

НИИ «НАУЧНЫЙ ЦЕНТР»

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДВК

КНИГА 2

КОМАНДНЫЙ ЯЗЫК СИСТЕМЫ

МОСКВА 1990 год

АННОТАЦИЯ

Данная книга представляет собой руководство пользователя по работе с командным языком системы, вспомогательными программами и программами работы с файлами.

Глава «КОМАНДНЫЙ ЯЗЫК СИСТЕМЫ» содержит описание каждой команды, ее формат, назначение и особенности использования.

Эта глава также включает описание краткого командного языка системы и командного языка пользователя.

В главе «ВСПОМОГАТЕЛЬНЫЕ ПРОГРАММЫ» описываются возможности и способы использования следующих программ:

- печать;
- сравнение текстовых файлов;
- модификация файлов;
- модификация объектных модулей;
- модификация загрузочных модулей;
- модификация текстовых файлов;
- сравнение двоичных файлов;
- форматирование;
- получение информации о состоянии системы.

Приведены тексты сообщений, описания этих сообщений и действия, которые необходимо предпринять по этим сообщениям.

В главе «ПРОГРАММЫ РАБОТЫ С ФАЙЛАМИ» описаны возможности и способы использования программы обмена, программы обслуживания устройств и программы получения справочника.

1. НАЗНАЧЕНИЕ КОМАНДНОГО ЯЗЫКА СИСТЕМЫ (КЯС)

Командный язык системы — это средство связи пользователя с операционной системой ФОДОС. Он дает возможность пользователю работать с одним из трех мониторов: монитором одного задания (SJ), монитором основного — фонового задания (FB) или монитором расширенной памяти (XM).

Командный язык системы позволяет управлять работой операционной системы ФОДОС-2, т. е. получать справочную информацию о системе, загружать в оперативную память системные программы и программы пользователя, назначать физическим устройствам логические имена, управлять выполнением основного (фонового) задания, выводить на периферийные устройства содержимое одного или нескольких файлов и выполнять ряд других функций.

2. СРЕДСТВА КОМАНДНОГО ЯЗЫКА СИСТЕМЫ

Командный язык системы (КЯС) представляет собой набор команд клавиатурного монитора.

Кроме КЯС пользователь может обращаться к монитору, используя программные запросы (см. [9]).

В данном разделе описан общий формат команд КЯС, а также основные средства, облегчающие использование КЯС.

В дальнейшем будем считать выражения «Команда КЯС», «Команда клавиатурного монитора» и «Команда монитора» эквивалентными.

Команды КЯС могут подаваться непосредственно с терминала, из косвенного файла, из потока заданий или из управляющего файла. Косвенный файл ниже описан подробно. Использование потока заданий см. в [12]. Управляющий файл (косвенный управляющий файл) — объединенный в виде файла набор команд КЯС и директив IND, которые должны последовательно выполняться. Подробно управляющий файл описан в разд. 6.

2.1. Формат команды

Команда КЯС задается с терминала:

- 1) — в виде полной строки, содержащей достаточную информацию для выполнения указанной команды;

Общий формат полной строки:

ком[/прк] входспф[/прк] выходспф[/прк]

общий формат краткой строки:

ком[/прк]

подсказка1 ? Входспф[/прк]

подсказка2 ? Выходспф[/прк]

где ком — имя команды;

/прк — один или несколько переключателей;

входспф — спецификация входного файла;

выходспф — спецификация выходного файла;

подсказка — подсказка системы пользователю — какую информацию должен ввести пользователь. Печатается, если в команде КЯС опущена спецификация входного или выходного файла или имя устройства.

Спецификация файла имеет формат: уст:имфайл.тип

где уст: — логическое или физическое имя устройства;

имфайл.тип — имя и тип файла;

- 2) — в виде краткой строки. После ввода краткой строки на терминале печатается подсказка пользователю, какую информацию надо ввести.

В строке команды допускается указывать до шести входных и до трех выходных файлов. Если команда не помещается на одной строке, с помощью символа дефис (-) можно продолжить ее на следующую строку. Количество символов в строке команды не должно превышать 80. Строка команды должна заканчиваться возвратом каретки.

В строке команды допустимо использовать круглые скобки для более компактной записи нескольких спецификаций файлов.

Примеры:

1. DX0: (TEST.LST,TESTA.LST,TESTB.LST)

система интерпретирует такую запись спецификаций, как

DX0:TEST.LST, DX0:TESTA.LST, DX0:TESTB.LST

2. .COPY DX:FIL(1,2,3).SYS/SYSTEM RK1:

данная строка команды эквивалентна следующей.

.COPY DX:FIL1.SYS,DX:FIL2.SYS,DX:FIL3.SYS/SYSTEM RK1:

Примечание. Существует, однако, ограничение при использовании круглых скобок — количество символов пол-

ной строки команды после раскрытия круглых скобок не должно превышать 80. Поэтому, если используются имена файлов из шести букв, допустимо указывать не более 5 файлов в строке команды.

В данном документе для каждой команды КЯС приведен ее формат. Условные обозначения, используемые в формате команды:

1) для переключателей, которые допускают цифровые аргументы, система воспринимает значения аргументов как десятичные числа, если не оговорено особо. Такие величины, как адреса памяти, система интерпретирует как восьмеричные числа.

2) Такая запись, как [NO]QUERY, представляет собой два взаимоисключающих переключателя NOQUERY и QUERY.

3) Подчеркнутые переключатели являются переключателями по умолчанию.

Не рекомендуется в строке команды указывать взаимоисключающие переключатели.

Монитор допускает сокращение имен команд и переключателей (разд. 3). Простым и правильным для всех имен является их сокращение до 4-х символов или до 6-и, если переключатель с префиксом NO.

2.2. Использование конструкций * и % в командах КЯС

Система допускает использование конструкций «Звездочка» (*) и «Процент» (%) вместо имени файла, типа файла или в качестве одного из символов имени или типа файла (так называемая, встроенная конструкция).

Примеры.

1) Символ * используется вместо имени файла.

Если в команде КЯС спецификация файла имеет вид:

*.MAC ,

в указанной операции будут использованы все файлы тома DK: с типом .MAC независимо от имен файлов.

2) Символ * используется вместо типа файла.

Если в строке команды спецификация файла имеет вид:

TEST.* ,

в указанной операции будут использованы все файлы тома DK: с любым типом и именем TEST.

3) Символ * используется в качестве имени и типа файла.

Если спецификация файла имеет вид:

. ,

в указанной операции будут использованы все файлы тома DK:.

4) Символ * используется вместо нескольких символов имени или типа файла.

Если в команде спецификация файла имеет вид:

A*B.MAC ,

в указанной операции будут использованы те файлы тома DK: с типом .MAC, у которых имя начинается буквой A и заканчивается буквой B.

5) Символ % всегда подразумевает один символ имени или типа входного файла.

Если спецификация входного файла имеет вид :

A%B.MAC ,

система использует в операции те файлы тома DK: с типом .MAC, длина имени у которых три символа. При этом первая буква имени файла — A, третья — B.

В табл. 1 перечислены все команды, которые допускают использование символов * и %.

Таблица 1

Имя команды	Использование символов * и % в спецификации входного файла	Использование символов * и % в спецификации выходного файла
1	2	3
COPY	X	X
DELETE	X	
DIFFERENCES	X	
DIRECTORY	X	
HELP	X	
PRINT	X	
PROTECT	X	
RENAME	X	X
TYPE	X	
UNPROTECT	X	

В командах КЯС, которые допускают использование символов * и %, можно опускать отдельные части спецификации файлов, т. к. система присваивает символ * пропущенным частям спецификации. В табл. 2 символом * обозначены ча-

сти спецификации файла, которые могут быть опущены в допустимых командах.

Таблица 2

Имя команды	Спецификация входного файла	Спецификация выходного файла
1	2	3
COPY, RENAME DIRECTORY PRINT, TYPE DELETE, PROTECT UNPROTECT	*.* DK: *.* *.LST Имфайл.*	*.*

Примеры:

1. .DIRECTORY

система интерпретирует данную команду как:

.DIRECTORY DK: *.*

По данной команде на терминале печатается листинг справочника тома DK:, включающий все файлы данного тома.

2. .PRINT DK:

система интерпретирует данную команду как:

.PRINT DK: *.LST

По данной команде на построочно-печатающем устройстве печатаются все файлы тома DK:, имеющие тип .LST.

2.3. Ввод с терминала и редактирование командной строки

Редактор командной строки (SL) предназначен для изменения строки команды КЯС, командной строки системной программы или других строк (например, строк подсказки) при вводе с терминала. Не используя SL, пользователь может изменить введенную строку двумя способами:

- удалить символы слева от курсора с помощью клавиши <ЗБ>;
- удалить всю напечатанную строку по CY/U и повторно ввести строку.

Указатель текущего положения — курсор (далее в тексте — курсор) размещается в строке за последним обработанным знаком, в настоящем документе обозначается —.

С помощью SL можно редактировать строку, передвигая курсор к различным позициям и удаляя или вставляя символы.

SL доступен для команд КЯС и для фоновых программ и не допустим для печатающего устройства.

