится в пределах от 00 до 23; *mm* и *ss* — минутой и секундой, значения которых находятся в пределах от 00 до 59 для обоих параметров, например:

T DATE=81.005,CLOCK=11.29.53

Время оператор определяет по имеющимся часам. Если в команде SET указано время, то оно является для системы временем отсчета работы ОС. Время широко используется во многих сообщениях системы. Если время не указано, то это эквивалентно заданию CLOCK=00.00.00 и отсчет времени работы ОС производится от нулевого значения.

В параметрах Q и PROC указываются физические адреса устройств, на которых находятся наборы данных SYS1.SYSJOBQE и SYS1.PROCLIB соответственно. Эти адреса необходимы для обращения к ним во время работы ОС. В наборе данных SYS1.SYSJOBQE организуются системные очереди заданий. В наборе данных SYS1.PROCLIB находится библиотека каталогизированных процедур. Обычно эти оба набора данных находятся на резидентном томе, и нет необходимости указывать их адреса. Однако если эти наборы находятся не на резидентном томе, то команда SET позволяет указать адрес устройства, на котором они будут установлены. Если в параметре Q указывается F, то система должна подготовить набор данных SYS1.SYSJOBQE к созданию очередей заново, т.е. очистить очереди от старой информации, например: Q = (133,F), где 133 — адрес устройства. Если используется резидентное устройство, то адрес можно опустить: Q=(F). При повторной загрузке ОС, например в случае аварийной остановки системы, параметры PROC и Q в формате команды SET можно опустить (по умолчанию параметра Q созданные ранее очереди заданий сохраняются). Это позволяет после перезагрузки системы продолжить работать с заданиями, поступившими в систему до сбоя, т.е. не надо повторно вводить их через считывающее устройство, что экономит время.

Параметр AUTO связан с сохранением или отменой оператором трех автоматических команд START, запускающих системные программы (RDR, INIT, WTR или DSO). Набор автоматических команд START задается при генерации конкретной ОС. О том, какие автоматические команды START в данной ОС выдаются, оператор узнает из трех сообщений, которые появляются в протоколе на консоли. Типичные примеры таких сообщений приведены в протоколе на рис. 7.5 в строках 10—12. В данном примере выдаются три автоматические команды START для запуска системных программ WTR и двух инициаторов INIT.A1 и INIT.A2. Если оператор хочет оставить эти команды, то он может либо опустить параметр AUTO, либо *aaa* заменить комбинацией символов YYY, где каждая буква сохраняет одну из трех автоматических команд START. С помощью параметра AUTO можно отменить любую из трех команд. Для отмены одной автоматической команды START буква *a*, соответствующая порядковому номеру выдачи этой команды в протоколе, заменяется символом N. Например, AUTO=NYU означает отмену первой автоматической команды START. Если надо отменить все автоматические команды, то следует указать AUTO=NNN или AUTO=NONE. Параметр AUTO используется один раз, только в первой команде SET после загрузки IPL.

В качестве примера применения команды SET рассмотрим протокол, приведенный на рис. 7.5, строка 14. Команда SET выдана в предположении: автоматические команды START оставить; почистить очередь; считать, что наборы данных SYS1.SYSJOBQE и SYS1.PROCLIB находятся на резидентном томе:

T DATE=80.294,CLOCK=13.25.00,Q=(,F)

В ответ на ввод команды SET система всегда выдает сообщение, приведенное на строке 17, рис. 7.5.

*00 IEF423A SPECIFY JOB QUEUE PARAMETERS — запрос информации об изменении системных параметров в наборе данных SYS1.SYSJOBQE.

Если системные параметры не будут изменяться, оператор должен ответить операционной системе (рис. 7.5, строка 18):

R 00,'U'

или нажать клавишу «КТ» на пульте пишущей машинки. В ответ система выдает сообщения, например такие, как приведены на рис. 7.5, строки 19—24:

IEE600I REPLY TO 00 IS; U — ответ оператора на запрос 00 принят системой; системные параметры не меняются.

Если системные параметры набора данных должны быть изменены, то выдается ответ в следующем виде:

R 00,'n,t,k,p'

где n — число записей на одной логической дорожке ($10 \leq n \leq 225$); t — число записей, выделяемых каждому инициатору ($t \leq 9999$); k — число записей, которое отводится для сохранения информации о заданиях, для инициирования которых требуется более t записей ($k \leq 9999$); p — число записей для сохранения сообщений оператору. Ниже приводится пример ответа:

R 00,'450,,20'

В ответ система выдает сообщение:

IEE600I ACCEPTED REPLY TO MSG 00 IS; '450,,20'

Далее следует сообщение IEF249I (строки 20—24, рис. 7.5) об установленных томах:

IEF249I FOLLOWING P/R AND PSV VOLUMES ARE MOUNTED — смонтированы тома P/R (P — личный, R — резервный) и RSV — резервный; SD001 ON 132 (P/R — PUB) — устройство 132 находится в оперативном состоянии; SD001 — том (личный резервный — общий (PUB));

UD002 ON 230 (RSV—PRV) — устройство 230 находится в оперативном состоянии; UD002 — том (резервный — личный);

UD001 ON 231 (RSV—PRV) — устройство 231 находится в оперативном состоянии; UD001 — том (резервный — личный);

LD001 ON 232 (RSV—PRV) — устройство 232 находится в оперативном состоянии; LD001 — том (резервный — личный).

7.4. Подготовка планировщика заданий MVT

Подготовка планировщика заданий к работе начинается после получения сообщения IEEI01A (рис. 7.5, строка 9 и рис. 7.8, (1)). Планировщик считывает и интерпретирует задания, устанавливает порядок их обработки, ини-

цирует и завершает задания, управляет выдачей выходных данных. Для выполнения своих функций планировщику заданий необходимы системные программы RDR, INIT, WTR и DSO (гл. 6), которыми он будет пользоваться при выполнении заданий. Самым простым случаем подготовки планировщика заданий является случай (рис. 7.8), когда в систему включаются автоматические команды START (2), о чем оператор получает сообщения на консоли

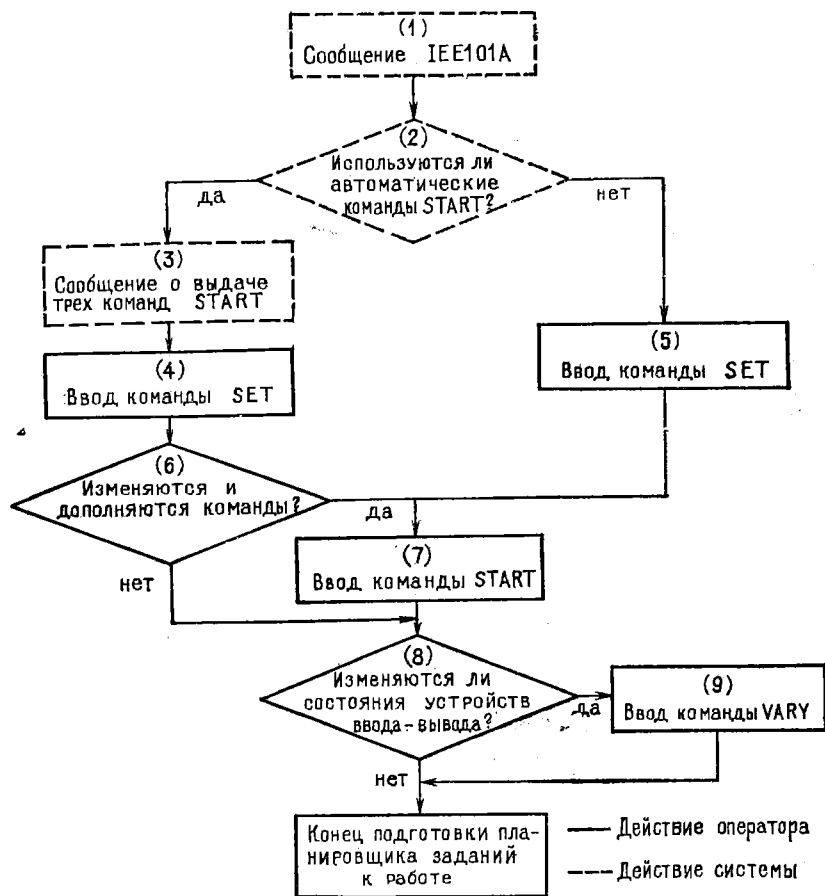


Рис. 7.8.

(3). Если системные программы не изменяются (6) и не добавляются (7), устройства, указанные в команде START, находятся в оперативном состоянии и их состояние не надо изменять (8), то оператору нужно выдать только одну команду SET (4). Наиболее сложный случай подготовки планировщика заданий, если автоматические команды не выдаются системой (2), оператор выдает команду SET (5), команды START (7) и изменяет состояния устройств, указанных в команде START (переводит их в оперативное состояние (9)). Возможны разные промежуточные варианты между этими двумя описанными случаями.

Пример подготовки планировщика заданий для управляющей программы MVT приведен на рис. 7.5. Генерацией системы предусмотрена выдача трех автоматических команд START, о чем становится известным оператору из сообщений (строки 10—12):

IEE103I S WTR,00F * — стартует системная программа вывода и ей назначается устройство 00F (АЦПУ);

IEE103I S INIT.A1 * — стартует системная программа инициатора INIT, которая имеет имя A1;

IEE103I S INIT.A2 * — стартует системная программа инициатора INIT с именем A2. Знак * справа говорит о том, что эти команды выдаются автоматически системой.

Пусть оператор решает сохранить автоматические команды. Тогда следует опустить оператор AUTO в команде SET (строка 14, рис. 7.5):

T DATE=80.294,CLOCK=13.25.00,Q=(,F)

После этой команды система выдает набор сообщений, который находится в прямой зависимости от набора автоматических команд START. В данном случае (строки 28—34):

IEF403I INIT STARTED TIME=13.26.29 — программа INIT стартовала в 13.26.29;

IEF403I WTR STARTED TIME=13.26.33 — программа WTR стартовала в 13.26.33;

IEF236I ALLOC.FOR WTR 00F — для системной программы WTR выделено устройство 00F;

IEF237I 00F ALLOCATED TO IEFRDER — устройство 00F выделено набору данных, описанному в DD с именем IEFRDER каталогизированной процедуры WTR;

IEF403I INIT STARTED TIME=13.26.36 — программа INIT стартовала в 13.26.36;

IEF429I INITIATOR'A1'WAITING FOR WORK — инициатор 'A1' ждет работы;

IEF429I INITIATOR'A2'WAITING FOR WORK — инициатор 'A2' ждет работы.

Для запуска программы системного ввода RDR, которая отсутствует в системе, оператор выдает команду START (рис. 7.5, строка 35):

S RDR, 00C

— стартует программа системного ввода RDR, которая будет работать с устройством ввода с перфокарт 00C.

В ответ система выдает сообщение (рис. 7.5, строки 36—38):

IEF403I RDR STARTED TIME=13.27.25 — программа RDR стартовала в 13.27.25.

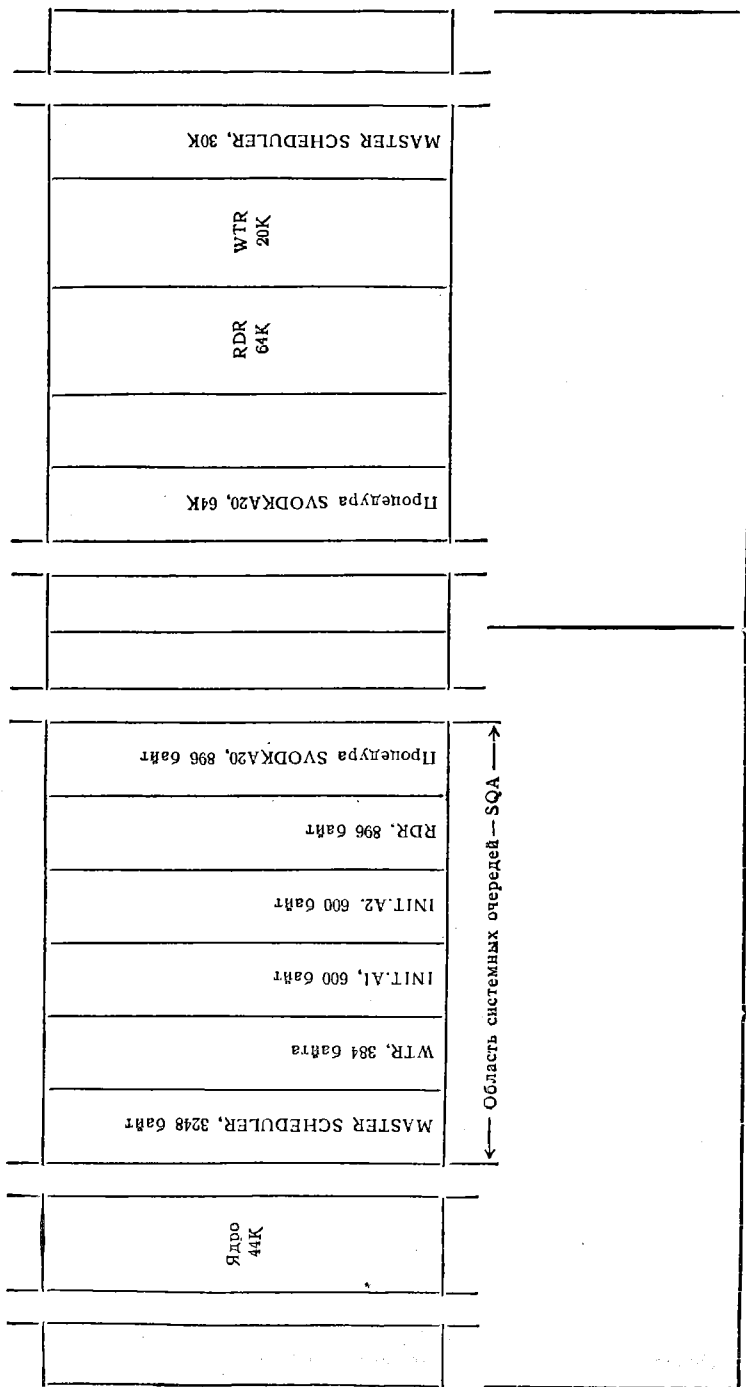
Ранее в строке 25 система сообщила, что устройство 00C находится в состоянии ONLINE:

IEE302I 00C ONLINE — в ответ на автоматическую команду VARY (рис. 7.5, строка 16).

В строках 37—39 выдаются сообщения IEF236I, IEF237I и *IEA000A на ввод команды START.

На этом подготовка планировщика заданий заканчивается. В результате в динамической области ОП появляются разделы, в которых будут находиться программы RDR и WTR, готовые к обработке заданий. На рис. 7.9

314K 398K 462K 482K 512K



Фиксированная область ОП
Динамическая область ОП
Рис. 7.9. Пример размещения системных программ RDR, WTR и процедуры SVODKA20 в ОП (MVT).

показано, как размещаются системные программы RDR, WTR, каталогизированная процедура SVODKA20 в ОП. Из рисунка видно, что программа системного ввода RDR занимает 64К, программа системного вывода WTR — 20К. Размер памяти для программы пользователя выделяется по мере необходимости, например для процедуры SVODKA20 выделено 64К. Разделы существуют до тех пор, пока в системе имеются системные программы и задания пользователя. В противном случае динамическая область ОП считается свободной. Инициаторы работают в разделах, минимальный размер которых равен 52К. Каждый INIT выделяется определенному заданию до его завершения. То же относится и к программе прямого системного вывода. Размер раздела для задания задается в параметре REGION оператора JOB или по умолчанию параметра REGION используется значение, установленное при генерации системы. Если система имеет объем памяти 256 К, то часто временно освобождают разделы, занимаемые RDR и WTR, чтобы выделить больше памяти для задания. Если для выполнения задания памяти не хватает, то процессор переходит в состояние ожидания.

Пример 7.1. При подготовке планировщика заданий к работе отменить первую автоматическую команду START для WTR (строка 10, рис. 7.5) в связи с неисправностью устройства 00F и заменить его на 00E. Для этого оператор должен выдать команду

T DATE=80.294,CLOCK=13.25.00,Q=(,F),AUTO=NYU

которая отменяет команду WTR. В ответ система повторяет команды, которыми разрешено пользоваться:

IEE103I S INIT.A1 *
S INIT.A1 *
IEE103I S INIT.A2 *
S INIT.A2 *

Далее оператор должен ввести новую команду START для WTR с новыми параметрами:

S WTR,00E — программа системного вывода WTR будет работать с устройством 00E (АЦПУ).

Система на эту команду отвечает сообщениями (см. п. 6.3 и рис. 6.2):

IEF403I WTR STARTED TIME=13.27.23 — программа стартовала в 13.27.23. Далее следуют сообщения:

IEF244I,IEF247I (два сообщения) и запрос IEF238A.

В ответ на запрос оператор должен подтвердить адрес устройства:

R xx,00E

где xx — номер запроса.

В ответ система выдает сообщения IEE600I, IEF236I, IEF237I и *IEA000A. Оператору остается перевести устройство 00F в автономное состояние:

V 00F,OFFLINE

В ответ система выдает сообщение:

IEF281I 00F NOW OFFLINE — устройство 00F переведено в автономное состояние OFFLINE. На этом процесс замены устройства, работающего в системной программой WTR можно считать законченным.

Для облегчения работы оператора автоматически могут выдаваться не только команды START, но и команды MONITOR и VARY, о чем он получает сообщения, например, как показано на рис. 7.5, строки 13, 15 и 16:

```
IEE103I MN JOBNAME$,T *  
IEE103I MN SPACE      *  
IEE103I V 00C,ONLINE  *
```

Команда VARY, переводящая устройство 00C в состояние ONLINE, обычно выдается оператором. Это связано с тем, что при первоначальной загрузке устройства ввода-вывода находятся в состоянии OFFLINE. В данном варианте загрузки она выдается автоматически.

Сообщение IEE360I (27, рис. 7.5) связано с работой программы SMF (системной мониторинной программы). Эта программа обеспечивает сбор и регистрацию информации, полезной для оценки использования ресурсов системы, и является дополнительным средством ОС.

7.5. Загрузка ядра и подготовка планировщика заданий к работе (MFT)

Загрузку ядра управляющей программы MFT рассмотрим на примере протокола, приведенного на рис. 7.10. Загрузка начинается с выдачи тех же сообщений (строки 1, 2), что и загрузка MVT (см. 7.4). Но, в отличие от загрузки управляющей программы MVT, при инициализации ядра MFT динамическая область памяти делится на разделы, размер и количество которых устанавливается при генерации системы. Всего может быть образовано до 52 разделов, из них: до трех разделов для программ системного ввода RDR, до 36 разделов для программ системного вывода WTR, до 15 разделов для программ пользователей. Если программы RDR и WTR помещены в самостоятельные разделы, то они называются *резидентными программами*, а если в разделы программ пользователя, то *транзитными*. Управляющая программа MFT может работать в двух режимах: без подзадач, т. е. когда в разделе выполняется только одна задача, и с подзадачами, когда их несколько. Максимальное число подзадач в задаче может достигать от 196 до 249. Каждая подзадача выполняется в том же разделе, что и создавшая ее задача. Каждому разделу присваивается приоритет. Раздел P0 является разделом с наивысшим приоритетом. Кроме того, разделам присваиваются классы. Разделам, в которых будут выполняться проблемные программы, может быть присвоено до трех классов (от A до O). Разделам, в которых будут находиться резидентные программы системного вывода, присваивается класс W. Разделам, в которых находятся транзитные программы системного вывода, может быть назначено до восьми классов (от A до Z и от 0 до 9). Разделам, в которых будут находиться системные программы ввода, присваивается класс R. Разделам, в которых будут находиться транзитные программы ввода, может быть присвоено до трех классов (от A до O). Программы прямого системного вывода DSO обычно размещаются в тех же разделах, что и проблемные программы. Количество программ DSO определяется числом доступных устройств вывода. Каждой программе DSO присваивается один класс (от A до Z или от 0 до 9). Размеры резидентных разделов: RDR — 16K, WTR — от 30 до 44K. Разделы проблемных программ пользователя могут быть двух размеров: большие и малые. Малые разделы

могут иметь размеры от 8К до 44К. На рис. 7.11 приведен один из возможных вариантов распределения динамической области памяти после окончания

Номер строки	Текст сообщений и команд
1	IEA218I MOD=1033 ASSUMED
2	IEA101A SPECIFY SYSTEM PARAMETERS FOR RELEASE 4.1 MFT
3	*00 IEE801D CHANGE PARTITIONS—REPLY YES/NO (LIST)
4	R 00,'YES'
5	IEE866I DEFINE COMMAND BEING PROCESSED
6	*00 IEE802A ENTER DEFINITION
7	R 00,'P1=(AE,10K,LAST),END'
8	IEE812I P 01 HAS 143360 EXCESS BYTES ADDED
9	IEE805I DEFINITION COMPLETED
10	IEE101A READY
11	IEE103I S WTR,P0,00F *
12	IEE103I S RDR,S,00C *
13	IEE103I S INIT.ALL *
14	T DATE=79.025,CLOCK=18.34.00,Q=(F),AUTO=YN
15	*00 IEF423A SPECIFY JOB QUEUE PARAMETERS
16	R 00,'U'
17	MN JOBNAMES
18	S RDR,S,00A
19	IEE103I S WTR,P0,00F *
20	IEE103I S INIT.ALL *
21	IEF351I SMF SYS1.MAN RECORDING NOT BEING USED
22	IEF403I WTR STARTED P01
23	IEF236I ALLOC. FOR WTR P0 P01
24	IEF237I 00F ALLOCATED TO IEFDRDR P01
25	IEF868I 00F WTR WAITING FOR WORK P00
26	IEF403I RDR STARTED P01
27	IEF236I ALLOC. FOR RDR S P01
28	IEF237I 00A ALLOCATED TO IEFDRDR P01
29	*IEA000A 00A,INT REQ,02,0200 4000,,,RDR

Рис. 7.10. Один из вариантов загрузки управляющей программы MFT.

инициализации системы. Часто в разделе P0 размещают программу резидентного вывода WTR.

P2	P1	P0
до трех классов П/П 8—44К	Класс R Резидентная программа RDR 30—44К	Класс W Резидентная программа WTR 16К
Младшие адреса		Старшие адреса

Рис. 7.11. Упрощенная схема распределения разделов ОП в динамической области после инициализации системы (режим MFT).

После того как NIP закончит работу, система выдает запрос (рис. 7.12 и рис. 7.10, строка 3):

•00 IEE801D CHANGE PARTITIONS—REPLY YES/NO (LIST) — будут ли переопределяться разделы ОП, сгенерированные системой, а также нужна ли оператору информация о разделах в виде списка (LIST); в запросе фор-

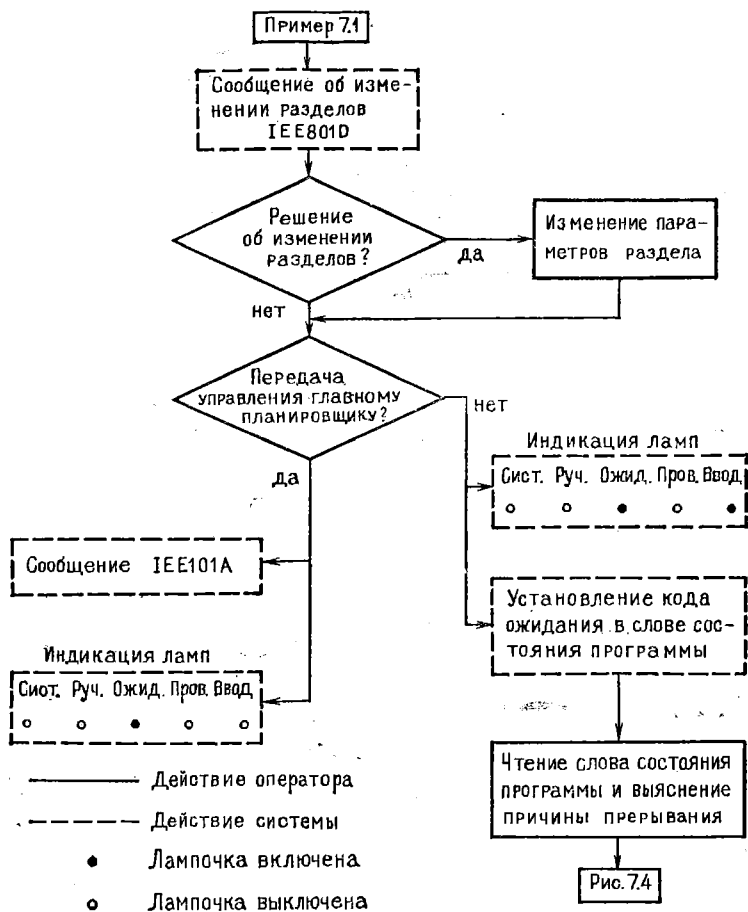


Рис. 7.12.

мулируются ответы: YES — да и NO — нет. На сообщение IEE801D оператор должен дать один из четырех ответов:

1. Если разделы, предусмотренные при генерации системы, остаются без изменения, то выдается ответ:

R 00,'NO'

Система выдает сообщения:

IEE805I DEFINITION COMPLETED — распределение закончилось;

IEE101A READY — система готова к работе.

2. Если разделы остаются без изменения, но на консоль следует выдать информацию о разделах в виде списка, то после получения запроса IEE801D

оператор должен ответить:

R 00,'NO,LIST'

разделы остаются без изменения, но система должна выдать информацию о разделах. В ответ система выдает сообщения IEE866I, IEE804I (их может быть несколько по количеству разделов) и IEE803I, например:

IEE866I DEFINE COMMAND BEING PROCESSED — система выдает распечатку параметров разделов, определенных при генерации;

IEE804I P0 = (WTR,16000),P1 = (AF,8000) IEE804I P2 = (DE,8000),P3 = (G,6000) — текущее распределение разделов: WTR в разделе P0 выделено 16000 байт, класс W; раздел P1 имеет объем 8000 байт и может обрабатывать задания классов A и F; раздел P2 имеет объем 8000 байт и может обрабатывать задания классов D и E; раздел P3 имеет объем 6000 байт и может обрабатывать задания класса G;

IEE816I CLASSES=AFGDE — будет сформировано 5 входных очередей с классами A, F, G, D и E;

*00 IEE803A CONTINUE DEFINITION — продолжите переопределение памяти в разделах.

Ответ оператора:

R 00,'END'

переопределение закончено. Система выдает сообщения:

IEE805I DEFINITION COMPLETED — распределение закончилось;

IEE101A READY — система готова к работе.

3. Если память в разделах будет переопределяться, но без выдачи информации о разделах, то оператор должен ответить (рис. 7.10, строки 4—10):

R 00,'YES'

В ответ система выдает сообщения:

IEE866I DEFINE COMMAND BEING PROCESSED — система будет ждать сообщения IEE805I или IEE814I, после чего возобновляется обработка команд;

*00 IEE802A ENTER DEFINITION — начните переопределение памяти.

После этого сообщения оператор может с помощью команды REPLY начать переопределение разделов, например:

R 00,'P1=(AE,10K,LAST),END' — выделить разделу P1 память 10K; раздел будет обслуживать задания с классами A и E; параметр LAST говорит о том, что раздел P1 является последним, а параметр END — о том, что больше переопределений не будет.

В ответ система выдает сообщения:

IEE812I P 01 HAS 337920 EXCESS BYTES ADDED — к разделу P1 под-
соединяется вся оставшаяся память;

IEE805I DEFINITION COMPLETED — распределение закончилось;

IEE101A READY — система готова к работе.

4. Если память в разделах будет переопределяться с выдачей данных, установленных при генерации системы, на консоль, то оператор выдает команду

R 00,'YES,LIST'

В ответ выдается сообщение главного планировщика IEE866I, IEE804I, IEE816I и запрос IEE802A. Ответ на запрос IEE802A и последующие действия оператора описаны в п. 3.

Таким образом, при работе с управляющей программой MFT оператор перед началом работы с заданиями знает количество и параметры разделов, определенных при генерации и после переопределения разделов, если оно было выполнено оператором, в ответ на запрос IEE801D.

После того как будет получено сообщение IEE101A, процессор перейдет в состояние ожидания. Лампочка «Ожид» на пульте оператора будет включена, а остальные выключены (рис. 7.12). В случае появления каких-либо ошибок на пульте оператора окажутся включенными лампочки «Ожид» и «Ввод». В этом случае оператору следует прочитать на пульте слово состояния программы и выяснить причину неполадки по технической документации.

После перехода процессора в состояние ожидания система выдает три автоматические команды START (строки 11—13, рис. 7.10). В данном примере оператор оставляет две из них WTR и INIT. Об этом он сообщает системе в команде SET, указав параметр AUTO=YN (строка 14). В ответ система снова повторяет автоматические команды START (строки 19 и 20). В строке 21 выдается сообщение IEE351I о том, что программа SMF не будет использоваться.¹ В строке 15 выдается запрос IEF423A об изменении системных параметров. На запрос оператор отвечает, что системные параметры изменяться не будут (строка 16). Кроме того, оператор выдает команду MONITOR и команду START, с помощью которой запускается системная программа RDR (строки 17 и 18). Система в ответ на запуск системных программ WTR и RDR выдает сообщения (строки 22—25). В сообщении IEF868I указывается, что программа WTR ждет работы. Остальные сообщения аналогичны сообщениям, рассмотренным в гл. 6, кроме того, что в них включены данные о номерах разделов.

Если оператору надо изменить размер и параметры какого-либо раздела после того, как закончилась подготовка планировщика заданий, то следует воспользоваться командой DEFINE. Формат команды DEFINE:

Команда	Параметры
{DEFINE} N	[LIST]

Фактически команда DEFINE запрашивает ОС о возможности распределения разделов, так как далее следует определенная процедура действий оператора, которая заключается в том, что он должен ввести некоторые описания разделов в ответ на сообщения ОС IEE866I и IEE802I, как было описано выше.

Параметр LIST приводит к распечатке разделов памяти на консоли, чтобы напомнить оператору о существующем распределении разделов. В сообщении указываются классы заданий, связанные с активными в данный момент разделами. Если используется квантование времени, то будут напеча-

таны разделы, охваченные квантованием. *Квант времени* — это стандартный интервал времени центрального процессора, который выделяется для решения задачи. По истечении этого времени центральный процессор передается другой задаче. В режиме с разделением времени между пользователями *квант времени* — это отрезок времени, который выделяется заданию, взаимодействующему с терминалом.

7.6. Загрузка ядра и подготовка планировщика заданий к работе (SVS)

Загрузка ядра управляющей программы SVS начинается так, как было описано в п. 7.2, но инициализация ядра NIP заканчивается тем, что ОП делится на две части: фиксированную и динамическую (рис. 7.13). Фиксированная область содержит резидентную часть управляющей программы — ядро, область системных очередей (SQA), общую область (LPA) и раздел

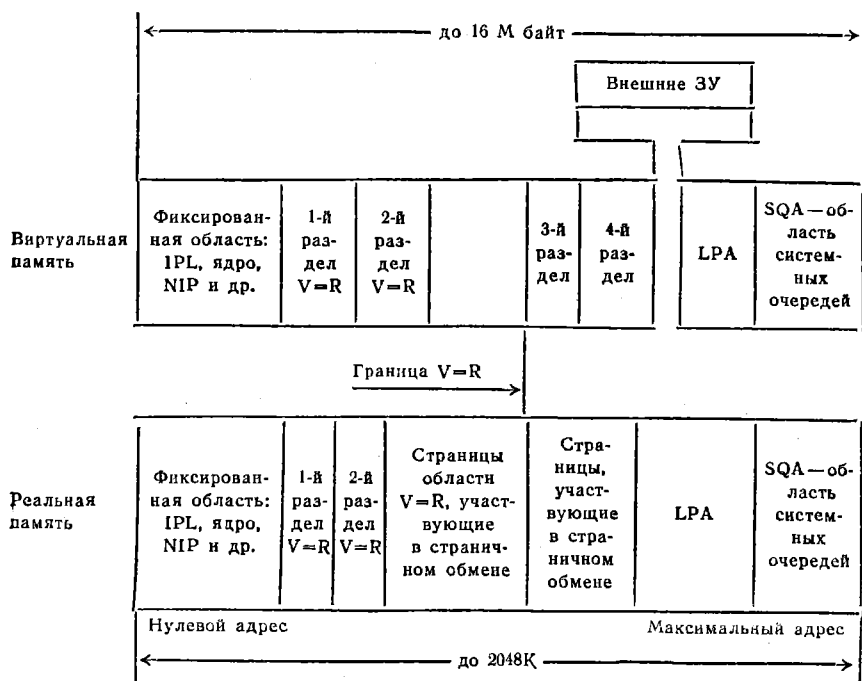


Рис. 7.13. Упрощенная схема разделения памяти на области при инициализации системы SVS.

главного планировщика (MASTER SCHEDULER). Динамическая область содержит разделы для выполнения заданий. На рисунке показана реальная ОП и ее организация и виртуальная память, которая получается путем добавления к ОП памяти внешних запоминающих устройств.

Введение *виртуальной памяти* связано с тем, что часто для выполнения задания не хватает оперативной памяти. Виртуальная память позволяет

устранить этот недостаток путем более рациональной организации и возможности динамического использования ОП. Вводится понятие памяти вычислительной системы, в которую входит память всех имеющихся запоминающих устройств ЭВМ. Запоминающие устройства (ОП, НМД, НМЛ и т. д.) разделяются на два уровня памяти в соответствии с их основной характеристикой — временем выборки. Навысший уровень памяти — это память, время выборки которой около 1 мкс. Время выборки внешних ЗУ находится в пределах от 10^5 до 10^3 мкс. *Оперативная память* (или часть ее) называется реальной памятью. *Виртуальной памятью* называется совокупность всех уровней памяти, как показано на рис. 7.13. Адреса, используемые для обращения к реальной памяти, называются реальными R, а к виртуальной памяти — виртуальными V.

Работа организуется так, чтобы операнды, участвующие в вычислении, в нужное время оказались в ОП. Для этого информация передается по страницам и называется *страничным обменом*. Страница реальной памяти называется *реальной страницей*, виртуальной памяти — *виртуальной страницей*, внешней памяти — *внешней страницей*. Память задачам выделяется *сегментами*.

Размер сегмента 64К, размер страницы 2К. Начальный адрес сегментов страницы кратен их размерам. Максимальный размер ОП может достигнуть 16 Мбайт, что значительно превосходит размер реальной памяти. Если обозначить размер фиксированной области памяти через V_{oc} , то память V, доступная для каждого пользователя, равна

$$V = 16 \text{ Мбайтов} - V_{oc}$$

В реальной памяти в каждый момент времени находятся используемые страницы виртуальной памяти, а неиспользуемые страницы хранятся во внешней памяти в системном страничном наборе данных SYS1.PAGE. Система обеспечивает преобразование адресов между виртуальной и реальной памятью.

В качестве примера на рис. 7.14 приведен протокол одного из вариантов загрузки управляющей программы SVS. Загрузка IPL, ядра и программы NIP заканчивается сообщением IEЕ101А (строка 26). Подготовка планировщика заданий начинается с выдачи сообщений об автоматических командах START (строки 27—29) и заканчивается сообщением IЕF237I (строка 77). В протоколе, кроме известных сообщений, разобранных выше в протоколах на рис. 7.5 и 7.10, встречаются сообщения, на которых остановимся ниже. Некоторые из них могут встретиться и при загрузке управляющих программ MVT и MFT.

Первые два сообщения IEA218I и IEA101A аналогичны таким же сообщениям в MVT и MFT. Далее выдается сообщение (строка 3):

IEA103I DATASET SYS1.DUMP NOT FOUND BY LOCATE — не найден набор данных SYS1.DUMP, в который обычно направляется дамп при возникновении критической системной ошибки.

В сообщении (строка 4):

IEA135A SPECIFY SYS1.DUMP TAPE UNIT ADDRESS OR NO — система просит сообщить адрес устройства на магнитной ленте, на которой находится набор данных SYS1.DUMP, или ответить NO, если он не будет использоваться.

Номер строки	Текст сообщений и команд
1	IEA218I MOD=1055 ASSUMED
2	IEA101A SPECIFY SYSTEM PARAMETERS FOR RELEASE 6.1 SVS
3	IEA103I DATASET SYS1.DUMP NOT FOUND BY LOCATE
4	IEA135A SPECIFY SYS1.DUMP TAPE UNIT ADDRESS OR NO
5	R 00,'286'
6	ITM000D * DDM INITIALIZATION REQUESTED? REPLY NO OR YES (,HOLD)
7	R 00,'NO'
8	ITM001I * DDM NOT SUPPORTED NOW
9	IEA153I HARDCOPY CONSOLE UNAVAILABLE
10	IEA150A SPECIFY HARDCOPY
11	R 00,'HARDCOPY=(SYSLOG)'
12	IEE140I SYSLOG H CMDS ALL
13	IEE140I 010/00C,00F N ALL 01 Z ALL
14	IEE140I 00C,00F/01F N 03 NONE
15	IEE140I 3A0/01F N SYS,I/O 04 Z,A-B 3-4,7,10
16	IEE140I 100/0A0 N ALL 05 Z,A-B ALL
17	IEE140I 0A0/014,00E A ALL 06 Z,A-B ALL
18	IEE140I 014,00E/01F N ALL 08 NONE
19	IEE140I 01F/060 M ALL 09 ALL
20	IEE140I 060/0D0 N ALL 10 Z,A ALL
21	IEE140I 0D0/002 N ALL 11 Z,A ALL
22	IEE140I 002/009 N ALL 12 Z,A-B ALL
23	IEE140I 009/010 N SYS,CONS 13 10
24	IEE140I 011/009 N NONE 14 NONE
25	IEE140I 003/01F N NONE 15 NONE
26	IEE101A READY
27	IEE103I S WTR,00E *
28	IEE103I S RDR,014 *
29	IEE103I S INIT *
30	*00 IEE114A DATE=79.111,CLOCK=14.04.59-REPLY WITH SET PARAMETERS OR U
31	R 00,Q=(,F),AUTO=NY Y
32	IEE600I REPLY TO 00 IS; Q=(,F), AUTO=NY Y
33	IEE118I SET PARAMETER(S) ACCEPTED
34	*01 IEF423A SPECIFY JOB QUEUE PARAMETERS
35	R 1,U
36	IEE600I REPLY TO 01 IS; U
37	*02 IFB010D 'IPL REASON SUBSYSTEM ID' OR 'U'

Рис. 7.14. Один из вариантов загрузки управляющей программы SVS.

Номер строки	Текст сообщений и команд
38	R 2,U
39	IEE600I REPLY TO 02 IS; U
40	IEE103I S RDR,014 *
41	IEE103I S INIT *
42	MN JOBNAME,S,T
43	IEE354I SMF PARAMETERS
44	PRM=SVS616
45	ALT=SVS616
46	SID=01
47	OPI=YES
48	JWT=015
49	BUF=400
50	MDL=55
51	MAN=ALL
52	EXT=YES
53	OPT=2
54	*03 IEE357A REPLY WITH SMF VALUES OR U
55	IEE041I LOG NOW RECORDING ON SYS1.SYSVLOGX ON 432
56	R 3,U
57	IEE600I REPLY TO 03 IS; U
58	IEE360I SMF NOW RECORDING ON SYS1.MANX ON 432
59	TIME=14.06.52
60	*04 MMS001A DEFINE STATUS OF CPU (H, L OR N) FOR
61	DIRECT CONTROL PROCESSING
62	R 4,N
63	IEE600I REPLY TO 04 IS; N
64	IEF403I INIT STARTED TIME=14.07.00
65	IEF403I RDR STARTED TIME=14.07.02
66	IEF244I RDR .014 . IEFORDER UNABLE TO ALLOCATE
67	IEF247I RDR ALLOCATION RECOVERY
68	IEF247I RDR 014 OFFLINE
69	*05 IEF238A REPLY DEVICE NAME OR 'CANCEL'
70	IEF429I INITIATOR 'INIT' WAITING FOR WORK
71	R 5,014
72	IEE600I REPLY TO 05 IS; 014
73	IEF236I ALLOC.FOR RDR 014
74	S WTR,00E,(SPACE=YES)
75	IEF237I 014 ALLOCATED TO IEFORDER
76	*IEA000A 014,INT REQ,02,0200,4000,,,RDR
77	IEF403I WTR STARTED TIME=14.07.34
78	IEF236I ALLOC FOR WTR 00E
79	IEF237I 00E ALLOCATED TO IEFORDER

Рис. 7.14. Один из вариантов загрузки управляющей программы SVS.

Оператор в данном случае сообщает адрес устройства:

R 00,286'

В сообщении ITM000D * (строка 6) задается вопрос о том, будут ли при прямом управлении (DDM) фиксироваться ошибки или нет. Оператор в данном случае отвечает нет — NO (строка 7). В ответ система выдает сообщение ITM001I (строка 8) о том, что DDM не будет использоваться. Далее система выдает сообщение IEA153I (строка 9) о том, что консоль оказалась недоступной. Так как система будет использовать сборный протокол (набор данных HARDCOPY — H сокращ.), то система выдает сообщение IEA150A (строка 10), на которое оператор дает ответ о том, что желательно иметь сборный протокол (строка 11). Далее система сообщает данные о конфигурации консолей. В строке 12 сообщается, что все системные команды и команды оператора (CMD5) будут записываться в системный журнал SYSLOG. В системе организован сборный протокол системы H. Для записи в него сообщений используются все имеющиеся пути (ALL).

Сообщение IEE114A (строка 30) об изменении системных параметров несколько видоизменено по сравнению с IEE423A в MVT и требует ответа от оператора (строка 31) о подготовке набора данных SYS1.SYSJOBQE и принятии или отмене автоматических команд START (см. п. 14 рис. 7.5). Далее система выдает сообщение IEE600I (строка 32) о том, что ответ с номером 00 принят системой.

В строке 37 выдается запрос системы IFB010D о том, что программа IPL будет использовать подсистему идентификаторов пользователя или стандартную подсистему идентификаторов. Оператор отвечает — стандартную (строка 38).

Далее следует сообщение IEE354I, где перечисляются параметры (строки 44—53), при которых работает системная мониторинная программа: PRM=SVS616 — серийный номер тома, на котором находится набор данных SMF; ALT=SVS616 — серийный номер тома, на котором находится альтернативный набор данных SMF; SID=01 — идентификатор системы, в которой работает SMF; OPI=YES — оператор может изменить параметры; JWT=015 — предельное количество минут времени ожидания задания; BUF=400 — размер буфера в байтах; MDL=55 — шифр модели ЭВМ; MAN=ALL — все записи заносятся в набор данных SMF; EXP=YES — подпрограммы выходов получают управление; OPT=2 — создаются записи типов 0—13; выдаются сообщения IEF373I и IEF374I. Запрос IEE357A (строка 54) позволяет изменить эти параметры или их оставить. В данном случае параметры не изменяются (строки 56). Система сообщает, что ответ принят (строка 57). Сообщение IEE041I информирует оператора о том, что запись сообщений производится в набор данных SYS1.SYSVLOGX, который находится на устройстве 432 (строка 55).