Для использования SL следует:

1. Загрузить драйвер SL, подав команду:

.INSTALL SL

если после этой команды было сообщение об ошибке, следует подать команду:

.SET SL SYSGEN

чтобы параметры драйвера SL соответствовали текущему монитору. Затем вновь подать команду:

.INSTALL SL

2. Позволить редактору SL определить тип используемого терминала или ввести тип используемого терминала. Чтобы SL определял сам тип терминала, следует подать команду:

.SET SL ASK

чтобы ввести тип терминала, следует подать:

.SET SL VT13

3. Разрешить использование SL, подав команду:

.SET SL ON

т. е. пользователь может работать с SL при вводе строки с терминала. По окончании работы с SL следует подать:

.SET SL OFF

чтобы при ответе на подсказки системных программ пользователь мог использовать SL, следует подать команду:

.SET SL TTYIN

ниже описаны действия служебных клавиш и их комбинаций в SL. Ограничитель редактируемой строки — <BK>.

Таблица 3

ДЕЙСТВИЯ СЛУЖЕБНЫХ КЛАВИШ И ИХ КОМБИНАЦИЙ

Служебная клавиша/ комбинация	Действие
1	2
<ПФ1> (Служебная 1)	Используется в комбинации с любой клавишей для многократного повторения действия этой клавиши

1	2
<ПФ2> (Справочная) <— (стрелка влево) —> (стрелка вправо) <ПФ1> <— <ПФ1> —> ↑ (стрелка вверх) <ПФ3> (Служебная 2) <ПФ1> <ПФ3> <ЗБ> <ПФ1> <ЗБ> СУ/Н <ПФ1> СУ/Н СУ/У <ПФ1> СУ/У СУ/Р СУ/В <ВК> <ПФ1> <ВК>	Используется для печати вспомога- тельной информации о клавишах Перемещает курсор на одну позицию влево Перемещает курсор на одну позицию вправо Перемещает курсор в начало строки Перемещает курсор в конец строки Воспроизводит на терминале послед- нюю строку, завершленную возвратом карепки Удаляет часть строки от курсора до конца строки Восстанавливает последнюю удален- ную строку Удаляет символ слева от курсора Восстанавливает последний удален- ный символ Меняет местами символ, помеченный курсором, и символ, стоящий справа от курсора Меняет местами символ, помеченный курсором, и символ, стоящий слева от курсора Удаляет левую часть строки между точкой монитора (* системной про- граммы) и курсором Восстанавливает удаленную часть строки Воспроизводит на терминале текущую строку то же что и после СУ/Р Выполняет команду Удаляет правую часть строки от кур- сора до последнего символа строки и выполняет команду

2.3.1. Действие клавиши <СЛУЖЕБНАЯ 1>

Клавиша <СЛУЖЕБНАЯ 1> используется только в ком-
бинации с другими клавишами для многократного повторе-
ния функций.

Если нажать какую-либо клавишу без предварительной печати клавиши <СЛУЖЕБНАЯ 1>, выполняется обычная функция нажатой клавиши.

Если же нажать клавишу <СЛУЖЕБНАЯ 1> и затем клавишу, действие клавиши будет выполняться многократно. Например, если нажать клавишу <СТРЕЛКА ВЛЕВО>, курсор передвигается на одну позицию влево. Если же нажать <СЛУЖЕБНАЯ 1> и <СТРЕЛКА ВЛЕВО>, курсор передвигается в начало строки.

2.3.2. Действие клавиши <СПРАВОЧНАЯ>

Клавиша <СПРАВОЧНАЯ> обеспечивает печать сообщений об ошибках и информацию о функциях клавиш, используемых в SL. Если клавишу <СПРАВОЧНАЯ> нажать один раз, SL печатает сообщение о последней ошибке. Если клавишу <СПРАВОЧНАЯ> нажать второй раз, SL печатает информацию о функциях клавиш, используемых в SL. Для возврата к первоначальному экрану следует нажать любую клавишу, кроме клавиши <СПРАВОЧНАЯ>. Все клавиши затем выполняются как функции SL.

2.3.3. Перемещение курсора

Клавиша <СТРЕЛКА ВЛЕВО> перемещает курсор на один или более знаков влево.

Клавиша <СТРЕЛКА ВПРАВО> перемещает курсор на один или более знаков вправо.

Клавиши <СЛУЖЕБНАЯ 1> и <СТРЕЛКА ВЛЕВО> используются для перемещения курсора в начало строки.

Клавиши <СЛУЖЕБНАЯ 1> и <СТРЕЛКА ВПРАВО> используются для перемещения курсора в конец строки.

Примеры:

1. COPY DX0:A.MAC DX1:B.MSG —

два последовательных нажатия клавиши <СТРЕЛКА ВЛЕВО> перемещают курсор на две позиции влево.

.COPY DX0:A.MAC DX1:B.MSG

2. .DELETE DX0:FILE.TXT

после нажатия 4 раза клавиши <СТРЕЛКА ВПРАВО> курсор перемещается на 4 позиции вправо:

.DELETE DX0:FILE.TXT

3. .RENAME DX0:A.TXT DX1:B.TXT

после нажатия клавиш <СЛУЖЕБНАЯ 1> и <СТРЕЛКА ВЛЕВО> курсор перемещается в начало строки:

`.RENAME DX0:A.TXT DX1:B.TXT`

4. `.COPY FIL1.MAC,FIL2.MAC,FIL3.MAC FILES.MAC`

после нажатия клавиш `<СЛУЖЕБНАЯ 1>` и `<СТРЕЛКА ВПРАВО>` курсор перемещается в конец строки:

`.COPY FIL1.MAC,FIL2.MAC,FIL3.MAC FILES.MAC` _

2.3.4. Действие клавиши `<СТРЕЛКА ВВЕРХ>`

Клавиша `<СТРЕЛКА ВВЕРХ>` используется в ответ на точку монитора или системной программы для воспроизведения на терминале последней введенной строки, завершенной `<ВК>`.

Строка, состоящая только из возврата каретки, не считается последней строкой.

После воспроизведения строку можно редактировать или выполнять напечатанную команду. Курсор расположен в конце воспроизведенной строки.

Пример:

`.LINK RTN1, RTN2, RTN3, RTN4, PROGM/EXECUTE/MAP`

После нажатия клавиши `<СТРЕЛКА ВВЕРХ>` на терминале печатается:

`LINK RTN1, RTN2, RTN3, RTN4, PROGM/EXECUTE/MAP` _

2.3.5. Действие клавиши `<СЛУЖЕБНАЯ 2>`

Клавиша `<СЛУЖЕБНАЯ 2>` стирает часть строки от курсора до конца строки.

Пример:

`.RENAME FIL1.MAC,FIL2.MAC DY1:MYFILE.MAC`

После нажатия клавиши `<СЛУЖЕБНАЯ 2>` строка имеет вид:

`.RENAME FIL1.MAC,FIL2.MAC`

2.3.6. Действие клавиш `<СЛУЖЕБНАЯ 1>` и `<СЛУЖЕБНАЯ 2>` или `<СЛУЖЕБНАЯ 1>` и `СУ/У`

Последовательное нажатие клавиш `<СЛУЖЕБНАЯ 1>` и `<СЛУЖЕБНАЯ 2>` или `<СЛУЖЕБНАЯ 1>` и `СУ/У` восстанавливает все, только что уничтоженные символы строки.

Пример:

После нажатия клавиши `<СЛУЖЕБНАЯ 2>` (см. пример п. 2.3.5.) строка имеет вид:

`.RENAME FIL1.MAC,FIL2.MAC` _

после нажатия клавиш `<СЛУЖЕБНАЯ 1>` и `<СЛУЖЕБНАЯ 2>` строка принимает прежний вид (какой был до нажатия клавиши `<СЛУЖЕБНАЯ 2>`):

`.RENAME FIL1.MAC,FIL2.MAC DY1:MYFILE.MAC`

2.3.7. Действие клавиши <ЗБ>

Клавиша <ЗБ> удаляет символ слева от курсора.

Пример:

.COPY A.MAC B.MAC

после нажатия клавиши <ЗБ> получаем строку:

.COPY A.MAC B.MA

используя клавиши <СЛУЖЕБНАЯ 1> и <ЗБ>, можно восстановить последний удаленный символ.

2.3.8. Действие комбинации клавиш <СУ> и <Н>

Комбинация клавиш <СУ> и <Н> (СУ/Н) меняет местами символ, под которым расположен курсор и символ, стоящий справа от курсора. При этом курсор также перемещается на новую позицию.

Пример:

Если строка имеет вид:

.COPY MYFIL1.SAV MYFIL2.SAV

то после нажатия СУ/Н, получаем строку:

.COPY MYFIL1.SAV MYFIL2.SAV

Последовательное нажатие клавиш <СЛУЖЕБНАЯ 1> и СУ/Н меняет местами символ, помеченный курсором, и символ, стоящий слева от курсора.

Пример:

Если строка имеет вид:

.COPY MYFIL1.SAV MYFIL2.SAV

то после нажатия клавиш <СЛУЖЕБНАЯ 1> и СУ/Н, получаем новую строку: .COPY MYFIL1.SAV MYFIL2.SAV

2.3.9. Действие комбинации клавиш <СУ> и <U>

Комбинация клавиш <СУ> и <U> (СУ/U) удаляет все символы между точкой (.) монитора или звездочкой (*) системной программы и символом, помеченным курсором.

Пример:

Если строка имеет вид:

.RENAME DX0:MAIN.MAC DX1:SUB1.MAC

то после нажатия СУ/U получаем:

..MAIN.MAC DX1:SUB1.MAC

для восстановления удаленных символов следует нажать клавиши <СЛУЖЕБНАЯ 1> и СУ/U.