Записи программы SMF заносятся в набор данных SYS1.MANX, которые находятся на устройстве 432 (строка 58). Далее следует запрос MMS001A, чтобы определить состояние процессора при прямых контрольных действиях (строка 59). Оператор должен выбрать одну из букв: H — главный процессор, L — вторичный, N — нейтральный. В ответе (строка 60) выбрана буква N. Сообщения в строках 62—71, 73 и 75 связаны с запуском системных про-

грамм RDR и INIT (см. рис. 7.5). Программа системного вывода WTR запускается по команде START (строка 72) и ее сопровождают сообщения (строки 75—77). На этом процесс подготовки планировщика заданий можно считать законченным. Если его сравнить с запуском системы для MVT, то он мало чем от него отличается.

7.7. Конец работы с ЭВМ

Для того чтобы закончить работу с ОС, следует остановить работу всех системных программ: P INIT, P RDR, P WTR и P DSO, а также прекратить выполнение заданий по командам *С* *имя задания*. Затем выдать команды WRITELOG и HALT, как показано на рис. 7.7, после чего выключить питание ЭВМ.

7.7.1. Команда HALT предназначена для подготовки системы к выключению питания, когда работа вычислительной системы под управлением ОС заканчивается. Подготовка к выключению питания состоит в сохранении статистических и других системных данных, которые из ОП пересылаются в набор данных SYSI.LOGREC. Формат команды HALT приводится ниже

Команда	Параметры
{HALT} Z	EOD

В формате команды HALT имеется только один обязательный параметр EOD. Успешное выполнение команды

Z EOD

заканчивается выдачей сообщения

IEE334I HALT EOD SUCCESSFUL — система записала данные системного счетчика внутренних ошибок устройств ввода-вывода в набор данных SYSI.LOGREC, т. е. подготовка к выключению питания прошла успешно.

После получения этого сообщения питание ЭВМ можно отключить. Кроме того, команда HALT закрывает системный журнал. При желании содержимое системного журнала может быть выведено оператором на АЦПУ перед выключением питания. Для этого необходимо перед командой HALT ввести команду WRITELOG.

7.7.2. Выключение питания. Прежде чем выключить питание системы, необходимо изъять все из входного потока, остановить выполнение заданий или дождаться их завершения. Отключение питания системы производится с помощью кнопки «Откл. пит.». Порядок отключения питания следующий: первой отключается оперативная память, далее — все остальные устройства в произвольном порядке. Если процессор находился в состоянии ожидания, то содержимое памяти (за исключением ключей защиты) остается без изменения.

7.8. Получение дампа

Команда DUMP позволяет получить дамп ОП и поместить его в набор данных SYS1.DUMP. Формат команды DUMP:

Ко-манда	Параметры
DUMP	COMM=(заголовок)

Заголовок представляет собой любой текст, содержащий от 1 до 100 символов и который будет являться идентификатором дампа. После выдачи команды, например:

DUMP COMM=(LIDA)

на консоли выдается сообщение, в ответ на которое оператор может указать те области ОП, с которых должен быть получен дамп, например:

R 05,ALL — получить дамп всей ОП, включая ядро ОС и области системных очередей.

Возможен другой ответ:

R номер,STOR=(нач.адрес 1, кон.адрес 1 [, нач.адрес 2, кон.адрес 2])
[,SDATA]

где указываются граничные адреса не более двух участков памяти: параметр SDATA позволяет получить дамп ядра ОС и области системных очередей. Адреса записываются либо шестнадцатеричными цифрами, либо десятичными (до 5 цифр), за которыми следует цифра K, например:

R 05,STOR=(00800K,00900K) или R 05,STOR=(0B7836,0B9836)

Возможен третий ответ:

R 05,SDATA — получение дампа ядра ОС и области системных очередей.

7.9. Получение информации о режиме работы технических средств контроля

Команда MODE позволяет получить сведения о режиме работы технических средств контроля и внести изменения в этот режим. Формат команды MODE:

Команда	Параметры
MODE	$\left\{ \begin{array}{l} \text{STATUS} \\ \text{RETRY} \quad \left\{ \begin{array}{l} \text{RECORD} \\ \text{QUIET} \end{array} \right\} \\ \text{MAIN} \quad \left\{ \begin{array}{l} \text{RECORD} \\ \text{QUIET} \\ \text{COUNT} \end{array} \right\} \\ \text{CONTROL} \quad \left\{ \begin{array}{l} \text{THRESHOLD} \\ \text{QUIET} \end{array} \right\} \end{array} \right\}$

Параметры обеспечивают следующие режимы: STATUS — получение сведений о режиме, в котором работают в настоящий момент времени средства контроля; RETRY RECORD — выдача сообщения оператору о том, что команды используются им повторно; RETRY QUIET — сообщения о повторном использовании команд оператором не выдаются; MAIN RECORD — получение сообщений от средств контроля и коррекции ошибок ОП (необходимо, чтобы ранее был установлен режим RETRY RECORD); MAIN QUIET — сообщения от средств контроля и коррекции ошибок ОП не выдаются; MAIN COUNT имеет то же назначение, что MAIN RECORD, но со счетом легких ошибок от 1 до 999 (число ошибок задается при генерации ОС; используются для ЕС-1055 и ЕС-1060); CONTROL THRESHOLD — получение сообщений только от нескольких первых легких ошибок, далее сообщения поступать не будут (используется только в ЕС-1035); CONTROL QUIET — сообщения об ошибках не выдаются (только для ЕС-1035).

Глава 8. УПРАВЛЕНИЕ ПОРЯДКОМ ПРОХОЖДЕНИЯ ЗАДАНИЙ В СИСТЕМЕ MVT

8.1. Общая характеристика команд

Все задания по способу выполнения можно разделить на два вида: задания, вводимые во входном потоке, и нестандартные каталогизированные процедуры*), которые находятся в библиотеке процедур SYS1.PROCLIB и вызываются с консоли. Для управления выполнением заданий используются команды START, STOP, CANCEL, MODIFY, RESET, HOLD и RELEASE.

Задания во входном потоке (рис. 8.1) запускаются с устройства ввода с помощью программы системного ввода RDR (п. 6.1.1). Программа RDR помещает задания во входную очередь JOB или очередь задержанных заданий HOLD. Из очереди JOB задания выбираются для выполнения инициатором INIT, а после выполнения результаты выводятся системными программами WTR или DSO. Обязательным условием выполнения задания является совпадение классов системных программ (INIT, WTR или DSO) с классами очереди JOB и SOUT, в которых находятся задания. Возможен случай, когда оператор специально переводит задание в класс, который системные программы не обслуживают, и задание не будет выполняться. Тем самым он задерживает

Рис. 8.1. Ввод задания PRIM во входном потоке.

его выполнение. Задержать выполнение задания можно также, если поместить задание в очередь HOLD. В очередь HOLD задание помещается программой RDR, если программист указал в операторе JOB параметр TYP-RUN=HOLD. Когда задания помещены в очередь JOB или HOLD, то оператор может перевести их из одной очереди в другую. Команда HOLD переводит задание из очереди JOB в очередь HOLD, а команда RELEASE выполняет обратный перевод. Команда STOP и CANCEL позволяет прекратить процесс обработки задания. Команды MODIFY и RESET позволяют изменить классы, приоритеты и параметры заданий. Такие изменения могут быть сделаны оператором для того, чтобы обеспечить более оперативное прохождение заданий в системе.

Нестандартные каталогизированные процедуры (далее: каталогизированные процедуры) запускаются по команде START. Они находятся постоянно в библиотеке процедур SYS1.PROCLIB и вызываются из нее по имени. Имя

*) Стандартные каталогизированные процедуры такие, как RDR, INIT, WTR и DSO рассмотрены в гл. 6.

каталогизированной процедуры — это имя раздела библиотеки SYS1.PROCLIB, содержащего эту процедуру. При вызове каталогизированной процедуры из SYS1.PROCLIB оператор указывает ее имя в команде START. Назначение команд STOP, CANCEL и MODIFY такое же, как при работе с заданиями во входном потоке. Вызванная по команде START нестандартная каталогизированная процедура выполняется в порядке, предусмотренном для системных программ.

8.2. Управление выполнением заданий, введенных во входном потоке

Управляющее воздействие команд на процесс выполнения заданий будем рассматривать в следующие моменты времени (табл. 8.1): задание вводится во входном потоке; задание находится в очереди (JOB или HOLD); задание выполняется под управлением инициатора; выходная информация находится

Таблица 8.1. Воздействие команд и системных программ,
на задание, введенное во входном потоке

Состояние задания	Команды					
	MODIFY	RESET	HOLD	RELEASE	STOP	CANCEL
Ввод задания с помощью RDR	—	—	—	—	—	+
Находится в очереди JOB	—	+	+	—	—	+
Находится в очереди HOLD	—	+	—	+	—	+
Задание выполняется с помощью INIT	+	—	—	—	+	+
Находится в очереди SOUT	—	+	+	+	—	+
Выводятся данные с помощью WTR или DSO	—	—	—	—	—	+

в выходной очереди SOUT; выходная информация выводится на устройство вывода.

8.2.1. Ввод заданий во входном потоке с перфокарт начинается с того, что оператор помещает колоду с перфокартами в приемный карман считывающего устройства и нажимает кнопку «Пуск». После нажатия кнопки «Пуск» управление получает программа системного ввода RDR (считаем, что программа RDR была ранее запущена по команде START, как показано в п. 6.1). Пусть, например, задание PRIM вводится с устройства 00С и имеет следующие параметры: в операторе JOB указан класс CLASS=A, приоритет PRTY=5. Программа RDR помещает задание PRIM во входную очередь JOB в класс A с приоритетом 5 (рис. 8.1).

Во входном потоке количество перфокарт не должно превышать 1000. Можно начать ввод заданий с некоторого известного задания. Для этого необходимо имя этого задания указать в команде START (п. 6.1). Следует

отметить, что если одновременно работают несколько программ RDR, то каждая вводит свой поток заданий.

8.2.2. Управляющее воздействие команд на задания, находящиеся во входной очереди JOB или в очереди задержанных заданий HOLD, определяется табл. 8.1. Чтобы убедиться, что задание находится в одной из этих очередей, следует выдать команду DISPLAY Q (см. п. 4.1).

Команда RESET служит для изменения класса и (или) приоритета задания, которое находится во входной очереди JOB или очереди задержанных заданий HOLD (рис. 8.2). Первоначально класс и приоритет присваиваются

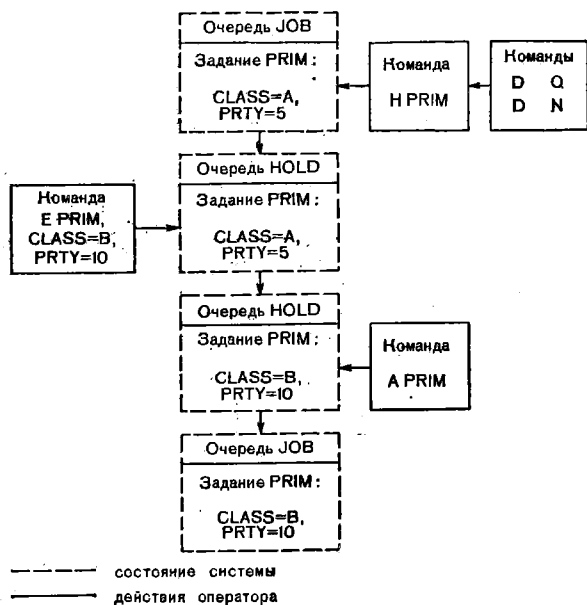


Рис. 8.2. Управляющее воздействие команд HOLD, RESET и RELEASE на задание, которое находится в очереди JOB или HOLD.

заданию программистом в операторе JOB с помощью параметров CLASS и PRTY (п. 3.3.2). В соответствии с этими параметрами система помещает задание в системные очереди (рис. 8.1). Команда RESET, формат которой приводится в табл. 8.2, позволяет изменить параметры задания, т. е. перевести его в другой класс и назначить новый приоритет. В формате команды RESET указывается имя задания, класс и (или) приоритет которого изменяются. В параметре CLASS указывается значение входного класса от А до О. В параметре PRTY назначается новый приоритет. Значения приоритетов находятся в пределах от 0 до 13 (наивысший приоритет).

Пример 8.1. В задании OLGA изменить приоритет на значение, равное 12:

E OLGA,PRTY=12

После того как система изменит приоритет, она выдает сообщение
 IEE304I OLGA JOB RESET — система изменила параметры задания OLGA.

Таблица 8.2. Форматы команд, которые используются для управления и изменения параметров заданий, находящихся в очередях JOB или HOLD

Команда	Параметры	Состояние задания
{RESET} {E}	<i>имя задания</i> {,PRTY= <i>приоритет</i> *} {,CLASS= <i>класс</i> }	Изменяется класс и приоритет
{CANCEL} {C}	<i>имя задания</i> [,ALL ,IN[= <i>класс</i>]]	Задание отменяется
{HOLD} {H}	{ <i>имя задания</i> } {Q[= <i>список</i>]}	Перевод задания из JOB в HOLD
{RELEASE} {A}	{ <i>имя задания</i> } {Q[= <i>список</i>]}	Перевод задания из HOLD в JOB

*) Можно указывать оба параметра одновременно.

Пример 8.2. Заданию PRIM назначить новый приоритет, равный 10, и класс B (рис. 8.2):

E PRIM,CLASS=B,PRTY=10

После изменения класса и приоритета система выдает сообщение

IEE304I PRIM JOB RESET

Команда CANCEL предназначена для немедленного аннулирования выполнения задания, находящегося в очередях JOB или HOLD. Формат команды приводится в табл. 8.2. В формате команды CANCEL обязательно указывается имя задания, выполнение которого должно быть отменено, например: C F08181 — где F08181 имя задания.

После аннулирования задания выдается сообщение системы

IEE301I F08181 CANCEL COMMAND ACCEPTED — команда CANCEL принята системой.

Последующие сообщения уточняют причину отмены задания, например:

IEF450I F08181. F01. ABEND S222 TIME=12.10.41 — выполнение пункта F01 задания F08181 прекращено; S222 — системный код завершения, показывающий, что задание отменил оператор; время выдачи команды CANCEL — 12.10.41.

Параметр ALL предписывает системе искать задание, которое подлежит аннулированию, во всех очередях, например:

C PRIM,ALL — PRIM — имя задания.

Ключевой параметр IN показывает, что задание следует искать во входной очереди JOB. Этим сужается область поиска задания, подлежащего аннулированию. *Класс* заменяется именем входного класса от A до O, в котором находится задание, например:

C PRIM,IN=F — аннулировать задание PRIM, которое находится во входной очереди JOB, в классе F.

Команда HOLD позволяет перевести задание из очереди JOB в очередь HOLD (рис. 8.2). Формат команды приводится в табл. 8.2. В формате коман-

ды HOLD указывается имя задания, которое необходимо перевести в очередь задержанных заданий.

Пример 8.3. Поставить в очередь задержанных заданий задание с именем RAB1:

H RAB1

В ответ система после перевода задания в очередь HOLD выдает сообщение:

IEE3301 RAB1 JOB HELD — задание RAB1 находится в очереди задержанных заданий.

В очередь HOLD задание может попасть помимо оператора в случае, если программист в своей программе в операторе JOB указал параметр TYPRUN=HOLD.

Ключевое слово Q используется в двух случаях: 1) если все 15 входных очередей переводятся в очередь задержанных заданий; 2) если необходимо задержать выполнение заданий определенных входных классов. В последнем случае указывается список классов от A до O. В списке можно указать любую комбинацию классов, но не более четырех. Если указываются два или более классов, то они должны быть разделены запятыми и заключены в общие круглые скобки.

Пример 8.4. Поставить в очередь задержанных заданий HOLD все задания, которые находятся во всех 15 входных очередях JOB:

H Q

В ответ система выдает сообщение:

IEE3321 QUEUE HELD — система временно приостанавливает выборку заданий для выполнения из всех входных очередей.

Пример 8.5. Поставить в очередь задержанных заданий все задания, которые находятся в классах B, C, D, E:

H Q=(B,C,D,E)

Ответ системы аналогичен ответу в примере 8.4.

Команда RELEASE переводит задание, или класс заданий, или все задания из очереди задержанных заданий HOLD в очередь JOB. Формат команды RELEASE приводится в табл. 8.2. В формате команды RELEASE сообщается имя задания, которое должно быть переведено из очереди задержанных заданий HOLD во входную очередь JOB. Если указывается параметр Q, то это значит, что все задания переводятся в очередь JOB. Список заменяется именами классов заданий. Обратный перевод из очереди HOLD в очередь JOB производится по команде RELEASE, причем формат команды RELEASE должен находиться в следующем соответствии с форматом команды HOLD: H имя задания требует команды A имя задания; H Q — команды A Q; H Q=список — команды A Q=список. Если задание было задержано программистом с помощью параметра TYPRUN=HOLD, то выдается команда A имя задания.

Пример 8.6. Задержать, а затем разрешить выборку всех заданий:

H Q

В ответ выдается сообщение системы:

IEE3321 QUEUE HELD — все задания переведены в очередь HOLD.

Разрешить выборку всех заданий:

A Q

Система в ответ выдает сообщение:

IEE333I QUEUE RELEASED — все задания могут быть реализованы, т.е. переведены в очередь JOB.

Если задание не найдено ни в одной из очередей при выдаче команд RESET, CANCEL, HOLD и RELEASE, то система выдает сообщение, например:

IEE316I F16 JOB NOT FOUND — задание F16 не найдено.

Это бывает в том случае, если задания нет в системе или началось его выполнение, или оно обрабатывается программами системного вывода.

8.2.3. Управляющее воздействие команд на задания, выбранные для выполнения инициатором INIT. Инициатор выбирает задание для выполнения из JOB, если имеется равенство класса инициатора и класса очереди (рис. 8.3). Команда MODIFY разрешает изменять параметры во время выполнения задания. Команды STOP и CANCEL позволяют остановить выполнение задания.

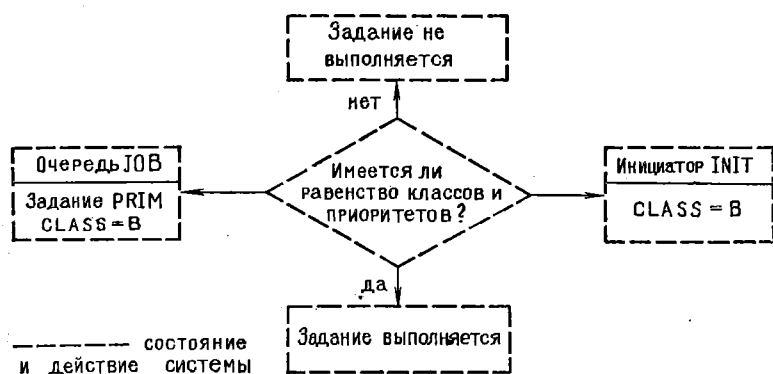


Рис. 8.3.

В общем случае порядок сообщений при работе с заданиями и действия оператора трудно предсказать. Поэтому остановимся только на некоторых наиболее простых случаях. Считаем, что команда MN JOBNAMES,T выдана. Если нужный том установлен на устройстве, которое находится в состоянии ONLINE, и при выполнении задания не произошло никаких сбоев, то система выдает два сообщения: IEF403I и IEF404I о начале и конце работы соответственно (рис. 8.4,а). Такой случай рассматривается в примере 8.7.

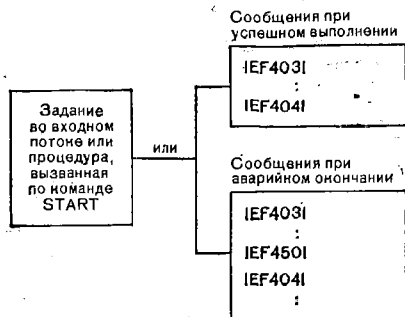
Пример 8.7. Ввод задания DOG во входном потоке приводит к появлению сообщений:

IEF403I DOG STARTED TIME=22.58.30 — задание DOG стартовало в 22.58.30;

IEF404I DOG ENDED TIME=23.00.26 — задание закончило выполнение в 23.00.26.

В более сложных случаях между сообщениями IEF403I и IEF404I могут появиться сообщения, требующие определенных действий оператора, например, как показано в пример 8.8.

а) Устройство с набором данных, используемых заданием или процедурой, находится в состоянии ONLINE.



б) Устройство с набором данных, используемым заданием или процедурой, находится в состоянии OFFLINE.

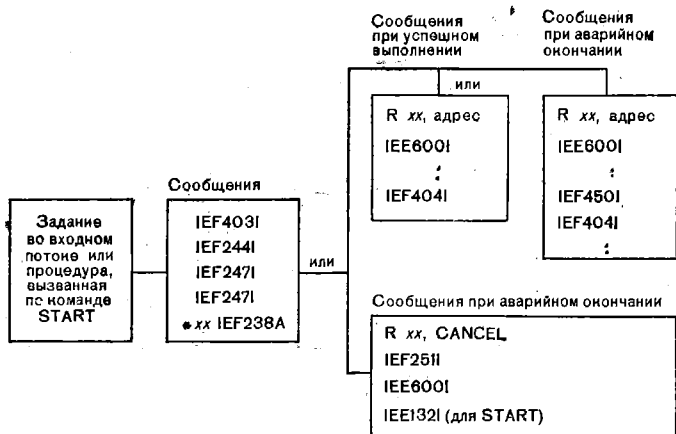


Рис. 8.4. Порядок выдачи сообщений при выполнении заданий.

Пример 8.8. Задание ZAG вводится во входном потоке:

IEF403I ZAG STARTED TIME=12.47.52 — задание ZAG стартовало в 12.47.52.

Далее следуют сообщения, запрос системы и ответ оператора:

IEF236I ALLOC.FOR ZAG LKED
IEF237I 00E ALLOCATED TO SYSPRINT
IEF861I FOLLOWING RESERVED DATA SET NAMES
UNAVAILABLE TO ZAG

IEF863I DSNAME=SYS1.PLILIB
•08 IEF864D 'REPLY 'RETRY' OR 'CANCEL' OR 'WAIT'
R 08, WAIT

IEE600I REPLY TO 08 IS; WAIT
IEF404I ZAG ENDED TIME=13.02.55

Сообщение IEF236I идентифицирует пункт LKED задания ZAG; IEF237I сообщает о том, что устройство 00E выделено набору данных с именем SYSPRINT. Сообщение IEF861I говорит о том, что набор данных, указанный в сообщении IEF237I DSNAME=SYS1.PL1LIB, зарезервирован для другого задания, выполняемого в настоящее время. Поэтому система выдает запрос оператору IEF864D. Оператору следует ввести ответ с помощью команды REPLY: повторить обращение к набору данных (RETRY); прекратить обработку задания (CANCEL) или приостановить обработку задания до того момента, когда набор данных освободится (WAIT). В данном случае оператор ответил WAIT. Последнее сообщение IEF404I говорит о том, что выполнение задания закончилось успешно.

Если выполнение задания прерывается из-за какой-либо ошибки в программе, то между сообщениями IEF403I, IEF404I и IEF450I (рис. 8.4, а) выдается сообщение о причине аварийного завершения, например, как показано в примере 8.9.

Пример 8.9. Введено задание OT во входном потоке. Система сообщает о том, что оно стартовало:

```
IEF403I OT STARTED TIME=19.29.40
```

Далее следуют сообщения:

```
IEF450I OT.ST1.GO ABEND S706 TIME=19.31.41
```

```
IEA703I 706 OT GO MODULE ACCESSED GO
```

```
IEF404I OT ENDED TIME=19.31.48
```

В сообщении IEF450I говорится о том, что задание OT закончилось аварийно с системным кодом S706. Далее следует сообщение IEA703I с уточнением причины аварийного завершения. И, наконец, сообщение IEF404I об окончании выполнения задания.

Если устройство, на котором находится набор данных, используемый заданием, находится в состоянии OFFLINE, то порядок выдачи сообщений несколько другой (рис. 8.4, б). В примере 8.10 показан один из возможных вариантов выполнения задания с успешным его окончанием.

Пример 8.10. Задание KADR введено во входном потоке. О старте задания система выдает сообщение:

```
IEF403I KADR STARTED TIME=06.26.20
```

Далее следуют сообщения, запрос системы и ответ оператора:

```
IEF244I KADR. .STEPLIB UNABLE TO ALLOCATE
```

```
IEF247I KADR ALLOCATION RECOVERY
```

```
IEF247I KADR 150,320,330,340,350,420,430,440,450,530,540
```

OFFLINE

```
*11 IEF238A REPLY DEVICE NAME,OR 'CANCEL'
```

```
R 11,150
```

```
IEE600I REPLY TO 11 IS; 150
```

```
IEF404I KADR ENDED TIME=06.46.03
```

В сообщении IEF244I указывается, что система не может распределить устройство для набора данных STEPLIB задания KADR. В сообщении IEF247I перечисляются устройства, которые находятся в состоянии OFFLINE и не могут быть выделены заданию KADR. В запросе IEF238A оператору

предлагается указать номер устройства с помощью команды REPLY или остановить выполнение задания. Далее следует ответ оператора с указанием номера устройства 150. Выполнение задания заканчивается успешно, о чем свидетельствует сообщение IEF404I.

Если оператор решил остановить выполнение задания, указав в команде REPLY параметр CANCEL:

R 11,CANCEL

то система выдаст бы следующие сообщения:

IEF251I KADR JOB CANCELED

IEE600I REPLY TO 11 IS; CANCEL

Команда MODIFY предназначена для изменения ключевых параметров. Формат команды приводится в табл. 8.3. В формате команды MODIFY указывается имя задания и новые ключевые параметры. Программист в своей

Таблица 8.3. Форматы команд, которые используются для управления и изменения параметров заданий, при работе инициатора INIT

Операция	Операнды
{MODIFY} {P}	имя задания, ключевые параметры
{STOP} {P}	имя задания
{CANCEL} {C}	имя задания [,DUMP]

программе должен предусмотреть возможность такого изменения ключевых параметров, иначе команда MODIFY не окажет никакого действия на задание.

Команда STOP позволяет остановить выполнение задания. Эта команда действует, если программист предусмотрел такую возможность в своей программе. Формат команды STOP приводится в табл. 8.3. В формате команды указывается имя задания.

Команда CANCEL прекращает выполнение задания. Формат команды CANCEL приводится в табл. 8.3. Обязательно в формате команды указывается имя задания. Кроме имени задания, можно указать параметр DUMP, который позволяет получить распечатку (дамп) участка памяти в момент отмены задания. Дамп — это некоторые таблицы. Полный дамп выдается, если в задании имеется оператор DD с именем SYSABEND или SYSUDUMP.

Пример 8.11. Прекратить выполнение задания DP:

C DP

В ответ система выдает сообщения:

IEE301I DP CANCEL COMMAND ACCEPTED — выполнение задания DP прекращено по команде CANCEL;

IEF450I DP. DP. ABEND S222 TIME=19.49.22 — выполнение задания DP прекращено по команде CANCEL, выданной оператором; системный код S222; время 19.49.22.

8.2.4. Управляющие воздействия команд на задания, которые находятся в очереди SOUT. После того как процессор закончит выполнение задания, инициатор помещает результаты вычисления (выходные наборы данных) и системные сообщения на устройство прямого доступа (рис. 6.1) и ставит задание в выходную очередь SOUT в соответствии с параметром SYSOUT оператора DD и параметром MSGCLASS оператора JOB. С помощью команды RESET можно изменить выходной класс задания, а по команде CANCEL отменить выдачу выходной информации.

Формат команды RESET приводится в табл. 8.4. Обязательным параметром является *имя задания*. Кроме имени задания, можно указать новый вы-

Таблица 8.4. Форматы команд, которые используются для управления и изменения параметров заданий, которые находятся в очереди SOUT

Операция	Операнды	Состояние задания
{RESET} {E}	<i>имя задания</i> [ALL OUT[=класс]]	Изменяется выходной класс
{CANCEL} {C}	<i>имя задания</i> {ALL OUT[=класс]}	Задание отменяется
{HOLD} {H}	Q=SOUT	Задерживаются все задания в очереди SOUT
{RELEASE} {A}	Q=SOUT	Разрешается выполнение заданий из очереди SOUT

ходной класс (от A до Z и от 0 до 9). В ответ на выдачу команды RESET система выдает сообщение IEE304I (см. пример 8.1).

Формат команды CANCEL приводится в табл. 8.4. Обязательным параметром является *имя задания*. Параметр ALL предписывает системе искать задание во всех очередях. Параметр OUT показывает, что задание следует искать в выходной очереди SOUT, а класс еще более сужает область поиска задания в выходной очереди. После аннулирования задания по команде CANCEL выдаются сообщения IEE301I и IEF450I, как было показано в п. 8.2.2.

Команды HOLD и RELEASE позволяют задержать выдачу выходной информации всей выходной очереди или разрешить выдачу информации. Формат команды приведен в табл. 8.4.

Пример 8.12. Аннулировать задание RRIM1, которое находится в классе B, очереди SOUT:

C PRIM1,OUT=B

8.2.5. Остановить выдачу выходной информации можно, воздействуя на работу программы системного вывода WTR и DSO, как описано в гл. 6.

8.2.6. В случае успешного окончания выполнения задания система выдает сообщение:

IEF403I PRIM ENDED TIME=16.55.36 — задание PRIM закончило работу в 16.55.36.

Если задание закончилось аварийно на любом из этапов, то выдается сообщение, в котором указывается причина аварийного завершения. Систем-

ные коды завершения представляют собой трехзначное шестнадцатеричное число, перед которым указывается буква S, например, S222 — оператор отменил задание без выдачи дампа. Перечень всех системных кодов аварийного завершения приведен в [2].

8.3. Управление выполнением нестандартных каталогизированных процедур

Управляющее воздействие команд на процесс выполнения нестандартной каталогизированной процедуры (далее: процедуры) во многом похоже на процесс, описанный в п. 8.2 при работе с заданиями, введенными во входном потоке. Основное отличие заключается в том, что процедура вызывается по команде START и в очереди JOB или HOLD она не попадает. Вызов запоминается в ЭВМ, и при первой же возможности процедура начинает выполняться. Процедура состоит из операторов EXEC и DD. В подполе операндов

Таблица 8.5. Управляющее воздействие команд на выполнение нестандартной каталогизированной процедуры

Состояние процедуры	Команды			
	START	MODIFY	STOP	CANCEL
1. Запуск	+	-	-	+
2. Процедура выполняется с помощью INIT	-	+	+	+
3. Данные выводятся с помощью WTR или DSO	-	-	-	+

EXEC и DD указываются параметры, описанные в гл. 3 и 9. В табл. 8.5 показано воздействие команд оператора на выполнение процедуры (ср. с табл. 8.1).

8.3.1. **Запуск процедуры** производится по команде START. Формат команды START приводится в табл. 8.6. В формате команды START указывается имя процедуры. Кроме того, может быть сообщен идентификатор, который представляет собой имя, состоящее из букв и цифр (первой должна быть буква). В идентификаторе должно быть не более восьми знаков. Ключевые параметры — это параметры, которые используются в операторе EXEC данной каталогизированной процедуры.

Порядок выдачи сообщений при работе с процедурами и действия оператора в каждом конкретном случае такие же, как при работе с заданиями, вводимыми во входном потоке (п. 8.2, рис. 8.4). Ниже остановимся на некоторых наиболее простых случаях выдачи сообщений при работе с процедурами. Если нужный том установлен на устройстве, которое находится в состоянии ONLINE, и при выполнении процедуры не произошло никаких сбоев, то система выдает два сообщения: IEF403I и IEF404I о начале и конце работы соответственно (рис. 8.4, а). Такой пример приводится ниже.

Пример 8.13. Запустить процедуру

S S,T=SNRCOP,B=URI,V=2

где T, B и V — ключевые параметры данной процедуры, которые сообщаются оператору программистом. В случае успешного запуска и выполнения процедуры система выдает сообщения:

IEF403I S STARTED TIME=17.30.40 — процедура S стартовала в 17.30.40;

IEF404I S ENDED TIME=17.31.01 — выполнение процедуры закончилось в 17.31.01.

В более сложных случаях между сообщениями IEF403I и IEF404I могут появиться сообщения, требующие определенных действий оператора.

Пример 8.14. Запущена процедура

S PC

выполнение которой заканчивается аварийно. Система выдает следующие сообщения:

IEF403I PC STARTED TIME=19.14.37

IEF236I ALLOC. FOR PC PC

IEF237I 00D ALLOCATED TO SYSPUNCH

IEF450I PC .PC .ABEND S806 TIME=19.14.46

IEF404I PC ENDED TIME=19.14.46

IEA703I 806=4 PC PC MODULE ACCESSED PUNCH

Сообщение IEF403I говорит о том, что процедура PC стартовала. В сообщении IEF236I указывается имя процедуры, для которой были распределены устройства. Следующее сообщение IEF237 описывает распределение устройства. В данном случае устройства с адресом 00D выделено для набора данных, описанного в операторе DD с именем SYSPUNCH. В сообщении IEF450I указывается о том, что выполнение процедуры PC закончилось аварийно (ABEND) с системным кодом завершения S806: программа не найдена в библиотеке. Далее следует сообщение IEF404I о том, что процедура PC закончила работу. И, наконец, сообщение IEA703I уточняет причину аварийного окончания.

Если устройство, на котором находится набор данных, используемый процедурой, находится в состоянии OFFLINE, то порядок выдачи сообщений несколько другой (рис. 8.4, б). Ниже в примере 8.15 показан возможный ход выполнения процедуры с успешным его окончанием.

Пример 8.15. Запущена процедура

S DP.DP, T=KAB682

Система выдает сообщения и запрос 03, на который следует ответ оператора:

IEF403I DP STARTED TIME=20.45.24

IEF244I DP .DP .TAPE UNABLE TO ALLOCATE

IEF247I DP ALLOCATION RECOVERY

IEF247I DP 180,181,280,281 OFFLINE

*03 IEF238A REPLY DEVICE NAME, OR 'CANCEL'

R 3,180

IEE600I REPLY TO 03 IS; 180

IEF234E R 180,DP

*IEF233A M 180, SCRTCH, NL, DP, DP

IEF404I DP ENDED TIME=20.51.55

Из всех сообщений остановимся на втором сообщении IEF247I, в котором перечисляются адреса устройств 180, 181, 280, 281, находящихся в OFFLINE, и из которых должен быть сделан выбор адреса устройства при ответе на запрос IEF238A. Остальные сообщения неоднократно рассматривались выше в примерах. Выполнение процедуры закончилось успешно. В случае аварийного окончания выдается сообщение IEF450I, как в примере 8.14.

Если оператор решил снять задание, то он должен в команде REPLY указать CANCEL. Такой случай приводится в примере 8.16.

Пример 8.16. Запущена процедура

S DP.DP,T=MVT676

Система выдает сообщения и запрос 05, на который следует ответ оператора:

```
IEF403I DP STARTED TIME=20.54.05
IEF244I DP .DP .DISK UNABLE TO ALLOCATE
IEF247I DP ALLOCATION RECOVERY
IEF247I DP 130,132,230,231,232,252 OFFLINE
*05 IEF238A REPLY DEVICE NAME,OR'CANCEL'
R 5,CANCEL
IEF251I DP JOB CANCELED
IEE600I REPLY TO 05 IS; CANCEL
IEE132I START COMMAND DEV ALLOC ERR
```

Остановимся только на сообщении IEF251I, в котором указывается, что задание закончено. В сообщении IEE132I указывается, что команда START выдана ошибочно.

Команда CANCEL позволяет отменить команду START в любой момент времени. Формат команды CANCEL приводится в табл. 8.6. В формате команды CANCEL указывается идентификатор, который был указан в команде START. Если идентификатор отсутствует, то в качестве идентификатора используется имя процедуры, как показано в примере 8.17.

Пример 8.17. Запущена процедура

S PRSMFZS,C=2,TOM=DL7002

В ответ система выдает сообщения, в том числе сообщение IEF504A с требованием смонтировать том:

```
IEF403I PRSMFZS STARTED TIME=07.23.50
IEF505I UNIT REQUIRED
IEF234E D 233,NEWZUB
*IEF504A M 5031,DP3010,,PRSMFZS,PRSMFZS
C PRSMFZS
IEE301I PRSMFZS CANCEL COMMAND ACCEPTED
IEF450I PRSMFZS .PRSMFZS .ABEND S222 TIME=07.29.01
IEE132I START COMMAND DEV ALLOC ERR
```

Сообщение IEF403I говорит о старте процедуры PRSMFZS; IEF505I предупреждает о том, что том, указанный в сообщении IEF234E должен быть снят с устройства 233. Для работы процедуры требуется том DP3010, который необходимо смонтировать на устройстве типа 5061. В данном слу-

чае, для того чтобы выполнить требования системы, оператор должен снять выполнение процедуры с помощью команды CANCEL. В ответ система выдает серию сообщений: IEE3011 — выполнение процедуры закончено по команде CANCEL; IEF4501 — завершение процедуры аварийное из-за выдачи команды CANCEL; IEE1321 — команда START выдана ошибочно.

8.3.2. Выполнение каталогизированной процедуры происходит под управлением INIT. Команда MODIFY позволяет изменить ключевые параметры каталогизированной процедуры. Формат команды MODIFY приводится в табл. 8.6. В формате команды MODIFY указывается имя каталогизированной процедуры. Идентификатор, как в команде START (п. 8.3.1). Ключевые

Таблица 8.6. Форматы команд, которые используются для управления и изменения параметров каталогизированных процедур

Операция	Операнды	Примечание
{START} {S}	имя[.идентификатор][.ключевые параметры]	При запуске и выборе задания INIT
{CANCEL} {C}	идентификатор	
{MODIFY} {F}	{имя.[идентификатор].[ключевые параметры]}	При работе INIT
{STOP} {P}	{имя.[идентификатор]}	

параметры — это новые значения параметров каталогизированной процедуры. Программист должен предусмотреть возможность динамического изменения параметров в программе, иначе команда не будет выполняться.

Команда STOP позволяет остановить выполнение каталогизированной процедуры. Эта команда действует, если программист предусмотрел эту возможность в своей программе. Формат команды STOP приводится в табл. 8.6. В формате команды STOP указывается имя каталогизированной процедуры и идентификатор, как в команде START.

Команда CANCEL, формат которой приводится в табл. 8.6, отменяет выполнение пункта задания, начатого INIT. Система на ввод команды CANCEL отвечает также как обычно.

Пример 8.18. Выдана команда

S COPY1

О запуске процедуры выдается сообщение IEF4031:

IEF4031 COPY1 STARTED TIME=16.12.09

Если оператор хочет снять выполнение процедуры COPY1, то выдается команда

C COPY1

Система в ответ выдает сообщения: IEF301I, IEF450I и IEE132I:

```
IEF301I COPY1 CANCEL COMMAND ACCEPTED  
IEF450I COPY1 .COPY 1 .ABEND S222 TIME=16.18.01  
IEE132I START COMMAND DEV ALLOC ERR
```

Интерпретация этих сообщений аналогична той, которая была приведена в примере 8.17.

8.3.3. Вывод результатов выполнения каталогизированной процедуры. Остановить вывод информации можно, выдав команду CANCEL, формат которой приводится в табл. 8.6. Кроме того, остановить вывод информации можно, воздействуя на системные программы WTR или DSO (см. гл. 6), вплоть до момента распределения устройств вывода заданию. Нельзя остановить вывод информации с помощью команды CANCEL после того, как закончилось распределение устройств вывода. Если все-таки надо остановить вывод информации, следует воздействовать на устройство вывода командой CANCEL, например:

C 00F

Вывод информации текущего задания закончится и система начнет выводить информацию следующего задания.

В данной главе будет продолжено рассмотрение языка управления заданиями, начатого в гл. 3.

9.1. Управляющий оператор JOB

В этом разделе рассматриваются параметры ADDRSPC, COND, NOTIFY, RD, RESTART, ROLL (см. табл. 3.2).

Параметр ADDRSPC используется при работе с управляющей программой SVS и позволяет задать вид памяти, с которым будет работать система: *реальный* или *виртуальный* (см. п. 7.6). В формате параметра

$$\text{ADDRSPC} = \{ \text{VIRT} \}$$
$$\text{ADDRSPC} = \{ \text{REAL} \}$$

где VIRT указывает, что раздел является частью виртуальной памяти; REAL — обычное (реальное) использование памяти. По умолчанию параметра ADDRSPC система принимает то значение, которое установлено в программе системного ввода RDR при генерации системы. Если оно там опущено, то система принимает его равным VIRT.

Условие прекращения выполнения задания задается с помощью параметра

$$\text{COND} = ((\text{код}, \text{условие}), \dots)$$

Код — это любое десятичное число от 0 до 4095. *Логическое условие* выбирается из набора следующих отношений: GT — больше (>), GE — больше или равно (≥), EQ — равно (=), NE — не равно (≠), LT — меньше (<) и LE — меньше или равно (≤). Для решения вопроса о прекращении выполнения задания система использует код возврата. *Код возврата* — это число, которое устанавливается системой (или программистом) при окончании выполнения очередного пункта задания. Код возврата может принимать любое значение от 0 до 4095. В системных программах используются стандартные коды возврата: 0 — ошибок нет, 4 — возможны ошибки (предостережение), 8 — обнаружены серьезные ошибки, 12 — обнаружены грубые ошибки и дальнейшее выполнение программы невозможно, 16 — обнаружены критические ошибки и нормальное выполнение программы невозможно, 20 — трансляция прекращена, так как обнаружены неустраняемые ошибки ввода-вывода. Планировщик заданий анализирует логическое условие, используя, с одной стороны, код, указанный в параметре COND, и, с другой стороны, код возврата, выдаваемый системой или программой пользователя. Если условие выполняется, то выполнение задания прекращается. Всего в параметре COND может быть указано до восьми условий. Если указывается одно условие, то оно

закрывается в скобки, а внешние скобки опускаются. Пусть в задании PRIM15 в операторе JOB указан параметр COND:

//PRIM15 JOB 121, 'АБРАМОВ', COND=(7,GT)

Пусть при окончании выполнения одного из пунктов этого задания будет выдан код возврата, равный 16. Тогда система будет проверять условие

Код	Условие	Код возврата
7	GT	16

Поскольку данное логическое условие неверно, то планировщик задания передает управление следующему пункту задания. Если, например, будет выдан код возврата, равный 4, то $7 > 4$, т.е. логическое условие выполняется и планировщик задания прекращает выполнение всех последующих пунктов задания. Значение кода возврата выдается программой в зависимости от того, правильно или неправильно закончится пункт задания. Если пункт задания закончился аварийно, то выдается сообщение IEF450I с указанием аварийного окончания пункта задания ABEND и кодом завершения [6]. Введение параметра COND позволяет закончить выполнение задания, как только появился пункт задания с аварийным окончанием.