2.3.10. Действие клавиши <СЛУЖЕБНАЯ 1> и <БК>

Последовательное нажатие клавиш <СЛУЖЕБНАЯ 1>

и <BK> удаляет правую часть строки от курсора до последнего символа строки и выполняет команду.

Пример:

Пусть строка имеет вид:

.COPY DX0:*.MAC DX1:*.BAC DY1:FILES.BAK

после нажатия клавиш <СЛУЖЕБНАЯ 1> и <BK> строка принимает вид: .COPY DX0:*.MAC DX1:*.BAC и выполняется.

2.3.11. Клавиша <BK> используется для выполнения строки независимо от положения курсора.

2.3.12. Действие комбинаций клавиш <CY> и <R> (<CY> и <W>)

Комбинации клавиш <CY> и <R> (CY/R) или <CY> и <W> (CY/W) используются для воспроизведения на терминале текущей печатаемой или последней напечатанной строки. Эти клавиши используются для проверки строки.

2.4. Косвенный файл — это объединенный в виде файла набор команд КЯС и командных строк любой из программ системы ФОДОС, которые должны последовательно выполняться. Косвенные файлы рекомендуется использовать для задач, которые требуют достаточно много машинного времени и при выполнении которых нет необходимости контролировать или вмешиваться в их решение.

2.4.1. Косвенный файл создается по команде КЯС EDIT/CREATE (п. 4.21). По умолчанию тип косвенного файла — .COM, но можно использовать любой другой тип. Каждая строка файла содержит команду КЯС или командную строку любой из программ системы ФОДОС и заканчивается возвратом каретки.

Перед выполнением определенных команд КЯС на терминале печатаются опрашивающие сообщения, на которые пользователь должен ответить.

Пример:

Перед выполнением команды INITIALIZE на TT: печатается: ARE YOU SURE?

Пользователь должен напечатать на терминале:

— Y и возврат каретки для выполнения операции;

— NO и возврат каретки для запрещения операции.

Существует три способа работы с подобными командами:

1) Можно запретить печать опрашивающих сообщений с помощью /NOQUERY. Применяется с командами, у которых /QUERY — переключатель по умолчанию.

2) Ответы на опрашивающие сообщения размещаются в косвенном файле на строках, следующих за командой, которая вызывает печать сообщения. В этом случае пользователь должен знать порядок появления опрашивающих сообщений.

3) Допускается также вводить ответы пользователя непосредственно с терминала. Применяется с командами, которые могут иметь различное число сообщений, особенно, если спецификация файла содержит символы * и %.

Примеры:

1. INITIALIZE/NOQUERY/VOLUMEID DK1:

TAPE6

TIMM

В данных строках косвенного файла за командой, которая вызывает печать двух сообщений на TT:, следуют два ответа, где TAPE6 — идентификатор тома (ID), TIMM — собственное имя владельца тома.

2. Косвенный файл можно изменить так, что пользователь будет вводить ответы с TT:.

INITIALIZE/NOQUERY/VOLUMEID DK1:

ΛC

Последняя строка косвенного файла воспринимается монитором при считывании как комбинация клавиш <CY> и <C>. Возврат каретки должен завершать строку ΛC как и всякую другую. Символы ΛC информируют систему, что ответы будут введены с TT:. Если ΛC — не последняя строка файла, то после ввода ответов с терминала выполнение косвенного файла продолжится.

Примечание. Нельзя в косвенном файле размещать ответы на опрашивающие сообщения за командой, которая вызывает разрушение данных на томе. Например, нельзя использовать следующие строки в косвенном командном файле:

INITIALIZE DY1:

Y

такого типа команды требуют вводить ответы на опрашивающие сообщения непосредственно с терминала. (Можно отменить необходимость вводить эти ответы по /NOQUERY).

2.4.2. Косвенный файл обрабатывается под управлением монитора SJ или в фоновом режиме монитора FB или XM.

Косвенный файл не может выполняться в качестве основного или системного задания.

Для обработки косвенного файла в ответ на точку монитора необходимо подать команду:

Ⓔ спф или Ⓐ Ⓔ спф

где Ⓐ — команда КЯС, указывающая, что файл должен выполняться как косвенный файл, если действует команда SET KMON IND (п. 4.44);

Ⓔ — команда КЯС, указывающая косвенный файл;

спф — спецификация косвенного файла.

Если действует команда SET KMON IND, то по команде Ⓔ спф указанный файл будет выполняться как управляющий файл (используя процессор косвенного управляющего файла). Для того, чтобы указанный файл выполнялся как косвенный командный файл, следует подать команду Ⓐ Ⓔ спф. Для запрещения обработки файла с помощью процессора косвенного управляющего файла надо подать команду SET KMON NOIND.

Команду Ⓔ спф можно разместить также в команде КЯС, если команда Ⓔ спф является последним элементом в строке, не включая комментарий.

Пример:

.DELETE/NOQUERY Ⓔ INDCT ! Комментарий

первая строка косвенного файла INDCT.COM должна содержать список удаляемых файлов. Если первая строка имеет вид:

*.BAK, то приведенная команда КЯС интерпретируется как:

.DELETE/NOQUERY *.BAK.

Необходимо соблюдать предосторожность при использовании командных строк программ в косвенном файле.

Пример:

R PIP

DX1:*.*=DX0:*.*

DX0:*.MAC/D

^C

PRINT DX0:*.LST

если устройство DX1: переполняется прежде, чем все файлы устройства DX0: будут скопированы, то вторая строка косвенного файла выполнится не полностью. В результате выполнения следующей строки будут стерты все файлы с типом .MAC с DX0:. Более безопасно использовать только команды КЯС в косвенном файле. В этом случае управление возвращается монитору после каждой команды и обработка косвенного файла может быть прекращена как только он

обнаружит ошибку. Поэтому более лучшим вариантом косвенного файла с теми же функциями что и вышеприведенный является:

```
COPY DX0:*. * DX1:*. *  
DELETE DX0:*.MAC  
PRINT DX0:*.LST
```

обычно каждая строка косвенного файла, выполняясь, печатается на терминале, и пользователь может следить за обработкой файла. По команде SET TT QUIET (п. 4.44) запрещается вывод на терминал строк косвенного файла. В этом случае на TT: будут печататься только те сообщения, в ответ на которые пользователь должен ввести какую-либо информацию. Для прекращения обработки косвенного файла в любое время надо дважды подать команду СУ/С. Управление возвращается монитору. Прекратить обработку косвенного файла можно также подавая команду СУ/С при ответе на сообщение монитора.

Допустимо вложение косвенных файлов до третьего уровня.

Пример:

после ввода с терминала командной строки

```
@ FIRST
```

обрабатывается косвенный файл FIRST.COM, состоящий из следующих строк:

```
DATE
```

```
TIME
```

```
COPY *.MAC *.BAK
```

```
@ SECOND
```

```
PRINT C
```

```
DIRECTORY/PRINTER DK:
```

Во время обработки косвенного файла FIRST выполняется обработка косвенного файла SECOND, который состоит из одной строки: MACRO/CROSSREFERENCE A+B+C/LIST

Примечания:

1. Точка (.) в начале строки команды в косвенном файле не ставится, также как и * для командных строк системных программ.

2. Запись ^C в вышеприведенных примерах означает два отдельных следующих друг за другом символа ^ (код 134) и C (код 103). Монитор при обработке косвенного файла воспринимает их как СУ/С (код 003). Такое представление облегчает пользователю чтение распечатки косвенного файла, полученной по командам TYPE или PRINT.

3. Для удобства работы пользователь может включить в косвенный файл комментарии. Ограничителем комментариев является восклицательный знак (!). Комментарии не распечатываются на терминале во время обработки монитором косвенного файла.

4. Для любого устройства произвольного доступа, указанного в спецификации косвенного файла в команде @, монитор автоматически загружает в память драйвер этого устройства (по умолчанию — DK:).

5. Если косвенный файл используется для выполнения программы, написанной на ассемблере, необходимо учитывать ограничения на применение запроса .EXIT с косвенными файлами (см. [9]).

3. ПУСК СИСТЕМЫ

Загрузить систему согласно [13]. В ответ появится сообщение: ФОДОС Ф В03.00

Дата [дд—мм—гг]?

Ввести текущую дату: день, порядковый номер месяца, год и подать команду возврат каретки.

Ответ: время [чч:мм:сс]

ввести текущее время: часы, минуты, секунды и подать команду возврат каретки.

Ответ: стартовый файл?

Печать: STARTS <BK>

ответ: SET TT SCOPE и т. д.

4. ОПИСАНИЕ КОМАНД КЯС

В данном разделе описаны назначение, формат и особенности использования всех команд оператора при работе с системой. Команды оператора приведены в порядке латинского алфавита.

Все перечисленные ниже команды являются допустимыми для всех мониторов за исключением команд ABORT, FRUN, SRUN, SUSPEND и RESUME, которые применяются с основным и системным заданиями и поэтому являются недопустимыми для монитора SJ. Обработка фонового задания выполняется под управлением монитора SJ.

Если в строке команды допущена ошибка или система не может определить, какое действие необходимо выполнить, на терминале печатается сообщение об ошибке. Сообщение об ошибке указывает, какая произошла ошибка и какая системная программа обнаружила эту ошибку.

Для каждой команды все допустимые переключатели приведены в таблицах. Переключатели, помеченные в таблице одинаковым количеством звездочек (*) являются взаимoisключающими. Переключатели в строке команды следуют за именем команды, если не оговорено особо.