Передача управления от пункта к пункту в задании возможна и без проверки кодов возврата, что соответствует случаю, когда в операторе JOB отсутствует параметр COND.

Параметр NOTIFY используется, если работает система ДУВЗ (диалоговый удаленный ввод заданий, гл. 12). Пользователь устанавливает связь с ЭВМ с удаленного терминала с помощью команды LOGON, выдаваемой оператором с абонентского пункта (АП). Терминал наиболее часто работает в системе разделения времени, т.е. используется некоторый способ распределения ресурсов ЭВМ во времени между несколькими пользователями. Система с разделением времени предоставляет большинству удаленных пользователей возможность взаимодействовать с ЭВМ одновременно и пошагово, позволяя отлаживать, выполнять программу или получать информацию из системы. Параметр NOTIFY позволяет оператору получить сообщение о завершении задания пользователя, которые выполнялось в пакетном режиме с удаленного терминала. Формат параметра:

NOTIFY=идентификатор пользователя

Идентификатор пользователя — это символ, содержащий до 7 знаков, который указывался в команде LOGON, сообщающей о начале сеанса работы с ЭВМ.

Рестарт заданий. В любой момент времени выполнение задания в системе может закончиться аварийно. Причиной аварийного завершения может быть ошибка в программе пользователя, сбой операционной системы и т. д. Рестарт позволяет повторно выполнить задание с определенного места: с начала любого пункта задания или с контрольной точки. В контрольной точке запоминается состояние наборов данных, необходимых для рестарта задания. Имеется два вида рестарта: *автоматический*, который задается параметром RD, и *отсроченный*, задаваемый параметром RESTART, который добавляется

а оператор JOB для повторно выполняемого задания. Во всех случаях любой вид рестарта может быть выполнен только с разрешения оператора.

Параметр RD позволяет с разрешения оператора повторить выполнение аварийно закончившегося задания с начала пункта задания (рестарт пункта задания), или с контрольной точки (рестарт контрольной точки). *Контрольная точка* устанавливается в программе пользователя с помощью макрокоманды супервизора СНКРТ. Формат параметра RD:

$$RD = \begin{Bmatrix} R \\ RNC \\ NC \\ NR \end{Bmatrix}$$

где R запрашивает автоматический рестарт с начала пункта задания; RNC запрашивает автоматический рестарт с начала пункта задания и подавляет действие макрокоманды СНКРТ; NC — автоматический рестарт не запрашивается и подавляется действие макрокоманды СНКРТ; NR — автоматический рестарт не запрашивается, но разрешается установить контрольную точку с помощью макрокоманды СНКРТ.

Если в параметре RD закодированы значения R и RNC, то оба они запрашивают у оператора разрешения повторно выполнить задание. Однако имеется некоторая разница в их использовании. Значение R запрашивает рестарт с начала пункта задания, если в программе совсем не используется макрокоманда СНКРТ или если используется одна или несколько макрокоманд СНКРТ, но ошибка появилась до начала выполнения макрокоманды СНКРТ, или если после выполнения макрокоманды СНКРТ ее действие подавлено ранее чем будет произведено повторение с контрольной точки. Если ошибка произошла после выполнения макрокоманды СНКРТ, то запрашивается автоматический рестарт, начиная с контрольной точки. Параметр RD=RNC запрашивает автоматический рестарт пункта задания, но, кроме того, запрещает все макрокоманды СНКРТ, которые имеются в программе.

Рестарт не запрашивается, если в параметре RD закодировано NC или NR. Действие параметров RD=NC и RD=NR необходимо рассматривать вместе с макрокомандой СНКРТ. Если в программе используется макрокоманда СНКРТ, то RD=NC действует так, как если бы параметр RD не указывался, а макрокоманды СНКРТ не использовались в программе. Параметр RD=NR не запрашивает автоматический рестарт, но позволяет макрокоманде СНКРТ установить контрольную точку, для того чтобы позже можно было повторить выполнение задания, но не с начала пункта, а с контрольной точки. Если параметр RD опущен и в программе нет макрокоманд СНКРТ, то автоматический рестарт не запрашивается. Если же в программе имеются макрокоманды СНКРТ, то автоматический рестарт запрашивается, несмотря на то, что параметр RD опущен.

При использовании параметра RD необходимо, чтобы пункты задания имели разные имена. Если пункты задания имеют одно и то же имя, то ОС попытается повторить пункт задания, в котором это имя встретилось в первый раз, независимо от того, какой пункт задания окончился аварийно. Кроме того, если запрашивается автоматический рестарт, то в операторе JOB должно быть указано: MSGLEVEL=(1,0), MSGLEVEL=(1,1) или MSGLEVEL=1.

Действия оператора при запросе автоматического рестарта. Если в программе пользователя указан параметр $RD=R$, или $RD=RNC$, то на консоли появляется сообщение, запрашивающее разрешение на автоматический рестарт, например:

*05 IEF225D SHOULD ALLA RESTART

где ALLA — имя задания.

С помощью команды REPLY оператор может дать один из трех возможных ответов. Первый ответ:

R 05,'YES' — автоматическое повторение задания разрешается.

В ответ система снова выдает запрос:

*06 IEF226A LOAD JOB ALLA INTO DEVICE 00C — система требует, чтобы задание было введено в ЭВМ с устройства 00C.

Но задание находится в системе, поэтому оператор должен снова ввести ответ:

R 06,'YES' — задание находится в системе.

Далее нужно выполнять действия в соответствии с запросами ОО.

Второй ответ оператора:

R 05,'NO' — повторение не разрешается оператором. В ответ система обрабатывает диспозицию набора данных так, как если бы рестарт не был запрошен.

Третий ответ оператора:

R 06,'HOLD' — откладывает выполнение задания на некоторое время до тех пор, пока не будет введена команда RELEASE или снято задание по команде CANCEL.

Параметр RESTART добавляется программистом в повторно выполняемое задание, которое ранее закончилось аварийно. При аварийном окончании оператору выдается сообщение, например:

IHJ008I OLGA RESTART — для задания OLGA предусматривается отсроченный рестарт.

Повторить задание можно либо с некоторого пункта задания, либо с контрольной точки, установленной макрокомандой CHKPT. Формат параметра RESTART (сокращенный):

RESTART=*имя пункта*[,*идентификатор контрольной точки*]

При рестарте с определенного пункта задания в параметре RESTART указывается имя пункта задания, которое было установлено при работе макрокоманды CHKPT во время первого выполнения задания. При повторении задания с начала первого пункта имя пункта заменяется символом *, например:

//PROG23 JOB MSGLEVEL=(1,1),RESTART=*

Слова *идентификатор контрольной точки* заменяются именем, которым названа контрольная точка в пункте задания макрокоманды CHKPT, например, CONTR1:

//PROG24 JOB MSGLEVEL=(1,1),RESTART=(*,CONTR1)

Выполнение программы начинается с команды, которая в программе следует непосредственно за макрокомандой CHKPT. При выполнении задания

с контрольной точки в повторно вводимое задание должен быть включен оператор DD, который описывает набор данных контрольных точек и имеет имя SYSDCHK. При повторении задания с начала пункта задания этот оператор DD можно не включать. В повторяемом пункте и в последующих пунктах задания нельзя делать ссылки назад к пунктам, предшествующим повторяемому, используя параметры: PGM и COND в операторе EXEC; SUBALLOС и VOLUME=REF в операторе DD, за исключением случая, когда в параметре VOLUME=SER указан номер тома.

Отсроченный рестарт не требует никаких действий со стороны оператора.

Параметр ROLL (управление сверткой-разверткой) предназначен для выделения дополнительного объема памяти пункту задания за счет того, что временно другой пункт задания передается во вспомогательную память, т.е. производится свертка другого пункта задания. Формат параметра ROLL:

$$\text{ROLL} = (\{ \text{YES} \} \{ \text{YES} \})$$
$$\text{ROLL} = (\{ \text{NO} \} \{ \text{NO} \})$$

Символы YES и NO имеют следующий смысл: YES,NO — каждый пункт задания может быть свернут, пункты задания не могут вызвать свертку другого пункта; YES,YES — каждый пункт задания может быть свернут, каждый пункт задания может вызвать свертку другого пункта задания; NO,YES — пункты задания не могут быть свернуты, каждый пункт задания может вызвать свертку другого пункта; NO,NO — пункты задания не могут быть свернуты, пункты задания не могут вызвать свертку другого пункта.

Если параметр ROLL опущен, то используется значение, установленное в процедуре системного ввода RDR. Ниже приведен пример записи параметра ROLL

```
//PRIM16 JOB 120,'СИЛИН В. М.',ROLL=(YES,YES).
```

Оператор не может изменить значение этого параметра.

9.2. Управляющий оператор EXEC

В этом разделе будет продолжено рассмотрение параметров оператора EXEC, начатое в п. 3.4.

Параметры PGM и PROC являются обязательными при записи оператора EXEC. Простейший формат этих параметров:

$$\{ \text{PGM} = \text{имя программы} \}$$
$$\{ \text{PROC} = \text{имя процедуры} \}$$

Выбирается один из параметров. Параметр PGM вызывает обрабатывающую программу, которая должна быть выполнена в данном пункте задания. Такой программой может быть, например, транслятор, редактор связей, программа пользователя, программа обслуживания и т. д., например:

```
//UT1 EXEC PGM=IEHDASDR
```

где IEHDASDR — имя системной программы обслуживания (гл. 10).

Параметр PROC в основном используется программистами для обращения к каталогизированным процедурам, которые находятся в системной библиотеке SYS.PROCLIB, или к процедурам во входном потоке. Возможны две

формы записи параметра: с указанием ключевого слова PROC и без него, например:

```
//PRIM17 JOB MSGLEVEL=(1,1)
//ASM EXEC PROC=ASMFC
```

где ASMFC — каталогизированная процедура для трансляции программы, написанной на языке ассемблера; ключевое слово PROC оставлено в формате параметра.

Параметр **PARM** имеет в простейшем случае следующий формат:

PARM=значения параметров

Он используется для изменения параметров, установленных в каталогизированных процедурах. Например, для программы обслуживания IEHINITT стандартным параметром является расстояние между заголовками при печати, которое равно 60 строкам. С помощью параметра PARM можно изменить это расстояние

PARM=LINECNT=*nn*

где *nn* заменяется на значение от 00 до 99.

Ниже приведен пример записи начала системной программы обслуживания для разметки магнитных лент:

```
//LENTA JOB 725, 'МАКСИМОВА'
//SMAG EXEC PGM=IEHINITT,PARM=LINECNT=80
```

```

.
.
.
//
```

Если параметр PARM опущен, то используются значения параметров, установленные в каталогизированной процедуре.

Учетная информация для пункта задания задается в простейшем случае параметром ACCT:

ACCT=(*учетная информация,...*)

Общее число значений учетной информации не должно превышать 142. Если в параметре ACCT указывается одно значение, то скобки опускаются, например ACCT=12051. Если какое-либо значение учетной информации содержит специальные символы, кроме дефиса, то он должен быть заключен в скобки, например:

```
//SHAG1 EXEC RGM=SUM1,ACCT=(0016,'OTD=10')
```

Параметр **ADDRSPC** используется при работе с управляющей программой SVS и задает вид памяти для пункта задания: реальный или виртуальный. Формат параметра такой же, как параметр ADDRSPC в операторе JOB (п. 9.1).

Параметр **COND** позволяет обойти некоторые пункты задания. В простейшем случае формат параметра COND:

COND=({ (код,условие)
(код,условие,имя пункта) },..., [EVEN]
[ONLY])

Если выполняется логическое условие

«код»«условие»«код возврата»

то пункт задания обходится. Приведем пример записи параметра COND:

```
//SHAG4 EXEC PGM=SUM1
```

·
·
·

```
//SHAG5 EXEC PGM=SUM2,COND=(8,LT,SHAG4)
```

Во втором операторе EXEC записано следующее условие: если код, равный 8, будет меньше кода возврата, который будет выдан по окончании пункта задания SHAG4, то система должна обойти пункт задания с именем SHAG5. Поскольку в параметре COND записано только одно условие, то внешние скобки в формате опускаются. Ниже приводится пример проверки двух условий:

```
//STEP1 EXEC PGM=SUM1
```

·
·
·

```
//STEP2 EXEC PGM=SUM2,COND=(8,LT,STEP1)
```

·
·
·

```
//STEP3 EXEC PGM=SUM3,COND=(8,LT,STEP1),(4,LT,STEP2)
```

В последнем операторе EXEC записаны два условия: если 8 меньше кода возврата, выданного пунктом задания STEP1, или если 4 меньше кода возврата, выданного пунктом задания STEP2, то следует обойти пункт задания STEP3. Если нужно сделать проверку кодов возврата всех предшествующих пунктов задания, то имя пункта в параметре COND не указывается, например:

```
//STEP1 EXEC PGM=LOGICAL1
```

·
·
·

```
//STEP2 EXEC PGM=LOGICAL2
```

·
·
·

```
//STEP3 EXEC PGM=LOGICAL3,COND=(4,LT)
```

Параметр COND позволяет не только обойти пункт задания, но и продолжить его выполнение, несмотря на то, что один или более пунктов задания закончились аварийно. Пункт считается аварийно законченным, если, например, сбой произошел в программе после того, как она получила управление. Аварийно может закончиться пункт во время планирования, например, из-за ошибок в управляющих предложениях языка управления заданиями или невозможности выделить нужное количество памяти. В аварийных случаях пункты задания обходятся, кроме случая, когда в COND указаны параметры EVEN или ONLY. Если указано EVEN, то это означает, что текущий пункт задания будет выполняться даже в случае аварийного завершения одного или более предшествующих пунктов задания, указанных в COND.

например,

```
//SHAG5 EXEC PGM=LOCAL,COND=(4,LE,SHAG2,EVEN)
```

— выполнить пункт задания SHAG5, даже если пункт задания SHAG2 закончится аварийно.

Если указан ONLY, то это означает, что текущий пункт задания будет выполняться только в случае аварийного завершения одного или более предшествующих пунктов задания, указанных в параметре COND, например:

```
//SHAG6 EXEC PGM=DEV,COND=(ONLY,(8,LT,SHAG2))
```

выполнить пункт задания SHAG6 только в том случае, если пункт задания SHAG2 закончится аварийно.

EVEN и ONLY являются взаимоисключающими параметрами. Они могут стоять перед, между и после условия проверки кодов, как показано в примерах выше.

Параметр RD позволяет с разрешения оператора повторить выполнение аварийно закончившегося задания с начала пункта задания или с контрольной точки, как было описано в п. 9.1.

Параметр ROLL предназначен для выделения дополнительного объема памяти пункту задания за счет того, что временно другой пункт задания передается во вспомогательную память, т.е. производится свертка другого пункта задания. Формат параметра ROLL полностью совпадает с форматом параметра ROLL, рассмотренным для оператора JOB в п. 9.1.

9.3. Управляющий оператор DD

В этом разделе будет продолжено описание оператора DD, которое начато в п. 3.5. Полный перечень параметров оператора DD приведен в табл. 3.4.

Связь с управляющей программой обеспечивают параметры * или DATA, COPIES, DUMMY, DYNAM, FCB, TERM, OUTLIM, UCS и DLM.

Позиционный параметр * сообщает о том, что далее следует набор данных во входном потоке. Набор данных не должен содержать внутри себя никаких управляющих операторов. Ниже приведен пример задания во входном потоке:

```
//PRIM JOB MSGLEVEL=(1,1)
//      EXEC PGM=IEHDASDR
//SYSIN DD      *
:
:
```

Параметры системной обслуживающей программы IEHDASDR

```
//
```

Параметры системной обслуживающей программы IEHDASDR являются набором данных во входном потоке (гл. 10).

Позиционный параметр DATA сообщает о том, что далее следует набор данных, который содержит управляющие операторы языка управления зада-

ниями (// в колонках 1 и 2). В остальном он аналогичен действию параметра *.

Параметр **COPIES** позволяет получить несколько копий выходных наборов данных. Формат параметра **COPIES**:

COPIES = *nnn*

где *nnn* — количество копий наборов данных, определенных параметром **SYSOUT**. По умолчанию выдается одна копия.

Позиционный параметр **DUMMY** обеспечивает обход операции ввода-вывода, т.е. система не выделяет внешнюю память и не устанавливает диспозиции наборов данных. Набор данных, который описывается оператором **DD**, с включением в него параметра **DUMMY**, по существу является фиктивным. Обход операций ввода-вывода в некоторых случаях приводит к экономии времени отладки сложной программы. Приведем пример использования параметра **DUMMY**:

//ASM.SYSPRINT DD DUMMY

для выходного набора данных **SYSPRINT** на шаге **ASM** не выделяются устройства ввода-вывода, т.е. отменяется выдача сообщений результата трансляции на АЦПУ.

Параметр **DYNAM** позволяет отложить установку тома до тех пор, пока он действительно не потребуется заданию. Пример использования параметра **DYNAM**:

//PTOM DD DYNAM

Параметр **FCB** определяет формат страниц для печати выходного набора данных (на устройстве **EC-7039**, АЦПУ). Формат параметра **FCB**:

FCB = (идентификатор [, **ALIGN**] [, **VERIFY**])

Идентификатор определяет формат страниц, должен начинаться с буквы и может содержать от 1 до 4 алфавитно-цифровых символов. Если указан параметр **VERIFY**, то система выдает запрос проверить визуально формат страниц. Параметр **ALIGN** запрашивает проверку формата перед печатью данных, например:

FCB = (АСРУ, **ALIGN**)

Параметр **TEPM** = **TS** запрашивает ввод-вывод через абонентский пункт (АП) в режиме разделения времени.

Параметр **OUTLIM** ограничивает число логических записей, которое выводится в выходном наборе данных. Формат параметра **OUTLIM**:

OUTLIM = *число*

где *число* — это максимальное число логических записей, которое может быть включено в выходной набор данных.

Параметр **UCS** задает набор знаков, используемых при печати набора данных на устройстве **EC-7039**. Формат параметра **UCS**:

UCS = (идентификатор [**FOLD**] [**VERIFY**])

Идентификатор определяется так же, как в параметре FCB; FOLD позволяет печатать прописные буквы, как строчные; VERIFY требует, чтобы оператор визуально проверил используемый при печати набор знаков. Пример записи параметра UCS:

UCS=(ACPY,,VERIFY)

Параметр DLM позволяет ввести новый ограничитель данных во входном потоке, отличный от /*. Формат параметра DLM:

DLM=*ограничитель*

где *ограничитель* — это два знака.

Использование каналов. Параметр AFF позволяет осуществить передачу наборов данных через тот же канал, который используется другим набором данных. Параметр AFF в простейшем случае имеет следующий формат:

AFF=*имя*dd

где *имя*dd должно быть определено ранее, например:

```
//SYSUT2 DD DSNAME=&SYSUT2,UNIT=SYSSQ,  
//          SPACE=(1700,(400,50))  
//SYSUT3 DD DSNAME=&SYSUT3,UNIT=SYSSQ,  
//          SPACE=(1700,(400,50)),AFF=SYSUT2
```

В этом примере в DD-операторе SYSUT3 будет использоваться тот же канал, который был использован набором данных, описанным в DD SYSUT2. Наборы данных должны быть определены в одном пункте задания. Возможна другая форма записи параметра AFF=SYSUT2 не в наборе данных SYSUT3, в отдельном операторе DD:

```
// DD AFF=SYSUT2
```

Параметр AFF следует использовать только в том случае, если это приведет к экономии времени. Следует отметить, что параметр AFF ограничивает возможности планировщика заданий в отношении распределения устройств и может привести к ненужному демонтажу томов. Если канал занят, то выдается запрос оператору. Удовлетворить запрос оператор может, если система имеет достаточное число свободных каналов. Оператор может присоединить к системе устройство, уничтожить запрос, уничтожить задание или ответить системе, что надо ждать до тех пор, пока устройство освободится. Если параметр AFF опущен, то система сама назначает любой доступный канал, который удовлетворяет требованию параметра UNIT.

Параметр SEP запрашивает раздельное использование каналов для наборов данных. В формате параметра

SEP=(*имя*dd,...)

можно указать до восьми имен DD операторов, для которых запрашиваются раздельные каналы. Ниже приводится пример использования параметра SEP в программе IETASM:

```
//ASM          EXEC PGM=IETASM  
//SYSLIB       DD DSNAME=SYS1.MACLIB,DISP=OLD  
//SYSUT1       DD UNIT=SYSSQ,SPACE=(400,(400,50))
```

```
//SYSUT2      DD UNIT=SYSSQ, SPACE=(400,(400,50))
//SYSUT3      DD UNIT=(SYSSQ, SEP=(SYSUT2, SYSUT1,
//              SYSLIB)), SPACE=(400,(400,50))
//SYSPRINT    DD SYSOUT=A
//SYSPRINT    DD UNIT=SYSCP
```

В операторе DD с именем SYSUT3 запрошен канал, который не должен совпадать с каналами, через которые передаются наборы данных с именами SYSUT2, SYSUT1 и SYSLIB.

Параметр QNAME используется при передаче наборов данных с использованием общего телекоммуникационного метода доступа. В формате параметра

QNAME=имя

указывается то же имя, что и в подполе названия макрокоманды TPROCESS, которая ставит сообщения в очередь.

Организация наборов данных. Параметр DCB используется при записи и поиске наборов данных. Формат параметра DCB:

DCB=(список характеристик)

где *список характеристик* насчитывает 20 ключевых подпараметров, которые в основном используются программистами. Для операторов наибольший интерес представляют два подпараметра: DEN и STACK. Подпараметр DEN задает плотность записи на дорожках диска: для 8 зн./мм — DEN=0, для 32 зн./мм — DEN=2 и для 64 зн./мм — DEN=3. Подпараметр STACK задает номер кармана в перфореторе или устройстве ввода с перфокарт: номер кармана 1 — STACK=1, номер кармана 2 — STACK=2. Подпараметры в DCB разделяются запятыми и заключаются в одни общие скобки, если их более одного. Если в DCB указывается один подпараметр, то общие скобки опускаются, например:

DCB=DEN=2

Параметр DDNAME позволяет продолжить описание набора данных в последующих операторах DD того же задания. Формат параметра DDNAME:

DDNAME=имяdd

где *имяdd* заменяется именем оператора DD, в котором будет продолжено описание набора данных, например:

//D1 DD DDNAME=DDOUT

Параметр DISP содержит информацию о состоянии набора данных (новый, старый, модифицируемый) и о том, что нужно сделать с набором данных по окончании выполнения пункта задания в случаях нормального и аварийного окончаний. Формат параметра DISP:

DISP=([состояние][, диспозиция][, условная диспозиция])

Первый параметр *состояние* определяет состояние набора данных по отношению к пункту задания: MOD — последовательный набор данных модифицируется; NEW — новый, выходной набор данных; OLD — старый (т.е. уже существующий) набор данных; SHR — старый набор данных, который можно одновременно использовать в нескольких заданиях (используется в условиях мультипрограммирования).

Второй подпараметр *диспозиция* может иметь одно из пяти значений: CATLG — каталогизировать набор данных в конце пункта задания; DELETE — уничтожить набор данных в конце пункта задания; KEEP — сохранить набор данных после окончания выполнения пункта задания; PASS — передать набор данных следующему пункту задания; UNCATLG — вывести набор данных из каталога в конце пункта задания.

Третий подпараметр *условная диспозиция* может иметь одно из четырех значений, с помощью которых сообщается, что надо сделать с набором данных в случае аварийного окончания пункта задания: CATLG — каталогизировать, DELETE — уничтожить, KEEP — сохранить, UNCATLG — вывести из каталога. Третий подпараметр может быть добавлен в соответствии со смысловым значением. Параметр DISP можно опустить, если набор данных создается заново (NEW) и уничтожается в конце пункта задания (DELETE). Ниже приведены примеры записи параметра DISP:

DISP=OLD — старый набор данных;

DISP=(NEW,PASS) — новый набор данных передать следующему пункту задания.

В параметре LABEL описываются метки и другие данные. Формат параметра LABEL:

$$\text{LABEL} = (\text{номер}[\text{,тип}][\text{,PASSWORD}][\text{,IN}][\text{,EXPDT}=\text{yy.ddd}][\text{,NOPWREAD}][\text{,OUT}][\text{,PETPDT}=\text{nnnn}])$$

где *номер* — это порядковый номер набора данных; слово *тип* заменяется на один из следующих: SL — набор данных имеет стандартные метки; SUL — набор данных имеет как стандартные, так и нестандартные метки; NSL — набор данных имеет нестандартные метки; NL — набор данных не имеет меток; BLP — опустить обработку метки.

Подпараметры PASSWORD и NOPWREAD обеспечивают защиту создаваемого набора данных с помощью пароля. *Пароль* представляет собой символическое имя, которое присваивается набору данных, и является секретным, в отличие от идентификатора. Без указания пароля в ответах на сообщения в дальнейшем нельзя обратиться к набору данных. Подпараметр PASSWORD обеспечивает защиту набора данных при чтении и при записи в набор данных, а NOPWREAD — только при записи.

Подпараметры IN и OUT сообщают о том, что набор данных вводится или выводится соответственно. Подпараметр EXPDT задает срок хранения набора данных: *yy* — год и *ddd* — день. Подпараметр PETPDT сообщает о периоде хранения, где *nnnn* заменяется числом дней хранения. Ниже приводится пример записи параметра LABEL для набора данных, который хранится на магнитной ленте, имеет порядковый номер 3 и должен быть сохранен в течение 100 дней:

LABEL=(3,,IN,PETPDT=100) — набор данных является входным, со стандартными метками (SL опущено), без пароля. Если параметр LABEL опущен, то это значит, что метки соответствуют стандартным.

Параметр SPACE позволяет выделить память на томе с прямым доступом в любом, удобном для ОС месте. Формат параметра SPACE:

$$\text{SPACE} = \left(\left\{ \begin{array}{l} \text{TRK} \\ \text{CYL} \\ \text{число} \end{array} \right\} [\text{, (количество [,приращение] [,справочник])}] \right)$$

где TRK — запрос памяти в дорожках, CYL — запрос памяти в цилиндрах, *число* — средняя длина блока в байтах; *количество* заменяется количеством памяти в выбранных единицах (дорожках, цилиндрах или блоках); *приращение* заменяется количеством дополнительной памяти сверх той, которая указана во втором параметре; *справочник* заменяется числом 256-байтовых записей для оглавления библиотечного набора данных. Ниже приводятся примеры записи параметра SPACE:

SPACE=(1700,(400,10)) — запрос в блоках: средняя длина блока 1700 байт, 400 блоков и по 10 дополнительных блоков (до 15 раз);

SPACE=(TRK,100) — запрос 100 дорожек;

SPACE=(CYL,10) — запрос 10 цилиндров;

SPACE=(TRK,200,10,5) — запрос 200 дорожек и по 10 дополнительных дорожек (до 15 раз), 5 блоков для оглавления библиотечного набора данных;

SPACE=(CYL,(10,5)) — запрос 10 цилиндров и по 5 дополнительных цилиндров (до 15 раз).

Параметр SPLIT позволяет распределить цилиндры между несколькими наборами данных. В операторе DD для первого набора данных указывается вся память, которая будет использоваться всеми наборами данных. Во втором и последующих операторах DD указывается часть памяти, которая необходима для этого набора данных. Формат параметра SPLIT (сокращенный):

SPLIT=*n*

где *n* — число запрашиваемых дорожек на цилиндре, например:

SPLIT=25

Такой способ распределения памяти позволяет в случае, если какой-либо набор данных исключается, использовать его память.

Параметр SUBALLOC используется совместно с параметром SPACE, который выделяет память для всех наборов данных, а параметр SUBALLOC запрашивает часть этой памяти. Формат параметра SUBALLOG:

$$\text{SUBALLOG} = \left(\left\{ \begin{array}{l} \text{TRK} \\ \text{CYL} \\ \text{число} \end{array} \right\}, (\text{количество} [\text{,приращение}]) \right)$$

где TRK, CYL, *число*, *количество* и *приращение* имеют то же назначение, что и в параметре SPACE. Пример записи параметра SUBALLOG:

//DD1 DD SPACE=(TRK,100)

основной запрос 100 дорожек

// DD SUBALLOG=(TRK,20)

выделение 20 дорожек из 100 основных.

Программы обслуживания*) представляют собой стандартные программы, которые выполняют сервисные функции ОС. По своему назначению программы обслуживания делятся на автономные, системные и наборов данных. Автономные программы обслуживания являются самостоятельными программами. Системные программы обслуживания и программы обслуживания наборов данных работают под управлением ОС и находятся в библиотеке SYS1.LINKLIB. Название автономных программ обслуживания начинается с идентификатора IBC, системных — с IEN и наборов данных — с IEB. Полный перечень программ обслуживания приведен в Приложении 11. Ниже кратко будут рассмотрены программы обслуживания IBCDASDI, IBCDMPRS, IENDASDR, IENHITT. Для каждой программы обслуживания требуется не менее 50K оперативной памяти.

10.1. Автономные программы обслуживания IBCDASDI, IBCDMPRS

Сгенерированная ОС чаще всего хранится на МЛ, которая передается пользователю вместе с техническими средствами ЭВМ. После получения магнитной ленты ОС переписывается на том прямого доступа, с которым в дальнейшем постоянно ведется работа. Чтобы переписать ОС на том прямого доступа, его сначала надо подготовить, т. е. проинициализировать, или, как говорят, разметить. Эта работа выполняется с помощью автономной программы IBCDASDI. После того как том будет проинициализирован, на него переписывают (восстанавливают) ОС с магнитной ленты. Сами автономные программы могут находиться на перфокартах или магнитной ленте, или томе прямого доступа. Они состоят из двух частей: загрузчика и самой программы. Ниже рассмотрим случай работы с программой, когда обе части ее находятся на одном и том же носителе — перфокартах.

10.1.1. Ввод автономной программы. На устройстве считывания с перфокарт установить колоду с перфокартами с автономной программой, нажать кнопки «Пуск» и «КТ». На пульте процессора с помощью переключателей следует установить адрес считывающего устройства (см. п. 7.1.2) и нажать кнопку «Загрузка». После нажатия кнопки «Загрузка» начинается считывание автономной программы в память и на консоли появляется запрос

*00 IBC105A DEFINE INPUT DEVICE — указать физический адрес считывающего устройства с перфокарт.

Оператор должен дать ответ, например:

R 00,'INPUT=6012' — адрес считывающего устройства 6012.

*) Ранее эти программы назывались утилитами.

Ответ следует закончить нажатием кнопки «КТ» на пишущей машинке. После того как программа будет помещена в память, процессор перейдет в состояние «ожидание», а последнее полуслово PSW будет содержать число FFFF₁₆.

10.1.2. Общие сведения об управляющих операторах при работе с автономными программами обслуживания. Обращение к автономной программе, когда она находится в памяти, производится с помощью управляющих операторов, которые должны располагаться в следующей последовательности:

Название	Операция	Операнды
[имя]	JOB MSG	'текст' TODEV = xxxx, TOADDR = ddd
	Операторы автономной программы обслуживания	
	END	

Оператор JOB сообщает системе о начале задания. В подполе операндов оператора JOB можно записать любой поясняющий «текст» с помощью символов ДКОИ. В операторе MSG сообщается типовое имя xxxx и физический адрес устройства ddd, на которое система будет выводить сообщения о работе программы, например: TODEV = 7030, TOADDR = 00F. Оператор MSG позволяет, например, получить сообщения о коде возврата, номерах сбойных дорожек и другую информацию. В случае успешного окончания работы автономной программы на устройство, указанное в MSG, будет выдано сообщение:

IBC163A END OF JOB — задание окончилось нормально и процессор переходит в состояние «ожидания». После MSG записываются операторы автономной программы, которые описывают те конкретные функции, которые она должна выполнить. Задание заканчивается оператором END. Одно задание обрабатывает один том.

10.1.3. Автономная программа IBCDASDI предназначена для проверки (инициализации) тома прямого доступа и назначении запасных (альтернативных) дорожек в случае выхода из строя основных дорожек. Полный перечень операторов и их параметров приведен в Приложении 12. Ниже кратко рассмотрим операторы DADEF, VLD, VTOCD и GETALT.

Оператор DADEF определяет том прямого доступа, который проверяется. Во время проверки на каждую дорожку записывается, а затем считывается стандартная комбинация сигналов для выявления дефектов. Если дефектов нет, то дорожка размечается стандартным способом (рис. 10.1). Если имеются дефектные дорожки, то они заменяются альтернативными. Если на томе окажется более семи дефектных дорожек, то такой том можно использовать только как рабочий. Желательно, чтобы резидентный том не имел дефектных дорожек. На рис. 10.1 показано, что на каждой дорожке помещается собственный адрес и нулевая запись R0. На нулевую дорожку помещаются записи IPL (R1 и R2) и метка тома R3. Остальное место резервируется для дополнительных меток. На первой дорожке помещается текст программы IPL (только в резидентном томе). Начиная со второй дорожки, можно размечать

таблицу оглавления тома VTOC. На остальных дорожках будут размещаться наборы данных.

Здесь приводится формат оператора DADEF для случая проверки тома с анализом поверхности дорожек:

Название	Операция	Параметры
[имя]	DADEF	$\text{TODEV}=\text{xxxx}, \text{TOADDR}=\text{ddd} \left\{ \begin{array}{l} \text{.VOLID}=\text{SCRATCH} \\ \text{.VOLID}=\text{номер} \end{array} \right\}$ $[, \text{IPL}=\text{YES}] [, \text{FLAGTEST}=\text{NO}] [, \text{PASSES}=\text{n}]$

В подполе названия записывается любое, произвольно выбранное символическое имя, которое используется в сообщениях. В параметре TODEV указывается типовое имя устройства прямого доступа, например TODEV=5050.

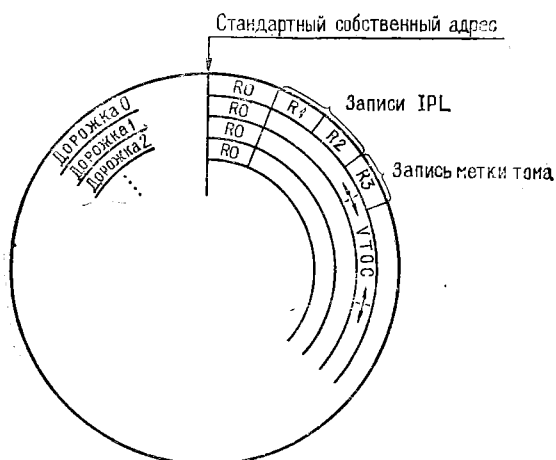


Рис. 10.1.

В параметре TOADDR записывается физический адрес устройства, например TOADDR=130. Параметр VOLID=SCRATSH указывает, что проверять серийный номер не следует (параметр обязателен в случае первичной инициализации). При повторной инициализации в параметре VOLID сообщается серийный номер тома, чтобы по ошибке не попал другой том. Параметр IPL= YES показывает, что на этот том должна быть помещена программа IPL, т. е. том будет использоваться как резидентный. Параметр FLAGTEST=NO обеспечивает полный анализ поверхности дорожек независимо от того, были ли дорожки на томе ранее отмечены как дефектные. Параметр всегда задается при первичной инициализации. По умолчанию этого параметра проверка дефектности дорожек выполняется в зависимости от состояния битов дефектности, т. е. если имеются дефектные дорожки, то они заново не проверяются. Параметр PASSES показывает, сколько раз должна быть повторена проверка каждой дорожки на дефектность. Значение n выбирается от 1 до 255. Если этот параметр опущен, то дорожка проверяется только один раз.

Пример 10.1. Проинициализировать том, который имеет серийный регистрационный номер PTOM1 и установлен на устройстве 130, с проверкой дефектности дорожек:

```
INIT JOB 'ИНИЦИАЛИЗАЦИЯ ДИСКА 130'
MSG   TODEV=7030,TOADDR=00F
DADEF TODEV=5050,TOADDR=130,VOLID=PTOM1,
                                           FLAGTEST=NO
END
```

Если том не имеет дефектных дорожек и работа автономной программы завершена нормально, то система выдает сообщение IBC163A.

Оператор VLD используется для того, чтобы присвоить тому новый серийный регистрационный номер при первичной или повторной инициализации. В формате VLD показан только один обязательный параметр.

Название	Операция	Параметры
[имя]	VLD	NEWVOLID=номер

В параметре NEWVOLID указывается новый серийный регистрационный номер тома.

Оператор VTOCD предназначен для выделения места под оглавление тома. Оператор имеет два обязательных параметра:

Название	Операция	Параметры
[имя]	VTOCD	STRTADR=nnnnn,EXTENT=nnnnn

В параметре STRTADR указывается номер дорожки (от 1 до 5 байт), начиная с которой должно располагаться оглавление тома. В параметре EXTENT задается количество дорожек, отводимое под оглавление, например STRTADR=5, EXTENT=10. Номер дорожки и количество дорожек записываются десятичными цифрами. Оглавление не рекомендуется размещать на дорожке 0 и на альтернативных дорожках.

Пример 10.2. Проинициализировать том, который имеет серийный регистрационный номер PTOM1 и установлен на устройстве 130, с проверкой дорожек на дефектность. Присвоить тому новый номер PTOM2. Оглавление тома начать с дорожки 5 и выделить для оглавления 10 дорожек. Сообщения о работе программы выдать на АЦПУ, адрес которого 00E. Программа имеет следующий вид:

```
INIT JOB 'INITIALIZATION DISKA 130'
MSG   TODEV=7030,TOADDR=00E
DADEF TODEV=5050,TOADDR=130,VOLID=PTOM1,
                                           FLAGTEST=NO
VLD   NEWVOLID=PTOM2
VTOCD STRTADR=5,EXTENT=10
END
```

В случае успешного выполнения программы система выдает сообщение IBC163A.

Пример 10.3. Выполнить первоначальную инициализацию тома с проверкой на дефектность дорожек. Том находится на устройстве 132 (типовое имя 5052). Тому присвоить имя PTOM1. Оглавление тома начать с дорожки 5 и под оглавление отвести 10 дорожек. Все сообщения о работе программы вывести на пишущую машинку с адресом 01F (типовое имя 7070). Программа запишется следующим образом:

```
INIT2 JOB    'DISK132'
      MSG    TODEV=7070,TOADDR=01F
      DADEF  TODEV=5052,TOADDR=132,VOLID=SCRATCH,      *
            FLAGTEST=NO
      VLD    NEWVOLID=PTOM1
      VTOCD  STRTADR=5,EXTENT=10
      END
```

Успешное выполнение программы заканчивается выдачей сообщения IBC163A.

Оператор GETALT позволяет заменить дефектную дорожку альтернативной. Список дефектных дорожек прилагается к каждому тому, начиная с момента его изготовления. Здесь приведен формат оператора GETALT, который проверяет дорожки и, если имеются дефектные, назначает вместо них альтернативные дорожки:

Название	Операция	Параметры
[имя]	GETALT	TODEV=xxx, TOADDR=ddd, TRACK=cccchhh VOLID=номер

Параметр TODEV задает типовое имя устройства, на котором находится том прямого доступа; параметр TOADDR — физический адрес устройства; параметр TRACK — номер дорожки, для которой запрашивается альтернативная дорожка. В параметре TRACK указывается cccc — номер цилиндра и hhhh — номер дорожки на цилиндре (16 с.с.). Параметр VOLID задает серийный регистрационный номер тома, чтобы исключить ошибку обращения к другому тому.

Пример 10.4. Проверить одну дорожку 004A0009 на томе PTOM2, который находится на устройстве 5050 (физический адрес 130). Если дорожка окажется дефектной, то назначить альтернативную. Программа запишется следующим образом:

```
ALT   JOB    'НАЗНАЧЕНИЕ АЛЬТЕРНАТИВНОЙ ДОРОЖКИ'
      MSG    TODEV=7030,TOADDR=00F
      STMT1  GETALT TODEV=5050,TOADDR=130,TRACK=004A0009,
            VOLID=PTOM2
      END
```

Если дорожка, на которую выдан запрос, не имеет дефекта, то выдается сообщение:

IBC109I TRACK CHK INDICATES TRACK IS GOOD — проверяемая дорожка хорошая.

Если обнаружена дефектная дорожка, то система выдает сообщение:
IBC1101 BAD TRACK 004A0009 — дорожка 004A0009 плохая.

Далее выдается одно из двух сообщений:

IBC1111 ALTERNATE 004B0007 — альтернативная дорожка имеет адрес 004B0007,
или сообщение:

IBC1111 ALTERNATE NONE — свободной альтернативной дорожки на том больше нет.

10.1.4. Программа IBCDMPRS предназначена для дампирования (снятия копии) и восстановления томов прямого доступа. Полный перечень операторов программы IBCDMPRS и их параметров приведен в Приложении 13. Ниже будут рассмотрены операторы DUMP и RESTORE.

Оператор DUMP служит для передачи данных с тома прямого доступа на магнитную ленту или том прямого доступа. Копируются все данные, включая данные альтернативных дорожек. Том прямого доступа, на который помещается копия, должен быть предварительно проинициализирован с помощью программы IBCDASDI и иметь серийный регистрационный номер. Формат оператора DUMP:

Название	Операция	Параметры
[имя]	DUMP	FROMDEV=xxxx, FROMADDR=ddd, TODEV=xxxx, TOADDR=ddd].VOLID=список]

В параметре FROMDEV указывается типовое имя устройства, а в параметре FROMADDR — физический адрес устройства, на котором находится том, с которого снимается копия. В параметре TODEV записывается типовое имя устройства, а в параметре TOADDR — физический адрес устройства, принимающего информацию (ленты или диска). Допустимыми типовыми именами являются 5050, 5056, 5010, 5055. В параметре VOLID указывается список серийных регистрационных номеров томов, чтобы не было ошибки при установке томов, принимающих информацию. Если принимающих томов несколько, то их серийные номера разделяются запятыми и список номеров заключается в общие скобки.

Пример 10.5. Снять копию с тома, находящегося на устройстве типа 5050 (физический адрес 132), на магнитную ленту, находящуюся на устройстве типа 5010 (физический адрес 281). Серийный номер тома PTOM4. Сообщения выдать на пишущую машинку типа 7070 (физический адрес 01F). Программа имеет следующий вид:

```
DUMP1 JOB 'КОПИЯ PTOM4'
MSG TODEV=7070, TOADDR=01F
DUMP FROMDEV=5050, FROMADDR=132,
      TODEV=5010, TOADDR=281, VOLID=PTOM4
END
```

В случае благополучного окончания снятия копии система выдает сообщение IBC163A.

Оператор RESTORE используется для восстановления данных на том прямого доступа с магнитной ленты. Этот оператор используется в случае,

если информация на томе прямого доступа по какой-либо причине испортилась. Формат оператора RESTORE:

Название	Операция	Параметры
[имя]	RESTORE	FROMDEV=xxx, FROMADDR=ddd, TODEV=xxx, TOADDR=ddd[, VOLID=номер]

В параметре FROMDEV указывается типовое имя устройства и в FROMADDR — физический адрес накопителя на магнитной ленте, с которого будет восстанавливаться ОС. В параметре TODEV записывается типовое имя устройства и в TOADDR — физический адрес устройства, на котором находится том, информация на который восстанавливается. В параметре VOLID указывается серийный номер тома, принимающего информацию.

Пример 10.6. Восстановить информацию на том SYSRES, который находится на устройстве типа 5050 (физический адрес 190) с магнитной ленты, которая находится на устройстве типа 5010 (физический адрес 282). Сообщения выдать на устройство типа 7030 (АЦПУ) с физическим адресом 00E. Вид программы:

```
RES JOB      'ВОССТАНОВИТЬ ОС'
MSG          TODEV=7030, TOADDR=00E
RESTORE FROMDEV=5010, FROMADDR=282,
           TODEV=5050, TOADDR=190, VOLID=SYSRES
END
```

В случае успешного выполнения программы система выдает сообщение IBC163A.

10.2. Системные программы обслуживания IENDASDR, IEHINITT

Системные программы обслуживания работают под управлением ОС. Их перечень приведен в Приложении 11. Они находятся в системной библиотеке SYS1.LINKLIB.

10.2.1. Управление системными программами обслуживания производится с помощью операторов языка управления заданиями и операторов управления системными программами обслуживания. Схема составления задания:

Название	Операция	Параметры
//имя // //SYSPRINT //имя ₁ : //имя _n //SYSIN	JOB EXEC DD DD : DD DD	PGM=имя системной программы обслуживания SYSOUT=A Параметры, которые описывают том 1, участвующий в работе системной программы обслуживания Параметры, которые описывают том n, участвующий в работе системной программы обслуживания •
Операторы системной программы обслуживания		
//		

Оператор JOB является началом задания. Оператор EXEC используется для указания в параметре PGM имени системной программы обслуживания. Оператор DD с именем SYSPRINT служит для описания класса системного вывода, в котором будут выдаваться сообщения о результате выполнения программы. Далее в операторах DD с именами 1 — *n* дается описание томов, участвующих в работе. В Приложении 14 приведены параметры оператора DD, которые используются в некоторых наиболее простых случаях.

10.2.2. Программа IEHDSR предназначена для разметки тома прямого доступа, копирования данных с тома прямого доступа на магнитную ленту или том прямого доступа, восстановления данных на том прямого доступа, а также может выполнять некоторые отдельные работы, связанные с подготовкой томов к работе с ОС. В программе используются шесть операторов, приведенных в Приложении 15. Ниже рассмотрим четыре из них: ANALYZE, GETALT, DUMP и RESTORE.

Оператор ANALYZE используется для проверки дорожек томов прямого доступа и разметки тома в стандартном формате (рис. 10.1). Этот оператор во многом аналогичен оператору DADEF независимой утилиты IEHDASDI. Формат ANALYZE приведен для случая, когда программа IPL не записывается на том; том проверяется с помощью стандартной комбинации сигналов один раз:

Название	Операция	Операнды
[имя]	ANALYZE	$\{TODD=(xxxx, \dots)\}, VTOC=nnnn,$ $\{TODD=(имя\ dd, \dots)\},$ $EXTENT=nnnnn, NEWVOLID=номер$ $[FLAGTEST=\{YES\}$ $\quad \quad \quad \{NO\}]$ $[PURGE=\{YES\}$ $\quad \quad \quad \{NO\}] [OWNERID=имя]$

В параметре TODD=(xxxx, ...) указывается физический адрес устройства (адреса устройств), содержащего том, иницируемый в первый раз. Устройство в этом случае должно находиться в автономном состоянии. Если том размечается повторно, то используется параметр TODD=(имяdd, ...), где «имяdd» определяет инициализируемый том (том). Том (тома) описывается DD-операторами, которые помещаются перед управляющими операторами программы обслуживания, как показано в п. 10.2.1.

В параметре VTOC указывается адрес первой дорожки, с которой начинается оглавление тома. Нельзя указывать нулевую дорожку. Дорожку 1 нельзя указывать, если на томе будет находиться текст программы IPL. В параметре EXTENT записывается длина оглавления тома в дорожках. В обоих параметрах VTOC и EXTENT нельзя указывать альтернативные дорожки. Запись числа дорожек производится десятичными числами. В параметре NEWVOLID обязательно указывается серийный регистрационный номер тома, проверяемого и размечаемого в первый раз. Если том проверяется повторно, т. е. том имеет уже серийный номер, то параметр NEWVOLID можно опустить. Параметр FLAGTEST имеет то же назначение, что и в программе IBCDASDI. В параметре OWNERID сообщается имя владельца тома или

любая другая идентифицирующая информация, которая помещается в запись R3 и может иметь до 10 символов кода ДКОИ (нельзя использовать пробелы и запяты). Параметр PURGE=YES показывает, что наборы данных на томе могут быть уничтожены, несмотря на то, что срок хранения их еще не истек. В этом случае система выдает запрос оператору, например:

*12 IEH807D 131 HAS UNEXPIRED DATA SETS, BCT009, UTIL, SHAG1 — в операторе ANALYZE указан параметр PURGE=YES, и программа встретила один или несколько наборов данных, срок хранения которых еще не истек; серийный номер тома BCT009, адрес устройства 131, имя задания UTIL, шаг задания SHAG1. Если оператор разрешает продолжить обработку тома, то он должен ответить:

R 12,'U'

Если работу утилиты следует прекратить, то выдается ответ:

R 12'T'

Операция ANALYZE прекращается, если закодировано PURGE=NO, либо параметр PURGE опущен и на томе встретился набор данных, для которого не истек еще срок хранения.

Пример 10.7. Выполнить первичную инициализацию тома на устройстве 131 и присвоить тому серийный регистрационный номер ACT006. Под оглавление выделить 40 дорожек, начиная с дорожки 10. Обеспечить проверку битов дефектности. Задать диспетчерский приоритет $np_1=10$ и $np_2=5$. Сообщения выдать на АЦПУ, класс — А. Программа запишется следующим образом:

```
//UORONA JOB(,2841,600),'ARXIV',MSGLEVEL=(1,1),TIME=1440
//          EXES PGM=IEHDASDR,DPRTY=(10,5)
//SYSPRINT DD SYSOUT=A
//SYSIN    DD *
           ANALYZE TODD=131,VTOC=00010,EXTENT=00040, *
           FLAGTEST=YES,NEWVOLID=ACT006
//
```

Перед выполнением программы IEHDASDR оператор должен перевести устройство 131 в автономное состояние. Для каждой дефектной дорожки будет автоматически назначена альтернативная, о чем будет выдано два сообщения, например:

IEH831I DEFECTIVE TRACK ON TODD=131 WAS 001E000D — номер дефектной дорожки 001E000D;

IEH832I ALTERNATE TRACK ON TODD=131 IS 00CA000E — номер альтернативной дорожки 00CA000E.

После успешного завершения работы программы IEHDASR система выдает сообщение:

IEH839I HIGHEST RETURN CODE ENCOUNTERED WAS 00 — наивысший код возврата равен нулю.

Пример 10.8. Пронициализировать один том, который находится на устройстве 130 и имеет серийный номер DISK3. Присвоить тому новый серийный номер PTOM3. Оглавление начать с дорожки 3, выделить для оглавления 10 дорожек. Указать имя владельца тома — ПЕТРОВА. Сообщения выдать на АЦПУ. Вид программы:

```
//INIT1    JOB 'ПОВТОРНАЯ ИНИЦИАЛИЗАЦИЯ ДИСКА'
//          EXES PGM=IEHDASDR
```



```
//SYSPRINT DD  SYSOUT=A
//M1 DD UNIT=130,VOL=SER=DISK3,DISP=OLD
//SYSIN DD *
ANALYZE TODD=M1,VTOC=00003,EXTENT=00010, *
NEWVOLID=PTOM3,OWNERID=ПЕТРОВА, *
FLAGTEST=NO
```

//

В операторе DD с именем M1 дано описание тома для случая, когда проверяется один том на одном устройстве (Приложение 14, п. 1). При последовательной проверке нескольких сменных томов параметры оператора DD приведены в Приложении 14, п. 2.

Пример 10.9. Пронумеровать повторно личный том, серийный номер которого неизвестен. Том находится на устройстве 132. Присвоить тому новый серийный номер PTOM12. Оглавление тома начать с дорожки 4 и выделить для оглавления 10 дорожек. Программа имеет следующий вид:

```
//INIT JOB 'ПОВТОРНАЯ ИНИЦИАЛИЗАЦИЯ ДИСКА,
// EXEC PGM=IEHDASDR
//SYSPRINT DD SYSOUT=A
//STDASD DD UNIT=(132,,DEFER),VOL=PRIVATE,
DISP=(NEW,KEEP)
//SYSIN DD *
ANALYZE TODD=STDASD,VTOC=4,EXTENT=10, *
NEWVOLID=PTOM12
```

//

В операторе DD с именем STDASD дано описание тома для случая запроса личного тома, серийный номер которого неизвестен (Приложение 14, п. 3).

Формат оператора LABEL приведен в Приложении 15.

Пример 10.10. Изменить серийный номер тома, который установлен на устройстве 231. Новый серийный номер DP6006. Вид программы:

```
//IN7 JOB (,1051,600),'ARXIV',MSGLEVEL=(1,1)
// EXEC PGM=IEHDASDR
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
LABEL TODD=231,NEWVOLID=DP6006
```

//

Если изменение прошло успешно, то в ответ система выдает сообщение:

```
IEH806I LABEL OF DDNAME=231 IS COMPLETE. VOLUME
SERIAL NO.=DP6006
IEH839I HIGHEST RETURN CODE ENCOUNTERED WAS 00 — наивыс-
```

ший код возврата равен нулю.

Оператор DUMP позволяет снять копию с тома прямого доступа или его части на один или более томов прямого доступа того же типа, на магнитную ленту или на АЦПУ. Полный формат оператора DUMP приведен в Приложении 15. Здесь приводится формат в сокращенном виде:

Название	Операция	Операнды
[имя]	DUMP	FROMDD= имя dd, TODD=(имя dd,...) [.CPYVOLID=[YES NO }] [.BEGIN=ccccchhhh] [.END=ccccchhhh]

В параметре FROMDD записывается имя оператора DD, которое описывает том, с которого копируется информация. В параметре TODD указывается одно или несколько имен операторов DD, описывающих тома, на которые помещаются копии. Параметр CPYVOLID=YES означает, что всем копируемым томам присваивается серийный номер копируемого тома. Если параметр CPYVOLID опущен или указано CPYVOLID=NO, то тома сохраняют свои серийные номера. В параметрах BEGIN и END указываются в 16 с.с. номер цилиндра (cccc) и номер головки (hhhh) соответственно первой и последней дампимых дорожек. По умолчанию этих параметров копия снимается, начиная с первой и кончая последней основной дорожкой. Альтернативные дорожки не дампятся, если они не заменяют основные. С помощью одной программы можно получить до шести копий.

Пример 10.11. Снять копию с тома PTOM5, находящегося на устройстве 130. Принимающий том находится на устройстве 132 и имеет серийный номер PTOM2. Программа запишется следующим образом:

```
//DUMP1      JOB    'ПОЛУЧЕНИЕ КОПИИ НА ТОМЕ
                                     ПРЯМОГО ДОСТУПА'
//           EXEC    PGM=IEHDASDR
//SYSPRINT DD  SYSOUT=A
//DISK1      DD      UNIT=130,VOL=(PRIVATE,,SER=PTOM1),
//           DISP=OLD
//DISK2      DD      UNIT=132,VOL=(PRIVATE,,SER=PTOM2),
//           DISP=OLD
//SYSIN      DD      *
               DUMP FROMDD=DISK1,TODD=DISK2
//
```

Пример 10.12. Снять копию с личного тома прямого доступа с серийным номером DP2008, с отложенным монтированием тома на устройстве 134. Копию передать на магнитную девятидорожечную ленту с плотностью записи 32 зн./мм, серийным номером TL7326 и именем файла DP2008. Магнитная лента находится на устройстве 283. Вид программы:

```
//DP2008PT   JOB    (,1051),'ARXIV',MSGLEVEL=(1,1),TIME=1440
//           EXEC    PGM=IEHDASDR
//SYSPRINT DD  SYSOUT=A
//D06        DD      UNIT (134,,DEFER),DISP=OLD,
//           VOL=(PRIVATE,,SER=(DP2008))
//T06        DD      UNIT=283,DISP=(NEW,PASS),
//           VOL=SER=TL7326,
//           VOL=SER=TL7326,DSN=DP2008,
//           DCB=(DEN=2), LABEL=(,SL)
//
```

```
//SYSIN DD *
      DUMP FROMD=D06,TODD=T06,PURGE=YES
```

```
//
```

В ответ на успешное выполнение программы система выдает сообщения
 IEH806I DUMP TO DDNAME=T06 IS COMPLETE — дамп выполнен;
 имя оператора DD — T06 —

IEH839I HIGHEST RETURN CODE ENCOUNTERED WAS 00 — код возврата нуль.

Пример 10.13. Снять копию с личного тома прямого доступа с серийным номером D00001, установленного на устройстве 134, на девятидорожечную магнитную ленту с плотностью записи 32 зн./мм. Магнитная лента имеет серийный номер L00001 и имя файла TAPE1. Она должна быть установлена на устройстве 283. Программа имеет следующий вид:

```
//DUMP2 JOB 'СНЯТИЕ КОПИИ С ТОМА НА ЛЕНТУ'
// EXEC PGM=IEHDASDR
//SYSPRINT DD SYSOUT=A
//DISK1 DD UNIT=134,DISP=OLD,VOL=SER=D00001,
//LENTA1 DD UNIT=283,DISP=OLD,DSN=TAPE1,
// VOL=SER=L00001,DCB=(DEN=2),
// LABEL=(2,SL)
//SYSIN DD *
      DUMP FROMDD=DISK1,TODD=LENTA1,PURGE=YES
//
```

Оператор DD с именем DISK1 описывает том прямого доступа, с которого снимается копия (Приложение 14, п. 1), а LENTA1 — том на МЛ (Приложение 14, п. 4).

Оператор RESTORE позволяет восстановить данные на том прямого доступа с магнитной ленты. Формат оператора RESTORE по форме полностью совпадает с форматом оператора DUMP. Назначение параметров описано в Приложении 15.

Пример 10.14. Восстановить данные с магнитной ленты: серийный номер TL7326, имя набора данных DP2008, лента девятидорожечная с плотностью записи 32 зн./мм, лентопротяжное устройство 283. Данные восстановить на томе прямого доступа, серийный номер которого D00002, установленном на устройстве 133. Программа запишется следующим образом:

```
//RESTORE1 JOB 'КОПИЯ С МАГНИТНОЙ ЛЕНТЫ НА ДИСК'
// EXEC PGM=IEHDASDR
//SYSPRINT DD SYSOUT=A
//LN2 DD UNIT=283,DISP=(OLD,PASS),
// VOL=SER=TL7326,DSN=DP2008,
// DCB=(DEN=2)
//DS2 DD UNIT=133,DISP=OLD,VOL=SER=D00002
//SYSIN DD *
      RESTORE FROMDD=LN2,TODD=DS2
//
```

В случае успешного выполнения программы система выдает сообщения:

IEH805I RESTORE TO DDNAME=DS2 IS COMPLETE. VOLUME
 SERIAL NO.=D00002

и сообщение IEH839I.

Пример 10.15. Восстановить данные с магнитной ленты: серийный номер L00003, имя набора данных TAPE3, лента девятидорожечная с плотностью записи 32 зн./мм, номер лентопротяжного механизма неизвестен. Копии получить на трех томах с серийными номерами D00003, D00004, D00005, тип устройства 5050 с отложенным монтированием томов. Тома личные. Программа имеет следующий вид:

```
//RESTORE2 JOB 'ВОССТАНОВЛЕНИЕ ДАННЫХ С ЛЕНТЫ'
//          EXEC PGM=IEHDASDR
//SYSPRINT DD SYSOUT=A
//LN3      DD UNIT=(5010,,DEFER),
//          VOL=SER=(L00003,,DEFER),
//          DSN=TAPE3,DCB=(DEN=2),DISP=OLD
//DS3      DD UNIT=(5050,,DEFER),DISP=OLD,
//          VOL=(PRIVATE,,SER=D00003)
//DS4      DD UNIT=(5050,,DEFER),DISP=OLD,
//          VOL=(PRIVATE,,SER=D00004)
//DS5      DO UNIT=(5050,,DEFER),DISP=OLD,
//          VOL=(PRIVATE,,SER=D00005)
//SYSIN    DD *
//          RESTORE FROMDD=LN3,TODD=(DS3,DS4,DS5)
//
```

В случае успешного выполнения система выдает сообщения IEH806I и IEH839I.

10.2.3. Программа IEHINITT служит для стандартной разметки магнитных лент, которая заключается в том, что на нее записывается группа меток тома. Каждый набор меток содержит начальную метку тома VOL1, фиктивную головную метку HDR1, которая представляет собой 80-байтовую запись четырех символов HDR1, 76 пробелов и программный маркер TM (рис. 10.2, а).

а) Разметка ленты

Начальная метка тома	
HDR1	76 пробелов
Ленточный программный маркер TM	

б) Разметка ленты после записи данных

Начальная метка тома	
HDR1	Информация ОС
HDR2	Характеристика набора данных
Головные метки пользователя (до восьми)	
Ленточный программный маркер TM	
Данные	
↓	

Рис. 10.2. Стандартная разметка магнитных лент.

Позже, когда размеченная лента используется для работы, операционная система уничтожает программный маркер, созданный программой IEHINITT, запись HDR1 заполняется данными, кроме того, создается запись HDR2, записываются головные метки пользователя, записывается программный маркер и далее следует набор данных, как показано на рис. 10.2, б.

В программе IEHINITT используется только один оператор INITT:

Название	Операция	Операнды
[имя]	INITT	SER=номер, OWNER='имя', [,NUMTAPE=n] [,DISP={REWING} UNLOAD}]

В подполе названия указывается *имя*, которое должно совпадать с именем, которое указывается в подполе названия оператора DD в управляющем предложении, которое используется перед описанием INIT, как показано в п. 10.2.1. В параметре SER сообщается серийный номер тома для первой или единственной размечаемой ленты.

Если размечаются несколько лент и номер каждой последующей ленты увеличивается на единицу по сравнению с предыдущей, то для них записывается один оператор INITT, в котором указывается только один серийный номер для первой ленты. Для всех последующих лент серийный номер будет формировать система, каждый раз увеличивая на единицу. Максимальное значение серийного номера 999999, за которым следует номер 000000. Для одной размечаемой ленты номер может содержать буквы и цифры; для нескольких — только цифры. В параметре OWNER записывается имя владельца ленты или идентификатор. Максимальная длина имени 10 знаков. Параметр NUMTAPE определяет число магнитных лент. Значение *n* находится в диапазоне от 1 до 255, хотя лентопротяжек 8. Если параметр NUMTAPE опущен, то размечается только один том. Параметр DISP=REWING указывает, что после записи меток ленту следует перемотать, но не разгружать, чтобы не нажимать на кнопку «Перемотка» на самом устройстве. Параметр DISP=UNLOAD означает, что после записи меток том перематывается и разгружается, т.е. после этого остается только снять магнитную ленту с устройства. По умолчанию принимается DISP=UNLOAD. В одном задании можно записать сколько угодно операторов INITT, т.е. произвести разметку любого числа магнитных лент.

Пример 10.16. Разметить одну магнитную ленту, серийный номер которой ML0001. Лента девятидорожечная, плотность записи 32 зн./мм. Мониторинг тома отложено. После разметки подготовить том к снятию. Фамилия владельца ленты ПЕТРОВА. Вид программы:

```
//LENI      JOB  'РАЗМЕТКА ОДНОЙ ЛЕНТЫ'
//          EXEC PGM=IEHINITT
//SYSPRINT DD  SYSOUT=A
//ML1       DD   UNIT=(5010,,DEFER),DCB=(DEN=2)
//SYSIN     DD   *
ML1         INITT SER=ML0001,OWNER='ПЕТРОВА'
//
```

В случае успешного выполнения программы система выдает сообщения:

```
SYSTEM SUPPORT UTILITIES IEHINITT
ML1 INITT SER=ML0001, OWNER= 'ПЕТРОВА',
VOL1ML00010 P.T. PETRO
```

где ML0001 означает, что начальная метка тома успешно записана на том; ML0001 — серийный номер тома; 0 — том не секретный. Во время разметки тома ошибок не было.

Пример 10.17. Разметить три магнитные ленты и присвоить им серийные номера 000001, 000002, 000003. Лента девятидорожечная, плотность записи 32 зн./мм. Монтирование томов отложено. Работает одно лентопротяжное устройство. После разметки подготовить том к снятию. Программа запишется следующим образом:

```
//LEN2      JOB  'РАЗМЕТКА ТРЕХ ЛЕНТ'  
//EXEC      PGM=IEHINITT  
//SYSPRINT DD  SYSOUT=A  
//ML2       DD  UNIT=(5010,1,DEFER),DCB=(DEN=2)  
//SYSIN     DD  *  
ML2        INITT SER=000001,NUMBTAPE=3  
//
```

Пример 10.18. Одну магнитную ленту разметить и присвоить ей серийный номер ML3. Лента находится на устройстве 280. Разметить еще пять магнитных лент и присвоить им номера от 000001 до 000005. Для разметки лент использовать три лентопротяжных устройства; монтирование томов отложенное. Тома после разметки подготовить к снятию, кроме тома ML3. Ленты девятидорожечные, плотность записи — 32 зн./мм. Вид программы:

```
//LEN3      JOB  'РАЗМЕТКА НЕСКОЛЬКИХ ТОМОВ'  
//          EXEC PGM=IEHINITT  
//SYSPRINT DD  SYSOUT=A  
//ML3       DD  UNIT=280,DCB=(DEN=2)  
//ML4       DD  UNIT=(5010,3,DEFER),DCB=(DEN=2)  
//SYSIN     DD  *  
ML3        INITT SER=ML3,DISP=REWIND  
ML4        INITT SER=000001,NUMBTAPE=5  
//
```

Глава 11. КОМАНДЫ ДИСПЛЕЙ-КОНСОЛИ И УПРАВЛЕНИЯ РЕЖИМАМИ КОНСОЛЕЙ

Оператор работает с ЭВМ с помощью консолей. К консолям относятся пишущие машины, алфавитно-цифровые и графические дисплеи, совместно используемые устройства ввода с перфокарт и вывода информации на печать (АЦПУ). При мультиконсольном режиме работы каждой консоли присваивается характеристика: основная (главная) консоль или альтернативная (запасная, резервная) консоль. Если основная или альтернативная консоль состоит из нескольких устройств, то она называется составной.

Основная консоль отмечается в сообщениях буквой *M*, например, в сообщении IEE140I, в колонке COND, которая часто выдается при загрузке ОС. С основной консоли разрешается выдавать все команды оператора и можно получить все виды сообщений. В процессе работы оператор может переопределить основную консоль. В случае выхода из строя основной консоли система переходит на альтернативную консоль. Характеристику консолей можно увидеть в сообщении IEE140I в колонке CONSOLE/ALT, например, как показано на рис. 7.5, в третьей строке сверху: пишущая машинка 01F является основной консолью, а алфавитно-цифровой дисплей 060 — альтернативной. С альтернативной консоли можно вводить не все команды оператора и на нее выдаются не все сообщения. Если в системе используется составная консоль, то адреса входящих в нее устройств перечисляются через запятую, как, например, показано на рис. 7.5: 00C,00F — составная консоль состоит из двух устройств; устройства ввода с перфокарт 00C и АЦПУ с адресом 00F. Пишущая машинка и АЦПУ обеспечивают получение копии текстов команд и сообщений на бумаге (или «твердой копии» — HARD COPY). Алфавитно-цифровой дисплей (далее называемый дисплей-консолью) позволяет получать визуальную информацию. Дисплей-консоль обладает рядом преимуществ по сравнению с пишущей машинкой. Так, например, она позволяет оператору исправить команду во время набора ее на клавиатуре, если была допущена ошибка. Одним из достоинств является быстрый вывод сообщений на экран. Параметры экрана и размещение информации на нем описаны в параграфе 2.2.

Назначение команд. Команды, которые используются при работе с консолями, по функциям можно разделить на две группы:

1. MONITOR, STOPMN, DISPLAY, CONTROL. MSGRT обеспечивают информационную связь с системой; команда CONTROL позволяет дополнительно устанавливать режим работы дисплей-консоли;

2. VARY предназначена для управления консолями.

11.1. Информационные команды

Команда MONITOR позволяет получить на экране дисплей-консоли динамическую информацию о выполняющихся заданиях. Команда STOPMN отменяет выдачу такой информации.

Форматы этих команд:

Команда	Параметры
$\{\text{MONITOR}\}$ $\{\text{MN}\}$	$A \left[.L = \begin{Bmatrix} a \\ cc \\ cca \end{Bmatrix} \right] ^*)$
$\{\text{STOPMN}\}$ $\{\text{PM}\}$	$A \left[.L = \begin{Bmatrix} a \\ cc \\ cca \end{Bmatrix} \right] ^*)$
*) Далее вводится обозначение $L=x$, где $x=a$, $x=cc$ или $x=cca$.	

В команде MONITOR символ A обозначает, что система должна обеспечить вывод информации на дисплей, а в команде STORMN — отменить выдачу информации. При вводе команды

MN A

появляется сообщение IEE102I такое же, как при выдаче команды D A (см. п. 4.1.1). По команде

PM A

эта информация снимается с экрана дисплей-консоли.

В параметре L можно указать: a — идентификатор области информации состояния от A до Z, cc — номер консоли, cca — номер консоли и области информации состояния, например:

MN A,L=12A — 12 является номером консоли, который берется из сообщения IEE140I или IEE250I в колонке ID; A — идентификатор области информации состояния.

Команда DISPLAY обеспечивает оператора информацией о состоянии системы. Формат команды DISPLAY:

Команда	Параметры	Примечание
$\{\text{DISPLAY}\}$ $\{\text{D}\}$	$\{\text{CONSOLES}\}$ $\{\text{C,K}\}$ $\{\text{PFK}\}$	В версии 6.1 параметр не используется

Информация о конфигурации системных консолей выдается по команде

D CONSOLES

Например:

```

IEE250I      CONSOLES
CONSOLE/ALT COND AUTH ID AREA ROUTCD
SYSLOG      H      CMDS
060/0C0      N      ALL 01 Z,A  1—11,16
0C0/01F      N      ALL 02 Z,A  1—11,16
01F/060      M      ALL 03      1—11,16
  
```


Это сообщение аналогично сообщению, выдаваемому системой при загрузке системы IEE140I (см. рис. 7.5).

Информация о возможностях команды CONTROL (сокращенно K) вы-
дается по команде

D C,K

Система в ответ на ввод команды выдает сообщение

IEE162I 10.18.22 K COMMAND 084

K	D,N	NUMBER MESSAGE LINES UNTIL DELETION REQUESTED
	,N,HOLD	NUMBER MESSAGE LINES ALL THE TIME
	,F	MOVE STATUS DISPLAY FORWARD ONE FRAME
	,H, HOLD	DYNAMIC DISPLAY—DO NOT UPDATE
	,U	UPDATE DYNAMIC DISPLAY
	,PFK	DISPLAY PROGRAM FUNCTION KEY LINE
K	E,SEG	ERASE MESSAGE LINES SPECIFIED IN SEG=NN OPTION
	,NN,NN	ERASE MESSAGE LINE NN OR NN THROUGH NN
	, F	ERASE FLAGGED MESSAGES
	, D	ERASE OUT—OF—LINE STATUS DISPLAY
	, N	ERASE MESSAGE LINE NUMBERS
	,PFK	ERASE PROGRAM FUNCTION KEY LINE
K	,S,REF	DISPLAY ALL SPECIFICATION VALUES
	,DEL=Y	DELETE MESSAGES AUTOMATICALLY
	,DEL=N	DO NOT DELETE MESSAGES AUTOMATICALLY
	,DEL=R	ROLL ALL MESSAGES
	,DEL=RD	ROLL ONLY DELETABLE MESSAGES
	,CON=Y	DELETE MESSAGES WITH VERIFICATION
	,CON=N	DELETE MESSAGES WITHOUT VERIFICATION
	,SEG=NN	DELETE NN LINES WHEN K E,SEG IS ENTERED
	,RNUM=NN	ROLL NN MESSAGE (DEL=R OR RD)
	,RTME=NNN	ROLL MESSAGE EVERY NNN SECONDS (DEL=R OR RD)
K	C,D,ID	CANCEL IN—LINE DISPLAY NUMBERED ID
K	M,REF	DISPLAY MASTER CONSOLE VALUES
	,UTME=NNN	UPDATE DYNAMIC DISPLAY EVERY NNN SECONDS
K	A,REF	DISPLAY ALL DISPLAY AREA DEFINITIONS
	,NONE	DELETE ALL DISPLAY AREA DEFINITIONS
	,LL	DEFINE DISPLAY AREA OF LENGTH LL
K	N,PEK=(NN,CMD='TEXT')	DEFINE PFK NN AS COMMAND
	,PFK=(NN,KEY=NN,NN,...)	DEFINE PFK NN AS LIST OF KEYS
	,CON=Y	DEFINE PFK AS CONVERSATIONAL
	,CON=N	DEFINE PFK AS NON=CONVERSATIONAL
K	V,USE=FC	VARY USE OF DEVICE TO FULL CAPABILITY
	,USE=SD	VARY USE OF DEVICE TO STATUS DISPLAY
	,USE=MS	VARY USE OF DEVICE TO MESSAGE STREAM

THE ROUTING OPERAND L=CCA IS ADDED TO A
K COMMAND TO DEFINE AREAS ON ANOTHER
CONSOLE TO CONTROL A DISPLAY IN A PARTI-
CULAR AREA, OR TO VARY THE USE OF A CONSOLE

В этом сообщении перечисляются все параметры команды CONTROL,
которые рассматриваются ниже.

1	—	D PFK			
2		IEE724I 18.27.44 PFK DEFINITION 956			Область
3		KEY# CON DEFINITION			
4		01 NO 10,3			
5		02 NO 10,4			
6		03 YES R 0,U;S BEGIN;S KPOC;R 1,U; X D N; X CJ (??);			Состояния
7		H=Q; X S;S CRJE,,(NONE,ABNO);A Q			информации
8		R 1,U;S BEGIN;S KPOC;R 2,COLD,NOREQ;			
9		S ?CR;S CRJE,,(FORM,NORM)			
10		X S PRTI			
11		06 YES D U,DASD,ONLINE			
12		07 YES M 231,VOL=(SL,PROM00)			
13		08 YES V 280,ONLINE			
14		09 YES V 280,OFFLINE			
15		10 NO K S,DEL=RD,RTME=1,CON=N;K E,1			
16		11 NO K D, F			
17		12 NO K E,D			
18					
19					
20					
21		IEE152I	ENTER CANCEL D C,K		Строки сообще- ний оператору
22	X D				
23		C,K			Область ввода команд
24		IEE163I	MODE=RD		Строка предо- стержений

← Текст сообщений →

Рис. 11.1. Расположение областей на экране дисплей-консоли ЕС-7069.

Чтобы удалить информацию, выданную в ответ на ввод команды

D C,K

следует ввести команду

K E,D

В ответ на ввод этой команды строки сообщений IEE162I исчезают с
экрана. Сама команда остается, и перед ней в колонке 3 появляется гори-
зонтальная черточка (—).

По команде

D PFK

выводится сообщение IEF724I (рис. 11.1) о том, какие команды закреплены
за функциональными клавишами.

Чтобы удалить это сообщение с экрана, следует выдать команду

К E,PFK

Команда CONTROL (К) предназначена для управления работой дисплей-консоли. Перечень команд, предназначенных для управления работой, можно получить по команде D C,K в сообщении IEEE162I, которое было приведено ранее.

1. Нумерация строк области состояния информации. Каждая строка дисплея имеет номер, который может постоянно высвечиваться на экране. Для этого следует выдать команду

К D,N,HOLD

Команда

К E,N

делает невидимыми номера строк на экране дисплея. Например, на рис. 11.1 пропадут номера строк с 1-го по 17-й.

Команда

К D,N

обеспечивает такой режим, когда номера строк пропадают вместе с удаляемыми сообщениями.

2. Управление функциональными клавишами. На пульте управления дисплей-консоли имеются функциональные клавиши, которые имеют номера от ПФ1 до ПФ12. По команде

К D,PFK

номера этих клавиш появляются на экране в строке 20.

1	2	3	4	5	6	7	8	9	10	11	12
---	---	---	---	---	---	---	---	---	----	----	----

Строка 20 экрана

Чтобы удалить это сообщение с экрана, следует выдать команду

К E,PFK

Команда CONTROL позволяет присвоить команду (команды) функциональной клавише, изменить номера клавиш и состав команд. Формат команды CONTROL для этого случая приводится ниже.

Команда	Параметры
{CONTROL} {K	$N, PFK = \left(\text{nn} \left\{ \begin{array}{l} \text{CMD} = \text{'команда [команда]'} \\ \text{KEY} = \text{ll [ttt, ...]} \end{array} \right\} \right) [CON = \{Y\}]$

Параметры N, PFK говорят о том, что команда, закрепленная за клавишей, будет изменена. Вместо nn указывается номер клавиши. В подпараметре CMD перечисляются команды (либо одна команда), которые закрепляются за клавишей nn. Подпараметр CON устанавливает режим работы команд: Y — режим ввода с подтверждением (по умолчанию система выбирает этот режим); N — режим ввода без подтверждения. Подпараметр KEY позволяет закрепить за клавишей nn команды, ранее закрепленные за

другими клавишами *ll*, *mm* ... В этом случае перечисляются старые номера клавиш. Всего может быть указано до 52 значений.

Пример 11.1. Закрепить за клавишей 1 команду VARY:

V 00F,ONLINE

Для этого необходимо выдать команду CONTROL:

K N,PFK=(01,CMD='V 00F,ONLINE')

После ввода команды K на экране никаких сообщений не появляется.

Пример 11.2. Закрепить за клавишей 2 команду, которая ранее была закреплена за клавишей 1. Для этого в формате команды следует указать подпараметр KEY:

K N,PFK=(02,KEY=01)

В ответ система никаких сообщений не выдает.

3. Определение размера области сообщений. Области сообщений предназначены для вывода сообщений, получаемых в ответ на ввод команд DISPLAY и MONITOR. По команде

K A,REF

на экран в строке 21 выдается сообщение:

IEEE1501 CHANGE OPTIONS IF DESIRED — изменить параметры, если требуется, а в строке 22 команда, например:

K A,10

В ответ оператор может оставить эту команду, которая появилась на экране, в строке 22, нажав клавишу ВВОД. Если оператор хочет изменить размер области (или областей), то изменение выполняется по команде

K A,*nn* [,*mm*,...][,L=*cc*]

где *nn* и *mm* — количество областей и размер каждой из них (в строках). Количество строк в области не может превышать восьми строк.

Пример 11.3. Представить область сообщений в виде четырех областей по четыре строки в каждой области. Для этого следует выдать команду

K A,4,4

Первой области будет присвоен идентификатор A, второй — B и т. д. Размеры областей сообщений отменяются по команде

K A,NOME[,L=*cc*]

4. Управление выводом информации на экран дисплея. По команде

K D,F[,L=*x*]

на экран дисплея выводится следующий кадр (сегмент) информации. Вместо *x* указывается номер дисплея и (или) идентификатор области сообщений. Если информации для вывода нет, то выдается сообщение в строке 21:

IEEE1581 REQUEST INCONSISTENT — NO DISPLAY ON SCREEN — нет сообщений для вывода на экран.

При автоматическом удалении сообщений с экрана бывают случаи, когда информацию нужно временно задержать на экране. Это делается с помощью команды

K D,H[,L=x]

Снова возобновить режим автоматического удаления сообщений с экрана следует по команде

K D,U[,L=x]

Если установлен неавтоматический режим, то выдается сообщение в строке 21:

IEE158I K REQUEST INCONSISTENT — DISPLAY IS NOT DYNAMIC

Если надо удалить информацию области сообщений с экрана, то для этого имеется несколько команд. Команда

K E,D[,L=x]

позволяет удалить с экрана только информацию, выданную по команде D C,K. В строке 21 появляется сообщение:

IEE151I DELETE REQUEST INCONSISTENT — NO DISPLAY ON SCREEN — удаляется сообщение с экрана.

Сегмент удаляется по командам

K E,SEG

По команде

K E,nn[,mm ...]

можно удалить определенную строку *nn* или группу строк, начиная с *nn* до *mm*. Например, после ввода команды

K E,1,7

в колонке 3 появляются вертикальные черточки (|) в строках сообщений с 1-й по 7-ю.

Сообщения, отмеченные вертикальной или горизонтальной чертой, удаляются по команде

K E,F

например, на рис. 11.1 в строке 1 удалится команда D PFK.

Если нужно отменить вывод на экран дисплея, то следует выдать команду

K C,D,идентификатор[,L=cc]

где указывается идентификатор в заглавной строке информации состояния, которую необходимо удалить, а *cc* заменяется номером дисплея. Номер дисплея опускается, если команда CONTROL относится к дисплею, с которого она вводится.

5. Вывод текущих значений параметров DEL, SEG, RNUM, RTME и CON. Параметр DEL устанавливает режим удаления сообщений с экрана: автоматический или неавтоматический; параметр SEG — размер

сегмента; параметр RNUM — размер свертываемого сообщения; параметр RTME — время свертывания; параметр CON — способ удаления сообщений: с подтверждением или без подтверждения. Все перечисленные выше параметры устанавливаются при генерации ОС. Значения этих параметров оператор может получить по команде

K S,REF

В ответ система в области ввода выдает сообщение:

IEEE1501 CHANGE OPTIONS IF DESIRED

Значения параметров задаются в появляющейся команде CONTROL:

K S,DEL=4,SEG=09,CON=Y,RNUM=19,RTME=046

где DEL=Y — установлено автоматическое удаление сообщений; SEG=09 — удаление девяти строк, когда действует команда K E,SEG (размер сегмента 9 строк); CON=Y — удаление сообщений с подтверждением; RNUM=19 — размер сегмента свертываемого сообщения равен 19 строкам; RTME=046 — время (число секунд) свертывания при выдаче команд K DEL=R или K DEL=RD. Это сообщение снимается повторным нажатием клавиши «ВВОД».

6. Настройка режима работы дисплея. Имеются четыре команды, относящиеся к установлению режима удаления сообщений с экрана дисплея:

K S,DEL=Y — автоматическое удаление;

K S,DEL=N — неавтоматическое (с помощью команд, перечисленных в п. 3, светового пера или курсора);

K S,CON=Y — удаление с подтверждением;

K S,CON=N — удаление без подтверждения.

Если информация удаляется сегментами (по команде K E,SEG), то размер сегмента устанавливается по команде

K S,SEG=nn

где nn — количество строк сегмента.

При удалении сообщений могут быть использованы различные методы свертки сообщений. По команде

K S,DEL=R

все сообщения свертываются. Команда

K S,DEL=RD

свертывает только удаляемые сообщения. Команда

K S,RNUM=nn

устанавливает размер сегмента свертывания (число строк nn) при работе команд K S,DEL=R или K S,DEL=RD.

Время свертывания задается командой

K S,RTME=nnn, где nnn — число секунд, определяющих интервал времени, через который происходит свертка экрана.

Команда

K M,REF позволяет получить данные о времени вывода информации состояния на строке 21, например:

IEEE1501 CHANGE OPTIONS IF DESIRED — система просит сообщить данные о времени выдачи информации состояния;

K M,UTME=030

Оператору следует указать новое время, например:

K M,UTME=045

и нажать клавишу «ВВОД».

7. Управление режимом работы дисплея. Имеются следующие режимы работы дисплей-консоли: команда

K USE,FC[,L=cc]

обеспечивает режим ввода-вывода сообщений и информации состояния. Возможно установление только режима вывода сообщений по команде

K USE,MS[,L=cc]

или вывода сообщений информации состояния по команде

K USE,SD[,L=cc]

Команда MSGRT направляет сообщения, получаемые в ответ на ввод команд DISPLAY, MONITOR и CONTROL, на определенную консоль и (или) в указанную область информации состояния дисплей-консоли. Формат команды MSGRT:

Команда	Параметры
{MSGRT} {MR}	{ ({ D=(операнд,...) [,MN=A] [,K] } MN=A [,D=операнд [,K]] K [,D=операнд [,MN=A]] } [,L=x]) }

Набор параметров команды MSGRT составлен так, что позволяет направлять сообщения как для одной информационной команды, так и для их разных комбинаций; параметр D=(операнд,...) используется при работе с командой DISPLAY. В скобках указываются ее операнды, параметр MN=A указывается при работе с командой MONITOR; параметр K — при работе с командой CONTROL. Вместо x указывается номер консоли и (или) идентификатор области информации состояния для дисплей-консоли.

11.2. Управление консолями

Команда VARY выдается с основной консоли и позволяет переопределить основную консоль. Формат команды VARY для этого случая:

Команда	Параметры
{VARY} {V}	{адрес {i=адрес, O=адрес}.MSTCONS

где указывается новый адрес основной консоли, а параметр MSTCONS по-

казывает, что команда VARY используется для назначения новой основной консоли, например:

V 0C0,MSTCONS

В ответ система выдает сообщение

IEE143I OLD=01F NEW=0C0 VALDCMD=ALL — адрес старой консоли (OLD)=01F (пишущая машинка); адрес новой консоли (NEW)—0C0 (дисплей-консоль); VALDCMD=ALL показывает, что разрешается использовать все команды основной консоли. Далее следует продолжение сообщения:

IEE143I ROUTCDE=ALL T=M H=N — разрешается использовать все маршрутные коды; T=M — признак главной консоли; H=N — для системного журнала консоль не изменилась. В результате пишущая машинка оказывается в неактивном состоянии.

Перед тем как выдать команду назначения новой основной консоли, следует убедиться, что она находится в активном состоянии, используя данные сообщения IEE140I или выдав команду D CONSOLES. В активное состояние устройство переводится по команде VARY, рассматриваемой ниже. Если новая основная консоль останется в неактивном состоянии, то система будет пытаться сама назначить новую основную консоль, выдав на все активные консоли сообщение:

IEE141A MASTER AND ALL ALTERNATES UNAVAILABLE — USSUE VARY MSTCONS — переключение главной консоли оказалось неудачным. Любой оператор, который получил это сообщение, может ввести команду VARY MSTCONS и тем самым объявить свою консоль или любую альтернативную как основную.

Если в качестве основной консоли назначается составная консоль, то в формате команды записываются: I=*адрес* — адрес устройства ввода; O=*адрес* — адрес устройства вывода.