Если не указано особо, все числовые значения, встречающиеся в командах клавиатурного монитора, должны быть десятичными.

Операционная система ФОДОС-2 позволяет удалить некоторые команды КЯС во время генерации системы.

Команда, указанная в строке команды, анализируется: является ли она командой КЯС; если нет, то — командой краткого командного языка системы (ККЯС) или командой командного языка пользователя (КЯП). Если команда не является допустимой ни для КЯС, ККЯС или КЯП, система печатает сообщение об ошибке. Подробное описание ККЯС и КЯП приведено в разд. 5.

4.1. Команда ABORT, поданная с системного терминала, используется для немедленного прекращения выполнения основного или системного задания, которому был назначен конкретный терминал (полезно ознакомиться с описанием команд FRUN или SRUN/TERMINAL:N в п. 4.25 и п. 4.47).

Формат команды: AB[ORT] имязд

где имязд — имя системного или основного задания.

Команда ABORT может быть введена в виде полной строки или краткой. После ввода краткой строки ABORT, на терминале печатается:

JOBNAME?

Следует подать имя задания и возврат каретки.

Если система работает под управлением монитора FB или XM и, если монитор не поддерживает системное задание, в качестве имени основного задания используется F. Если монитор поддерживает системное задание, следует подавать имя задания, выполнение которого надо прекратить.

4.2. Команда ASSIGN используется для назначения устройству логического имени.

Формат команды: AS[IGN] фимуст лимуст

где фимуст — постоянное имя устройства. Список постоянных имен устройств операционной системы ФОДОС приведен в документе [1]. Вместо фимуст можно указывать логическое имя устройства.

Лимуст — логическое имя устройства, представляющее собой буквенно-цифровое имя длиной от одного до трех сим-

волов (пробел и табуляция недопустимы для логического имени).

Если в строке команды опущено:

— постоянное имя устройства (фимуст), на терминале печатается:

PHYSICAL DEVICE NAME?

— Логическое имя устройства (лимуст), на терминале печатается:

LOGICAL DEVICE NAME?

Пользователь должен ввести с терминала постоянное или логическое имя устройства и возврат каретки.

Если по командам ASSIGN одно и то же логическое имя назначено двум разным устройствам, то второе назначение отменяет предыдущее. По одной команде ASSIGN можно присвоить логическое имя только одному устройству. Недопустимыми логическими именами являются — BA и SY.

С помощью нескольких команд ASSIGN можно присвоить несколько логических имен одному устройству.

По команде SHOW можно узнать, какие логические имена были присвоены устройствам по команде ASSIGN.

Пример:

.ASSIGN LP: OUT:

по данной команде построчно-печатающему устройству присваивается логическое имя OUT:

4.3. По команде В устанавливается база перемещения.

формат команды: В[адр]

где адр — адрес базы (восьмеричный). Если указанный адрес — нечетный, система делает его четным, уменьшая на единицу. Если в строке команды адрес опущен, система устанавливает адрес базы на 0.

Команда В не влияет на выполнение программы. Если используется команда Е или D, монитор складывает адрес, указываемый в этих командах, и базу перемещения для получения адреса ячейки, к которой будет обращение. Если адрес базы равен адресу загрузки модуля, то относительные адреса листинга модуля могут использоваться для обращения к ячейкам этого модуля.

Примеры:

1. В

по этой команде адрес базы устанавливается на 0

2. В 1000.

.В 1001

обе эти команды устанавливают адрес базы на 1000.

4.4. Команда BACKUP. По команде BACKUP создается копия большого файла или тома на несколько томов меньшей емкости и наоборот — копирование содержимого нескольких томов в один большой файл или том. Формат команды:

BACKUP[/прк] входспф выходспф

где /прк — один или два переключателя команды BACKUP;

входспф — спецификация копируемого файла/имя копируемого тома; выходспф — имя тома/спецификация файла, куда выполняется копирование.

Таблица 4

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ BACKUP

Переключатель	Функция
1	2
/DEVICE	Копирует содержимое одного тома на несколько томов меньшей емкости. Емкость тома, куда выполняется копирование, должна быть меньше емкости копируемого тома. /DEVICE используется с /RESTORE для копирования содержимого нескольких томов на один том
/RESTORE	Копирует в один файл содержимое нескольких томов, созданных по команде BACKUP
Никакого	Копирует содержимое большого файла на несколько томов

Если после выполнения команды BACKUP том является частью большого файла (тома большой емкости), то использовать этот том можно только для хранения информации. После копирования содержимого нескольких томов в один файл (том) можно опять использовать созданный файл (том).

В команде BACKUP нельзя использовать конструкцию * и %.

Копирование всего тома на несколько томов меньшей емкости выполняется одним файлом, имя которого совпадает

с именем устройства, с которого выполняется копирование, и тип файла по умолчанию .BUP (если не указаны имя и тип файла).

Устройство произвольного доступа может быть использовано в качестве устройства куда и откуда выполняется копирование. Магнитная лента может быть использована только в качестве устройства, куда выполняется копирование при копировании одного тома (файла) на несколько томов меньшей емкости или в качестве устройства, откуда происходит копирование, при копировании нескольких томов в один том (файл).

Копирование одного тома (файла) по команде BACKUP выполняется на несколько томов обязательно меньшей емкости. В противном случае, если выполняется копирование тома (файла) на том равной с ним емкости, печатается сообщение об ошибке.

При копировании тома (файла) на несколько томов меньшей емкости после заполнения каждого тома система печатает подсказку:

MOUNT NEXT OUTPUT VOLUME IN <УСТ>; CONTINUE?

Следует заменить том следующим и напечатать Y и возврат каретки. Как только том становится полным, система печатает порядковый номер заполненного тома, чтобы пользователь мог пометить снятый том. Так повторяется до тех пор, пока том (файл) не будет полностью скопирован.

Том, на какой выполняется копирование, должен быть инициализирован по команде INIT/BACKUP (п. 4.27). Если пользователь установил том, который:

- инициализирован не по INIT/BACKUP;

- не является устройством произвольного доступа;

- уже содержит файлы, система сообщает об ошибке.

Пользователь может заменить том или позволить системе инициализировать уже установленный том.

При копировании нескольких томов в один файл, если не указано имя копируемого файла, система использует в качестве имени файла имя указанного тома и тип файла по умолчанию .BUP. Если не указано имя создаваемого большого файла, система использует имя и тип копируемого файла. При создании одного большого файла система печатает подсказку установить каждый последующий том. Если томы устанавливаются в неверном порядке или установленный том содержит не тот файл, система сообщает об этом и повторно печатает подсказку установить следующий том. О заверше-

нии создания одного большого файла система также сообщает.

Примеры:

1. .BACKUP/DEVICE RK0: DX0:

в данном примере том RK0: выводится на дискеты с одинарной плотностью DX: файлом RK.BUP.

2. .BACKUP RK0:MYPROG.MAC DX0:

MOUNT OUTPUT VOLUME IN DX0;; CONTINUE?Y

?BUP—I—CREATING VOLUME <N>

MOUNT NEXT OUTPUT VOLUME IN DX0;; CONTINUE?Y

?BUP—W—NOT A BACKUP VOLUME DX0:

DX:/?BUP INITIALIZE; ARE YOU SURE?Y

?BUP—I—BAD BLOCK SCAN STARTED...

?BUP—I—NO BAD BLOCKS DETECTED

?BUP—I—CREATING VOLUME <N>

MOUNT NEXT OUTPUT VOLUME IN DX0;; CONTINUE?Y

?BUP—I—CREATING VOLUME <N>

В данном примере файл MYPROG.MAC копируется на дискеты с одинарной плотностью DX:. Во время операции система обнаружила, что вторая дискета инициализирована не по INIT/BACKUP; система выполняет инициализацию установленного тома.

3. .BACKUP/DEVICE RK0: DX0:RK.BUP

MOUNT OUTPUT VOLUME IN DX0;; CONTINUE?Y

?BUP—I—CREATING VOLUME <N>

MOUNT NEXT OUTPUT VOLUME IN DX0;; CONTINUE?Y

?BUP—W—NOT A BACKUP VOLUME DX0:

DX:/?BUP INITIALIZE; ARE YOU SURE?Y

?BUP—I—BAD BLOCK SCAN STARTED...

?BUP—I—NO BAD BLOCKS DETECTED

?BUP—I—CREATING VOLUME <N>

MOUNT NEXT OUTPUT VOLUME IN DX0;; CONTINUE?Y

?BUP—I—CREATING VOLUME <N>

В данном примере том RK0: копируется на дискеты с одинарной плотностью файлом с именем RK.BUP.

4. .BACKUP/DEVICE/RESTORE DX0:RK.BUP RK0:

В данном примере файл RK.BUP копируется с нескольких дисков с одинарной плотностью на том RK0:.

4.5. Команда BASIC. По команде BASIC в оперативную память вызывается интерпретатор языка БЕЙСИК.

Формат команды: BASIC

Команда BASIC не допускает использование переключателей и спецификаций файла. Описание языка БЕЙСИК см. в документе [10].

4.6. Команда BOOT. По команде BOOT загружается нужный монитор и передается ему управление. Команда BOOT используется также для загрузки в память новой копии монитора.

Формат команды: BO[OT]/[прк] спф

где /прк — переключатель;

спф — спецификация файла монитора или имя тома устройства.

Таблица 5

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ BOOT

Переключатель	Функция
1	2
/FOREIGN	Загружает предыдущую версию монитора ФОДОС В02.00. По этому переключателю не сохраняется дата и время
/WAIT	Используется, если конфигурация вычислительной системы содержит один привод или если пользователь хочет загрузить том с привода, который занят системным томом. После подачи команды с переключателем /WAIT, система инициирует выполнение команды, но затем приостанавливается и печатает: MOUNT INPUT VOLUME IN <УСТ>; CONTINUE? где <УСТ> — имя тома устройства. Следует установить том, содержащий нужный файл монитора, и печатать: — Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции;

1	2
	— N (или любую строку, начинающуюся с N) для запрещения операции. После отрицательного ответа на терминале печатается: MOUNT SYSTEM VOLUME IN <УСТ>; CONTINUE? Следует установить системный том и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ вызывает повторную печать сообщения. СУ/С для прекращения операции — система печатает точку. Для использования /WAIT на системном томе должен быть файл DUP.SAV

Если в строке команды опущена спецификация файла, на терминале печатается: DEVICE OR FILE?

Пользователь должен ввести с терминала имя файла монитора или имя тома и возврат каретки.

Команда BOOT выполняет одну из следующих функций:

— перезагрузку системы с определенного тома, если в строке команды указано имя тома;

— перезагрузку определенного файла, если в строке команды указано имя файла монитора.

При использовании команды BOOT можно не вводить повторно время и дату. Однако, во время перезагрузки часы системы могут отстать на несколько секунд.

В команде BOOT можно использовать не только стандартные имена мониторов.

Примечание. Если команда BOOT подается в то время, когда основное или системное задание выполняет ввод—вывод на системный том, то система может «зависнуть». Поэтому, следует завершить выполнение задания перед подачей команды BOOT.

Примеры:

1. .BOOT DK:

по данной команде загружается монитор, загрузчик которого записан на DK:.. Можно изменить загрузчик тома DK: по команде КЯС: COPY/BOOT.

2. .BOOT DX0:FMONSJ

по данной команде загружается монитор FMONSJ.

3. .BOOT/WAIT DX0:

MOUNT INPUT VOLUME IN DX0:; CONTINUE?Y

по данной команде загружается монитор, загрузчик которого записан на DX0:..

4.7. Команда CLOSE делает постоянными все файлы открытые в текущий момент времени в фоновом задании.

Формат команды: CL[OSE]

команда CLOSE не допускает переключателей и аргументов.

Команда CLOSE используется после:

— печати CY/C, т. е. после прекращения выполнения фонового задания;

— неожиданного прекращения выполнения программы.

Команда CLOSE не действует на файлы основного задания и на файлы на MT:; а также на дополнительные каналы ввода—вывода (определенные с помощью программного запроса .CDFN).

4.8. Команда COMPILE. По команде COMPILE вызываются один или более языковых трансляторов (АССЕМБЛЕР, ДИБОЛ или ФОРТРАН) для трансляции указанных файлов. Формат команды:

COM[PILE][[/прк] спф[, ... спф] [/прк]

где /прк — один или несколько переключателей;

спф — спецификация файла, который будет транслироваться.

Таблица 6

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ COMPILE

Переключатель	Функция
1	2
Переключатели общие для всех трансляторов: /ALLOCATE:N	Используется с /LIST или /OBJECT для резервирования области памяти на томе для выходного файла. Аргумент, N, указывает число

1	2
/LIST[:спф] /OBJECT[:спф] /NOOBJECT Переключатели для трансляции с языка DIBOL: /ALPHABETIZE /BUFFERING /CROSSREFERENCE /DIBOL /LINENUMBERS /NOLINENUMBERS	блоков области памяти для размещения файла; N — целое число от 1 до 65535, может быть равно также —1. (В этом случае на томе для выходного файла отводится наибольшая свободная область памяти) Используется для получения листинга при трансляции; спф — спецификация файла листинга. Если спецификация .LIST следует за именем команды, то листинг печатается на LP: Используется для получения объектной программы; спф — спецификация объектного файла Запрещает создание объектного файла Располагает по алфавиту таблицу символов и таблицу меток Указывает, что будет использоваться простая буферизация при вводе — выводе. Обычно используется двойная буферизация Создает таблицу перекрестных ссылок в листинге. Т. к. система не печатает листинг по умолчанию, надо указать /LIST для получения распечатки таблицы перекрестных ссылок Вызывает DIBOL для трансляции указанных файлов Разрешает вывод номеров внутренней последовательности в объектный код Запрещает вывод номеров внутренней последовательности в объектный код

1	2
/LOG /ONDEBUG /PAGE:N /TABLES /WARNINGS /NOWARNINGS Переключатели для трансляции с языка ФОРТРАН: /CODE:XXX /DIAGNOSE /EXTEND /FORTRAN /HEADER	Создает список сообщений об ошибках, полученных при трансляции Включает таблицу символов в объектный код Определяет количество строк в странице листинга; N — любое десятичное число от 1 до 32768. По умолчанию N=66 Включает в листинг таблицу символов и таблицу имен. Используется только с /LIST Разрешает вывод предупреждающих сообщений об ошибке Запрещает вывод предупреждающих сообщений об ошибке Используется для получения объектного кода определенного вида (XXX), исходя из конкретной аппаратной конфигурации. Аргумент XXX — трехбуквенное сокращение вида кода (допустимые значения — EAE, EIS, FIS и THR). Подробное описание видов кода программы и их функций см. в документе [8] Разрешает вывод расширения поля генерированного (поточного) кода листинга для определения ошибок трансляции и их исправления Разрешает расширение строк исходной программы до 80 символов Вызывает ФОРТРАН для трансляции указанных файлов Разрешает вывод поля режима работы транслятора

1	2
/I4 /LINENUMBERS /NOLINENUMBERS /ONDEBUG /RECORD:N /SHOW:N /STATISTICS /SWAP /NOSWAP /UNITS:N /VECTORS	Резервирует два слова памяти для целых переменных. По умолчанию резервируется одно слово памяти Разрешает вывод номеров внутренней последовательности в объектный код Запрещает вывод номеров внутренней последовательности в объектный код Разрешает трансляцию строк отладки. (Строка отладки имеет букву D в первой колонке). По умолчанию строки отладки рассматриваются как комментарии Определяет максимальный размер записи в байтах для операторов форматного ввода-вывода ($4 < N < 4095$). По умолчанию $N = 132$ Управляет печатью основных полей листинга. Аргумент N — код, определяющий формат листинга. В табл. 9 описаны аргументы N и их значения ($0 \leq N \leq 7$) Разрешает вывод поля статистики листинга Разрешает свопинг программы USR во время выполнения программы пользователя Запрещает свопинг программы USR во время выполнения программы пользователя Определяет максимальное число логических устройств ($1 \leq N \leq 16$), которые могут быть открыты одновременно. По умолчанию $N = 6$ Разрешает векторизацию многомерных массивов. Подробное описание векторизации см. в документе [8]

1	2
/NOVECTORS /WARNINGS /NOWARNINGS Переключатели для трансляции с языка Ассемблер: /CROSSREFERENCE [:тип[, ... :тип]] /DISABLE:влч [, ... влч] /ENABLE:влч [, ... влч] /LIBRARY	Запрещает векторизацию многомерных массивов Разрешает вывод предупреждающих сообщений об ошибках Запрещает вывод предупреждающих сообщений об ошибках Создает определенное поле таблицы перекрестных ссылок в листинге. Аргумент тип представляет собой одно — символьный код, указывающий, какие поля таблицы перекрестных ссылок должны быть включены транслятором в листинг. В табл. 8 перечислены все аргументы и их назначения. Т. к. система не печатает листинг по умолчанию, надо указать /LIST для получения распечатки таблицы перекрестных ссылок Используется для определения директивы .DSABLE. Табл. 7 содержит аргументы влч и их назначения Используется для определения директивы .ENABLE. Табл. 7 содержит список аргументов влч и их назначения. Директивы .DSABLE и .ENABLE описаны в документе [2] Используется для определения файла макробиблиотеки. Перед обращением к системной макробиблиотеке, SYSMAC.SML, транслятор просматривает все макробиблиотеки пользователя. В строке команды

1	2
/MACRO /SHOW:влч /NOSHOW:влч	следует за спецификацией файла макроопределений Вызывает АССЕМБЛЕР для трансляции указанных файлов Уточняет использование директивы .LIST. В табл. 9 описаны аргументы влч и их использование Уточняет использование директивы .NLIST. В табл. 9 описаны аргументы влч и их использование. Директивы .LIST и .NLIST описаны в документе [2]

По умолчанию тип файла листинга — .LST, тип объектного файла — .OBJ, исходного файла — .MAC.

Для независимой трансляции нескольких исходных файлов спецификации файлов разделяются запятыми. Знак плюс (+) между спецификациями используется для трансляции нескольких исходных файлов в один объектный файл с именем первого входного файла и типом .OBJ. Однако, пользователь может указать любое имя для объектного файла по /OBJECT[:спф].

Переключатели команды COMPILE (/LIST и /OBJECT) имеют различное назначение в зависимости от занимаемого места в строке команды. Переключатели, следующие за именем команды, действуют на всю строку. Если переключатели следуют за спецификацией файла, то они действуют только на этот файл (или группу файлов, соединенных знаками плюс).

Команда COMPILE может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается: FILES?

Пользователь должен ввести с терминала спецификацию исходного файла и возврат каретки.