Команда VARY в формате, приведенном ниже, изменяет состояние и характеристики всех консолей, кроме основной и консоли, которая используется для работы со сборным протоколом:

Команда	Параметры
{VARY} {V}	(<i>адрес 1</i> [, <i>адрес 2</i>]...),CONSOLE [,AUTH= <i>x</i>] [,ROUT= <i>y</i>] [,ALTCONS= <i>адрес</i>]

В формате команды указываются адреса периферийных устройств, состояние и характеристики которых изменяются. Параметр CONSOLE переводит устройство в состояние активной консоли. Параметр AUTH=*x* показывает, какие команды можно вводить с данного устройства. Значение *x* заменяется на следующие параметры: ALL — выдаются все команды, кроме команд V MSTCONS, V HARDCPY, MODE и DUMP; INFO — информационные команды; SYS — команды управления ОС; IO — команды управления периферийными устройствами; CONS — команды управления консолями (эти команды содержат параметр L). Параметр ROUT=*y* задает маршрутные коды сообщений, которые должны выдаваться на данную консоль. Значение *y* заменяется на параметры: ALL — все маршрутные коды от 1 до 16;

NONE — отменяет ранее определенные маршрутные коды; (число 1[число 2] ...) — список маршрутных кодов, составленных из значений от 1 до 16. Если указывается один маршрутный код, то скобки опускаются. Назначение новой альтернативной консоли производится с помощью параметра ALTCONS, в котором указывается адрес этой консоли.

Пример 11.4. Назначить в качестве основной консоли алфавитно-цифровой дисплей 0C0. Пусть ранее в качестве основной консоли (M) использовалась пишущая машинка 01F и альтернативной консолью было устройство 060 (см. рис. 7.5).

Поскольку дисплей-консоль 0C0 находится в неактивном состоянии (N), то следует перевести ее в активное состояние с помощью команды

V 0C0,CONSOLE

Для проверки можно выдать команду

D CONSOLES

Далее следует выдать команду

V 0C0,MSTCONS

В ответ получим сообщение IEE143I, которое было приведено выше. Выдать команду с новой основной консоли 0C0

D CONSOLES

Из сообщения IEE250I видно, что пишущая машинка 01F, находится в неактивном состоянии. Обратный перевод выполняется по командам

V 01F,CONSOLE

V 01F,MSTCONS

С целью проверки можно выдать команду D CONSOLES.

Команда VARY позволяет переопределить устройство и изменить условия выдачи сборного протокола системы с основной консоли. Формат команды VARY для этого случая:

Команда	Параметры
{VARY} {V}	[адрес [SYSLOG].HARDCPY [CMDS ,NOCMS ,INCHDS ,STCMS ,OFF] [ROUT={ALL NONE (число 1, число 2,...)}]]

Здесь указывается адрес нового устройства, который будет использоваться для вывода содержимого сборного протокола (устройство должно быть активным; можно использовать устройство вывода составной консоли; нельзя использовать дисплей-консоль). Параметр SYSLOG показывает, что в качестве сборного протокола используется системный журнал. Если системный журнал не предусмотрен при генерации системы, то команда VARY не воспринимается системой. Параметр HARDCPY показывает, что команда

VARY используется для изменения условий работы сборного протокола. Параметр CMDS позволяет занести в сборный протокол все команды оператора и ответы на них, а NOCMDS запрещает это делать. Параметр INCMDS вносит в сборный протокол все команды оператора и ответы на них, кроме сообщений в ответ на команды DISPLAY и MONITOR. Параметр STCMDS выполняет те же действия, что и INCMDS, но не вносит сообщений в ответ на ввод команды MONITOR. Параметр OFF отменяет введение системного журнала. В параметре ROUT указывается: ALL — в сборный протокол заносятся все сообщения; NONE — сообщения не заносятся; (число 1, число 2, ...) — список маршрутных кодов сообщений, которые должны вноситься в сборный протокол (значения маршрутных кодов от 1 до 16). Если опущен адрес или SYSLOG, то изменяются условия работы ранее определенного сборного протокола.

Глава 12. ДИАЛоговые удаленный ввод заданий

Система диалогового удаленного ввода заданий (ДУВЗ) предназначена для работы с терминальными устройствами, например такими, как дисплей ЕС-7927 и ЕС-7066, которые далее будем называть *абонентскими пунктами*.

Таблица 12.1. Команды оператора, непосредственно работающего с ЭВМ

Команда	Назначение
BDR CST	Изменяет содержимое раздела, где хранятся сообщения абонентов
CENOUT	Выводит результаты задания, выполняемого в ДУВЗ
[MOD]F[Y]	Изменяет состояние канала связи системы ДУВЗ
MSG	Обеспечивает связь с абонентами
SHOW	Позволяет получить информацию о текущем состоянии системы ДУВЗ
S[TART]	Запускает систему ДУВЗ
[STO]P	Останавливает работу системы ДУВЗ
USERID	Управляет списком абонентов ДУВЗ

С абонентских пунктов задания для выполнения передаются на ЭВМ в пакетную обработку. Пакетная обработка обеспечивает последовательное выполнение заданий, т. е. каждое последующее задание начинает выполняться после завершения предыдущего. Возможна работа в диалоговом режиме. *Диалоговый режим* — это режим, при котором запрос с дисплея приводит к ответу со стороны ЭВМ, и наоборот.

При работе с системой ДУВЗ в библиотеке SYS1.PROCLIB должна находиться процедура ее запуска CRJE, а для передачи заданий в пакетную обработку — процедура RDR CRJE. Система ДУВЗ выполняется как системная задача (см. гл. 6), и ей выделяется отдельная зона в ОП. Одновременно в системе ДУВЗ могут работать до 32 абонентов. Задания абонентов, которые передаются для пакетной обработки выполняются в фоновых зонах ОП, т. е. зонах, выделенных для заданий, вводимых с помощью команды ДУВЗ SUBMIT, которая рассматривается ниже.

Работу с системой ДУВЗ можно разделить на следующие этапы: запуск ДУВЗ, подключение к ЭВМ дисплеев и работа операторов с системой ДУВЗ. Необходимо обратить внимание на то, что с системой ДУВЗ работают два оператора: один непосредственно на ЭВМ, а второй — на абонентском пункте, где установлены удаленные дисплеи. Оператора, работающего на абонентском пункте, далее будем называть *абонентом*. Оператор, работающий непосредственно на ЭВМ, пользуется командами, перечень которых приведен в

Таблица 12.2. Команды абонента

Команда	Назначение
C[ANCEL]	Отменяет задание
CO[NTINUE]	Продолжает вывод
D[ELETE]	Удаляет набор данных
E[DIT]	Редактирует набор данных
EX[EC]	Выполняет группу команд
{LISTB[C] LB[C]}	Запрашивает копии сообщений
{LISTD[S] LDS}	
L[ISTLIB]	Выдает информацию о наборе данных ДУВЗ
LOG[OFF]	Выдает оглавление библиотеки
LO[GON]	Заканчивает сеанс
O[UTPUT]	Начинает сеанс
SE[ND]	Выдает результаты задания
ST[ATUS]	Посылает сообщение
SU[BMIT]	Запрашивает информацию о состоянии задания
T[ABSET]	Передает задание в пакетную обработку
V[ARY]	Устанавливает табуляцию
	Изменяет характеристики ввода-вывода (версия 6.1)

табл. 12.1, а абонент работает с командами, перечень которых приведен в табл. 12.2.

12.1. Команды оператора, непосредственно работающего с ЭВМ

Управление системой ДУВЗ. Система ДУВЗ запускается с помощью команды START как обычная нестандартная каталогизированная процедура (см. гл. 7). Формат команды START:

Команда	Параметры
{START} {S}	имя процедуры [.идентификатор]... $\left(\begin{Bmatrix} \text{FORM} \\ \text{NFMT} \\ \text{NONE} \end{Bmatrix} \begin{Bmatrix} \text{.ABNO} \\ \text{.NORM} \end{Bmatrix} \right)$ [.ключевые параметры]

Указывается имя процедуры ДУВЗ, под которым она хранится в библиотеке процедур SYS1.PROCLIB. Идентификатор абонента начинается с буквы и может содержать от 1 до 7 буквенно-цифровых символов. После трех запятых обязательно указывается состояние набора данных SYS1.SYSJOBQE, в котором организуется системная очередь заданий. Параметр FORM указывается, если очередь организуется заново; NFMT — старая очередь сохраняется; NONE — операционная система не перезагружалась после последнего останова ДУВЗ. Параметр ABNO используется, если нужно восстановить ранее отведенную область памяти; параметр NORM используется, если выделяется новая область памяти. В формате могут быть указаны ключевые параметры, если они отличаются от тех, которые установлены при генерации системы ДУВЗ. Ключевые параметры перечисляются в произвольном порядке

через запятые. Значения ключевых параметров сообщаются оператору системным программистом. На рис. 12.1 показан пример запуска каталогизированной процедуры с именем CRJE123 и идентификатором CRJE. Указаны параметры FORM, NORM и ключевые параметры T1 и L.

```
S CRJE123.CRJE,...(FORM,NORM),T1=RUZA02,L=0D1
IEF244I CRJE123.CRJE.LINE2 UNABLE TO ALLOCATE
IEF247I CRJE123 ALLOCATION RECOVERY
IEF247I CRJE123 063 OFFLINE
IEF403I CRJE123 STARTED TIME=19.55.13
*04 IEF238A REPLY DEVICE NAME OR 'CANCEL'
R 04,'063'
IEE600I TO 04 IS; 063
INH223I CRJE NOW ACTIVE
INH219I OUT OF MAIN STORAGE
D A
IEE102I 19.56.27 ACTIVE DISPLAY
STRADDR ENDADDR SQA R SUBT NAME 1 NAME2 NAME3
01984K 02048K 03648 03 MASTER SCHEDULER
00000K 00000K 00592 00 INIT
01884K 01984K 03467 01 CRJE123 CRJE
```

Рис. 12.1. Запуск системы ДУВЗ.

Перечень сообщений, которые выдает система в ответ на ввод команды START, зависит от того, в каком состоянии находится дисплей, который будет работать с ДУВЗ. Если он находится в состоянии OFFLINE, то выдаются сообщения, которые приведены на рис. 12.1. Сообщения планировщика заданий IEF244I, IEF247I и IEF403I были рассмотрены ранее. Последнее сообщение IEF403I говорит о том, что процедура CRJE123 стартовала. Далее система выдает запрос 04 о том, чтобы оператор указал адрес устройства, с которым будет работать система. Оператор указывает адрес устройства 063. Система принимает ответ, о чем выдает сообщение IEE600I. Сообщение INH223I говорит о том, что ДУВЗ активен. И, наконец, в сообщении INH219I говорится о том, что оперативная память для ДУВЗ выделена. Команда D A выдана для того, чтобы иметь представление об объеме памяти, которая выделяется для системы ДУВЗ.

Если дисплей находится в состоянии ONLINE, то сразу выдается сообщение IEF403I и далее сообщения INH219I, IEE223I, приведенные на рис. 12.1.

Команда STOP позволяет остановить работу системы ДУВЗ. Формат команды STOP:

Команда	Параметры
{STOP} {P}	[имя процедуры.] идентификатор

В формате команды указывается имя процедуры ДУВЗ и идентификатор, как в команде START. Например, пусть надо остановить работу системы ДУВЗ, запущенную выше по команде START (рис. 12.1). Выдается команда

P CRJE

В ответ система выдает сообщение:

INH230I CRJE CLOSED DOWN — работа системы ДУВЗ остановлена.

Команда CENOUT позволяет вывести результаты задания, которое выполняется в системе ДУВЗ, с помощью программы системного вывода WTR.

Формат команды CENOUT:

Команда	Параметры
CENOUT	J=задание, С=класс

В параметре J указывается имя задания, выходную информацию которого необходимо передать на устройство системного вывода с помощью программы WTR. В параметре С указывается класс выходной очереди, которая будет использоваться при выводе результатов этого задания. В ответ на ввод команды CENOUT, например:

CENOUT J=KLIMAT, C=A

система выдает сообщение:

INH214I CENOUT KLIMAT — результаты выполнения задания KLIMAT будут выводиться в соответствии с командой CENOUT.

Команда MODIFY позволяет изменить состояние каналов связи с системой ДУВЗ: перевести их в оперативное или автономное состояние. Формат команды MODIFY:

Команда	Параметры
{MODIFY} {F}	[имя процедуры.] идентификатор, { ^A _D } = (адрес 1, ...,)

В формате команды указывается имя процедуры и идентификатор такие же, как и в команде START; параметр А переводит канал (каналы), связывающий ЭВМ с удаленным дисплеем в оперативное состояние; параметр D переводит канал (каналы) в автономное состояние. В скобках указывается адрес (адреса) удаленного дисплея. Ниже приводится пример команды перевода канала в автономное состояние:

F CRJE, D=071 — адрес удаленного дисплея 071

В ответ система выдает сообщение:

INH244I LINE DEACTIVATED 071 — линия канала, связывающая ЭВМ с устройством 071, неактивная.

При вводе команды

F CRJE, A=071

выдается сообщение:

INH243I LINE BEING ACTIVATED 071 — линия канала, связывающая ЭВМ с устройством 071, активная.

Информационные команды. Команда SHOW позволяет вывести на консоль информацию о состоянии системы ДУВЗ в текущий момент времени.

Формат команды SHOW:

Команда	Параметры
SHOW	$\left\{ \begin{array}{l} \text{JOBS [имя задания]} \\ \text{USER [идентификатор абонента]} \\ \text{ACTIVE [NUMBER]} \\ \text{BRDCST} \\ \text{MSGs [идентификатор абонента]} \\ \text{LERB [адрес]} \\ \text{SESS [идентификатор абонента]} \\ \text{SESSREL [идентификатор абонента]} \end{array} \right\}$

Параметр JOBS позволяет получить информацию на консоли обо всех выполняемых заданиях системы ДУВЗ. Если указать имя задания, то выдается информация только об этом задании. В ответ на ввод команды

SHOW JOBS,INNTP

выдается сообщение:

INH237I JOB COMPLETE INNTP D071551 — в системе имеется задание INNTP; идентификатор абонента D071551.

Если указанного задания нет в системе, то выдается сообщение:

INH211I JOB NOT IN SYSTEM INNTP

Параметр USER обеспечивает получение сведения об абонентах системы ДУВЗ, а если указать идентификатор абонента, то выдаются сведения именно об этом абоненте.

Параметр ACTIVE позволяет получить информацию о всех активных абонентах системы ДУВЗ, которые установили связь с системой с помощью команды LOGON (табл. 12.2), например:

SHOW ACTIVE

INH201I ACTIVE CRJE USER MAG1 0D2

INH201I ACTIVE CRJE USER UBD1 0D1

INH201I ACTIVE CRJE USER SPF1 0D3

Активные абоненты имеют идентификаторы MAG1, UBD1 и SPF1; номера удаленных дисплеев 0D2, 0D1 и 0D3.

Параметр NUMBER позволяет дополнительно получить сведения о количестве активных абонентов системы ДУВЗ, например:

INH245I 3 ACTIVE CRJE USERS — в системе ДУВЗ имеется три активных абонента.

Если активных абонентов нет, то выдается сообщение:

INH245I NO ACTIVE CRJE USERS

Параметр BRDCST выводит на консоль все сообщения, которые хранятся в разделе BRDCST библиотеки CRJE.SYSLIB. В этот раздел помещаются сообщения оператора, выданные абоненту по команде MSG в моменты, когда абонент неактивен.

Параметр MSGS позволяет вывести на консоль все сообщения, которые были выданы оператором с помощью команды MSG и не были переданы

абонентам системы ДУВЗ. Если указывается идентификатор абонента, то выводится сообщение, относящееся к этому абоненту. При выдаче команды

SHOW MSGS

выдаются сообщения, например:

INH203I START MSG CRJE MESSAGES — начало сообщений ДУВЗ;

D07155I CRJE MSG FR CENTRAL ТЕКСТ — сообщение, выданное с главной консоли — ТЕКСТ;

INH213I END OF CRJE MESSAGES — конец сообщений ДУВЗ.

Информация об ошибках в каналах связи, которые имели место при передаче данных между абонентами системы ДУВЗ и ЭВМ, выдается, если указан параметр LERB. Адрес, указанный дополнительно к параметру LERB, показывает, что необходимо выдать информацию об ошибках только этого канала.

Параметр SESS выдает информацию о всех абонентах системы ДУВЗ в момент выполнения ими команд LOGON и LOGOFF (табл. 12.2). В ответ на ввод команды

SHOW SESS

система выдает сообщение:

INH249I SHOW SESS IN EFFECT — в моменты выполнения команд LOGON и LOGOFF будут выдаваться сообщения IEF125I и IEF126I, например:

IEF126I U2112100 LOGGED OFF TIME=10.04.13 — абонент с идентификатором U212100 начал работу в 09.49.04;

IEF126I U2112100 LOGGED OFF TIME=10.04.13 — абонент с идентификатором U2112100 закончил работу в 10.04.13.

Параметр SESSREL отменяет команду SHOW SESS для всех абонентов. Если указывается идентификатор, то отменяется выдача сообщений только для этого абонента. При мультиконсольном режиме рекомендуется перед остановкой работы системы ДУВЗ выдать команду SHOW SESSREL. В ответ на ввод этой команды система выдает сообщение:

INH250 SHOW SESS RELEASED — команда выполнена.

Команда MSG предназначена для связи оператора с абонентами. Формат команды MSG:

Команда	Параметры
MSG	{M='текст' [U=идентификатор] [Q]} {D=идентификатор абонента}

В параметре M указывается текст передаваемого сообщения (до 40 символов). В параметре U сообщается идентификатор абонента. Параметр Q позволяет занести сообщение в набор данных CRJE.SYSLIB в раздел сообщений BRDCST, если абонент в момент выдачи команды MSG был неактивен. Фактически это сообщение будет передано абоненту после того, как он выдаст команду LOGON. Если параметр Q опущен, то сообщение теряется. Указание идентификатора абонента в параметре D уничтожает все сообщения

данного абонента в наборе данных CRJE.SYSLIB в разделе BRDCST. Приведем пример выдачи команды:

MSG M='ТЕКСТ'

В ответ система выдает сообщение:

INH256I MSG QUEUED FOR DELIVERY — сообщение поставлено в очередь для передачи абоненту.

Команда BRDCST предназначена для внесения изменений в раздел BRDCST. Формат команды BRDCST:

Команда	Параметры
BRDCST	$\left\{ \begin{array}{l} \text{nnnn, 'текст'} \\ \text{'текст'} \\ \text{nnnn} \\ \text{DELETE} \end{array} \right\}$

В первом формате указывается номер записи *nnnn* в разделе BRDCST, в которую помещается текст сообщения. Текст сообщения должен содержать не более 40 символов (апостроф в тексте удваивается). Если указывается только текст сообщения, то он добавляется в этот раздел. Если указывается только номер записи, то это значит, что данная строка удаляется из раздела. Параметр DELETE используется в случае, если надо удалить все записи из раздела BRDCST.

Пример выдачи команды:

BRDCST 1200 — удалить строку 1200.

В ответ система выдает сообщение:

INH225I CRJE BROADCAST MESSAGE(S) DELETED 1200 — из раздела BRDCST удалена строка 1200.

Команда USERID позволяет управлять списком абонентов. Формат команды USERID:

Команда	Параметры
USERID	$\left\{ \begin{array}{l} A=(\text{идентификатор, пароль}) \\ D=(\text{идентификатор, пароль}) \\ S \\ R \end{array} \right\}$

В параметре A (ADD) указывается идентификатор абонента и его пароль (от 1 до 8 буквенно-цифровых символов, первый из которых должен быть буквой) для включения в список абонентов системы ДУВЗ. В параметре D (DELETE) указывается идентификатор абонента и пароль для исключения его из списка абонентов системы ДУВЗ. Параметр S (SUPPRESS) запрещает неактивным абонентам включаться в работу. Команды абонентов LOGON в этом случае не воспринимаются. Параметр R (RESUME) отменяет команду USERID S. Приведем примеры выдачи команд:

USERID R

В ответ система выдает сообщение:

INH229I LOGONS RESUMED — команды LOGON воспринимаются.

В ответ на ввод команды

USERID S

система сообщает:

INH221I LOGONS SUPPRESSED — команды LOGON не воспринимаются.

12.2. Команды абонента

Для обеспечения всех видов работ с дисплея предусмотрены три режима: РК (режим команд), РР (режим редактирования) и РВ (режим ввода). Каждый режим характеризуется возможностью работы с определенным типом данных. Режимы РР и РВ предназначены для работы с активными наборами данных. *Активным набором данных* называются данные, которые на время их обработки помещаются в активную область памяти, отведенную ДУВЗ. В качестве данных могут быть любые наборы, в том числе оформленные как задание. Режим работы РК предполагает работу, не связанную с обработкой активного набора данных. В каждом режиме используется определенный набор команд, который приведен в табл. 12.3. После выполнения каждой

Таблица 12.3. Режимы работы команд абонента

Команды абонента	РК		РР		РВ	
	режим работы	режим после выполнения	режим работы	режим после выполнения	режим работы	режим после выполнения
CANCEL	+	РК	—	—	—	—
CONTINUE	+	РК	—	—	—	—
DELETE	+	РК	—	—	—	—
EDIT	+	РР или РВ	—	—	—	—
EXEC	+	РК	—	—	—	—
LISTBC	+	РК	—	—	—	—
LISTDC	+	РК	—	—	—	—
LISTLIB	+	РК	—	—	—	—
LOGOFF	+	РК	—	—	—	—
LOGON	+	РК	—	—	—	—
OUTPUT	+	РК	—	—	—	—
SEND	+	РК	—	—	—	—
STATUS	+	РК	—	—	—	—
SUBMIT	+	РК	—	—	—	—
TABSET	+	РК	—	—	—	—
VARY	+	РК	—	—	—	—
Подкоманда INPUT	—	—	+	РВ	—	—
Подкоманда END	—	—	+	РК	—	—
Подкоманды EDIT, кроме INPUT и END	—	—	+	РР	—	—
Ввод пустой строки	—	—	+	РВ	+	РР
Ввод текста записей	—	—	—	—	+	РВ

Примечание. «+» — команда используется;
«—» — команда не используется.

команды режим дисплея может либо измениться, либо остаться без изменения. В исходном состоянии, т. е. после того, как система ДУВЗ стартовала и дисплей включен, на дисплее устанавливается режим команд. На первой строке экрана появляется сообщение:

CRJE NOW ACTIVE — ДУВЗ активен.

После появления этого сообщения и курсора на следующей строке можно начинать пользоваться командами абонента.

Общая характеристика команд. Команды абонента можно разделить на две группы. Первая группа команд предназначена для обеспечения работы дисплея с системой, установления режимов работы самого дисплея и обеспечения связи между оператором ЭВМ и абонентом (или между абонентами).

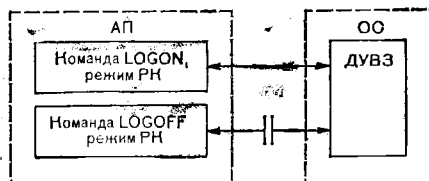


Рис. 12.2. Связь АП с системой ДУВЗ.

К этой группе команд можно отнести:

- 1) команды связи с системой: LOGON и LOGOFF;
- 2) команды связи между оператором и абонентом (или абонентами) SEND и LISTBC;
- 3) команды управления полями экрана дисплея TABSET и VARY;
- 4) команды изменения режимов работы: подкоманды INPUT и END команды EDIT, и команда ВВОД, которая вводится с пульта дисплея с помощью клавиши «ВВ».

Ко второй группе команды можно отнести команды:

- 1) управления выполнением заданий: SUBMIT, STATUS, OUTPUT, CONTINUE, CANCEL;
- 2) работы с наборами данных в библиотеке абонентов: CRJE,SYSLIB:LISTDS, LISTLIB, DELETE;
- 3) редактирования: все подкоманды команды EDIT, кроме INPUT и END.

Связь дисплея с системой обеспечивается с помощью команд LOGON и LOGOFF. Команды выдаются в режиме РН, и после выполнения этот режим работы дисплея не изменяется. Команда LOGON*) служит для подсоединения дисплея к системе ДУВЗ (рис. 12.2). Формат команды LOGON:

Команда	Параметры
{LOGON} {LO}	идентификатор пароль А (учетная информация) [BC][M] [NOBC][NOM]

*) Здесь и далее между параметрами указываются один или несколько пробелов.

Идентификатор пользователя и пароль составляются, как в команде USERID. В параметре A (ACCT) сообщается учетная информация абонента, которая может содержать до 33 символов. Если в учетной информации имеются пробелы или запятые, то она заключается в одиночные апострофы. Параметр BC позволяет получить сообщения на дисплее, выдаваемые оператором с ЭВМ по команде BRDCST, а параметр NOBC сообщает системе ДУВЗ, что такие сообщения не нужны. По умолчанию система считает, что задан параметр BC, т.е. сообщения нужны. Параметр M (MSGID) указывает, что системные сообщения выдаются с их идентификаторами, а NOM (NOMSGID) — отменяет выдачу идентификаторов сообщений.

В случае правильного подключения дисплея к ЭВМ, после выдачи команды, например:

LO A0C1/USER M

на экране дисплея абонента появляются сообщения:

INH327 ***CRJE LINE A0C1 82.097 TIME 02:45***
INH312 NO BROADCAST MSGS

В первом сообщении говорится о том, что дисплей абонента A0C1 подсоединен к системе ДУВЗ, и далее следуют дата и время. Во втором сообщении — о том, что нет никаких указаний от оператора, работающего непосредственно на ЭВМ. На следующей строке в первой колонке устанавливается курсор, показывающий место, начиная с которого можно вводить команды.

Команда LOGON выдается в следующих случаях:

- 1) после подсоединения удаленного дисплея к системе ДУВЗ, т.е. после получения абонентом сообщения CRJE NOW ACTIVE;
- 2) после выдачи команды LOGOFF и получения сообщения, например:

INH354 LOGOFF 13:15 SESSION TIME=0008 MIN

об окончании работы с абонентом;

- 3) во время работы другого абонента, который начал работу ранее, в режиме команд. При выдаче команды LOGON этот абонент будет отключен.

Команда LOGOFF служит для отключения абонента от системы ДУВЗ после того, как сеанс работы абонента с ЭВМ закончился. Формат команды LOGOFF:

Команда	Параметры
{LOGOFF} {OFF}	

В формате команды параметры не указываются. В ответ на ввод команды

OFF

система выдает сообщение на АП:

INH354 LOGOFF 18:10 SESSION TIME=0003 MIN

в котором указывается время окончания работы абонента с ЭВМ,

Отключение абонента на ЭВМ происходит также после выдачи каждой очередной команды LOGON.

Связь между оператором ЭВМ и абонентом (или между абонентами). Команда SEND позволяет передать сообщение оператору ЭВМ или другому абоненту системы ДУВЗ. Формат команды SEND:

Команда	Параметры
{SEND} {SE}	'текст' { [O (идентификатор пользователя) [L]] } [U (идентификатор) [N]] }

Текст сообщения должен содержать не более 40 символов (апострофы в тексте удваиваются). Параметр O (OPERATOR) показывает, что сообщение передается оператору ЭВМ. Параметр L (LOGON) сохраняет сообщение до того момента, пока абонент не будет подсоединен к системе ДУВЗ. После подсоединения АП к системе сообщение будет передано оператору ЭВМ. Параметр U (USER) показывает, что сообщение передается другому абоненту. Параметр N (NOW) добавляется в случае, когда сообщение следует передать, только если абонент активен.

Пример выдачи команды SEND другому абоненту:

SE 'TEKST' U(D071551) — идентификатор D071551. После ввода команды SEND курсор перемещается на следующую строку.

По команде LISTBC абонент получает копии сообщений, выданных оператором с помощью команды MSGS. Формат команды LISTBC:

Команда	Параметры
{LISTBC LISTB LBC LB }	

В формате команды параметры отсутствуют. Если сообщения не выдавались, то система сообщает:

INH312 NO BROADCAST MSGS

Управление полями экрана дисплея. Команда TABSET позволяет установить номера позиций полей на дисплее, которые имеют средства табуляции. Команда TABSET используется в режимах РК и PP, и состояние режима дисплея не изменяет. Формат команды TABSET:

Команда	Параметры
{TABSET} {T}	[номер позиции 1,...] [IN OFF] [OUT]

Всего можно установить до 10 полей. Первая позиция поля имеет номер 0. Параметр IN (INPUT) указывается, если информация вводится с дисплея. При нажатии клавиши горизонтальной табуляции курсор устанавливается на позиции поля, которые определены в команде TABSET. Параметр

OUT (или OUTPUT) показывает, что информация выводится на дисплей из ЭВМ в соответствии с определенными позициями, установленными по команде TABSET. Параметр OFF означает, что табуляция отменена.

Команда VARY позволяет изменить некоторые параметры работы дисплея и используется в режимах PK и PP, как команда TABSET. Формат команды VARY:

Команда	Параметры
{VARY} {V}	{ {DC=x} {DL=y} {HT=z} {EOM=m} {HT=z} {WAIT=nnn} {LIME=hhh} {MODE={PAGE/ROLL}} }

Параметр DC=x задает знак отмены символа, который используется в случае ошибочного ввода символа (по умолчанию используется @). Параметр DL=y задает знак отмены строки (по умолчанию система использует знак []). Параметр HT=z задает знак табуляции (по умолчанию используется []). Параметр EOM=m задает знак конца строки при вводе информации с пульта оператора (по умолчанию используется знак %). В параметре WAIT задается время ожидания в мкс между двумя последовательными операциями

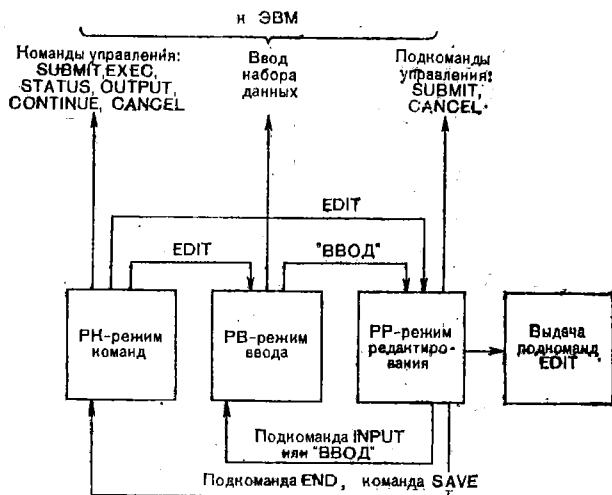


Рис. 12.3. Схема управления режимами работы дисплея на АП.

записи на дисплей в режиме свертки (по умолчанию принимается 250, т. е. 2,5 с). В параметре LINE=hhh устанавливается максимальное число символов в строке для пишущих машинок. В параметре MODE задается режим вывода: PAGE — постраничный вывод, ROLL — режим свертки (если этот режим предусмотрен при генерации системы ДУВЗ).

Команды изменения режимов работы дисплея. На рис. 12.3 приведена схема изменения режимов, с помощью команд абонента: EDIT, подкоманд INPUT, END и SAVE, и команды ВВОД, которая выдается путем нажатия

клавиши ВВ на пульте дисплея. Ниже рассмотрим режимы работы дисплея, виды работ, какие можно выполнять в каждом режиме, способы перехода от режима к режиму.

1. Создание нового набора данных и его ввод в библиотеку CRJE.SYSLIB начинается с установления режима РК с последующим переходом в режим РВ. Переход от режима РК к режиму РВ осуществляется по команде EDIT, в которой указан параметр NEW. Передача нового набора данных производится через активную область ОП.

2. Редактирование ранее созданного набора данных обеспечивается командой EDIT, но в ней должен быть указан параметр OLD. Команда EDIT выдается, когда на дисплее установлен режим команд. После выполнения команды EDIT с параметром OLD режим дисплея изменяется: дисплей устанавливается в режим редактирования. Обработка набора данных выполняется в активной области памяти. Формат команды EDIT:

Команда	Параметры
{EDIT} {E}	$\left\{ \begin{array}{l} \text{имя набора данных } \{ (\text{идентификатор пользователя } [\text{пароль}]) \} \\ \text{'имя набора данных (имя раздела)'} \\ \text{[NEW] [NUM] [LONG] [S]} \\ \text{[OLD] [NONUM] [Lnn[n]] [NOS]} \\ \text{PL/I [число 1 число 2] [C48] [C60]} \\ \text{FORT } \{ \begin{array}{l} G \\ H \end{array} \} \\ \text{DSLIST} \\ \text{CLIST} \\ \text{DATA} \\ \text{TEXT} \end{array} \right\}$

В формате команды указывается имя набора данных, создаваемого вновь или редактируемого в ДУВЗ или ОС ЕС. Кроме того, может быть указан идентификатор пользователя ДУВЗ, в библиотеке которого находится редактируемый, незащищенный набор данных ДУВЗ. Если набор данных защищен, то указывается пароль защиты. Можно вместо идентификатора и пароля указать имя раздела библиотечного набора данных. Параметр NEW используется, если создается новый набор данных; параметр OLD указывается, если старый набор данных редактируется. Параметр NUM показывает, что в позициях 73—80 каждой строки (записи) набора данных указаны номера строк по порядку. Если набор данных новый, то ему присваивается характеристика SEQ. Если строки не нумеруются, то указывается параметр NONUM, а набору данных присваивается характеристика NOSEQ. Параметр LONG указывается в случае, если длина строки (записи) равна 120 байт. Параметр Lnn[n], где nn или nnn — длина строки (записи), также используется для указания длины строки. Максимальный размер строки не должен превышать 144 байт. Параметры LONG и Lnn[n] используются только в модификации 1 версии ОС 6.1. Параметр S (SCAN) задает автоматический синтаксический контроль вводимых строк набора данных, а параметр NOS (NOSCAN) указывает, что строки не контролируются. Последний параметр в формате команды EDIT задает тип записей в наборе данных. Выбирается

одно из значений. По умолчанию принимается параметр DATA. Это означает, что работа будет вестись с набором данных, содержащим только данные. Параметры PL/1 и FORT используются для проведения синтаксического контроля строк набора данных, составленных на языках PL/1 и фортран соответственно. К параметру PL/1 можно добавить еще номера позиций строки *число 1* и *число 2*, которые показывают левую и правую границы предложения (по умолчанию этого параметра принимаются значения 2 и 72 соответственно). *Число 1* должно быть больше двух, и если в позициях 73—80 указываются номера строк, то не превышать 72. Параметры C48 (CHAR48) или C60 (CHAR60) определяют, какой алфавит используется: 48-символьный или 60-символьный. Параметр FORT используется, если набор данных создается на языке фортран. Буквы G или H задают уровень транслятора. Параметр CLIST показывает, что каждая строка набора данных содержит команду или подкоманду ДУВЗ, кроме LOGON, LOGOFF и EXEC. Если указан параметр DSLIST, то он показывает, что набор данных ДУВЗ содержит список имен наборов данных ДУВЗ. Параметр TEXT указывает, что набор данных представляет собой данные, но, в отличие от параметра DATA, вводимые строчные буквы не переводятся в прописные.

После того как по команде EDIT с параметром NEW дисплей установился в режим PB и на экране появился номер первой строки и курсор рядом с цифрой, например:

10 —

можно начинать вводить с пульта дисплея текст набора данных. Текст набирается по одной строчке. После того как строчка текста введена на экран, следует нажать клавишу ВВ (ВВОД), после чего текст будет передан в активную область памяти.

3. Для перевода дисплея из режима ввода в режим редактирования следует ввести команду ВВОД, путем нажатия клавиши ВВ. В режиме ввода система всегда сообщает абоненту номер следующей строки активного набора данных, под которым она будет введена в активную область памяти. При переходе из режима ввода в режим редактирования этот номер пропадает и появляется курсор в первой позиции экрана дисплея.

4. Обратный перевод из режима редактирования в режим ввода также обеспечивается вводом команды ВВОД, т. е. нажатием клавиши ВВ. На экране дисплея появляется номер следующей строки активного набора данных, которая будет вводиться абонентом. Переход из режима редактирования в режим ввода можно также выполнить с помощью подкоманды INPUT команды EDIT. Подкоманда INPUT используется в случае, если нужно ввести одну или несколько строк в ранее созданный набор данных или изменить текст строк набора данных. Формат подкоманды INPUT:

Подкоманда	Параметры
{INPUT} I	[номер строки] [{приращение} I] [P R] [NOP]

В формате подкоманды INPUT указывается номер первой строки изменяемой (или добавляемой) части набора данных. Если номер первой строки опущен, то это значит, что новый текст будет добавляться к концу существующего набора данных. Приращение показывает, как будут изменяться номера строк. Если приращение не указывается, то система изменяет номера с приращением, равным 10, например 10,20,... и т. д. Параметр I показывает, что новые строки не должны замещать существующие строки. Параметр R указывается, если вводимые строки должны замещать существующие строки, начиная с номера строки, указанного в команде. Параметр P (PROMPT) выдает на дисплей номер очередной строки, а параметр NOP (NOPROMPT) запрещает выдачу номеров строк, хотя в обоих случаях вводимые строки нумеруются системой.

Пример выдачи команды INPUT:

I 21 1 I — перевести дисплей в режим ввода, начиная со строки 21 (номер первой строки равен 21, к нему затем прибавляется приращение, равное 1). На экране дисплея появляется номер строки 21. Абонент вводит на строку 21 «новый текст I» и нажимает клавишу ВВ. Система выдает номер строки 22 и т. д. Пусть, например, абонент после появления строки 25 решил перейти снова в режим редактирования. Для этого нажимается клавиша ВВ.

5. Переход из режима редактирования в режим команд осуществляется с помощью подкоманды END команды EDIT. В результате этого перевода полностью уничтожается активный набор данных в ОП, который был ранее указан в команде EDIT. Формат подкоманды END:

Подкоманда	Параметры
{END} {E}	

В подполе параметров ничего не указывается. В ответ на ввод подкоманды END система устанавливает курсор на следующую строчку.

Если нужно перейти в режим команд и сохранить активный набор данных, находящийся в ОП, то следует предварительно выдать подкоманду SAVE команды EDIT. Формат подкоманды SAVE:

Подкоманда	Параметры
{SAVE} {S}	[имя набора данных] [K [EY] (пароль)]

Кроме имени набора данных, в формате подкоманды может быть указан пароль, если набор данных защищен. Если имя набора данных опущено, то система сохраняет набор данных, который ранее был указан в команде EDIT.

В ответ на ввод подкоманды SAVE, например S NAME1, где NAME1 — имя набора данных, система устанавливает курсор на следующую строчку.

Пример 12.1. Ввести задание во входной поток ЭВМ. Исходный режим дисплея — РК.

Из исходного состояния вводится команда

E UV NEW

и нажимается клавиша ВВ. Дисплей переходит в режим ввода. На экране дисплея появляется номер первой строки

10 —

и курсор, который показывает позицию, начиная с которой можно начинать вводить первую строку задания. После того как текст будет введен на экран, следует нажать клавишу ВВ. Текст первой строки будет передан в активную область памяти. На экране дисплея появится следующий номер 20. После того как текст задания будет введен в активную область памяти, можно:

а) передать задание во входной поток ЭВМ; для этого следует перевести дисплей в режим редактирования нажатием клавиши ВВ, выдать команду SUBMIT и еще раз нажать клавишу ВВ;

б) передать задание во входной поток ЭВМ и сохранить текст задания в библиотеке CRJE.SYSLIB; для этого следует перевести дисплей в режим редактирования нажатием клавиши ВВ, а потом выдать команду SAVE;

в) уничтожить текст нового задания; для этого следует перевести дисплей в режим редактирования, как было показано выше, а затем выдать команду END.

Пример 12.2. К старому набору данных UV добавить новый текст в 22—24 строки. Исходный режим дисплея — РК. Из исходного состояния выдается команда

E UV OLD

и нажимается клавиша ВВ. Дисплей переходит в режим редактирования, и на экране появляется курсор. Поскольку надо добавить текст, начиная со строки 22, то следует выдать команду

I 22 I P

В ответ система переходит в режим РВ и на экране появляется номер строки

22 —

строки, начиная с которой следует вводить текст программы. После того как текст строки 22 введен, надо нажать клавишу ВВ и так продолжать ввод до строки 25.

Управление выполнением заданий из режима команд РК. В этом режиме используются команды SUBMIT, STATUS, OUTPUT, CONTINUE, CANCEL и EXEC (см. рис. 12.3). Команда SUBMIT передает задание (или несколько заданий) для выполнения во входной поток в ЭВМ для пакетной обработки. Если абонент хочет получить информацию о задании, следует выдать команду STATUS. Результаты выполнения задания выдаются по команде OUTPUT. Если по какой-то причине выдача результатов прекратилась, то выдается команда CONTINUE, которая просит продолжить выдачу результатов. Команда CANCEL отменяет выполнение задания, введенного с данного дисплея. Команда EXEC предназначена для ввода задания, которое содержит команды абонента с характеристикой CLIST.

Формат команды SUBMIT:

Команда	Параметры
SUBMIT	<i>имя [(идентификатор пароль)] [имя]...</i>

Параметр *имя* заменяется именем набора данных, который вводится во входном потоке, либо набора данных, который описан в команде EDIT с параметром DSLIST (в этом случае вводится список имен наборов данных ДУВЗ, содержимое которых должно быть подсоединено к входному потоку). В скобках могут быть указаны идентификатор или идентификатор и пароль, если набор данных защищен. Пример выдачи команды SUBMIT:

SU JCL # ZAGRO.

В ответ система сообщений не выдает.

Следует отметить, что режим работы дисплея ПК после ввода команды SUBMIT не изменяется.

С помощью команды STATUS запрашивается информация о задании, переданном с помощью команды SUBMIT. Формат команды STATUS:

Команда	Параметры
{STATUS} {ST}	[<i>имя задания</i>]

Если имя задания опущено, то запрашивается информация о всех заданиях, переданных по команде SUBMIT с данного дисплея. Пример выдачи команды STATUS приводится ниже:

ST MONOREAD

В ответ система выдает сообщение, например:

INH385 STATUS MONOREAD EXECUTING — задание MONOREAD выполняется.

Команда OUTPUT служит для получения на дисплее результатов выполнения задания, переданного на ЭВМ с помощью команды SUBMIT. Формат команды OUTPUT:

Команда	Параметры
{OUTPUT} {O}	<i>имя задания</i> {MSG NOSMSG}

В формате команды указывается имя задания. Параметр MSG обеспечивает получение сообщений планировщика заданий в дополнение к основным результатам. Параметр NOSMSG используется, если сообщения планировщика заданий не нужны (версия 6.1). Пример команды:

O MASHA

где MASHA — имя задания.

В ответ система выдает на дисплей сообщение планировщика заданий и операторы языка управления заданиями. Последним сообщением выдается: INK361 END OF DATA — конец данных.