Существует три способа определения в строке команды нужного языкового транслятора. Для этого пользователь должен указать:

1. Переключатель (/MACRO, /FORTRAN или /DIBOL)
2. Тип файла (.MAC, .FOR или .DBL)

3. Если в строке команды не указан ни переключатель, ни тип файла, система ищет в справочнике файл с указанным именем и типом .MAC и затем с типом .DBL. Если нужный файл в справочнике существует, вызывается соответствующий транслятор. Если — нет, система предполагает, что файл имеет тип .FOR и вызывает транслятор ФОРТРАН.

Примеры:

1. .COMP A.MAC+B.MAC/LIST,MYLIB/LIBR+
+C.MAC/LIST/OBJECT

по данной команде транслируются:

— A.MAC и B.MAC вместе, создавая B.LST и A.OBJ

— C.MAC, удовлетворяя аргументы директивы .MCALL из MYLIB.MAC и SYSMAC.SML, создавая C.OBJ и C.LST.

2. .COMPILE/LIST:TT: A.MAC

по данной команде листинг будет печататься на TT: .

3. .COMP/LIST:RK3: A.MAC

по данной команде в результате трансляции создается RK3:A.LST и A.OBJ на DK:.

4. .COMPILE/MACRO A/LIST:B

.COMPILE/MACRO/LIST:B A

после выполнения обеих команд образуются файлы A.OBJ и B.LST на DK:

Нельзя задавать строку команды подобную следующей:

.COMPILE/LIST:FILE2 A.MAC,B.MAC

Таблица 7

Аргумент	Действие по умолчанию	Разрешает (/ENABLE:влч)/ Запрещает (/DISABLE:влч)
1	2	3
ABS	Запрещение	Вывод в абсолютном двоичном формате
AMA	Запрещение	Относительные адреса транслировать как абсолютные
CRF	Разрешение	Вывод листинга таблицы перекрестных ссылок
FPT	Запрещение	Усечение чисел с плавающей запятой без округления (/ENABLE:FPT), с округлением (/DISABLE:FPT)

1	2	3
GBL	Разрешение	Неопределенные имена транслировать как глобальные
LC	Разрешение	Прием входной информации, содержащей буквы русского алфавита
LCM	Запрещение	Зависимость верхнего и нижнего регистров от условных директив .IF IND и .IF DIF
LSB	Запрещение	Временно прервать блок локальных меток новой программной секцией, а затем продолжить прерванный блок локальных имен
MCL	Запрещение	Поиск макроопределения во всех макробibliothеках, если в исходном файле встретился код неопределенной операции
PNC	Разрешение	Включение кодов в объектный модуль
REG	Разрешение	Переопределение регистров, назначенных по умолчанию

Таблица 8

Аргумент	Тип секции
1	2
S	Имена пользователя
R	Имена регистров
M	Имена макрокоманд
P	Постоянные имена (команды, директивы)
C	Имена программных секций
E	Сообщения об ошибках
—	Эквивалентно аргументам :S:M:E

Таблица 9

Аргумент	Действие по умолчанию	Разрешает (/SHOW:влч)/ Запрещает (/NOSHOW:влч)
1	2	3
SEQ	Разрешает	Включать в листинг порядковые номера строк исходной программы
LOC	Разрешает	Печать значения счетчика адреса
BIN	Разрешает	Печать объектных кодов в восьмеричном виде
BEX	Разрешает	Печать объектных кодов в восьмеричном виде, расположенных на нескольких строках
SRC	Разрешает	Печать операторов исходной программы
COM	Разрешает	Печать комментария
MD	Разрешает	Печать макроопределений и расширений областей повторений
MC	Разрешает	Печать макрокоманд и расширений областей повторений
ME	Запрещает	Печать макрорасширений
MEB	Запрещает	Печать операторов макрорасширений, порождающих объектные коды
CND	Разрешает	Печать блоков условной трансляции с невыполненными условиями
LD	Запрещает	Действие директив, .LIST и .NLIST исходной программы, в которых нет аргументов
TOC	Разрешает	Печать таблицы содержания
TTM	Запрещает	Выбирать формат печати объектных кодов и таблицы имен
SYM	Разрешает	Печать таблицы символов

Таблица 10

Код, определяющий формат листинга	Формат листинга
1	2
0	Вывод диагностики
1 или SRC	Вывод исходной программы и диагностики

1	2
2 или MAP	Вывод карты памяти и диагностики
3	Вывод диагностики, исходной программы и карты памяти
4 или COD	Вывод генерированного поля и диагностики
7 или ALL	Вывод диагностики, исходной программы, карты памяти и генерированного кода

4.9. Команда COPY. Команда COPY позволяет:

- копировать файл(ы);
- объединять набор файлов в один файл;
- копировать один том на другой том;
- копировать загрузчик системы;
- копировать содержимое одного тома на другой том одним файлом или наоборот;
- копировать содержимое одного тома большей емкости на несколько томов меньшей емкости.

Формат команды:

COPY[Y][/прк] входспф[...входспф][/прк] выходспф[/прк]

где /прк — один или несколько переключателей;

входспф — спецификация входного (копируемого) файла;

выходспф — спецификация выходного файла.

Таблица 11

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ COPY

Переключатель	Функция
1	2
/ALLOCATE:N	Резервирует область памяти на томе для выходного файла. Аргумент, N, указывает число блоков области памяти для размещения файла; N — целое число от 1 до 65535, может быть равно также —1 (в этом случае на томе для выходного файла отводится наи-

1	2
/ASCII *	большая свободная область памяти). Переключатель следует за спецификацией выходного файла Копирует файлы в кодах КОИ-7, игнорируя коды ПУС,ЗБ; СУ/З воспринимается как логический конец файла. Недопустим с переключателем /VERIFY
/BEFORE[дата]	Копирует все файлы, созданные до указанной даты
/BINARY *	Копирует файлы двоичного формата (файлы типа .OBJ и .LDA) с проверкой на контрольную сумму. При возникновении ошибки печатается сообщение об ошибке и операция прерывается. По этому переключателю нельзя копировать библиотечные файлы из-за ошибок контрольной суммы. Эти файлы следует копировать в режиме отображения памяти. Недопустим с переключателем /VERIFY
/BOOT[:уст]	Копирует загрузчик системы из файла монитора в блоки 0 и 2—5 указанного тома устройства с произвольным доступом. Нельзя объединять /BOOT с другими переключателями. Допускается переименовывать файлы монитора и по команде COPY/BOOT использовать не только стандартные имена файлов монитора. Аргумент уст — имя устройства, для которого создается системный том. Аргумент используется при создании системного тома для устройства другого типа
/CONCATENATE	Объединяет набор входных файлов и копирует их в виде одного выходного файла. Конструкция * и % недопустима в спецификации выходного фай-

1	2
	ла. Переключатель /CONCATENATE полезен для объединения набора объектных модулей в один файл для использования его редактором связей или библиотекарем /DATE[:дата] Копирует файлы с указанной датой создания. Если файл не имеет дату создания, используется текущая системная дата /DELETE Удаляет входные файлы после завершения копирования. По COPY/DELETE не печатаются опрашивающие сообщения перед выполнением операции (следует использовать /QUERY для печати опрашивающих сообщений). Если входспф и выходспф совпадают, входной файл не удаляется /DEVICE Копирует «Блок в блок» содержимое одного тома на другой в режиме отображения памяти без изменения структуры файлов и расположения их на томе. Томы, откуда и куда выполняется копирование, не должны содержать плохих блоков. Если во время операции система обнаружит плохие блоки, печатается сообщение об ошибке и операция повторяется. Если будет напечатано только одно сообщение об ошибке, можно предположить, что передача завершена правильно. Если емкость одного тома меньше емкости другого, система копирует только то число блоков, которое может содержать том меньшей емкости (т. е. может быть скопирован весь справочник копируемого тома, но не все файлы этого тома). Перед копированием тома большей емкости на том мень-

1	2
	шей емкости печатается опрашивающее сообщение: Уст./COPY;ARE YOU SURE? следует подать Y и возврат каретки для выполнения операции. На магнитную ленту можно скопировать только одну копию тома диска Используется с переключателями /START:N и /DEVICE и указывает последний копируемый блок тома. Переключатель /END:N должен следовать за спецификацией входного файла. Аргумент N — десятичный номер блока. Если не указан аргумент, N, система предполагает последний блок тома Копирует все файлы тома кроме указанных Используется с переключателем /DEVICE для копирования содержимого одного тома на другой том одним файлом или наоборот. При копировании файла образа тома на другой том загрузчик системы и справочник на выходном томе заменяются соответствующими блоками из входного файла. При копировании на MT: надо указывать имя файла в выходной спецификации Игнорирует входные ошибки, которые происходят при передаче файла и продолжает копирование. Передача данных происходит по одному блоку в один прием Если строка команды не содержит переключателей или содержит /IMAGE, выполняется копирование файлов в режиме отображения памяти. Не сле-
/END[:N]	
/EXCLUDE	
/FILES	
/IGNORE	
* /IMAGE -----	