Команда CONTINUE используется, если вывод результатов, начатых по команде OUTPUT, прерываются. Формат команды CONTINUE:

Команда	Параметры
{CONTINUE} {CO}	{H[ERE]} {B[EGIN]} {N[EXT]}

Параметр H (HERE) указывает, что вывод результатов следует продолжать с того места, где было прерывание. Параметр B (BEGIN) сообщает, что вывод результатов следует продолжать с начала данного набора данных или с прерванного блока сообщений планировщика заданий. Параметр N (NEXT) указывает, что вывод результатов следует продолжать с начала следующего набора данных или со следующего блока сообщений планировщика заданий.

Команда CANCEL отменяет выполнение задания, введенного ранее абонентом. Формат команды CANCEL:

Команда	Параметры
{CANCEL} {C}	<i>имя задания</i>

В формате команды указывается только имя задания, которое должно быть отменено. После выдачи команды CANCEL режим работы дисплея не изменяется. Пример выдачи команды CANCEL:

C MODIF,

где MODIF — имя задания.

В ответ на ввод команды CANCEL, после того как планировщик снимет задание, система выдает абоненту сообщение:

INK000 JOB CANCELLED MODIF — задание MODIF снято.

Команда EXEC предназначена для выполнения группы команд, которая находится в наборе данных ДУВЗ с характеристикой CLIST. Формат команды EXEC:

Команда	Параметры
{EXEC} {EX}	<i>имя</i> [L NOL]

где *имя* — это имя набора данных, как в команде SUBMIT. Параметр L (LIST) указывается, если желательно, чтобы каждая выполняемая команда абонента выводилась на дисплей, а параметр NOL (NOLIST) означает, что выполняемые команды абонента не будут выводиться. По умолчанию система

считает, что выполняемые команды не выводятся на дисплей. В наборе данных могут находиться любые команды и подкоманды, кроме команд LOGON, LOGOFF и EXEC. В ответ на ввод команды, например:

EX LUBA L

где LUBA — имя набора данных, система сообщений не выдает. Однако сообщения появляются по мере того, как команды абонента встречаются в наборе данных. Например, пусть в наборе данных встречается команда STATUS. Тогда появится сообщение, например:

INH385 STATUS LUBA EXECUTING — задание LUBA выполняется.

Работа с наборами данных из библиотеки абонентов CRJE.SYSLIB. Команда LISTDS позволяет получить информацию о наборе данных, который находится в библиотеке абонентов ДУВЗ CRJE.SYSLIB. Формат команды LISTDS:

Команда	Параметры
$\left\{ \begin{array}{l} \text{LISTDS} \\ \text{LISTD} \\ \text{LDS} \end{array} \right\}$	<i>имя набора данных</i> [S[H]]

Если в команде указано только имя набора данных, то выдается информация о характеристике содержания набора данных и о том, что в позициях 73—80 каждой записи имеются номера строк. Параметр S (STATUS) запрашивает размер набора данных в байтах и пароль этого набора. Параметр H (HISTORY) запрашивает дату создания набора данных, дату внесения последних изменений, количество обращений к нему со времени создания по числу обращений по командам EDIT OLD, SUBMIT, EXEC и MERCE. В ответ на ввод команды

LDS TVSS,

где TVSS — имя набора данных, выдается сообщение:

INH320 DSNAME TYPE LEN LNO

INH320 TVSS DATA 80 YES

В колонке DSNAME сообщается имя набора данных TVSS, в колонке TYPE — параметр DATA команды EDIT, в колонке LEN — длина записей в байтах — 80, в колонке LNO указывается параметр YES.

С помощью команды LISTLIB в режиме редактирования на дисплее можно получить имена всех наборов данных, которые находятся в библиотеке пользователя ДУВЗ CRJE.SYSLIB и информацию по каждому набору данных. Формат команды LISTLIB:

Команда	Параметры
$\left\{ \begin{array}{l} \text{LISTLIB} \\ \text{L} \end{array} \right\}$	[S[H]]

Параметры S (STATUS) и H (HISTORY) имеют то же назначение, что и в команде LISTDS. В ответ на ввод команды L система выдаст сообщения, например:

```

INK320 DSHAME TYPE LEN LNO
INK320 CHIFF DATA 80 YES
INK320 DINPF DATA 80 YES

```

В колонке DSHAME указываются номера заданий, TYPE—характеристика данных, LEN — длина строк (80 байт), LNO — YES характеристика набора данных SEQ.

Команда DELETE удаляет набор данных из библиотеки CRJE.SYSLIB. Формат команды DELETE:

Команда	Параметры
{DELETE} D	{имя набора данных {имя набора данных [(имя раздела)]}'}

Имя раздела указывается для библиотечного набора данных. Первой составляющей в имени набора данных должен быть идентификатор абонента. Например, для уничтожения набора данных DINPF нужно выдать команду

D DINPF

В ответ система устанавливает курсор на следующую строчку. Убедиться в том, что набор данных DINPF уничтожен, можно, выдав подкоманду LISTLIB.

Режим редактирования. В табл. 12.4 приведен перечень подкоманд команды EDIT, предназначенных для редактирования и других видов работ. Все подкоманды можно разделить на такие же группы, как и команды абонента. В первой группе команд можно отнести:

1) подкоманду связи между оператором и абонентом (или абонентами): SEND;

2) подкоманду управления полями экрана дисплея: TABSET;

3) подкоманды изменения режимов работы дисплея: INPUT и END.

Ко второй группе отнесем подкоманды:

1) редактирования: CHANGE;

2) работы с наборами данных DELETE, FIND, LIST, MERGE, RENUMBER, SAVE и SCAN;

3) управления заданиями: SUBMIT и CANCEL.

Формат подкоманды SEND и TABSET полностью совпадает с форматами команд SEND и TABSET, которые были рассмотрены ранее. Подкоманды INPUT и END были рассмотрены выше в связи с рассмотрением вопроса изменения режимов работы дисплея.

Редактирование. Считаем, что на дисплее установлен режим редактирования. Необходимо отредактировать новый или старый набор данных. Ранее был рассмотрен порядок перехода из режима команд в режим редактирования для нового набора данных (команда EDIT, с указанием параметра NEW, и команда BBOD). Перейти от режима команд в режим редактирования для старого набора данных можно сразу по команде EDIT с указанием в ней параметра OLD.

Т а б л и ц а 12.4. Перечень подкоманд команды EDIT

Подкоманды	Назначение
Неявная подкоманда *) CA[NCEL] C[HANGE] D[DELETE]	Отменяет замещение или ввод строк в активном наборе данных Снимает ранее введенное задание Выполняет контекстное редактирование Удаляет строку или группу строк из активного набора данных
E[ND]	Переводит дисплей из режима редактирования в режим команд
F[IND]	Находит в активной области строку, в которой впервые встречается заданная последовательность символов (версия ОС 6.1)
I[INPUT]	Переводит дисплей из режима редактирования в режим ввода
L[IST]	Распечатывает набор данных и приводит к выдаче на дисплей определенных строк из активного набора данных
M[ERGE]	Используется для объединения и копирования строк активного набора данных и набора данных в библиотеке ДУВЗ.
REN [UMBER] S[AVE]	Перенумерует строки в активном наборе данных Сохраняет в библиотеке ДУВЗ копию активного набора данных
SC[AN]	Запрашивает синтаксический анализ операторов в активном наборе данных
SE[ND]	Посылает сообщение оператору ЭВМ или другому пользователю ДУВЗ
SUB[MIT]	Передаёт одно или несколько заданий в пакетную обработку
TAB[SET]	Устанавливает логический формат для ввода и физический для вывода на дисплей, оборудованный средствами табуляции

*) Неявная подкоманда — это случай, когда мнемокод подкоманды отсутствует, т. е. в формате указываются только параметры.

Подкоманда контекстного редактирования CHANGE позволяет заменить последовательность символов в строке или группе строк активного набора данных. Формат подкоманды CHANGE:

Подкоманда	Параметры
{CHANGE} C	номер строки [номер строки] Δ текст 1 Δ текст 2 [A[LL]]

В формате подкоманды указывается номер строки или номера группы строк, которые должны быть изменены. Вместо параметра *текст1* указывается старый текст, а вместо параметра *текст2* — новый текст. Длина текстов не должна превышать 40 символов. Знак Δ является разделителем. В качестве разделителя может быть использован любой нецифровой символ ДКОИ, но не запятая. Разделитель не должен встречаться в текстах. Если разделителем является дефис, то им не должен заканчиваться текст. В этом случае следует

ввести в текст дополнительный пробел в качестве последнего символа. Параметр ALL указывает, что при каждом обращении к активному набору производится изменение старого текста. Если параметр ALL опущен, то текст изменяется только один раз. Пример выдачи команды CHANGE:

C 120\DINPF\DIRRF\

в строке 120 заменяется старый текст DINPF на новый текст — DIRRF.

В ответ система выдает сообщение:

INH3771 PROCESSING COMPLETE4 LINE TRUNCATED, где число 4 сообщает о количестве внесенных изменений.

Работа с активными наборами данных. Подкоманда FIND находит в активной области строку, где впервые встречается нужный текст. Формат команды FIND:

Подкоманда	Параметры
{FIND} F	[номер строки] Δ искомый текст Δ

В формате подкоманды указывается номер строки в активной области ОП, начиная с которой следует начать поиск. Если номер строки не указан, то поиск начинается с начала активной области. Искомый текст может содержать не более 56 символов. В качестве разделителя Δ можно использовать любой символ, кроме пробела, но не встречающийся в искомом тексте, например:

F \//_____DISP=(NEW,CATLG)\

найти указанный текст. В ответ система выдает найденный текст и номер строки, в которой он встречается:

100\//_____DISP=(NEW,CATLG)\

Подкоманда LIST позволяет вывести текст определенных строк из активной области на дисплей. Формат подкоманды LIST:

Подкоманда	Параметры
{LIST} L	[номер строки 1 [номер строки 2]] [NUM NONUM]

В формате подкоманды указывается номер строки 1, начиная с которой следует выводить текст, и номер строки 2 — последняя строка в тексте, которая выводится на дисплей. Если не указан номер строки 2, то выводится только одна строка. Параметр NUM указывается, если вместе с текстом нужно вывести номера выводимых строк, а параметр NONUM — если номера строк текста не выводятся. После выдачи команды, например:

L 80

выдается текст строки 80:

_____80.// «текст строки»

При выдаче команды

L 80 100

выводятся три строки в номерах 80, 90 и 100.

Подкоманда DELETE позволяет удалить строку или группу строк из активного набора данных. Формат подкоманды DELETE:

Подкоманда	Параметры
{DELETE} D	[номер строки 1 [номер строки 2]]

В формате подкоманды указываются номера первой и последней строк *номер строки 1* и *номер строки 2* соответственно, которые удаляются. Если указан *номер строки 1*, то удаляется весь текст, начиная с этой строки. Если параметры в команде отсутствуют, то удаляется только одна последняя строка. После выдачи команды, например:

D 110

где 110 номер удаляемой строки, содержащий, например, текст

110 //R

система удаляет этот текст.

Неявная подкоманда (в формате отсутствует мнемокод подкоманды) используется для удаления, замещения или ввода строки в активном наборе данных. Формат подкоманды:

Подкоманда	Параметры
	<i>номер строки 1</i> {текст}

В формате неявной подкоманды указывается номер удаляемой или вставляемой строки. Текст указывается для вставляемой строки. Длина текста может иметь в длину до 80, 72 или 144 символов в зависимости от того, какая характеристика длины задана была в команде EDIT: NOSEQ, SEQ или LONG соответственно. Если текст в подкоманде отсутствует, то удаляется одна строка, номер которой указан в команде.

Подкоманда SCAN запрашивает синтаксический анализ операторов в активном наборе данных, если в команде EDIT указаны параметры PL/1 или FORT, т. е. ведется работа с наборами данных, написанными на языке PL/1 или фортран. Формат подкоманды SCAN:

Подкоманда	Параметры
{SCAN} {SC}	[номер строки 1 [номер строки 2]] {ON OFF}

В формате подкоманды указывается *номер строки 1*, начиная с которого следует вести синтаксический контроль, и *номер строки 2*, на которой этот

контроль следует закончить. Если указан только *номер строки* 1, то проверяется весь текст, начиная с этого номера и до конца. Если номера строк опущены, то проверяется весь текст. Параметр ON указывается, если все последующие строки, вводимые в режиме ввода, будут синтаксически проверяться, а параметр OFF — если проверка не нужна.

Подкоманда RENUMBER позволяет перенумеровать строки в активном наборе данных. Формат подкоманды RENUMBER:

Подкоманда	Параметры
{RENUMBER} {R}	[номер строки 10] [приращение 10]

В формате подкоманды указывается номер строки, начиная с которой текст будет перенумерован. По умолчанию система нумерует строки, начиная с десятой, с приращением 10. Можно указать приращение номеров, которое отличается от десяти. Если набор данных имеет характеристику SEQ, которая была указана в команде EDIT, то номер помещается в позиции с 73-й по 80-ю. В ответ на ввод команды R сообщений не выдается.

Подкоманда MERGE используется для соединения активного набора данных с набором данных из библиотеки пользователя ДУВЗ и для копирования активного набора данных. Формат команды MERGE:

Подкоманда	Параметры
{MERGE} {M}	{имя набора данных (идентификатор пароля)} [номер строки 1 номер строки 2] [номер строки]

Параметры *имя набора данных*, *идентификатор* и *пароль* такие же, как в команде SUBMIT. Звездочка * указывается в случае, если копируемые строки находятся внутри активного набора данных. *Номер строки 1* показывает начало текста, а *номер строки 2* — конец текста, который будет вставлен в активный набор данных. Если номера строк опущены, то вставляется весь набор данных целиком. *Номер строки* указывается, если в активный набор данных вставляются новые строки. Если номер строки опущен, то строки добавляются к концу активного набора данных. Например, пусть следует вставить строки с 40 по 140 из набора данных ADD, который находится в библиотеке CRJE.SYSLIB:

M ADD 40 140

Сообщения в ответ на ввод команды не выдаются.

Подкоманда SAVE, предназначенная для сохранения в библиотеке CRJE копии активного набора данных, была рассмотрена ранее.

Управление выполнением задания. После того как набор данных отредактирован, он вводится из режима редактирования в ЭВМ с помощью подкоманды SUBMIT (см. рис. 12.3).

Формат подкоманды SUBMIT:

Подкоманда	Параметры
{SUBMIT} {SU	{ <i>имя набора данных</i> } {(идентификатор <i>пароль</i>)} [<i>имя набора данных</i>]

Параметры *имя набора данных*, *идентификатор* и *пароль* такие же, как в команде SUBMIT. Параметр * указывается, если активный набор данных передается в ЭВМ во входном потоке или если активный набор данных имеет характеристику DSLIST, которая была указана в команде EDIT. Всего можно перечислить до 10 имен.

Подкоманда CANCEL предназначена для отмены ранее введенного задания. Она аналогична команде CANCEL, рассмотренной выше, за исключением сокращенного написания кода подкоманды (см. табл. 12.4).

Глава 13. РАБОТА ОПЕРАТОРА С ПРОГРАММОЙ КРОС

13.1. Общие сведения

Программа КРОС включается в состав ОС и выполняется под ее управлением. Она используется в системах MFT, MVT и SVS. КРОС облегчает работу оператора за счет общего уменьшения числа команд по сравнению с ОС. Программы пользователей выполняются в так называемых *логических разделах*. Система оперирует понятиями фиктивных устройств ввода-вывода, т.е. на самом деле отсутствующих на реальной ЭВМ. Эти фиктивные устройства в случае необходимости используются системой при планировании работы с системными программами ввода RDR и вывода WTR. Сами системные программы ввода и вывода, а также инициаторы запускаются автоматически по мере необходимости (в противоположность неавтоматическому запуску, см. гл. 6). За счет этого увеличивается производительность ОС. Основная часть команд КРОС предназначена для работы с устройствами вывода. Во многих случаях при объеме памяти ЭВМ более 256К выгоднее иметь один раздел КРОС и один раздел для выполнения проблемных программ, чем два раздела для проблемных программ. Степень увеличения производительности за счет использования КРОС зависит от типа ЭВМ и решаемых на ней задач.

Программа КРОС состоит из постоянной (резидентной) части, которая помещается системой в ОП, и нерезидентной, которая находится на резидентном диске в наборе данных SYS1.KROCOLIB. Для очередей и контрольных точек требуется промежуточный набор данных SYS1.KROCASE на диске (дисках), который имеет серийный номер SPOOL1 (или SPOOLx, где $x = 2, \dots, n$).

13.2. Язык управления заданиями в КРОС

Операторы языка управления заданиями ОС, рассмотренные в гл. 3 и 9, практически полностью, без изменения, используются в программах, которые выполняются на ЭВМ под управлением КРОС. Дополнительно введены некоторые изменения в параметр *учетная информация* в операторе JOB. Кроме того, введены четыре новых оператора. MESSAGE — выдать сообщение оператору, PRIORITY — задать приоритет, ROUTE — указать устройство для вывода информации, SETUP — перечислить необходимые тома на шаге выполнения задания.

Параметр *учетная информация* задается в следующем виде: (номер, комната[время] [строки] [карты] [номер формата] [количество экземпляров] [журнал] [размер страницы])

В параметре обязательно указываются подпараметры *номер* и *комната*, в которых можно указать до четырех алфавитно-цифровых знаков. Остальные

данные могут быть опущены. В них задается следующая информация: *время* — время выполнения задания в минутах; *строки* — объем вывода информации на печать в тысячах строк; *карты* — количество выводимых перфокарт; *номер формата* — специальный формат печати (от одного до четырех знаков); *количество экземпляров* — количество экземпляров, которое выдается на печать (до двух знаков; по умолчанию — один экземпляр); *журнал* — любая буква, кроме N, если журнал ведется, и буква N, если журнал не ведется (или по умолчанию); *размер страницы* — количество строк на одной странице при выводе информации на АЦПУ. Если необязательные данные не указываются, то используются значения, установленные при генерации КРОС.

Оператор PRIORITY назначает приоритет. Значения приоритета задаются числами от 0 до 15 или указывается знак *. Формат оператора PRIORITY:

Операция	Параметры
/* PRIORITY	приоритет

Идентификатор /* указывается в первых двух колонках, а значение приоритета — в колонках 16 и 17. Пример записи этого оператора PRIORITY:

/* PRIORITY 5

Если используется знак * или оператор PRIORITY отсутствует, то приоритет вычисляется системой в соответствии со значением, указанным в параметре PRTY оператора JOB. Оператор PRIORITY помещается в задании перед оператором JOB.

Оператор ROUTE позволяет указать устройство для вывода информации. Формат оператора ROUTE:

Операция	Параметры
/* ROUTE	{PRINT PUNCH} {LOCAL PRINTER _n PUNCH _n }

Первый параметр PRINT означает, что информация выводится на печать, а PUNCH — на перфоратор. Запись первого параметра должна начинаться с колонки 10. Второй параметр задает устройство: LOCAL — любое локальное (не удаленное) устройство, PRINTER_n — печатающее устройство с номером *n*, PUNCH_n — перфоратор с номером *n*. Запись второго параметра должна начинаться с колонки 16. Оператор (операторы) ROUTE должен располагаться сразу же за оператором JOB.

Оператор MESSAGE используется для выдачи сообщений оператору во время ввода задания. Формат оператора MESSAGE:

Операция	Параметры
/* MESSAGE	текст

Текст сообщения может быть записан с 12-й по 71-ю колонку. Если оператор MESSAGE будет помещен между заданиями, то в сообщение включается имя устройства ввода; если оператор MESSAGE помещается после оператора JOB или ROUTE, то следует включить в сообщение имя задания, например:

/* MESSAGE ASSEMB ВЫЗВАТЬ ИВАНОВА

Оператор SETUP служит для указания номеров томов. Формат оператора SETUP:

Операция	Параметры
/* SETUP	регистрационные номера томов

Эти номера выводятся на консоль оператора во время ввода задания, чтобы сообщить ему о том, какие тома требуются заданию. Само задание помещается в очередь задержанных заданий HOLD. Номера томов записываются в колонках 16—71 и разделяются запятыми. В задании он помещается после операторов JOB, ROUTE и MESSAGE. Пример записи оператора SETUP:

/* SETUP PTOM135,PTOM136,PTOM137 — заданию требуются тома PTOM135, PTOM136, PTOM137.

13.3. Общая характеристика команд КРОС

Перечень команд, используемых в КРОС, приведен в табл. 13.1. Справа в таблице указаны команды ОС, которые аналогичны командам КРОС.

Ниже показан формат команд, если они вводятся с пульта пишущей машинки или дисплей-консоли:

Команда	Параметры
⌘ <i>мнемокод команды</i>	<i>параметры, разделенные запятыми</i>

Все команды начинаются с идентификатора ⌘. Мнемокод команды следует непосредственно за идентификатором, хотя можно сделать один или несколько пробелов. В подполе операндов указываются параметры, разделенные запятыми. Параметры могут следовать непосредственно за мнемокодом команды.

По назначению команды КРОС можно разделить на следующие группы:

- управление системой КРОС: ⌘S и ⌘P;
- информационная связь с системой: ⌘D;
- управление устройствами ввода-вывода: ⌘S, ⌘P, ⌘Z, ⌘C, ⌘E, ⌘I, ⌘F, ⌘N, ⌘T;
- управление логическими разделами: ⌘S, ⌘P, ⌘T;
- управление порядком прохождения заданиями: ⌘C, ⌘P, ⌘H, ⌘A, ⌘T.

Команды оператора КРОС и ОС во входном потоке. При работе с КРОС команды оператора ОС (табл. 3.5) должны быть помещены внутри задания. Команды КРОС (табл. 13.1) должны быть размещены

Таблица 13.1. Команды КРОС

№ п/п	Команда КРОС	Назначение	Аналогичная* команда в ОС
1	⊗A	Разрешить выборку заданий	{RELEASE} A
2	⊗B	Продублировать вывод нескольких страниц	—
3	⊗C	Прекратить процесс	{CANCEL} C
4	⊗D	Получить информацию	{DISPLAY} D
5	⊗E	Продублировать вывод информации	—
6	⊗F	Пропустить несколько страниц при выводе информации	—
7	⊗H	Временно задержать выборку заданий	{HOLD} H
8	⊗I	Прекратить вывод информации на печать	—
9	⊗N	Выдать дополнительный экземпляр выходной информации	—
10	⊗P	Остановить процесс	{STOP} P
11	⊗S	Начать процесс	{START} S
12	⊗T	Изменить классы и приоритеты; поместить задание в очередь	—
13	⊗Z	Остановить работу устройства на некоторое время	—

*) Остальные команды в ОС не имеют аналогов в КРОС.

между заданиями. Если они попадут внутрь задания, то система их игнорирует. Команда записывается с 4-й по 71-ю колонки. В колонках 1—3 должен быть указан идентификатор /*⊗. Если в колонке продолжения 72 указан символ N, то текст команды не выводится на консоль.

13.4. Управление программой КРОС

Программа КРОС запускается во время загрузки системы с помощью команды START, как каталогизированная процедура (см. гл. 8):

S КРОС

При запуске программы КРОС должны соблюдаться следующие условия: программы системного ввода, вывода и инициатор должны быть остановлены (это значит, что при запуске ОС в параметре SET должно быть указано AUTO=NONE или AUTO=NNN); устройства ввода-вывода должны находиться в состоянии КОМПЛЕКС; поставлен диск (диски) для набора данных SYS1.KROCASE, который используется в КРОС в качестве промежуточной памяти.

В ответ на ввод команды S КРОС система выдает запрос:

*02 X SPECIFY КРОС OPTIONS — — — — КРОС VERSION 3.1 — задать параметры программы КРОС.

Параметры выбираются из следующего набора: FORMAT — необходимо форматизовать набор данных SYS1.КРОСАСЕ; NOFMT — форматизация набора данных SYS1.КРОСАСЕ не нужна; COLD — очереди создаются заново (предполагается, что задан FORMAT); WARM — сохраняется старый (предыдущий порядок очереди; REQ — КРОС начинает работу только после ввода команды X S (до ввода X S оператор может изменить параметры, изменить состояние устройств ввода-вывода и т. д.); NOREQ — КРОС начинает работу сразу же после выполнения подготовительных работ; U — использовать стандартный режим КРОС (NOFMT, WARM, REQ); NONE — так же, как U. Имеются еще режимы, которые оператор может ввести только с разрешения системного программиста.

Стандартным ответом при первом запуске является ответ:

R 02,'FORMAT'

В ответ система выдает сообщение:

IEE600I REPLY TO 02 IS; FORMAT SPOOL1 IS BEING FORMATTED — ответ 01 принят системой; формат в наборе данных SYS1.КРОСАСЕ форматизован.

При повторных запусках оператор отвечает:

R 02,'U'

Далее система выдает сообщение:

IEE600I REPLY TO 02 IS; 'U'

и

ENTER КРОС REQUEUSTS

— ответ 02 принят; будут использованы установленные ранее параметры КРОС. После получения ответа IEE600I можно вводить команды КРОС, в частности команду

X S

которая активизирует все функции КРОС. Параметры в команде отсутствуют. В случае принятия команды системой она выдает сообщение:

X 19.01.32 OK — команда принята, время 19.01.32;

X*19.01.32 ALL AVAILABLE FUNCTIONS COMPLETE — все установленные функции КРОС активизированы полностью.

Для того чтобы остановить работу программы КРОС, следует выдать команду

XР КРОС

В ответ система выдает сообщение, например:

IEF161I IAO READER CLOSED — программа RDR с адресом IAO закрыта.

При работе с программой КРОС используются как команды ОС (гл. 4—8), так и команды КРОС.

13.5. Информационная связь с системой

С помощью команды $\mathcal{Q}D$ оператор может получить одноразовую, текущую информацию. Формат команды:

Команда	Параметры
$\mathcal{Q}D$	$\left\{ \begin{array}{l} A \\ N [\text{очередь}] \\ Q [\text{очередь}] \\ J \text{ номер задания} \\ \text{'имя'} \\ I [n] \\ D \\ U \\ F \end{array} \right\}$

По команде *)

$\mathcal{Q}D A$

оператор получает информацию о заданиях, которые выполняются в данный момент времени. В ответ система выдаст сообщение, например:

$\mathcal{Q}*18.44.27 \text{ JOB } 8 \text{ PRINTZK ON PRINTER2 PRIO } 6$ — текущее время 18.44.27; задание 8 с именем PRINTZK выводится на АЦПУ (номер 2) с приоритетом 6. Пример еще одного сообщения:

$\mathcal{Q}*16.11.15 \text{ JOB } 68 \text{ RUID EXECUTING J PRIO } 7$ — текущее время 16.11.15; задание 68 с именем RUID обрабатывается инициатором в классе J с приоритетом 7.

Если выполняющихся заданий нет, то выдается сообщение:

$\mathcal{Q}*16.11.15 \text{ NO ACTIVE JOBS}$ — в системе нет активных заданий.

Список имен заданий, находящихся во всех очередях, выдается по команде

$\mathcal{Q}D N$

Если задания в очередях имеются, то система в ответ на ввод этой команды выдает сообщения, например:

$\mathcal{Q}*20.01.40 \text{ JOB } 17 \text{ PRINTZK AWAITING EXEC M PRTO } 7$ — время 20.01.40; задание 17 с именем PRINTZK ждет выполнения в классе M, приоритет 7;

$\mathcal{Q}20.01.40 \text{ 1 PERCENT SPOOL UTILIZATION}$ — 1% использования диска SPOOL1.

Список имен заданий, которые находятся в определенных очередях, выдается по команде

$\mathcal{Q}D N[\text{очередь}]$

где вместо параметра *очередь* указывается одно из следующих значений: XEQ — очередь JOB; XEQ x — очередь JOB класса x ; PRT — очередь SOUT при выводе на печать АЦПУ; PUN — очередь SOUT при выводе на

*) Здесь и далее при записи команд КРОС допускается ввод параметра непосредственно за командой без пробела.

перфокарты; HOLD — очередь задержанных заданий HOLD. Пусть, например, надо получить имена заданий, которые находятся в очереди JOB. Тогда следует выдать команду

QD N, XEQ

В ответ система выдает сообщение, аналогичное предыдущему.

Если в очереди (очередях) нет заданий, то выдается сообщение:

*16.15.06 0 PERCENT SPOOL UTILIZATION — 0% использования диска SPOOL1.

Информация о количестве заданий в очередях, если они там имеются, выдается по команде

QD Q

например:

*18.57.27 1 XEQ J — одно задание в очереди JOB класса J;

*18.57.27 4 PERCENT SPOOL UTILIZATION — 4% использования диска SPOOL1.

Можно запросить информацию о количестве заданий, которое находится в определенной очереди. В этом случае указывается

QD Q, очередь

где параметр *очередь* принимает значения: XEQ, XEQx, PRT и PUN, например:

QD Q, XEQ — запрос о количестве заданий в очереди JOB.

Если в очередях заданий нет, то система выдает те же сообщения, что и при вводе команды QD N.

По команде

QD J номер задания

выдается информация о заданиях, их номерах и состоянии, например:

QD J10

В ответ система выдает сообщение, например:

*19.04.34 JOB 10 LOAD09 EXECUTING J PRIO 7 — задание 10 с именем LOAD09 выполняется в классе J с приоритетом 7.

Аналогичную информацию о заданиях можно получить и по команде

QD 'имя задания'

например:

QD 'KAPTA'

В ответ система выдает сообщение, например:

*17.25.52 JOB 81 KAPTA AWAITING PRINT0 PRIO9 — время 17.25.52; задание 81 с именем KAPTA ждет АЦПУ с номером 0, приоритет 9. И далее, как только результаты задания начинают выводиться на АЦПУ, выдается сообщение:

*17.28.30 JOB 81 PRINTING ON PRINTER1 — задание 81 выводится на АЦПУ с номером 1.

Информация об инициаторах выдается при вводе команды

QD I

в виде сообщений, например:

⌘*16.13.47 INIT1 (INACTIVE)=F

:

⌘*16.13.47 INIT10 (INACTIVE)=FIJKLMNOA

Инициатор 1 свободен и может обрабатывать задания в классе F. Инициатор 10 также находится в свободном состоянии и может обрабатывать задания в классах F, I, J, K, M, N, O, A.

Можно получить данные только об одном каком-либо инициаторе, указав в команде его номер, например:

⌘D I12

Система выдает сообщение, аналогичное предыдущему.

Информация о всех устройствах прямого доступа выдается по команде

⌘D D

В ответ система выдает сообщения, например:

⌘*16.49.01 133 PPOMO5

⌘*16.49.01 134 PPOMO4

⌘*16.49.01 220 APP

⌘*16.49.01 221 TSTLIB

На устройстве 133 установлен том PPOMO5, на 134 — том PPOMO4, на 220 — том APP, на 221 — том TSTLIB.

Информация о всех устройствах, кроме устройств прямого доступа, выдается по команде

⌘D U

В ответ система выдает сообщения, например:

⌘*16.17.55 READER1 00C INACTIVE

⌘*16.17.55 READER2 *** DRAINED

⌘*16.17.55 PRINTER1 00F INACTIVE

⌘*16.17.55 PRINTER2 *** DRAINED

⌘*16.17.55 PUNCH1 *** DRAINED

⌘*16.17.55 TAPE1 *** DRAINED

⌘*16.17.55 INTRDR1 160 INACTIVE

⌘*16.17.55 INTRDR2 161 INACTIVE

⌘*16.17.55 INTRDR3 162 INACTIVE

Устройство считывания READER1 с адресом 00C свободно (INACTIVE); устройство считывания READER2 не включено (***) и работа с этим устройством запрещена (DRAINED); АЦПУ (PRINTER1) с адресом 00F свободно; АЦПУ (PRINTER2) не включено; перфоратор PUNCH1 не включен; магнитная лента TAPE1 не включена; устройства считывания с дисков INTRDR1, INTRDR2 и INTRDR3 с адресами 160, 161 и 162 свободны.

При вводе команды

⌘D F

выдается информация о количестве заданий, для которых требуются специальные форматы вывода (см. команду ⌘T).

Если таких заданий нет, то выдается сообщение:

⌘*16.50.03 FORMS QUEUE EMPTY — заданий, требующих специальных форматов, нет.

Сведения о запросах, на которые оператор не ответил, оператор получает по команде D R (см. п. 4.1.1). Информация о верхней и нижней границах областей, занимаемых системными очередями, и размер ОП выдаются по команде D SQA (см. п. 4.1.1). Из информационных команд оператор может пользоваться командами MONITOR, STOP, STOPMN, REPLY, LOG, WRITELOG. В команде REPLY могут быть использованы дополнительные форматы: а) мнемонический код команды можно опустить; б) запятую в формате можно опустить; в) если вводимый текст содержит только латинские буквы, то апострофы опускаются; г) если вводимый текст начинается с цифр, то следует поставить пробел или запятую между номером запроса и самим текстом. Пусть, например, оператором получен запрос:

*02 IEF238A REPLY DEVICE NAME,OR 'CANCEL'

В ответе оператор решил указать номер устройства 281:

2 281

Система сообщает о том, что ответ оператора на запрос 02 принят:

IEE600I REPLY TO 02 IS: 281

13.6. Управление устройствами ввода-вывода

В распоряжении оператора имеются команды ⌘S, ⌘P, ⌘Z, ⌘C, ⌘E, ⌘I, ⌘B, ⌘F, ⌘N, ⌘T, пользуясь которыми он может управлять устройствами ввода-вывода.

В качестве параметров в командах, работающих с устройствами ввода-вывода, используются определенные символические имена, перечень которых приведен в табл. 13.2.

Таблица 13.2. Символические имена для устройств ввода-вывода

Полное имя	Сокращенное имя	Название устройства
CONSOLE n	CON n	Консоль КРОС с номером n
	CON	Консоль ОС ЕС
PRINTER n	RDI n	Устройство ввода с дисков и МЛ с номером n
	PRT n	Печатающее устройство с номером n (АЦПУ)
	PUN n	Перфокарточное устройство с номером n для вывода результатов
READER n	RDR n	Перфокарточное считывающее устройство с номером n для ввода потока заданий
TAPE n	TPE n	Накопитель на магнитной ленте с номером n для ввода потока заданий

Значение n , которое указывается в именах устройств в табл. 13.2, может изменяться от 1 до некоторого максимального значения, которое устанавливается при генерации КРОС. Для дисплей-консолей, внутренних устройств

ввода и перфокарточных устройств вывода следует использовать только сокращенное имя.

Устройства ввода-вывода при работе с КРОС могут находиться в одном из четырех состояний: ACTIVE — устройство занято обработкой очередного задания; DRAINING — устройство занято обработкой очередного задания (после окончания обработки переводится в состояние DRAINED); INACTIVE — устройство свободно (при обращении к нему переводится в состояние ACTIVE); DRAINED — работа с устройством запрещена.

В табл. 13.3 приведены форматы команд αS , αP , αC и αZ , которые используются при запуске и остановке устройств ввода-вывода.

Таблица 13.3. Форматы команд αS , αP , αC и αZ

Команда	Параметры	Исходное состояние	Состояние после выполнения команды
αS	TRЕп, адрес	DRAINING, DRAINED	Устройство ввода в ACTIVE
	устройство [,устройство,...]	DRAINING	Устройство вывода в ACTIVE
		DRAINED	Устройство вывода в INACTIVE
αP	устройство [,устройство,...]	ACTIVE	DRAINING
		INACTIVE	DRAINED
αZ	устройство [,устройство,...]	ACTIVE	ACTIVE
		INACTIVE	INACTIVE
αC	устройство [,устройство,...]	ACTIVE	ACTIVE
		INACTIVE	INACTIVE

Команда αS разрешает выделить устройство (устройства) обрабатывающей программе. Формат команды αS приведен в табл. 13.3; в нем указывается символическое имя (имена), взятое из табл. 13.2. В формате команды можно указать до пяти устройств. В формате команды, предназначенном для запуска магнитной ленты, указывается параметр TRЕп и физический адрес устройства.

Команда αS переводит устройства ввода в состояние ACTIVE, а устройства вывода — в ACTIVE или INACTIVE. Далее система просматривает выходную очередь SOUT, и если там имеется задание для вывода информации, то переводит устройство в состояние ACTIVE. Пусть устройство ввода RDR1 находится в состоянии DRAINED. Перевести его в состояние ACTIVE:

αS RDR1

В ответ система выдает сообщения:

α *15.39.36 ОК — команда принята.

*IEA000I 00C,INT REQ,02,0200,4000,,,КРОС 550 — устройство считывания не готово к считыванию данных с перфокарт.

Команда $\text{Q}P$ позволяет остановить работу устройства (устройств). Формат команды приведен в табл. 13.3. В формате команды перечисляются устройства, работа которых должна быть остановлена. По этой команде устройство, если оно находилось в состоянии ACTIVE, переводится в состояние DRAINING, а если находилось в состоянии INACTIVE, то переводится в состояние DRAINED. Сообщения, выдаваемые системой, несколько отличаются. Так, например, при вводе команды

$\text{Q}P \text{ PRT1}$

система выдает сообщение, если устройство находилось в состоянии ACTIVE:

$\text{Q}*15.35.00 \text{ OK}$ — команда принята системой.

Если устройство останавливается из состояния INACTIVE, то выдается сообщение, например:

$\text{Q}P \text{ PRT2}$

$\text{Q}*15.36.47 \text{ OK}$ — команда принята системой:

$\text{Q} 15.36.47 \text{ PRINTER2 IS DRAINED}$ — устройство переведено в состояние DRAINED. Аналогичные сообщения выдаются при останове других типов устройств.

Команда $\text{Q}Z$ позволяет немедленно остановить работу устройства. Формат команды приведен в табл. 13.3. Состояние устройства не изменяется.

После выдачи команды Z, например:

$\text{Q}Z$

выдается сообщение:

$\text{Q}*06.58.09 \text{ OK}$

Работа на устройстве может быть продолжена, если выдать команду

$\text{Q}S \text{ PRT1}$

Команда $\text{Q}C$ прекращает выдачу информации на печать. Устройство переходит к выводу следующего результата по порядку. Здесь приводится пример выдачи команды $\text{Q}C$:

$\text{Q}C \text{ PRT1}$

В ответ система выдает сообщения:

$\text{Q}*19.38.31 \text{ OK}$

$\text{Q} 19.38.31 \text{ PRINTER1 DELETED}$

Устройство остается в состоянии ACTIVE.

Все команды, которые описаны ниже, относятся к некоторым режимам работы устройств вывода, расширяющих их возможности.

Команда $\text{Q}T$ имеет формат, задающий для перфокартного устройства вывода или АЦПУ некоторые параметры:

Операция	Параметры
$\text{Q}T$	$\text{устройство } [C=1] \left[.F = \begin{Bmatrix} R \\ S \\ A \\ \text{nnnn} \end{Bmatrix} \right] \left[.S = \begin{Bmatrix} Y \\ N \end{Bmatrix} \right]$

Параметр $C=1$ означает, что печать будет производиться через строку (пропуск одной строки). Отменить этот режим можно с помощью команды $\text{X}S$. Параметр F задает специальные форматы при печати: R — стандартная форма с возможностью изменения по требованию; S — стандартная форма; A — специальная форма; $nnnn$ — от 1 до 4 цифр, указывающих номер формата. Параметр S показывает, как будут выводиться страницы-разделители на АЦПУ или перфокарты-разделители на перфокарточном устройстве: Y — разделители будут выводиться; N — разделители не будут выводиться.

Команда $\text{X}E$ позволяет прекратить вывод информации текущего задания и начать его с самого начала. Формат команды $\text{X}E$:

Команда	Параметры
$\text{X}E$	<i>устройство</i> [<i>устройство</i> ,...]

В ответ на ввод команды

$\text{X}E \text{ PRT1}$

система выдает сообщение, например:

$\text{X}*13.06.58 \text{ OK}$

$\text{X}13.06.58 \text{ PRINTER1 RESTARTED}$ — печатающее устройство 1 начнет вывод информации с самого начала.

Команда $\text{X}I$ прерывает печать текущей страницы. Формат команды $\text{X}I$

Команда	Параметры
$\text{X}I$	PRTn_1 [$\text{PRTn}_2, \dots$]

Далее печать будет повторена с предыдущей страницы. При выдаче команды

$\text{X}I \text{ PRT1}$

система выдает сообщения:

$\text{X}*13.44.35 \text{ OK}$

$\text{X}13.44.35 \text{ PRINTER1 SUSPENDED}$ — печатающее устройство 1 повторяет печать с предыдущей страницы;

$\text{X}13.44.35 \text{ JOB 285 PRINTING ON PRINTER1}$ — результаты задания 285 выводятся на печатающее устройство 1.

Команда $\text{X}B$ позволяет повторить несколько последних страниц, например, в случае замятия бумаги. Формат команды $\text{X}B$:

Команда	Параметры
$\text{X}B$	PRTn_1 [$\{ \text{счетчик} \}_D$] [$\text{PRTn}_2, \dots$]

В параметре *счетчик* указывается количество страниц, которое следует повторить (от 1 до 9999). Если в параметре указать число более 9999, то вывод информации будет повторен с самого начала, так же, как если

указать параметр D. Здесь приводится пример ввода команды XB и сообщения, выдаваемые системой в ответ:

```
XB PRT1,D
X*22.13.44 OK
X 22.13.44 PRINTER1 BACKSPACED
```

Команда XF предназначена для того, чтобы пропустить вывод нескольких страниц при печати или прекратить вывод текущего задания. Формат команды XF :

Команда	Параметры
XF	$\text{PRTn} [\{ \text{счетчик} \}] [\text{PRTn}, \dots]$

Параметр *счетчик* задается так же, как в команде B. Если задается число, большее 9999, то вывод информации прекращается и начинается вывод информации следующего задания. Вывод информации прекращается также, если указать параметр D.

После выдачи команды, например:

```
XF P RT1
```

выдаются сообщения:

```
X*14.05.38 OK
X 14.05.38 PRINTER1 FWD — SPACED — вывод текущего задания
FWD на печатающем устройстве 1 прекращен.
```

Команда XN позволяет вывести дополнительный экземпляр задания, во время печати которого выдается эта команда. Формат команды XN :

Команда	Параметры
XN	<i>устройство</i> [<i>устройство</i> , ...]

Вывод текущей информации заканчивается, после чего он будет повторен. В ответ на ввод команды, например:

```
XN PRT1
```

система выдает сообщения:

```
X*17.35.56 OK
X 17.35.56 PRINTER1 REPEATED — информация, выводимая с
печатающего устройства 1, будет повторена.
```

13.7. Управление логическими разделами

Чтобы узнать, какое количество инициаторов действует в системе, следует выдать команду XD I . Ниже рассматриваются команды, управляющие работой инициаторов.

Команда XS запускает инициаторы во всех логических разделах. Формат команды XS :

Команда	Параметры
XS	$I[n]$

Если указывается только параметр I , то запускаются инициаторы во всех логических разделах. Если добавляется параметр n , где n — номер логического раздела, то запускается инициатор в каком-то одном конкретном логическом разделе. В случае успешного запуска инициаторов по команде, например:

$\text{XS } I$

система выдает сообщение:

X*19.56.08 OK — команда принята системой.

Команда XP прекращает работу инициатора, т.е. делает логический раздел неактивным. Формат команды XP :

Команда	Параметры
XP	$I[n]$

Если указывается только параметр I , то останавливается работа инициаторов во всех логических разделах. Если добавляется номер раздела n , то переводится в неактивное состояние только один раздел (прекращается работа только одного инициатора). При успешном выполнении команды, например:

$\text{XP } I$

система выдает сообщение:

X*19.55.48 OK — команда принята системой.

Команда XT позволяет изменить классы, обрабатываемые инициаторами. Первоначально классы приписываются инициаторам при генерации КРОС. Формат команды XT :

Команда	Параметры
XT	$I_n, \text{ классы}$

Буква n обозначает номер логического раздела (инициатора). Значение классов выбирается из букв от А до О. Какие классы приписаны инициаторам, можно узнать по команде $\text{XD } I$. Максимальное количество классов, которое может быть приписано, определяется при генерации КРОС, но не более восьми классов.

Пример 13.1. Инициатор INIT1 обрабатывает задания класса F. Информация об этом получена по команде XD I:

XD*16.13.47 INIT1 (INACTIVE)=F

Изменить класс F на классы A и B. Для изменения следует выдать команду

XT II,AB

В случае успешного выполнения команды система выдает сообщение:

XD*16.15.55 OK — команда принята системой.

Информацию об изменении можно получить, повторно, выдав команду

XD I

XD*16.16.06 INIT1 (INACTIVE)=AB

13.8. Управление порядком прохождения заданий

Все задания по способу выполнения можно разделить на два вида: задания, вводимые во входном потоке, и нестандартные каталогизированные процедуры, которые находятся в библиотеке процедур SYS1.PROCLIB и вызываются с консоли. Для управления выполнением заданий, которые введены во входном потоке и помещены в очереди JOB или HOLD, используются команды XH, XA, XC, XP, XT.

Управление заданиями, введенными во входном потоке. Управляющее воздействие команд на процессы выполнения заданий в КРОС будем рассматривать, только когда задания находятся в очереди JOB (или HOLD), так как в дальнейшем они выполняются автоматически с помощью инициатора INIT и программы WTR. Считаем, что имеются свободные логические разделы и устройства вывода подготовлены к работе.

Ввод заданий во входном потоке с перфокарт. Прежде чем начать ввод, надо стартовать устройство ввода по команде XS, которая была рассмотрена в п. 13.6. Устройство ввода по этой команде переводится в состояние INACTIVE. После этого следует поместить колоду с перфокартами в приемный карман считывающего устройства и нажать кнопки «Пуск» и «Конец картотеки». После нажатия кнопки «Пуск» управление получает программа системного ввода RDR, автоматически запускаемая системой. Одновременно можно вводить задания с нескольких считывающих устройств и магнитных лент. Введенные задания помещаются в очередь JOB (или HOLD), которая находится в наборе данных SYS1.KPOCACE. Каждому заданию присваивается номер от 1 до 999. Этот номер указывается в сообщениях, выдаваемых системой.

Пусть задания вводятся с устройства 00A. Тогда, если на устройстве нет заданий для ввода, то система выдает сообщение, например:

*IEA000A 00A,INT REQ,02,0200,4000,,,KPOC550

Интерпретация этого сообщения такая же, как в примере 6.1 при запуске программы системного ввода. Если при вводе была обнаружена ошибка в операторах языка управления заданиями, то выдается сообщение, например:

IEF453I MTP0 JOB FAILED — JCL ERROR

Если RDR не прочитал перфокарту с оператором JOB, то выдается сообщение, например:

⌘*20.02.16 READER1 SKIPPING FOR JOB CARD — в устройстве считывания READER1 обнаружена ошибка в операторе JOB.

Команда ⌘C используется для прекращения ввода текущего задания во входном потоке. Формат команды ⌘C приведен в табл. 13.3. После выдачи команды ⌘C прекращается ввод задания, которое вводится в этот момент времени, и система переходит к вводу следующего задания.

С помощью команды ⌘T можно задание, вводимое с устройства, поместить в очередь задержанных заданий HOLD. Формат команды ⌘T:

Команда	Параметры
⌘T	<i>устройство, N</i>

В ответ на ввод команды ⌘T выдается сообщение, например:

⌘T RDR2,H

⌘*19.13.56 ОК — команда принята.

Управление очередями заданий. Задания, которые находятся в очереди JOB автоматически, в порядке их номеров выбираются инициаторами.

Команда ⌘H позволяет задержать выборку заданий и перевести их в очередь HOLD. В очередь HOLD можно перевести задания и из выходной очереди SOUT. Формат команды ⌘H:

Команда	Параметры
⌘H	{ A J номер задания 1,... номер задания 5}

Параметр A задерживает выборку всей очереди JOB. Параметр J позволяет задержать выборку определенных заданий. При вводе команды ⌘H, например:

⌘H A

система выдает сообщение:

⌘*19.33.09 ОК — команда принята.

При вводе команды

⌘H J125

выдается аналогичное сообщение, например:

⌘*19.32.05 JOB 125 ASSEMB AWAITING EXEC F PRIO 3 HOLD

Команда QA снова возвращает задержанные задания в свои очереди.
Формат команды QA :

Команда	Параметры
QA	$\{J\}$ (номер задания 1, ..., номер задания 5)

Параметр J переводит задания из очереди HOLD в очередь JOB или SOUT. Если указываются номера заданий, то в очереди JOB или SOUT переводятся именно эти задания.

При вводе команды, например:

QA J

система выдает сообщение:

QA*19.33.43 J

При вводе команды, например:

QA J125

выдается сообщение:

$\text{QA*19.31.32 JOB 125 RELEASE}$

Команда QT позволяет изменить класс или приоритет задания или определить начало порядка нумерации заданий. Формат команды QT :

Команда	Параметры
QT	$Jn \left[\begin{matrix} P=\text{приоритет} \\ C=\text{класс} \end{matrix} \right]$

Первый параметр Jn является обязательным. Если указывается только один этот параметр, то система всем вводимым заданиям присваивает номера, начиная с n . В параметре P указывается новое значение приоритета для задания с номером n . Это новое значение устанавливается следующим образом: если указывается число без знака (от 0 до 15), то оно означает новый приоритет; если со знаком «+», то к старому приоритету прибавляется это значение, образуя новое значение приоритета; если со знаком «-», то от старого значения приоритета вычитается это значение, также образуя новое значение приоритета. В параметре C указывается новое значение класса от A до Z или от 0 до 9.

При вводе команды

QT J=125, C=F

выдается сообщение:

$\text{QA*19.25.37 JOB 125 ASSEMB AWAITING EXEC F PRIO 1 HOLD}$

Если задание уже обрабатывается системой, то его приоритет и класс не изменяются.

Команда αP удаляет задания из очереди. Если задание обрабатывается системной программой, то оно удаляется после того, как система закончит его обработку. Если задание не выполнялось, то перед удалением из очереди распечатываются его управляющие операторы языка управления заданиями. Формат команды αP :

Команда	Параметры
αP	J (номер задания 1,..., номер задания 5)

При вводе команды, например:

αP J124

система выдает сообщение:

α *19.06.25 JOB 124 ASSEMB AWAITING PRINTO PRIO7

и далее сообщение:

α *19.06.25 JOB 124 PRINTING ON PRINTER1 — задание 124 выводит данные на АЦПУ PRINTER1.

Команда αC удаляет задание из очереди. Если задание начало обрабатываться инициатором, то оно удаляется после того, как закончится работа инициатора. Формат команды αC :

Команда	Параметры
αC	J (номер задания 1,..., номер задания 5)

При вводе команды, например:

αC J125

система выдает сообщение, то же что при вводе команды αP . Следует отметить, что если удаляются задания из очереди или вводимые системной программой RDR, то на печать выдаются операторы языка управления заданиями.

Задание можно уничтожить с помощью команды CANCEL (см. гл. 8). Если выполнение команды снимается по команде CANCEL, например:

C PRINTZK

то система выдает сообщения такие же, как были описаны в гл. 8, например:

```
IEE301I PRINTZK CANCEL COMMAND ACCEPTED
IEC202E K,282,ZAPROS,NL,PRINTZK,IEFPROC
IEF450I PRINTZK.IEFPROC.ABEND S222 TIME=19.43.58
```

МИНИМАЛЬНЫЙ КОМПЛЕКТ ЭВМ

Наименование блоков	Шифры				
	1030	1033	1035	1045	1060
Процессор	2030	2433	2635	2645	2060
Центральный процессор					
МК	4430	в процес-	2435	2445	4012.01
СК		соре			4035.03
ОП	3203	3203	3235	3206	3206 (2 шт.)
НМД	5056 (2 шт.)	5056М	5061 (3 шт.)	5061 (4 шт.)	5056М (10 шт.) 5061 (6 шт.)
УУ НМД	5551	5551М	5561	5561	5551М (2 шт.)
НМЛ	5010 (4 шт.)	5017 (4 шт.)	5017.03 (6 шт.)	5017 (6 шт.)	5017 (8 шт.)
УУ НМЛ	5511	5517	5517	5517	5517
Ввод с ПК	6012	6012	6012	6012 (2 шт.)	6019 (2 шт.)
Вывод на ПК	7012	7012	7010	7010	7010 (2 шт.)
Ввод с ПЛ	6022	6022	6022	6022	6022 (2 шт.)
Вывод на ПЛ	7022	7022	7022	7022	7022 (2 шт.)
АЦПУ	7032	7032	7032	7032	7032 (2 шт.)
Пишущая машинка	7077	7077	7077	7077	7077 (2 шт.)
Подготовка данных на ПК	9011	9011	9011	9011	9011
Подготовка данных на МЛ	9020	9024	9024	9024	9024
Дисплей					7066 (4 шт.)
Пульт оператора			1535	1535	1501.01
Пультый НМЛ			5009	5009	5009

ЧИСЛОВАЯ ТАБЛИЦА ДЛЯ ПЕРЕВОДА ЧИСЕЛ ИЗ ДЕСЯТИЧНОЙ СИСТЕМЫ СЧИСЛЕНИЯ В ДВОИЧНУЮ И ОБРАТНО

2^n	n	2^n	n	2^n	n
1	2	1	2	1	2
1	0	2 048	11	4 194 304	22
2	1	4 096	12	8 388 608	23
4	2	8 192	13	16 777 216	24
8	3	16 384	14	33 554 432	25
16	4	32 768	15	67 108 864	26
32	5	65 536	16	134 217 728	27
64	6	131 072	17	268 435 456	28
128	7	262 144	18	536 870 912	29
256	8	524 288	19	1 073 741 824	30
512	9	1 048 576	20	2 147 483 648	31
1 024	10	2 097 152	21	4 294 967 296	32

В колонке (1) находится набор десятичных чисел, которые получены при вычислении 2^n , для $n \leq 32$. Степень n показывает место цифры, если оно будет выражено двоичными разрядами.

ТАБЛИЦА ДВОИЧНОГО КОДА ДЛЯ ОБМЕНА ИНФОРМАЦИИ (ДКОИ)

Первая, зонная шестнадцатеричная цифра кода

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0				Пробел	&	-		ц	я	я	ь	{	}	\		0
1					/			а	і	-	ы	A	J			1
2								в	к	з	а	B	K	S		2
3								с	l	t	ш	C	L	T		3
4								д	m	и	э	D	M	D		4
5								е	n	v	ц	E	N	V		5
6							ю	f	с	w	Ч	F	O	W		6
7							а*	g	p	x	Ъ	G	P	X		7
8							б	h	о	у	Ю	H	O	У		8
9								i	т	z	A*	Y	K	Z		9
A				[	]		:	д	к	p*	Б	X*	H*	T*		з
B				.	⊗	,	#	е*	л	с*	Ц	И	O*	У		ш
C				<	*	%	@	ф	м	т	Д	И	І	Ж		э
D				(	)	-	'	г	н	у*	E*	K*	Я	B*		щ
E				+	;	>	=	х*	о*	ж	Ф	Л	P*	Ь		ч
F				!	Λ	?	"	и	п	в	Г	M*	C*	Ы		

Примечания. 1. В кодовой позиции 5F допускается «—» (логическое ИИ);

2. В кодовой позиции B7 допускается прописная буква «Ъ»;

3. Знаком * отмечены русские прописные и строчные буквы АВЕКМНОРСТХаеорсух, вместо которых можно использовать латинские прописные и строчные буквы АВЕКМНОРСТХаеорсух, если шрифты их визуально неотличимы.

Таблица 8-битового кода для обмена информацией (КОИ-8)

Первая, зонная шестнадцатеричная цифра кода

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Вторая шестнадцатеричная цифра кода	0		Пробел	0	@	P		p					ю	п	Ю	П
	1		!	1	A	Q	a	q					а*	я	А*	Я
	2		"	2	B	R	b	r					б	р*	Б	Р*
	3		#	3	C	S	c	s					ц	с*	Ц	С*
	4		☒	4	D	T	d	t					д	т	Д	Т*
	5		%	5	E	U	e	u					е*	у*	Е*	У
	6		&	6	F	V	f	v					ф	ж	Ф	Ж
	7		'	7	G	W	g	w					г	в	Г	В*
	8		(	8	H	X	h	x					х*	ь	Х*	Ь
	9		)	9	I	V	i	y					и	ы	И	Ы
	A		*	:	J	Z	j	z					й	э	Й	З
	B		+	:	K	l	k	{					«	ш	К*	Ш
	C		,	<	L	\	l						л	э	Л	Э
	D		-	=	M	l	m	}					м	щ	М*	Ш
	E		.	>	N	^	n	-					н	ч	Н*	Ч
	F		/	?	O	-	o						о*	ь	О*	

- Примечания: 1. В кодовой позиции 5Е допускается «—» (логическое НЕ).
 2. В кодовой позиции ЕF допускается прописная буква «Ъ».
 3. Знаком * отмечены русские прописные и строчные буквы АВЕКМНОРСТХасорсух, вместо которых можно использовать латинские прописные и строчные буквы АВЕКМНОРСТХасорсух, если шрифты их визуально неотличимы.

ОБОЗНАЧЕНИЕ И НАИМЕНОВАНИЕ СПЕЦИАЛЬНЫХ ЗНАКОВ

Обозначение	Наименование	Позиция в таблицах (16 с. с.)	
		ДКОИ	КОИ-8
	Пробел	40	20
[	Квадратная скобка левая	4A	5B
.	Точка	4B	2E
<	Меньше	4C	3C
(	Круглая скобка левая	4D	28
+	Плюс	4E	2B
!	Восклицательный знак	4F	2I
&	Знак амперсанда	50	26
]	Квадратная скобка правая	5A	50
¤	Знак денежной единицы	5B	24
*	Звездочка	5C	2A
)	Круглая скобка правая	5D	29
;	Точка с запятой	5E	3B
^	Логическое НЕ	5F	5E
-	Минус	60	2D
/	Дробная черта	6I	2F
	Вертикальная черта	6A	7C
,	Запятая	6B	2C
%	Проценты	6C	25
_	Подчеркивание	6D	6F
>	Больше	6E	3E
?	Вопросительный знак	6F	3F
:	Двоеточие	7A	3A
#	Номер	7B	23
@	Коммерческое ЭТ	7C	40
'	Апостроф	7D	27
=	Равно	7E	3D
"	Кавычки	7F	22
—	Черта сверху	A1	7E
{	Фигурная скобка левая	C0	7B
}	Фигурная скобка правая	D0	7D
\	Обратная дробная черта	E0	5C

ЮБЦФТАДСЕННДННВРПТХУТЪМЗШЩЧ									
1 2	4	6	8	10	12	14	16	18	20
00000	00000	00000	00000	00000	00000	00000	00000	00000	00000
11111	11111	11111	11111	11111	11111	11111	11111	11111	11111
22222	22222	22222	22222	22222	22222	22222	22222	22222	22222
33333	33333	33333	33333	33333	33333	33333	33333	33333	33333
44444	44444	44444	44444	44444	44444	44444	44444	44444	44444
55555	55555	55555	55555	55555	55555	55555	55555	55555	55555
66666	66666	66666	66666	66666	66666	66666	66666	66666	66666
77777	77777	77777	77777	77777	77777	77777	77777	77777	77777
88888	88888	88888	88888	88888	88888	88888	88888	88888	88888
1 2	4	6	8	10	12	14	16	18	20
00000	00000	00000	00000	00000	00000	00000	00000	00000	00000
11111	11111	11111	11111	11111	11111	11111	11111	11111	11111
22222	22222	22222	22222	22222	22222	22222	22222	22222	22222
33333	33333	33333	33333	33333	33333	33333	33333	33333	33333
44444	44444	44444	44444	44444	44444	44444	44444	44444	44444
55555	55555	55555	55555	55555	55555	55555	55555	55555	55555
66666	66666	66666	66666	66666	66666	66666	66666	66666	66666
77777	77777	77777	77777	77777	77777	77777	77777	77777	77777
88888	88888	88888	88888	88888	88888	88888	88888	88888	88888
1 2	4	6	8	10	12	14	16	18	20
00000	00000	00000	00000	00000	00000	00000	00000	00000	00000
11111	11111	11111	11111	11111	11111	11111	11111	11111	11111
22222	22222	22222	22222	22222	22222	22222	22222	22222	22222
33333	33333	33333	33333	33333	33333	33333	33333	33333	33333
44444	44444	44444	44444	44444	44444	44444	44444	44444	44444
55555	55555	55555	55555	55555	55555	55555	55555	55555	55555
66666	66666	66666	66666	66666	66666	66666	66666	66666	66666
77777	77777	77777	77777	77777	77777	77777	77777	77777	77777
88888	88888	88888	88888	88888	88888	88888	88888	88888	88888
1 2	4	6	8	10	12	14	16	18	20
00000	00000	00000	00000	00000	00000	00000	00000	00000	00000
11111	11111	11111	11111	11111	11111	11111	11111	11111	11111
22222	22222	22222	22222	22222	22222	22222	22222	22222	22222
33333	33333	33333	33333	33333	33333	33333	33333	33333	33333
44444	44444	44444	44444	44444	44444	44444	44444	44444	44444
55555	55555	55555	55555	55555	55555	55555	55555	55555	55555
66666	66666	66666	66666	66666	66666	66666	66666	66666	66666
77777	77777	77777	77777	77777	77777	77777	77777	77777	77777
88888	88888	88888	88888	88888	88888	88888	88888	88888	88888

Код для перфорации перфокарт КПК-12. Русский алфавит и специальные знаки.

ШИФРЫ (ТИПОВЫЕ ИМЕНА), ФИЗИЧЕСКИЕ АДРЕСА И ГРУППОВЫЕ ИМЕНА УСТРОЙСТВ ВВОДА-ВЫВОДА

Названия устройств, их шифры	Возможные физические адреса	Групповые имена
Накопители на магнитной ленте (НМЛ)—5010, 5012, 5014, 5015, 5016, 5017, 5019, 5021, 5022	180—187 280—287	SYSSQ
Накопители на сменных магнитных дисках (НСМД)—5050, 5052, 5055, 5056, 5058, 5060, 5061	130—137 190—197 230—237 290—297 330—337 390—397	SYSSQ SYSDA
Устройство ввода с перфокарт (считывающее устройство)—6010, 6012, 6013, 6016	00A, 00C	SYSIN
Устройство вывода на перфокарты—7010, 7012	00B, 00D	SYSCP
Устройство ввода с перфолент—6020, 6022, 6021, 6122	008	
Устройство вывода на перфоленту—7020, 7022, 7122	009	
Алфавитно-цифровые печатающие устройства (АЦПУ)—7030, 7031, 7033, 7035, 7031 с двумя трактами бумаги, 7038	00F, 00E	SYSPRINT
Пишущая машинка—7070, 7073, 7074, 7172	01F	
Алфавитно-цифровой дисплей комплекса ЕС-7906	060—063, 050—053, 0D0—0D3	

ПРЕФИКСЫ СИСТЕМНЫХ СООБЩЕНИЙ *)

Префикс	Программа, которая выдает сообщение
IBC	Независимые программы обслуживания
IEA	Супервизор
IEB	Утилиты для обработки наборов данных
IEC	Супервизор ввода-вывода
IEE	Главный планировщик
IEF	Планировщик заданий
IEH	Системные программы обслуживания
IEI	Программы библиотеки генерации системы
IEK	Транслятор фортран оптимизирующий
IEM	Транслятор ПЛ/1
IER	Программа сортировка-объединение
IES	Транслятор РПГ
IEU	Ассемблер
IEW	Редактор, оверлейный супервизор
IEX	Транслятор алгол-60
IEY	Стандартный транслятор фортран
IFB	Программа регистрации состояния
IFC	Программа регистрации состояния и печати
IFD	Неавтономная тестовая программа
IFF	Программа обеспечения графических устройств
IGC	Транзитные SVC-программы
IGE	Программы обработки ошибок ввода-вывода
IGF	Программа обработки машинных сбоев
IGG	Программы обеспечения ввода-вывода
INA	Системные управляющие блоки
INB	Программы библиотеки макроопределений
INC	Программа библиотеки фортрана
IND	Программа библиотеки ПЛ/1
INE	Сообщения об ошибках во время выполнения программы
INJ	Программа контрольная точка-рестарт
ING	Программа анализа и обновления
INI	Программа библиотеки алгола
IKF	Транслятор кобод

*) Перечень префиксов сокращенный.

СТАНДАРТНАЯ ПРОЦЕДУРА СИСТЕМНОГО ВВОДА RDR

```
//IEFPROC EXEC PGM=IEFIRC,REGION=nnnnK,
//      PARM='bpptitooommmilicccrlssssssssaaaaefh'
//IEFRDER DD UNIT=устройство,LABEL=(run),
//      VOLUME=SER=SYSIN,
//      DCB=(список параметров) [,DSNAME=имя,DISP=OLD]
//IEFPDSI DD DSNAME=SYSI.PROCLIB,DISP=SHR
//IEFDATA DD UNIT=устройство,
//      SPACE=(единицы,(величины) [,RLSE,CONTIG]),
//      VOLUME=SER=серийный номер тома,
//      DISP=(состояние, диспозиция),
//      DCB=(список параметров),DSORG=PS
```

Оператор EXEC с именем IEFPROC обращается к программе системного ввода IEFIRC. В параметре REGION указывается объем памяти для RDR. В параметре PARM: *b*—значения параметра ROLL, необходимость указания учетной информации и фамилии программиста в операторе JOB; *pp*—значения параметра PRTY (00-14; *ttt*—значение параметра TIME; *ooo*—максимальное число дорожек, отводимых для набора данных SYSOUT; *mmm*—дополнительное число дорожек, отводимых для набора данных SYSOUT; *iii*—значение параметра DRPTY (0-255); *sss*—размер памяти для пункта задания; *r*—указывает диспозицию команд, считанных из входного потока; *l*—метод обработки меток; *ssssssss*—стандартное устройство для SYSOUT; *aaaa*—группы команд оператора, которые выполняются, если они считаны во входном потоке; *ef*—значения параметра MSGLEVEL; *h*—значение параметра MSGCLASS.

Оператор DD с именем IEFRDER описывает входной поток, IEFPDSI—библиотеку процедур SYSI. PROCLIB; IEFDATA—набор данных, который используется для промежуточного запоминания данных из входного потока (размер блока 80 байт).

Пример стандартной процедуры системного ввода RDR:

```
//IEFPROG EXEC PGM=IEFIRC,REGION=48K,
//      PARM='80103005001024905010 SYSDA bbbE00001A'
//IEFRDER DD UNIT=5010,LABEL=(,NL),
//      VOLUME=SER=SYSIN,DISP=OLD
//      DD DCB=(BLKSIZE=80,LRECL=80,
//      BUFL=80,BUFNO=1,RECFM=F)
//IEFPDSI DD DSNAME=SYSI.PROCLIB,DISP=SHR
//IEFDATA DD UNIT=SYSDA,
//      SPACE=(80,(500,500),RLZE,CONTIG,
//      DCB=(BLKSIZE=80,LRECL=80,
//      BUFL=80,BUFNO=2,RECFM=F,
//      DSORG=RD)
```

Значения PARM: 8—для параметра ROLL задается (YES,NO), учетная информация и имя программиста указывать не обязательно; 01—приоритет PRTY (по умолчанию); 030—процессорное время (в минутах) выполнения пункта задания; 050—число дорожек, отводимое наборам данным SYSOUT; 010—дополнительное число дорожек, отводимое наборам данных SYSOUT; SYSOUT; 249—значение диспетчерского приоритета DRPTY; 050—размер раздела в К байт для пунктов заданий; 1—выдача команды в программу планирования команд; 0—обход обработки меток лент; SYSDAbbb—устройство для SYSOUT; 'E000—режим MVT без MCS, в котором разрешается вводить все команды, кроме HALT,MODE и SWAP; 01—значения параметра MSGLEVEL (по умолчанию); A—значение параметра MSGCLASS (по умолчанию).

СТАНДАРТНАЯ ПРОЦЕДУРА ИНИЦИАТОРА INIT

```
//      IEFPROC EXEC PGM=IEFIIC,
//      PARM='x [(n)] [x1 (n1)...] [LIMIT=k]'
```

Оператор EXEC с именем IEFPROC обращается к программе инициатора с именем IEFIIC; *x*—класс задания (A—O), может быть указано до восьми классов; *n*—принудительный приоритет (0—15), который будет приписан всем заданиям указанного класса; *k*—предельный приоритет (0—15), т. е. наивысший приоритет, с которым инициатор будет обрабатывать задания. Значение *n* используется в том случае, если оно не более *k*. Если *n* опущено, то система определяет приоритет из операторов EXEC, JOB, каталогизированной процедуры RDR, пока оно не превысит значения *k*. Если в каталогизированной процедуре указан принудительный приоритет *n*, то он перекрывает приоритеты, указанные в EXEC, JOB, каталогизированной процедуре RDR.

Пример стандартной процедуры INIT:

```
//IEFPROC EXEC PGM=IEFIIC,
//      PARM='A,LIMIT=13'
```

В данной каталогизированной процедуре указано, что она будет обрабатывать задания класса A с предельным приоритетом, равным 13.

СТАНДАРТНАЯ ПРОЦЕДУРА СИСТЕМНОГО ВЫВОДА WTR

```
//IEFPROC EXEC PGM=IEFSD080,
//      REGION=nnnnK,PARM='сххххххх,имя'
//IEFRDED DD      UNIT=устройство; LABEL=(,тип),
//      VOLUME=(,счётчик томов),
//      DSNNAME=,DISP=(NEW,KEEP),
//      DCB=(список параметров)
```

Оператор EXEC с именем IEFPROC обращается к программе системного вывода IEFSD080. В параметре PARM буква *c* заменяется: P—(АЦПУ), C—устройство на перфокартах; буквы *хххххххх* заменяются выходными классами от 1 до 8 (A—Z, 0—9); *имя*—имя программы, которая формирует разделитель заданий в выходном потоке (если *имя* опущено, то эта программа не используется); в параметре REGION указывается объем памяти, занимаемой процедурой WTR.

Пример стандартной процедуры системного вывода:

```
//IEFPROC EXEC PGM=IEFSD080,
//      REGION=20K,PARM='PA'
//IEFRDER DD      UNIT=7030,VOL=(,35),
//      DSNNAME=SYSOUT,DISP=(NEW,KEEP),
//      DCB=(BLKSIZE=133,LRECL=133,
//      BUFL=133,BUFNO=2,RECFM=FM)
```

СТАНДАРТНАЯ ПРОЦЕДУРА ПРЯМОГО СИСТЕМНОГО ВЫВОДА DSO

```
//IEFPROC EXEC PGM=IEFDSO,REGION=8K,
//          PARM=(сх,имя,////////)
//IEFRDED DD UNIT=имя,DSN=имя набора,
//          DISP=(NEW,KEEP),LABEL=(,SL),
//          VOL=(...число томов),
//          DCB=(список параметров)
```

Оператор EXEC с именем IEFPROC обращается к программе прямого системного вывода IEFDSO. В параметре с заменяется: P—АЦПУ, C—перфоратор; x—класс SYSOUT, который должен быть обработан. Если класс указан и в процедуре и в команде START, то берется класс, указанный в команде START. Имя заменяется на имя программы, которая формирует разделитель выходных наборов данных (выходных потоков). Буквы *////////* заменяются на значения входных классов от А до О.

Пример процедуры DSO:

```
//IEFPROC EXEC PGM=IEFDSO,REGION=8K,
//          PARM=(PA,,A)
//IEFRDER DD UNIT=5010,DSN=SYSOUT,
//          DISP=(NEW,KEEP),LABEL=(,SL),
//          VOL=(...05),DCB=(BUFNO=3)
```

Приложение 11

ПЕРЕЧЕНЬ ПРОГРАММ ОБСЛУЖИВАНИЯ

Группа	Название	Назначение
Автономные программы	IBCDASDI IBCDMPRS	Инициализирует том прямого доступа Снимает копию с тома прямого доступа на МЛ и том прямого доступа
Системные программы	IEHDASDR IEHINITT IEHIOSUP	Инициализирует том прямого доступа Подготавливает МЛ к работе Модифицирует адреса в библиотеке супервизора
	IEHLIST	Распечатывает системные управляющие данные
	IEHMOVE	Копирует наборы данных с последовательной организацией, библиотечной и прямой
	IEHPROGM	Создает и обслуживает системные управляющие данные
Программы наборов данных	IEBCOMPR IEBCOPY	Сравнивает наборы данных Копирует наборы данных с тома прямого доступа
	IEBDG	Создает набор текстовых данных, который используется как вспомогательное средство отладки
	IEBEDIT	Создает выходной набор данных, содержащий набор заданий и шагов заданий
	IEBGGENER	Преобразует набор данных в библиотечный и копирует записи из последовательного набора данных
	IEBISAM	Снимает копию с индексно-последовательного набора данных
	IEBPTCH	Печатает и перфорирует записи из библиотечного и последовательного наборов данных
	IEBUPDTE	Вносит изменения в символьную библиотеку, т. е. библиотеку, содержащую программы на одном из символьных языков

УПРАВЛЯЮЩИЕ ОПЕРАТОРЫ АВТОНОМНОЙ ПРОГРАММЫ ОБСЛУЖИВАНИЯ IBCDASDI

Оператор	Параметры	Назначение
DADEF—определяет том прямого доступа, подлежащий инициализации	TODEV=xxxx TOADDR=ddd IPL=YES { VOLID=номер } { VOLID=SCRATCH } { FLAGTEST=NO } { PASSES=n } [BYPASS=YES]	Задается типовое имя устройства прямого доступа Задается физический адрес устройства На том должна быть помещена программа IPL Указывается серийный регистрационный номер тома (от 1 до 6 символов) при повторной инициализации Указывается, если серийный номер тома не проверяется Проверяются все дорожки на дефектность Проверяется каждая дорожка на дефектность n раз (по умолчанию 1 раз) Не проверяются дорожки на дефектность (по умолчанию проверяются и дефектным дорожкам присваиваются альтернативные)
VLD—задает информацию для начальной метки и для резервирования дополнительных меток тома	NEWVOLID=номер { VOLPASS=1 } { VOLPASS=0 } [OWNERID=имя] [ADDLABEL=n]	Задается серийный регистрационный номер тома (от 1 до 6 символов) Устанавливается бит разграниченного доступа к тому в 1 Устанавливается бит разграниченного доступа к тому в 0 (по умолчанию —0) Указывается имя владельца тома (от 1 до 10 символов) Резервируется место для n дополнительных меток, где n=1—7 (по умолчанию —0)
VT OCD—задает информацию для расположения оглавления тома	STRTADR=nnnnn EXTENT=nnnnn	Задается адрес дорожки, начиная с которой должно быть расположено оглавление тома (10 с.с.) Задается длина оглавления тома в дорожках (10 с.с.)
[IPLTXT]		Отделяются управляющие операторы автономной программы от текстовых операторов программы IPL
[LASTCARD]		Отделяется задание IBCDASDI или пакет заданий IBCDASDI на устройстве ввода от других управляющих операторов
GETALT—присваивает альтернативную дорожку	TODEV=xxxx TOADDR=ddd TRACK=cccchhh VOLID=номер	Задается типовое имя устройства прямого доступа Задается физический адрес устройства прямого доступа Задается адрес дорожки, для которой запрашивается альтернативная дорожка (16 с.с.) Указывается серийный регистрационный номер тома, чтобы избежать ошибки при установке тома

Оператор	Параметры	Назначение
	[FLAGTEST=NO] [PASSES= <i>n</i>] [BYPASS=YES]	Указывается, что дорожка должна проверяться на дефектность Проверяется каждая дорожка на дефектность <i>n</i> раз (по умолчанию — 1 раз) Не проверяются дорожки на дефектность (по умолчанию дефектным дорожкам присваиваются альтернативные)
Примечание. Операторы DADEF и GETALT взаимно исключающие.		

УПРАВЛЯЮЩИЕ ОПЕРАТОРЫ ПРОГРАММЫ IBSDMPRS

Оператор	Параметры	Назначение
DUMP — идентифицирует исходный и принимающие тома	FROMDEV= <i>xxxx</i> FROMADDR= <i>ddd</i> TODEV= <i>xxxx</i> TOADDR= <i>ddd</i> [VOLID= <i>список</i>]	Задается типовое имя устройства, на котором установлен том, с которого снимается копия Задается физический адрес устройства, на котором установлен том, с которого снимается копия Задается типовое имя устройства, на котором смонтирован том, принимающий информацию Задается физический адрес устройства, на котором смонтирован том, принимающий информацию Указываются серийные регистрационные номера томов, на которые передаются данные
[VDRL] — указывает пределы частичного дампа	BEGIN= <i>nnnnn</i> [END= <i>nnnnn</i>]	Указывается адрес дорожки относительно начала тома, начиная с которой будет выполняться копирование Указывается адрес последней копируемой дорожки относительно начала тома
RESTORE — идентифицирует исходный и принимающий том	FROMDEV= <i>xxxx</i> FROMADDR= <i>ddd</i> TODEV= <i>xxxx</i> TOADDR= <i>ddd</i> VOLID= <i>номер</i>	Задается типовое имя устройства, на котором установлен том, содержащий восстанавливаемые данные Задается физический адрес устройства, на котором установлен том, содержащий восстанавливаемые данные Задается типовое имя устройства, на котором смонтирован том, на который будут передаваться восстанавливаемые данные Задается физический адрес устройства, на котором смонтирован том, на котором будут передаваться восстанавливаемые данные Указывается серийный регистрационный номер тома прямого доступа, на котором восстанавливаются данные

**ПАРАМЕТРЫ, КОТОРЫЕ УКАЗЫВАЮТСЯ В УПРАВЛЯЮЩИХ
ОПЕРАТОРАХ DD, ПРИ РАБОТЕ С СИСТЕМНЫМИ ПРОГРАММАМИ
ОБСЛУЖИВАНИЯ IENDASDR И IENINITT**

№ п/п	Описание тома	Набор параметров
1	Один том на устройстве	UNIT= <i>адрес</i> , VOL= <i>SER=номер</i> , DISP= <i>OLD</i>
2	Несколько сменных томов с отложенным монтированием	UNIT=(<i>адрес</i> ,,DEFER), VOL=(PRIVATE,...), DISP=(NEW,KEEP)
3	Неизвестный серийный номер с отложенным монтированием тома	INIT=(<i>адрес</i> ,,DEFER), VOL=PRIVATE, DISP=(NEW,KEEP)
4	Один том на МЛ (при работе с DUMP и RESTORE)	UNIT= <i>адрес</i> , VOL= <i>SER=номер</i> , LABEL=(...), DISP=(...,KEEP) DCB=(DEN= <i>x</i>), DSN= <i>имя</i>
5	Один том на МЛ с отложенным монтированием тома	UNIT=(<i>адрес</i> , <i>n</i> ,DEFER), VOL=(PRIVATE,...)
6	Несколько томов при работе с INITT	UNIT=(<i>адрес</i> , <i>n</i> ,DEFER), DCB=(DEN= <i>x</i>)

УПРАВЛЯЮЩИЕ ОПЕРАТОРЫ ПРОГРАММЫ IENDASDR

Оператор	Параметры	Назначение
ANALYZE—размечает тома прямого доступа	$\left\{ \begin{array}{l} TODD=(xxx,...) \\ TODD=(имя dd,.) \end{array} \right\}$ VTOC= <i>nnnn</i> EXTENT= <i>nnnn</i> [NEWVOLID= <i>номер</i>] [IPLDD= <i>имя dd</i>] $\left[\begin{array}{l} FLAGTEST= \left\{ \begin{array}{l} YES \\ NO \end{array} \right\} \end{array} \right]$ [PASSES= <i>n</i>]	Указывается один или несколько физических устройств, на которых находятся инициализируемые тома Указывается одно или несколько имен операторов DD, в которых дается описание томов прямого доступа, которые размечаются Задается начальный адрес оглавления тома VTOC (10 с.с.) Задается длина оглавления тома в дорожках (10 с.с.) Присваивается новый серийный регистрационный номер тому от 1 до 6 символов (по умолчанию сохраняется старый номер) Указывается имя оператора DD, определяющего набор данных, содержащий программу IPL Проверяется, была ли ранее дорожка отмечена как дефектная Не проверяется, была ли ранее дорожка отмечена как дефектная (по умолчанию принимается YES) Задается проверка каждой дорожки на дефектность <i>n</i> раз от 1 до 255 (по умолчанию 1 раз)

Оператор	Параметры	Назначение
ANALYZE — размечает тома прямого доступа	[OWNERID= <i>имя</i>] [PURGE={ YES NO}]	Определяется имя владельца тома (от 1 до 10 символов) Стираются все наборы данных, для которых не прошел еще срок хранения Не стираются наборы данных, срок хранения которых еще не прошел (по умолчанию принимается <u>NO</u>)
LABEL — изменяет серийный номер тома с прямым доступом	{ TODD= <i>xxxx</i> TODD= <i>имя dd</i> NEWVOLID= <i>номер</i> [OWNERID= <i>имя</i>]	Указывает физический адрес устройства, на котором находится том прямого доступа, серийный номер которого изменяется Указывается имя оператора DD, в котором дано описание тома прямого доступа Присваивается новый серийный номер тому Определяется имя владельца тома (от 1 до 10 символов)
GETALT — назначает альтернативную дорожку	TODD= <i>имя dd</i> TRACK= <i>cccc hhhh</i>	Указывается имя оператора DD, в котором дается описание тома прямого доступа, в котором назначается альтернативная дорожка Указывается номер цилиндра (<i>cccc</i>) и номер головки (<i>hhhh</i>) для обрабатываемой дорожки в 16 с.с.
DUMP — снимает копию с тома прямого доступа	FROMDD= <i>имя dd</i> TODD=(<i>имя dd,...</i>) [CPYVOLID={ YES NO}] [BEGIN= <i>cccc hhhh</i>] [END= <i>cccc hhhh</i>] [PURGE={ YES NO}]	Указывается имя оператора DD, в котором дается описание тома, с которого делается копия Указываются имена операторов DD, в которые помещается копия Присваивается серийный номер тому — YES, сохраняется старый — NO Указывается номер цилиндра (<i>cccc</i>) и номер головки (<i>hhhh</i>) первой дампруемой дорожки (16 с.с.) Указывается номер цилиндра (<i>cccc</i>) и номер головки (<i>hhhh</i>) последней дампруемой дорожки Как в параметре ANALYZE
RESTORE — восстанавливает том прямого доступа	TODD=(<i>имя dd,...</i>) FROMDD= <i>имя dd</i>	Указываются имена операторов DD, в которых дается описание томов прямого доступа, которые должны быть восстановлены Указывается имя оператора DD, в котором дается описание тома, с которого идет восстановление информации

Оператор	Параметры	Назначение
RESTORE — восстанавливает том прямого доступа	$\left[\text{CPYVOLID} = \begin{Bmatrix} \text{YES} \\ \underline{\text{NO}} \end{Bmatrix} \right]$ $\left[\text{PURGE} = \begin{Bmatrix} \text{YES} \\ \underline{\text{NO}} \end{Bmatrix} \right]$	Восстановленным томам присваивается серийный номер, который имеет дампируемый том, если указан YES; сохраняется собственный номер тома, если указан NO Как в параметре ANALYZE
FORMAT — инициализирует том прямого доступа	TODD=(имя dd,...) VTOC=nnnnn EXTENT=nnnnn [NEWVOLID=номер] [IPLDD=имя dd] [OWNERID=имя] $\left[\text{PURGE} = \begin{Bmatrix} \text{YES} \\ \underline{\text{NO}} \end{Bmatrix} \right]$	Указываются имена операторов, в которых дается описание томов прямого доступа, которые размечаются Задается начальный адрес оглавления тома VTOC (10 с.с.) Задается длина оглавления тома в дорожках (10 с.с.) Присваивается новый серийный регистрационный номер тому от 1 до 6 символов (по умолчанию сохраняется старый номер) Указывается имя оператора DD, определяющего набор данных, содержащих программу IPL Имя владельца тома (от 1 до 10 символов) Стираются все наборы данных, для которых не прошел еще срок хранения Не стираются наборы данных, срок хранения которых еще не прошел (по умолчанию принимается NO)

СВОДНЫЙ ПЕРЕЧЕНЬ ФОРМАТОВ КОМАНД ДЛЯ ОПЕРАТОРА

1. Форматы команды оператора ОС

Операция	Операнды
$\left\{ \begin{matrix} \text{CANCEL} \\ \text{C} \end{matrix} \right\}$	$\left\{ \begin{matrix} \text{адрес} \\ \text{имя задания} \\ \text{идентификатор} \end{matrix} \left[\begin{matrix} \text{,ALL} \\ \text{,DUMP} \\ \text{,IN[=класс]} \\ \text{,OUT[=класс]} \end{matrix} \right] \right\}$
$\left\{ \begin{matrix} \text{DEFINE} \\ \text{N} \end{matrix} \right\}$	[LIST]
$\left\{ \begin{matrix} \text{DISPLAY} \\ \text{D} \end{matrix} \right\}$	$\left\{ \begin{matrix} \text{A} \\ \text{N [=список]} \\ \text{Q [список]} \\ \text{R} \end{matrix} \right\}$