1	2
/INFORMATION	дует копировать по /IMAGE файлы исходного и двоичного форматов, т. к. в этом случае не происходит проверки, предусмотренных по /ASCII и /BINARY Используется, если пользователь не уверен, есть ли на томе указанные в строке команды файлы. Если указанный файл не найден — печатается информационное сообщение об ошибке, но выполнение операции не прекращается — выполняется копирование найденных файлов
/LOG	Печатает на терминале список файлов, включенных в операцию COPY. Этот список печатается по умолчанию, если спецификации файлов содержат конструкцию * и %
/NOLOG	Запрещает печать на ТТ: списка скопированных файлов
/MULTIVOLUME	Копирует файлы с одного тома большей емкости на один или более томов меньшей емкости следующим образом: сначала копируются файлы первого сегмента справочника. Как только том будет полный, система печатает: MOUNT NEXT OUTPUT VOLUME IN уст.;CONTINUE? Следует заменить выходной том следующим, подать Y (или любую строку, начинающуюся с Y) и возврат каретки. Система начинает копирование на установленный том с первого не скопированного файла сегмента. Если подать N (или любую строку, начинающуюся с N) или два раза SU/C, копирование прерывается и монитор печатает точку (.). Любой другой от-

1	2
/NEWFILES /POSITION:N	вет приводит к повторной печати сообщения. Система продолжает копировать файлы 1-го сегмента справочника до полного заполнения выходного тома или пока не будут скопированы все файлы сегмента. Когда все файлы 1-го сегмента будут скопированы, будут копироваться файлы следующего сегмента. Копирование будет выполняться таким образом до тех пор, пока все указанные файлы копируются. /MULTIVOLUME недопустим при копировании с ленты Копирует файлы, имеющие текущую дату создания Копирует файлы на или с магнитной ленты; в строке команды следует за спецификацией выходного файла. Переключатель позволяет управлять работой с лентой: перематывать ленту и выполнять операцию с указанной точки. По умолчанию $N=0$. Для операций копирования с ленты /POSITION:N устанавливает следующие действия: $N=0$ лента перематывается, и система ищет указанный файл. Если указать больше одного файла, лента перематывается перед каждым поиском. Если спецификация файла содержит конструкцию * и %, лента перематывается только один раз, и система копирует все соответствующие файлы. ----- $N>0$ система ищет на томе файл с -----

1	2
	порядковым номером N. Если имя найденного файла совпадает с указанным, система копирует его. В противном случае печатается сообщение об ошибке. Если спецификация файла содержит конструкцию * и % система ищет файл с порядковым номером N, а затем начинается поиск соответствующих файлов. N=-1 система начинает поиск с текущей позиции ленты. Если текущая позиция не является началом ленты, возможно, что указанный файл не будет найден, хотя на томе он может быть. Для операций копирования на ленту /POSITION:N устанавливает следующие действия: N=0 лента перематывается перед копированием каждого файла. Если на томе уже хранится файл с таким же именем и типом, на ТТ: печатается предупреждающее сообщение и копирование не выполняется. N>0 система ищет на томе файл с порядковым номером N или логический конец ленты — что встретится первым. Затем на том записывается указанный файл. Если указать более одного файла или спецификация файла содержит конструкцию * и %, лента не перематывается перед записью каждого файла, и система не проверяет наличие на томе файла с указанным именем.

1	2
/SETDATE[дата]	Присваивает указанную дату всем файлам, которые система копирует. Если аргумент дата не указан, используется текущая системная дата. Обычно, система сохраняет дату создания файла, т. к. система всегда использует текущую дату при копировании файлов на магнитную ленту и при копировании с MT: — дату создания файлов, то этот переключатель недопустим для операций с магнитной лентой
/SINCE[дата]	Копирует все файлы, созданные в течение или после указанной даты
/SLOWLY	Копирует файлы по одному блоку в один прием. Рекомендуется использовать этот переключатель, если предыдущая операция копирования не выполнялась из-за ошибки чтения или записи. Переключатель повышает вероятность безошибочного выполнения операции
/START[:N]	Используется с /DEVICE и с /END:N для определения номера начального блока. Аргумент, N, представляет собой десятичный номер блока. Переключатель /START:N следует или за именем входного устройства/спецификацией входного файла, чтобы указать начальный блок копируемого тома или за спецификацией выходного файла/именем выходного устройства, чтобы указать начальный блок выходного тома. Если не указан аргумент N, система предполагает первый блок тома (нулевой)

1	2
/SYSTEM	Копирует системные файлы. Без этого переключателя системные файлы исключаются из всех операций
/VERIFY	Проверяет, соответствует ли выходной файл (том) входному файлу (тому), если операция копирования выполнялась между устройствами справочной структуры с произвольным доступом. Если файлы (тома) различны, печатается сообщение об ошибке. Недопустим с переключателями /ASCII и /BINARY
/WAIT	Используется для копирования с одного тома на другой, если конфигурация вычислительной системы содержит один привод или пользователь хочет использовать один привод, или если конфигурация вычислительной системы содержит два привода, но системный том не является ни входным ни выходным томом. С переключателем /WAIT допустим переключатель /DEVICE, когда входной и выходной том являются томами устройств разного типа. Использование /WAIT требует присутствия на системном томе файлов PIP.SAV и DUP.SAV. При выполнении копирования с использованием /WAIT, система печатает опрашивающие сообщения. Ответы могут быть: — Y или любая строка, начинающаяся с Y, и возврат каретки для продолжения операции; — N или любая строка, начинающаяся с N, и возврат каретки или два СУ/С для прекращения выполнения операции печати монитором точки (.).

1	2
	1) Если в наличии один привод, то после вывода полной строки команды: 1. Система инициирует выполнение команды, но затем приостанавливается и печатает: MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя устройства, куда надо установить входной том. Пользователь должен установить входной том и напечатать Y и возврат каретки. 2. Система продолжает операцию копирования и печатает следующее сообщение на терминале: MOUNT OUTPUT VOLUME IN <УСТ>;CONTINUE? после этого сообщения следует: — удалить с устройства входной том — установить выходной том — напечатать Y и возврат каретки. 3. В зависимости от размера файла(ов) система может повторить шаги 1 и 2 несколько раз для завершения передачи файла(ов). После чего система печатает: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? после этого сообщения необходимо: — удалить с устройства выходной том — установить системный том — напечатать Y и возврат каретки. Система печатает точку — операция завершена. 2) Если имеется два привода, то после ввода полной строки команды:

1	2
	1. Система печатает: MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? следует установить входной том на первый привод и напечатать Y и возврат каретки. 2. Затем система печатает: MOUNT OUTPUT VOLUME IN <УСТ>;CONTINUE? следует установить выходной том на другой привод и напечатать Y и возврат каретки. 3. В отличие от копирования с одним приводом, шаги 1 и 2 не повторяются. После завершения передачи файла(ов) печатается MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? следует: — удалить с устройства входной (выходной) том — установить системный том — напечатать Y и возврат каретки. Система печатает точку — операция завершена.

Если используется /DEVICE, в качестве спецификации входного (выходного) файла указывается имя тома.

В строке команды можно указать только одно имя выходного файла.

Спецификации входных файлов в строке команды разделяются запятыми. Знак плюс (+) между спецификациями используется для объединения набора входных файлов в один. Действие знака плюс аналогично действию переключателя /CONCATENATE.

Пример:

.COPY A.MAC+B.MAC C.MAC

по данной команде файлы А.МАС и В.МАС объединяются и копируются под именем С.МАС на DK:.

Допускается использовать конструкцию * и % в спецификации входного (выходного) файла. Но нельзя использовать встроенную конструкцию в спецификации выходного файла. Если в спецификации входного файла используется конструкция *, то соответственно имя или тип выходного файла также должны быть обозначены через * (для всех операций, кроме /CONCATENATE).

Команда COPY может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается:

FROM?

TO?

Пользователь должен ввести с терминала спецификацию входного (выходного) файла и возврат каретки.

Некоторые переключатели используют аргумент дата. Синтаксис, определяющий аргумент дата, следующий:

[:дд][:ммм][:гг] ,

где дд — день (десятичное число от 1 до 31);

ммм — месяц (первые три буквы названия месяца);

гг — год (десятичное число от 82 до 99).

Значение даты по умолчанию соответствует текущей системной дате. Если опущена любая часть даты (дд, ммм, гг), система использует соответствующую часть текущей системной даты. Например, если указать только :82 и текущая системная дата 4 MAY 1983, система использует 4:MAY:82. Если текущая системная дата не установлена, она считается нулевой.

Примеры:

1. .COPY/ASCII DK0:MATR.MAC DK1:TEST.MAC/ALLOCATE:50

по этой команде копируется файл MATR.MAC с DK0: на DK1: под именем TEST.MAC (предварительно резервируется 50 блоков на DK1: для выходного файла).

2. .COPY/BINARY ANALIZ.OBJ DK1:*.*

По этой команде копируется файл двоичного формата с DK: на DK1:.

3. .COPY/DEVICE RK0:/START:0/END:500 RK1:ADAM.MAC/START:501

по этой команде копируются блоки с 0 по 500 с тома RK0: на том RK1: с блока 501 в виде файла ADAM.MAC.

4. .COPY/DEVICE/FILES DX0: DX1:FLOPPY.BAC
по этой команде том DX0: копируется на DX1: в виде файла FLOPPY.BAC.

5. .COPY/BOOT:DX DY:FMONSJ.SYS DY1:
по этой команде на том DY1: копируется загрузчик системы для устройства DX: .

6. .COPY/BEFORE:4:FEB:83 *.MAC DX0:*.MAC
По данной команде с устройства DK: файлы с типом .MAC, созданные до 4 февраля 1983 года, копируются на DX0:.

7. .COPY/DELETE JSPROG.SAV DX1:JSPROG.SAV
по данной команде файл JSPROG.SAV копируется на DX1: и затем удаляется с DK: .