Операция	Операнды
{DISPLAY} {D}	$\left\{ \begin{array}{l} \text{SQA} \\ \text{T} \\ \text{U} \left[\begin{array}{l} \text{,DASD} \\ \text{,GRAPHIC} \\ \text{,TAPE} \\ \text{,TP} \\ \text{,UR} \end{array} \right] \left[\begin{array}{l} \text{,OFFLINE} \\ \text{,ONLINE} \end{array} \right] [\text{,адрес}] [\text{,nnn}] \\ \text{имя задания} \end{array} \right\}$
DUMP	COMM=(заголовок)
{HALT} {Z}	EOD
{HOLD} {H}	{имя задания} {Q [=список]}
{LOG} {L}	'текст'
MODE	$\left\{ \begin{array}{l} \text{CONTROL} \left\{ \begin{array}{l} \text{,QUIET} \\ \text{,THRESHOLD} \end{array} \right\} \\ \text{MAIN} \left\{ \begin{array}{l} \text{,COUNT} \\ \text{,QUIET} \\ \text{,RECORD} \end{array} \right\} \\ \text{RETRY} \left\{ \begin{array}{l} \text{,QUIET} \\ \text{,RECORD} \end{array} \right\} \\ \text{STATUS} \end{array} \right\}$
{MODIFY} {F}	$\left\{ \begin{array}{l} [\text{INIT.}] \text{ идентификатор, CLASS=параметр} \\ [\text{WTR.}] \text{ идентификатор} \left\{ \begin{array}{l} \text{,CLASS=классы} \\ \text{,PAUSE}=\left\{ \begin{array}{l} \text{DATASET} \\ \text{FORM} \end{array} \right\} \end{array} \right\} \\ [\text{DSO.}] \text{ идентификатор} \left\{ \begin{array}{l} \text{,JOBCLASS=классы заданий} \\ \text{,OUTCLASS=выходной класс} \end{array} \right\} \\ \text{имя задания, ключевые параметры} \\ [\text{процедура.}] \text{ идентификатор} [\text{,ключевые параметры}] \end{array} \right\}$
{MONITOR} {MN}	$\left\{ \begin{array}{l} \text{DSNAME} \\ \text{JOBNAMES [T]} \\ \text{SPACE} \\ \text{STATUS} \end{array} \right\}$
{MOUNT} {M}	адрес {,VOL=(NL,номер)} [USE={PUBLIC PRIVATE STORAGE}]
{RELEASE} {A}	{Q [=список]} имя задания
{REPLY} {R}	номер, 'текст'

Операция	Операнды
{RESET} E	имя задания [,PRTY=приоритет] [,CLASS=класс] [,OUT=класс]
{SET} T	DATE=yy.ddd [,CLOCK=hh.mm.ss] [,Q=([адрес] [,F])] [,PROC=адрес] [,AUTO=aaa]
{START} S	процедура [.идентификатор] [,параметры] RDR [.идентификатор] ,адрес [,номер тома] [,имя задания] [,ключевые параметры] WTR [.идентификатор] ,адрес [,номер тома] [,выходные классы] [,ключевые параметры] DSO [.идентификатор] ,адрес [,номер тома] [,классы] [,ключевые параметры] INIT [.идентификатор] [,,,параметры]
{STOP} P	[процедура.] {идентификатор} адрес имя задания DSNAME SPACE STATUS JOBNAMES
{STOPMN} PM	{ DSNAME SPACE STATUS JOBNAMES }
{SWAP} G	{ OFF ON xxx,yyy }
{SWITCH} I	SMF
{UNLOAD} U	адрес
{VARY} V	(адрес 1, адрес 2,...) { ,ONLINE ,OFFLINE ,PATH,адрес,{ONLINE ,OFFLINE} }
{WRITELOG} W	[класс CLOSE]

2. Форматы команд для дисплей-консолей и управления консолями

Операция	Операнды
$\{\text{CONTROL}\}_K$	$\left[\begin{array}{l} D \left\{ \begin{array}{l} ,N [,HOLD] \\ ,F [,L=x] \\ ,H [,L=x] \\ ,U [,L=x] \\ ,PFK \end{array} \right\} \\ E \left\{ \begin{array}{l} ,SEG \\ ,nn [,mm] \\ ,F \\ ,D [,L=x] \\ ,N \\ ,PFK \end{array} \right\} \\ S \left\{ \begin{array}{l} ,REF \\ ,DEL = \left\{ \begin{array}{l} Y \\ N \\ R \\ RD \end{array} \right\} \\ ,CON = \left\{ \begin{array}{l} Y \\ N \end{array} \right\} \\ ,SEG = nn \\ ,RNUM = nn \\ ,RTME = nnn \end{array} \right\} \\ C \quad ,D, \text{идентификатор} [,L=cc] \\ M \left\{ \begin{array}{l} ,REF \\ ,UTME = nnn \end{array} \right\} \\ A \left\{ \begin{array}{l} ,REF \\ ,NONE \quad \quad \quad [,L=cc] \\ ,nn,mm... \quad \quad [,L=cc] \end{array} \right\} \\ N, PFK = \left(nn \left\{ \begin{array}{l} ,CMD = 'команда [,команда,...]' \\ ,KEY = ll [,mm...] \end{array} \right\} \right) [,CON = \left\{ \begin{array}{l} Y \\ N \end{array} \right\}] \\ V, USE = \left\{ \begin{array}{l} FC \\ SD \\ MS \end{array} \right\} [,L=cc] \end{array} \right]$
$\{\text{DISPLAY}\}_D$	$\{\text{CONSOLES}\}_C, K, PFK$
$\{\text{MONITOR}\}_{MN}$	$A [,L=x]$
$\{\text{MSGRT}\}_{MR}$	$\left\{ \left(\left\{ \begin{array}{l} D = (\text{операнд}, ...) [,MN=A] [,K] \\ MN=A [,D=\text{операнд} [,K]] \end{array} \right\} [,L=x] \right) \right\}$
$\{\text{STOPMN}\}_{PM}$	$A [,L=x]$

Операция	Операнды
{VARY} {V}	$\left\{ \begin{array}{l} \{ \text{адрес} \\ \text{I=адрес, O=адрес} \}, \text{MSTCONS} \\ (\text{адрес 1}, [\text{адрес 2}], \dots), \text{CONSOLE } [\text{AUTH=x}], [\text{ROUT=y}] \\ [\text{ALTCONS=адрес}] \\ \left[\begin{array}{l} \text{адрес} \\ \text{[SYSLOG]}, \text{HARDCPY} \left[\begin{array}{l} \text{,CMDS} \\ \text{,NOCMDS} \\ \text{,INCMDS} \\ \text{,STCMDS} \\ \text{,OFF} \end{array} \right] \end{array} \right\} [\text{ROUT} = \left\{ \begin{array}{l} \text{ALL} \\ \text{NONE} \\ \text{(число 1,...)} \end{array} \right\}] \end{array} \right\}$
Примечание. $x = \left\{ \begin{array}{l} a \\ cc \\ csa \end{array} \right\}$	

3. Форматы команд для работы с системой ДУВЗ

Операция	Операнды
BDR CST	1. Команды для местного управления ЭВМ $\left\{ \begin{array}{l} \text{DELETE} \\ \text{пппп} \\ \text{пппп, 'текст'} \\ \text{'текст'} \end{array} \right\}$
CENOUT	J=задание, C=класс
{MODIFY} {F}	[имя процедуры.] идентификатор, $\left\{ \begin{array}{l} A \\ D \end{array} \right\} = (\text{канал 1,...})$
MSG	$\left\{ \begin{array}{l} D=\text{идентификатор абонента} \\ M=\text{'текст'} \\ U=\text{идентификатор абонента} \end{array} \right\} [Q]$
SHOW	$\left\{ \begin{array}{l} \text{ACTIVE [NUMBER]} \\ \text{BRDCST} \\ \text{JOBS [имя задания]} \\ \text{LERB [адрес]} \\ \text{MSGS [идентификатор абонента]} \\ \text{SESS [идентификатор абонента]} \\ \text{SESSREL [идентификатор абонента]} \\ \text{USER [идентификатор абонента]} \end{array} \right\}$
{START} {S}	имя процедуры [идентификатор] ... $\left(\left\{ \begin{array}{l} \text{FORM} \\ \text{NFMT} \\ \text{NONE} \end{array} \right\} \left\{ \begin{array}{l} \text{,ABNO} \\ \text{,NORM} \end{array} \right\} \right)$ [ключевые параметры]
{STOP} {P}	[имя процедуры.] идентификатор

Операция	Операнды
USERID	$\left\{ \begin{array}{l} A[DD] = (\text{идентификатор, пароль}) \\ D[DELETE] = (\text{идентификатор, пароль}) \\ S[UPPRESS] \\ R[ESUME] \end{array} \right\}$
2. Команды для удаленного управления системой ДУВЗ с АП	
{CANCEL} C	имя задания
{CONTINUE} CO	$\left\{ \begin{array}{l} B[BEGIN] \\ H[HERE] \\ N[EXT] \end{array} \right\}$
{DELETE} D	$\left\{ \begin{array}{l} \text{имя набора данных} \\ \text{имя набора данных [(имя раздела)]} \end{array} \right\}$
EDIT	$\left\{ \begin{array}{l} \text{имя набора данных [(идентификатор пользователя [ключ])]} \\ \text{имя набора данных [(имя раздела)]} \\ \left[\begin{array}{l} \text{NEW} \\ \text{OLD} \end{array} \right] \left[\begin{array}{l} \text{NUM} \\ \text{NONUM} \end{array} \right] \left[\begin{array}{l} \text{LONG} \\ \text{Lnn [n]} \end{array} \right] \left[\begin{array}{l} \text{S[CAN]} \\ \text{NOS[CAN]} \end{array} \right] \left[\begin{array}{l} \text{CLIST} \\ \text{DATA} \\ \text{DSLST} \\ \text{FORT} \left\{ \begin{array}{l} G \\ H \end{array} \right\} \\ \text{PL/I (параметры)} \\ \text{TEXT} \end{array} \right]$
{EXEC} EX	имя набора данных $\left[\begin{array}{l} L[IST] \\ \text{NOL[IST]} \end{array} \right]$
{LISTBC LISTB LBC LB}	
{LISTDS LISTD LDS}	имя набора данных $S[TATUS]$ $H[ISTORY]$
{LISTLIB L}	$S[TATUS]$ $H[ISTORY]$
{LOGOFF OFF}	
{LOGON LO}	идентификатор пользователя/пароль $\left[\begin{array}{l} \text{BC} \\ \text{NOBC} \end{array} \right] \left[\begin{array}{l} M[SGID] \\ \text{NOM[SGID]} \end{array} \right]$ $A[CCT]$ (учетная информация)

Операция	Операнды
{OUTPUT} O	имя задания {MSG NMSG}
{SEND} SE	'текст' { U[SER] (идентификатор пользователя) N[OW] O[PERATOR] (число) L[OGON] }
{STATUS} ST	[имя задания]
{SUBMIT} SUB	имя набора данных [(идентификатор пароль)] [имя]...
{TABSET} T	[число...] [IN[PUT] OFF] [OUT[PUT]]
{VARY} V	{ [DC=x] [DL=y] [HT=z] [EOM=m] [HT=z] [WAIT=nnn] [LINE=hhh] [MODE={PAGE ROLL}] }
3. Подкоманды команды EDIT	
	номер строки [текст]
CA[NCEL]	имя задания
C[HANGE]	номер строки [номер строки] Δ текст1 Δ текст2 [A[LL]]
D[ELETE]	номер строки [номер строки]
E[ND]	
F[IND]	[номер строки] Δ последовательность символов Δ
I[NPUT]	[номер строки] [[приращение] [I]] [P[ROMPT] R NOP[ROMPT]]
L[IST]	[номер строки [номер строки]] [NUM NONUM]
M[ERGE]	{ имя набора данных [(идентификатор пароль)] } [номер строки1 номер строки2] [номер строки]
REN[UMBER]	[номер строки [приращение]] 10 10
S[AVE]	[имя набора данных] [K[EY] (пароль)]
SC[AN]	[номер строки [номер строки]] {ON OFF}

Операция	Операнды
ZE[ND]	'текст' { U[SER] (идентификатор пользователя [N[OW] O[PERATOR] (число) [L[OGON]] }
SUB[MIT]	{ имя набора данных } [(идентификатор пароль)] [имя набора данных]
TAB[SET]	[число...] [IN[PUT] OFF] [OUT[PUT]]

4. Форматы команды для работы с КРОС

Операция	Операнды
⊗A	{ J J (номер задания 1...[номер задания 5]) }
⊗B	PRT _{n₁} [. {счетчик}] [,PRT _{n₂} ,...]
⊗C	{ устройство1 [,устройство 2,...] J (номер задания 1...[номер задания 5]) }
⊗D	{ A N [,очередь] Q [,очередь] J номер задания 'имя' I [n] D U F }
⊗E	устройство 1 [,устройство 2,...]
⊗F	PRT _{n₁} [. {счетчик}] [,PRT _{n₂} ,...]
⊗H	{ A J (номер задания 1...[номер задания 5]) }
⊗I	PRT _{n₁} [,PRT _{n₂} ,...]
⊗N	устройство 1 [,устройство 2,...]
⊗P	{ имя КРОС I [n] [устройство 1 [,устройство 2,...] J (номер задания 1...[номер задания 5]) }

Приложение 16 (продолжение)

Операция	Операнды
αs	$\left\{ \begin{array}{l} \text{устройство 1 [,устройство 2,...]} \\ \text{TREN, адрес} \\ \text{I [n]} \end{array} \right\}$
αt	$\left\{ \begin{array}{l} \text{устройство, H} \\ \text{устройство [, C=I]} \left[.F = \left\{ \begin{array}{l} R \\ S \\ A \\ \text{пппп} \end{array} \right\} \right] \left[.S = \left\{ \begin{array}{l} Y \\ N \end{array} \right\} \right] \\ \text{Jn [,P=приоритет]} \\ \text{I [n], классы} \end{array} \right\}$
αz	устройство 1 [,устройство 2,...]

ЛИТЕРАТУРА

1. Вычислительная техника и обработка данных. Терминологический толковый словарь фирмы IBM/Пер. с англ. Т. Тер-Микаэляна. — М.: Статистика, 1978, 231 с.
2. Данилочкин В. П., Митрофанов В. В., Одинцов Б. В., Пеледов Г. В. Операционная система ОС ЕС. Справочное пособие/Под ред. Л. Д. Райкова. — М.: Статистика, 1980, 480 с.
3. Джермейн К. Программирование на ИБМ/360/Пер. с англ. под ред. В. С. Штаркмана. — М.: Мир, 1971 и 1973, 872 с.
4. Дроздов Е. А., Комарницкий В. А., Пятибратов А. П. Электронные вычислительные машины Единой системы. — М.: Машинностроение, 1976, 672 с.
5. Единая система ЭВМ/Под ред. А. М. Ларионова. — М.: Статистика, 1974, 136 с.
6. Ерофеев В. И., Меркушов Ю. П., Першиков В. И., Соколов А. П., Средства отладки программ в ОС ЕС ЭВМ. — М.: Статистика, 1979, 245 с.
7. Катцан Г. Вычислительные машины системы 370/Пер. с англ. под ред. В. К. Левина и Л. Д. Райкова. — М.: Мир, 1974, 307 с.
8. Пеледов Г. В., Райков Л. Д. Введение в ОС ЕС ЭВМ. — М.: Статистика, 1977, 120 с.
9. Рейтборт И. М. Пособие для оператора ЕС ЭВМ. — М.: Статистика, 1979, 192 с.
10. Система документации единой системы ЭВМ/Под ред. А. М. Ларионова. — М.: Статистика, 1976, 328 с.
11. Система математического обеспечения ЕС ЭВМ/Под ред. А. М. Ларионова. — М.: Статистика, 1974, 216 с.
12. Шелихов А. А., Селиванов Ю. П. Вычислительные машины. Справочник. Изд. 2-е, перераб. и доп./Под ред. В. В. Пржиялковского. — М.: Энергия, 1978, 224 с.
13. Электронная вычислительная машина ЕС-1020/Под ред. А. М. Ларионова. — М.: Статистика, 1975, 128 с.
14. Электронная вычислительная машина ЕС-1030/Под ред. А. М. Ларионова. — М.: Статистика, 1977, 256 с.
15. Электронная вычислительная машина ЕС-1050/Под ред. А. М. Ларионова. — М.: Статистика, 1976, 304 с.
16. Эшли Р., Фернандес Д. Язык управления заданиями/Пер. с англ. Т. Н. Михайловой. — М.: Мир, 1981, 173 с.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

Абонент 171
Абонентский пункт 171
Аварийное завершение 11, 132
Адрес байта 6
— канала 91, 93
— резидентного устройства SYSRES 91
— устройства 49, 91, 93
Адреса постоянно распределенные в ОП 13
— устройств физические 220
Адресное слово канала CAW 13
Алфавит 34, 218
АЦПУ 7

Байт 6
Бит 6
Бланк кодирования 35
Блок ОП 8
— управления ЭВМ 6, 7

Ввод автономной программы обслуживания 144
— заданий 74, 116
— — с перфокарт 117, 120
— команды оператора с пульта ПМ 16
— пустой строки 178
— системный 74
— текста записей 178
Версия ОС 33
Виды заданий 37
Включение дисплея 19
Внешнее ЗУ 108
Время выполнения задания 42
— — пункта задания 45
Входной поток заданий 43, 116
Вывод результатов 130
— системный 74
— сообщений на ПМ 17

Главный индекс 93
— планировщик MASTER SCHEDULE 32, 95, 101, 108
Граница набора данных 46

Дамп 113, 124
Динамическое перемещение тома 72
Дисплей-консоль 17, 159
Дисплейный комплекс дистанционный 17
— — локальный 17
Дорожка диска 23, 146
ДКОИ 34, 215
ДУВЗ 50, 132, 171

Единица измерения емкости памяти 6
Емкость памяти 5, 6
ЕС ЭВМ 3, 5

Загрузка альтернативного ядра 90
— ОС 90, 92
— программы начальной загрузки IPL 90, 91, 93, 95, 108, 109
— управляющей программы 90
— ядра управляющей программы MFT 103
— — — MVT 92, 93
— — — SVS 108

Загрузчик 30
Задание 32
Задача 32
Задержка выполнения задания 43
Запись 146
Запрос автоматического рестарта 134
— системы 52
Запуск ДУВЗ 173
— КРОС 200
— программы DSO 87
— — INIT 82
— — RDR 76
— — WTR 85

Идентификация задания 39, 43
— последовательности предложений 35
Изменение режима работы дисплея 182, 184, 185
Имена стандартных каталогизированных процедур системных программ 75
Имя групповое 49, 200

Имя набора данных 49

— символическое 37

— типовое 49, 200

Индекс 93

Интерфейс ввода-вывода 6, 7, 8, 15

Канал мультиплексный 7, 8, 17

Каналы 6, 7, 140

— селекторные 6, 7, 17

Каталогизированная процедура 41

Каталогизированные процедуры не-
— стандартные 41, 116, 126

— — стандартные 75

Кбайт 6

Квант времени 107, 108

Клавиатура пульта управления дис-
— плеем 20, 21

Класс задания 40

— входной 40

— выходной 49, 85

— выходных системных сообщений
41

— прерываний 14

Ключ защиты памяти 12

— — программы 12

Код 131, 137

— возврата 131, 137

— завершения 11

— маршрутный 95

— состояния ожидания 11, 14, 92,
105

— прерывания 14

— символов алфавита 34

КОИ-8 34, 216

Колонка продолжения предложения
35

Команды абонента ДУВЗ 172

— — CANCEL 172, 178, 179, 186,
187, 234

— — CONTINUE 172, 178, 179,
186, 187, 234

— — DELETE 172, 178, 190, 234

— — EDIT 172, 178, 183, 234

— — EXEC 172, 178, 186, 187,
234

— — LISTBC 172, 178, 179, 181,
234

— — LISTDS 172, 178, 189, 234

— — LISTLIB 172, 178, 189, 234

— — LOGOFF 172, 178—180, 234

— — LOGON 132, 172, 178, 179,
234

— — OUTPUT 172, 178, 179, 186,
187, 235

— — SEND 172, 178, 179, 181,
235

— — STATUS 172, 178, 179, 186,
187, 235

— — SUBMIT 172, 178, 179, 186,
235

— — TABSET 172, 178, 179, 181

Команды абонента ДУВЗ VARY 172,
178, 179, 181, 235

— автоматические 97, 99, 107, 112,
196

— дисплей-консоли 159

— — CONTROL 159, 161, 163, 232

— — DISPLAY 159, 160, 232

— — MONITOR 159, 160, 232

— — MSGRT 159, 167, 232

— — STORMN 159, 160, 232

— — VARY 159, 167—169, 233

— KPOC 198

— XA 198, 199, 211, 236

— — XB 198, 199, 204, 207, 236

— — XC 198, 199, 204—206, 213,
236

— — XD 198, 199, 201, 286

— — XE 198, 199, 204, 207, 236

— — XF 198, 199, 204, 207, 236

— — XH 198, 199, 211, 286

— — XI 198, 199, 204, 207, 236

— — XN 198, 199, 204, 207, 236

— — XP 198, 199, 204—206, 209,
212, 236

— — XS 198, 199, 204, 205, 209, 237

— — XT 198, 199, 204—206, 209,
211, 212, 237

— — XZ 198, 199, 204—206, 237

— оператора ЭВМ 50

— — CANCEL 50, 51, 65, 70, 75,
77, 81, 82, 86, 116, 117, 119, 121,
124—126, 128—130, 199, 213, 229

— — — CONTROL 50, 51, 232

— — — DEFINE 50, 51, 90, 107,
229

— — — DISPLAY 50, 51, 53, 199,
229

— — — DUMP 50, 51, 90, 113, 230

— — — HALT 50, 51, 90, 96, 113, 230

— — — HOLD 50, 51, 116—119, 125,
199, 230

— — — LOG 50, 51, 53, 63, 230

— — — MODE 50, 51, 90, 114, 230

— — — MODIFY 50, 51, 75, 77, 82,
83, 85, 88, 116, 117, 121, 124, 126,
129, 230

— — — MONITOR 50, 51, 53, 59,
230

— — — MOUNT 50, 51, 65, 67, 230

— — — MSGRT 50, 51

— — — RELEASE 50, 51, 116—120,
125, 199, 230

— — — REPLY 50, 51, 61, 230

— — — RESET 50, 51, 116—119,
125, 231

— — — SET 50, 51, 90, 96, 99, 231

— — — START 50, 51, 75—77, 82, 85,
87, 99, 116, 126, 129, 199, 231

— — — STOP 50, 51, 53, 61, 75, 77,
80, 82, 84, 86, 89, 116, 117, 121,
124, 126, 129, 199, 231

Команды абонента ЭВМ: SWITCH
 50, 51, 231
 — — — STOPMN 50, 51, 53, 61, 80,
 231
 — — — SWAP 50, 51, 65, 72, 231
 — — — UNLOAD 50, 51, 65, 70, 231
 — — — VARY 50, 51, 65, 99, 231
 — — — WRITELOG 50, 51, 53, 64,
 96, 113, 231
 — оператора, работающего в ДУВЗ
 171
 — — — — BDR CST 171, 177, 233
 — — — — CENOUT 171, 174,
 233
 — — — — MODIFY 171, 174, 233
 — — — — MSG 171, 176, 233
 — — — — SHOW 171, 174, 233
 — — — — START 171, 172, 233
 — — — — STOP 171, 173, 233
 — — — — USERID 171, 177, 234
 Комплекс программ технического об-
 служивания 29
 Комплектация моделей ЕС 7
 Комплекты ЭВМ 214
 Консоль альтернативная 159
 — основная (главная) 93, 159
 КПК-12 36, 218, 219
 Курсор 19

Листинг 36

Магнитная головка записывающая 26
 — — считывающая 26
 — лента 26
 Макрокоманда CHKPT 133, 134
 Макрокоманда ввода-вывода 32
 — супервизора 30, 32
 — управления данными 30, 32
 Маркер конца 26
 — начала 26
 — программный ТМ 156
 Маска внешних прерываний 12
 — мультиплексного канала 12
 — прерывания от схем контроля ма-
 шины 12
 — системы 12
 Маски селекторных каналов 12
 Метка начальная 156
 — тома 146
 — фиктивная головная 156
 Модем 17
 Модуль загрузочный 31
 — исходный 30
 — объектный 30
 Мультиплексор передачи данных 17
 Мультипрограммная обработка зада-
 чий 32

Набор данных 46
 — — активный 178, 192
 — — библиотечный SYS1.SYSJOBQE
 74, 96, 97, 112, 172
 — — — SYS1.LINKLIB 47, 144, 150
 — — — SYS1.LOGREC 113
 — — — SYS1.PAGE 112
 — — — SYS1.KPOCACE 196, 199,
 200, 210
 — — — SYS1.KPOCOLIB 196
 — — — SYS1.PAGE 109
 — — — SYS1.PROCLIB 41, 74, 96,
 97, 116, 171, 172, 210
 — — — SYS1.SYSLOG 62, 95, 112,
 169
 — — — SYS1.SYSLOGX 62, 112
 — — — SYS1.SYSLOGY 62, 112
 — — временный 46
 — — входной 46
 — — выходной 46
 — — новый 183
 — — промежуточный 46
 — — старый 183
 — — CRJE.SYSLIB 179, 183, 189
 — — JOBLIB 47
 — — HARDCOPY 112, 159, 169
 — — STEPLIB 47
 — — SYSABEND 46, 47, 124
 — — SYSIN 46
 — — SYSPRINT 46
 — — SYSUDUMP 46, 47, 124
 Назначение команд оператора ЭВМ
 50
 Накопители на магнитных лентах 7,
 8, 25
 — — сменных магнитных дисках 7,
 8, 23
 Номер канала 91
 — системного сообщения 52
 Область ввода команд оператора 17
 — выдачи предостережений 17
 — динамическая 95, 101, 108
 — общая LPA 95, 108
 — памяти 12
 — системных очередей SQA 95, 96,
 101, 108
 — сообщений 17
 — состояния информации 17
 — фиксированная 95, 101, 108
 — экрана дисплея 17
 Обмен страничный 109
 Обращение к памяти 6
 Объем памяти задания 42
 — — пункта задания 45
 Обязанности оператора 3
 Ограничение длины поля дисплея 22
 Операнд 34
 Оператор 34
 Оператор-комментарий 34

Оператор конца задания 34
 — — набора данных 34
 — начала задания JOB 34, 37, 131
 — пункта задания EXEC 34, 135
 — описания данных DD 34, 138
 Операторы программы IBCDASDI 145

— — — DADEF 145, 146, 225
 — — — GETALT 145, 148, 225
 — — — VLD 147, 225
 — — — VTOCD 145, 147, 225
 — — IBCDMPRS 149, 226
 — — — DUMP 149, 226
 — — — RESTORE 149, 150, 226
 — — IEHDASDR 150, 151, 227
 — — — ANALISE 151, 152, 227
 — — — DUMP 151, 153, 228
 — — — GETALT 151, 228
 — — — RESTORE 151, 155, 228
 — — IEHINIT 150, 156, 227
 — — — INITT 156, 227

Операторы языка управления KPOC 196

— — — MESSAGE 196, 197
 — — — PRIORITY 196, 197
 — — — ROUTE 196, 197
 — — — SETUP 196, 198
 — ЯУЗ 34

Операции арифметические 6
 — логические 6

Операция 34

ОС ЕС 29

Основание системы счисления 8

Очередь входная JOB 75, 96, 116—118, 120, 210

— выходная SOUT 75, 84, 96, 117, 125

— задержанная HOLD 43, 75, 96, 116—118, 120, 210

Пакет HCMД 23, 91

— прикладных программ 29

Память виртуальная 108

— оперативная 6, 8, 13, 95, 108

— реальная 108

Параметры ключевые и позиционные 36

— оператора JOB 38
 — — — ния программиста 38, 40
 — — — учетная информация 36, 39
 — — — ADDRSPC 38, 131
 — — — CLASS 38, 40
 — — — COND 38, 131
 — — — MSGCLASS 38, 41, 46, 84, 125
 — — — MSGLEVEL 38, 41, 46, 84, 133

— — — NOTIFY 38, 131, 132

— — — PRTY 38, 40

— — — RD 38, 131—133

Параметры оператора JOB REGION 38, 42

— — — RESTART 38, 131, 132, 134

— — — ROLL 38, 131, 135

— — — TIME 38, 42

— — — TYPRUN 38, 43

— — — EXEC 44

— — — ACCT 44, 136

— — — ADDRSPC 44, 136

— — — DPTTY 44

— — — COND 44, 136

— — — PARM 44, 136

— — — PGM 44, 135

— — — PROC 44, 135

— — — RD 44, 138

— — — REGION 44, 45

— — — ROLL 44, 138

— — — TIME 44, 45

— — — DD 138

— — — * 48, 138

— — — AFF 48, 140

— — — DATA 48, 138

— — — DCB 48, 141

— — — DDNAME 48, 141

— — — DISP 48, 141

— — — DLM 48, 138, 140

— — — DSNAME 47, 48, 49

— — — DUMMY 48, 138, 139

— — — DYNAM 48, 138, 139

— — — COPIES 48, 138, 139

— — — FCB 48, 138, 139

— — — LABEL 48, 142

— — — OUTLIM 48, 138, 139

— — — QNAME 48, 141

— — — SEP 48, 140

— — — SPACE 48, 142

— — — SPLIT 48, 143

— — — SUBALLOC 48, 143

— — — SYSOUT 41, 47—49, 84, 125

— — — TERM 48, 138, 139

— — — UNIT 47—49

— — — UCS 48, 138, 139

— — — VOLUME 47, 48

Перевод целых чисел из одной системы счисления в другую 9, 125

Перечень программ обслуживания 224

Перфокарта 34, 218

Печатающий механизм ПМ 15

Пишущая машинка 7, 15

Планировщик заданий 32, 90

— — MFT 103

— — MVT 90, 98

— — SVS 108, 109

Подготовка планировщика заданий к работе 90, 96

— — — — MFT 103

— — — — MVT 98

— — — — SVS 108

Подготовка ЭВМ к работе 90
 Подканалы 7
 Подключение дисплея к системе 19
 Подкоманды команды EDIT 191
 — — — неявная подкоманда 191,
 193, 235
 — — — CANCEL 190, 191, 195,
 235
 — — — CHANGE 190, 191, 235
 — — — DELETE 190, 191, 193, 235
 — — — END 178, 179, 182, 185, 190,
 191, 235
 — — — FIND 190—192, 235
 — — — INPUT 178, 179, 182, 184,
 190, 191, 235
 — — — LIST 190—192, 235
 — — — MERGF 190, 191, 194, 235
 — — — RENUMBER 190, 191, 194,
 235
 — — — SAVE 182, 185, 190, 191,
 235
 — — — SCAN 190, 191, 193, 235
 — — — SEND 190, 191, 236
 — — — SUBMIT 190, 191, 194, 236
 — — — TABSET 190, 191, 236
 Подполе названия оператора DD 47
 — — — EXEC 44
 — — — JOB 38
 Подполя: название, операции, операн-
 дов, комментарии 35
 Поле идентификации 35
 — оператора 35
 Порядковый номер системного сооб-
 щения 52
 Порядок обработки заданий 74
 Правила записи операторов 34
 — — управляющих операторов ЯУЗ
 34
 Прерывания 11, 13
 — ввода-вывода 13
 — внешние 13
 — замаскированы 12
 — от обращения супервизора 13
 — от схем контроля машины 13
 — программные 13
 Префиксы системных сообщений 52,
 221
 Приоритет диспетчерский 44
 — задания 40
 — принудительный 83
 Программа 43, 135
 — главного планировщика заданий
 32, 96, 98
 — инициатора INIT 33
 — КРОС 196
 — начальной загрузки 13
 — обрабатывающая 30, 31, 74
 — планировщика заданий 32, 96, 98
 — проблемная 30, 31, 101, 104
 — прямого системного вывода DSO
 33

Программа резидентная 103
 — сервисная 30, 31
 — системного ввода RDR 33
 — — вывода WTR 33
 — сортировки-объединения 30, 31
 — транзитная 103
 — управления данными 30, 31, 32
 — — заданиями 30, 31, 32
 — — задачами 30, 31, 32
 — управляющая 11, 31
 — NIP 92, 95, 104, 108, 109
 Программное обеспечение 5
 — — общее 29
 — — специальное 29
 Программы обслуживания 30, 31,
 144
 — — автономные 144, 224
 — — наборов данных 144, 224
 — — системные 144, 224
 — — — IBCDASDI 144, 145, 224,
 225
 — — — IBCDMPRS 144, 149, 224,
 226
 — — — IEHDASDR 144, 151, 224,
 227
 — — — IEHINITT 144, 156, 224,
 227
 Прохождение заданий на ЭВМ 116
 Процедура 43, 135
 — инициатора INIT 223
 — прямого системного вывода DSO
 224
 — — системного ввода RDR 222
 — — — вывода WTR 223
 — — RDRCRJE 171
 Процессор 6, 8
 Пульт управления дисплеем 19
 — — НМЛ 27
 — — НСМД 24
 — — пишущей машинкой 15
 — — процессором 8
 Пункт задания 32, 43

Работа пульта ПМ 27
 Раздел логический 196
 Редактор связей 30, 31
 Режим ввода РВ 178, 183
 — диалоговый 171
 — команд РК 178, 183
 — монопольный 7
 — мультиконсольный 159
 — мультиплексный 7
 — — мультипрограммный с фикси-
 рованным числом задач 33
 — работы НСМД 25
 — — однозадачный 33
 — — с переменным числом задач
 33

Режим с разделением времени 108
 — с разными алфавитными кодами 12
 — редактирования PP 178, 183, 190
 Резидентное устройство 91
 Резидентный том 93, 95
 Рестарт автоматический 132, 134
 — заданий 132
 — отсроченный 132

Связь дисплея с системой 179
 — между оператором и абонентом 181

Сегмент 109
 Семейство ЭВМ «Ряд 1» 5
 — «Ряд 2» 5, 22
 Серийный номер тома 49
 Система двоичная 8, 9
 — прерываний 11
 — программирования 29
 — программного обеспечения 29
 — стандартного соединения 7
 — счисления 8
 — управления заданиями 32
 — шестнадцатеричная 9, 10
 Системная библиотека 75
 — очередь 74
 — программа DSO 74, 75, 77, 87, 97, 99, 103, 116, 125, 126, 130
 — — INIT 74, 75, 77, 82, 97, 99, 101, 116, 121, 126, 129
 — — RDR 74—77, 81, 97, 99, 101, 104, 116, 117
 — — WTR 49, 74, 75, 77, 84, 97, 99, 101, 103, 104, 117, 125, 126, 130
 Системные программы 4
 Системный журнал SYSLOG 62
 Слово состояния канала CSW 13
 — — программы PSW 6, 7, 11, 12, 14, 92

— — — новое 13
 — — — старое 13
 — — — текущее 13

Сообщения системные 11
 Состав выходной информации 41
 Состояние «задача» 12, 30
 — «ожидание» 11, 12, 93, 95, 107

— супервизора 12
 — устройства автономное 65
 — — ввода-вывода 71
 — — оперативное 65

Страница виртуальная 109
 — внешняя 109
 — реальная 109

Строки перфокарты зонные и цифровые 36

Структура логическая ЭВМ 6
 Супервизор 32

Таблица сложения двоичных чисел 9
 Твердая копия HARDCOPY 159, 169
 Тетрада 11

Технические характеристики дисплейного комплекса 17

— — НМЛ 25
 — — НСМД 23
 — — ПМ 15

Типы наборов данных 46

Том 47

— системный резидентный SYSRES 95

Трансляторы 30, 31

Управление выполнением заданий
 ДУВЗ 186, 194

— данными 30
 — заданиями 30, 32
 — задачами 30
 — командами 6
 — логическими разделами 208
 — полями экрана дисплея 181
 — порядком прохождения заданий в КРОС 210

— программой КРОС 199
 — прохождением заданий MVT 116

— режимами работы дисплея в ДУВЗ 182

— устройствами ввода-вывода 65
 — — — в КРОС 204

Управляющая программа PCP 33

— — MFT 33, 38, 44, 51, 90, 109, 196

— — MVT 33, 38, 44, 51, 54, 74, 90, 196

— — SVS 33, 38, 44, 51, 90, 131, 196

— система 29

Управляющие операторы автономной программы обслуживания 145, 225, 226

— языки 30

— — системных программ обслуживания 150, 227

Уровень системных сообщений 41

Условие логическое 131, 137

Условные обозначения 36

Устройство ввода с перфокарт 7

— — с перфолент 7

— ввода-вывода 6, 8, 49, 65

— вывода 7

— контрольно-считывающее 7

— подготовки данных 7

— управления 6, 7, 15

— — пишущей машинкой 15

Учетная информация 39, 196

— — дополнительная 39

Форматы команд оператора 34, 50
— оператора DD 34, 46
— — EXEC 34, 44
— — JOB 34
— операторов ЯУЗ 34
— системных сообщений 50
Фотоселектор 21

Характеристика ЕС ЭВМ 5
— команд абонента 179
— — оператора 50

Шаг задания 32

ЭВМ ЕС 5
— конфигурация 7
Экран дисплея 17
— неформатизированный 18
— форматизированный 18

Ядро управляющей программы 90,
93, 95, 109
Язык программирования 30
— управления заданиями 29, 30, 32,
181
— — — в КРОС 196

ОГЛАВЛЕНИЕ

Предисловие	3
Глава 1. Общие сведения о ЕС ЭВМ	5
1.1. Характеристики ЭВМ	5
1.2. Структура ЭВМ	6
1.3. Комплектация моделей	7
1.4. Системы счисления	8
1.5. Система прерываний	11
Глава 2. Устройства ввода-вывода	15
2.1. Пультовая пишущая машинка	15
2.2. Дисплей-консоль оператора	17
2.3. Накопители на сменных магнитных дисках	23
2.4. Накопители на магнитных лентах	25
Глава 3. Краткие сведения об операционной системе и языке управления заданиями	29
3.1. Основные понятия операционной системы	29
3.2. Система управления заданиями	32
3.3. Управляющий оператор JOB	37
3.4. Управляющий оператор EXEC	43
3.5. Управляющий оператор DD	46
3.6. Общая характеристика команд оператора	50
Глава 4. Информационная связь с системой	53
4.1. Получение информации о состоянии системы и заданий	53
4.2. Отмена непрерывной выдачи информации	61
4.3. Выдача ответов на запросы системы	61
4.4. Работа с системным журналом	62
Глава 5. Управление устройствами ввода-вывода	65
5.1. Изменение состояния устройств ввода-вывода	65
5.2. Резервирование устройства ввода-вывода	67
5.3. Отмена зарезервированного устройства	70
5.4. Разгрузка устройства ввода-вывода	70
5.5. Динамическое перемещение тома	72
Глава 6. Управление системными программами (режим MVT)	74
6.1. Управление программой системного ввода (RDR)	76
6.2. Управление инициатором (INIT)	82
6.3. Управление программой системного вывода (WTR)	84
6.4. Управление программой прямого системного вывода (DSO)	87
Глава 7. Загрузка ОС	90
7.1. Подготовка ЭВМ к работе	90
7.2. Загрузка ядра управляющей программы (MVT)	93
7.3. Команда SET	96
7.4. Подготовка планировщика заданий (MVT)	98

7.5. Загрузка ядра и подготовка планировщика заданий к работе (MFT)	103
7.6. Загрузка ядра и подготовка планировщика заданий к работе (SVS)	108
7.7. Конец работы с ЭВМ	113
7.8. Получение дампа	114
7.9. Получение информации о режиме работы технических средств контроля	114
Глава 8. Управление порядком прохождения заданий в системе MVT	116
8.1. Общая характеристика команд	116
8.2. Управление выполнением заданий, введенных во входном потоке	117
8.3. Управление выполнением нестандартных каталогизированных процедур	126
Глава 9. Более сложные вопросы языка управления заданиями	131
9.1. Управляющий оператор JOB	131
9.2. Управляющий оператор EXEC	135
9.3. Управляющий оператор DD	138
Глава 10. Программы обслуживания	144
10.1. Автономные программы обслуживания IBCDASDI, IBCDMPRS	144
10.2. Системные программы обслуживания IEHDASDR, IEHINITT	150
Глава 11. Команды дисплей-консоли и управления режимами консолей	159
11.1. Информационные команды	159
11.2. Управление консолями	167
Глава 12. Диалоговый удаленный ввод заданий	171
12.1. Команды оператора, непосредственно работающего с ЭВМ	172
12.2. Команды абонента	178
Глава 13. Работа оператора с программой КРОС	196
13.1. Общие сведения	196
13.2. Язык управления заданиями в КРОС	196
13.3. Общая характеристика команд КРОС	198
13.4. Управление программой КРОС	199
13.5. Информационная связь с системой	201
13.6. Управление устройствами ввода-вывода	204
13.7. Управление логическими разделами	208
13.8. Управление порядком прохождения заданий	210
Приложение 1. Минимальный комплект ЭВМ	214
Приложение 2. Числовая таблица для перевода чисел из десятичной системы счисления в двоичную и обратно	215
Приложение 3. Таблица двоичного кода для обмена информацией (ДКОИ). Таблица 8-битного кода для обмена информацией (КОИ-8)	215
Приложение 4. Перфокарты	218
Приложение 5. Шифры (типовые имена), физические адреса и групповые имена устройств ввода-вывода	220
Приложение 6. Префиксы системных сообщений	221
Приложение 7. Стандартная процедура системного ввода RDR	222
Приложение 8. Стандартная процедура инициатора INIT	223
Приложение 9. Стандартная процедура системного вывода WTR	223

Приложение 10. Стандартная процедура прямого системного вывода DSO	224
Приложение 11. Перечень программ обслуживания	224
Приложение 12. Управляющие операторы автономной программы обслуживания IBCDASDI	225
Приложение 13. Управляющие операторы программы IBCDMPRS	226
Приложение 14. Параметры, которые указываются в управляющих операторах DD, при работе с системными программами обслуживания IEHDASDR и IEHINITT	227
Приложение 15. Управляющие операторы программы IEHDASDR	227
Приложение 16. Сводный перечень форматов команд для оператора	229
Литература	238
Предметный указатель	239

*Зоя Петровна Вострикова,
Сергей Тихонович Митин*

КУРС ОПЕРАТОРНОГО ОБСЛУЖИВАНИЯ ОС ЕС ЭВМ

Редактор *О. Ю. Меркадер.*

Техн. редактор *С. Я. Шкляр.* Корректоры: *О. А. Сигал, Н. Б. Румянцева.*

ИБ № 12082

Сдано в набор 20.10.82. Подписано к печати 18.08.83. Т-17654. Формат 60×90^{1/16}. Бумага тип. № 2. Литературная гарнитура. Высокая печать. Условн. печ. л. 15,5. Уч.-изд. л. 18,88. Тираж 90 000 экз. Заказ № 449. Цена 70 коп.

Издательство «Наука»
Главная редакция физико-математической литературы
117071, Москва, В-71, Ленинский проспект, 15.

Ленинградская типография № 2 головное предприятие ордена Трудового Красного Знамени Ленинградского объединения «Техническая книга» им. Евгении Соколовой Союзполиграфпрома при Государственном комитете СССР по делам издательств, полиграфии и книжной торговли. 198052, г. Ленинград, Л-52, Измайловский проспект, 29.