8. .COPY/INFORMATION DY0: (FIL1,FIL2,FIL3).TXT DY1:*. *
?PIP—I—FILE NOT FOUND DY0:FIL2.TXT
в данном примере файлы FIL1.TXT и FIL3.TXT копируются на DY1:, файл FIL2.TXT не найден.

9. .COPY/MULTIVOLUME DK:*. * DX0:
(список скопированных файлов)
MOUNT NEXT OUTPUT VOLUME IN DX0: ;CONTINUE?Y
(список скопированных файлов)
MOUNT NEXT OUTPUT VOLUME IN DX0: ;CONTINUE?Y
по данной команде все файлы диска DK: копируются на несколько дисков.

4.10. Команда CREATE. По команде CREATE создается или расширяется файл с указанным именем, расположением и размером на указанном томе. Формат команды:

CR[EATE] спф[/прк]

где спф — спецификация файла;

/прк — один или несколько переключателей. Все переключатели команды следуют за спецификацией файла.

Таблица 12

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ CREATE

Переключатель	Функция
1	2
/ALLOCATE:N	Резервирует область памяти на томе для создаваемого файла. Аргумент, N, указывает число блоков области па-

Примеры:

1. .DIRECTORY/FULL DX0:

```
05—SEP—83
MYPROG MAC 36P 19—JUL—83 TM .MAC 25 27—JUL—83
VTMAC .MAC 7 19—JUL—83 SYSMAC .MAC 41 19—JUL—83
<UNUSED> 25 MONSJ .SYS 67 19—JUL—83
TT .SYS 2 19—JUL—83 DX .SYS 3 19—JUL—83
LELA .LBM 1 05—JUL—83 BUILD .MAC 80 19—JUL—83
<UNUSED> 199
9 FILES, 262 BLOCKS
224 FREE BLOCKS
```

по следующей команде CREATE расширяется файл BUILD.MAC на 20 блоков:

.CREATE DX0:BUILD.MAC/EXTENSION:20

2. В следующем примере показана процедура создания файла по команде CREATE. Используется также команда DIRECTORY/DELETED для определения расположения всех удаленных файлов на DX0:.

.DIRECTORY/DELETED DX0:

05-JUL-83

```
SWAP.SYS 25 19-JUL-83 117 EMPTY.FIL 179 31-AUG-83 315
0 FILES 0 BLOCKS
```

204 FREE BLOCKS

.CREATE DX0:SWAP.SYS/START:117/ALLOCATE:25

по команде CREATE восстанавливается файл SWAP.SYS, начиная с блока 117 и резервируется под файл 25 блоков.

4.11. Команда D. По команде D указанные значения записываются в память, начиная с указанной ячейки.

Формат команды: D адр=влч[,...влч]

где адр — адрес (восьмеричный), который при сложении со значением базы перемещения образует фактический адрес (абсолютный), куда будут записываться новые значения;

влч — новое содержимое адреса.

Если не указать влч, система запишет в указанную ячейку 0. Если указать несколько значений влч через запятую, система запишет эти значения в последовательные ячейки памяти, начиная с адреса, указанного в строке команды.

Команда D допускает использование адреса байта и адреса слова в строке команды, но команда всегда выполняется, как если бы был указан адрес слова (если указать нечетный адрес, система уменьшает его на единицу, т. е. делает четным). По команде D все значения влч запоминаются как слова.

Две или большее количество смежных запятых в строке команды используются для записи нулей в указанные ячейки памяти. Нельзя указывать адрес вне области фонового задания.

Примеры:

1. D 300=,,,

по этой команде в ячейки 300—306 записываются нули.

2. B 1000

.D 500=2503,22

по данным командам адрес базы перемещения устанавливается на 1000 и в ячейку 1500 записывается 2503, в ячейку 1502 — 22.

4.12. Команда DATE используется для установки или проверки текущей даты системы. Формат команды:

DATE[TE][дд—мм—гг]

где дд — день (десятичное число от 1 до 31);

мм — первые три символа имени месяца (на английском языке);

гг — год (десятичное число от 82 до 99).

Вводить дату следует после первичной загрузки системы. Система использует эту дату для вновь созданных файлов, для файлов, которые копируются на магнитную ленту, и для файлов листинга.

Монитор FB и ХМ автоматически изменяет дату каждые сутки в полночь. Монитор SJ увеличивает дату при наличии поддержки монитором таймера. Для правильного изменения даты в конце месяца монитор должен иметь соответствующую поддержку.

Для распечатки текущей даты системы, надо подать с терминала команду DATE без аргумента.

Пример:

.DATE

18—SEP—82

4.13. Команда DEASSIGN отменяет логическое имя устройства.

Формат команды: DEA[SSIGN] лимуст

где лимуст — логическое имя устройства.

Если в строке команды указывается логическое имя, которое не было присвоено устройству, система печатает сообщение об ошибке.

Примеры:

1. DEASSIGN INP:

по данной команде отменяется логическое имя устройства INP: .

2. DEASSIGN

по данной команде отменяются все логические имена физических устройств (кроме DK: и SY:).

4.14. Команда DELETE. По команде DELETE удаляются указанные файлы с диска. Формат команды:

DEL[ETE][/прк] спф[...спф]

где /прк — один или несколько переключателей;

спф — спецификация удаляемого файла.

Таблица 13

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ DELETE

Переключатель	Функция
1	2
/BEFORE[дата]	Удаляет файлы, созданные до указанной даты
/DATA[дата]	Удаляет файлы с указанной датой создания
/ENTRY	Удаляет задание из очереди. Используется, когда QUEUE выполняется как основное или системное задание (более подробно о программе QUEUE см. в документе [4]). В командной строке с /ENTRY указывается имя задания
/EXCLUDE	Удаляет все файлы на томе кроме указанных в строке команды
/INFORMATION	Используется, если пользователь не уверен, есть ли на томе указанные в строке команды файлы. Если указанный файл не найден — печатается информационное сообщение об ошибке, но выполнение операции не прекращается — удаляются найденные файлы

1	2
/LOG	Печатает на терминале список файлов, которые были удалены текущей командой.
/NEWFILES	Если указан /LOG, система не опрашивает пользователя о включении каждого файла в операцию
/QUERY	Удаляет только файлы, имеющие текущую дату системы
----	Опрашивает пользователя о включении в указанную операцию каждого файла.
	Этот переключатель является переключателем по умолчанию, если спецификация файла содержит конструкцию * и %. Пользователь должен напечатать на терминале:
	— Y и возврат каретки, если файл надо включить в операцию удаления;
	— NO и возврат каретки, если файл исключается из операции удаления.
	Если используется /LOG, то действие по умолчанию переключателя /QUERY отменяется. Необходимо с /LOG указать /QUERY, чтобы сохранить действие этого переключателя
/NOQUERY	Запрещает опрос пользователя о включении указанного файла в операцию
/SINCE[дата]	Удаляет все файлы, созданные в течение или после указанной даты
/SYSTEM	Удаляет системные файлы (.SYS).
	Без этого переключателя системные файлы исключаются из операции удаления и печатается сообщение, напоминающее об этом
/WAIT	Используется, если конфигурация вычислительной системы содержит один привод или пользователь хочет ис-

1	2
	пользовать один привод. После подачи строки команды с переключателем /WAIT, система инициирует выполнение команды, но затем приостанавливается и печатает: MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя тома устройства. Следует установить том, содержащий удаляемый файл, и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Печать N (или любой строки, начинающейся с N) или два последовательных нажатия СУ/С, прерывают операцию и управление возвращается монитору. Любой другой ответ вызывает повторную печать сообщений. После завершения операции система печатает сообщение: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? Следует установить системный том и подать Y (или любую строку, начинающуюся с Y) и возврат каретки. Если подан другой ответ, система повторно печатает сообщение

В строке команды можно указать на более шести спецификаций файлов. Спецификации файлов в строке команд разделяются запятыми.

Команда DELETE задается с терминала в виде полной строки или краткой. Если задана краткая строка команды, на терминале печатается:

FILES?

Пользователь должен напечатать на терминале спецификацию удаляемого файла и возврат каретки. Если пользова-

тель случайно удалил файл, то, действуя немедленно, можно его восстановить. Процедура восстановления описана в документе [4].

Для удаления защищенного файла перед командой DELETE необходимо подать команду UNPROTECT, COPY/NOPROTECTION или RENAME/NOPROTECTION (п. 4.53, п. 4.9, п. 4.39). Защита файла обозначается буквой «Р», которая следует за размером данного файла в записи справочника.

Примеры:

1. DELETE/EXCLUD DX0:*.SAV
?SYS—W—NO .SYS ACTION
FILES DELETED:
DX0:ABC.OLD ?Y
DX0:AAF.OLD ?Y
DX0:COMB. ?Y

по данной команде с DX0: удаляются все файлы, кроме файлов с типом .SAV и системных файлов.

2. DELETE/NEWFILES DX1:*.BAK
FILES DELETED:
DX1:MERGE.BAK ?Y

По данной команде удаляются файлы с типом .BAK, созданные сегодня.

3. DELETE/ENTRY MILLER

по данной команде задание MILLER удаляется из очереди. Если QUEUE печатает задание, когда подается команда на удаление этого задания, то печатающийся файл задания печатается до конца и лишь после этого задание удаляется.

4. DELETE/WAIT RK0:FILE.MAC
MOUNT INPUT VOLUME IN RK0;;CONTINUE?Y
RK0:FILE.MAC?Y
MOUNT SYSTEM VOLUME IN RK0;;CONTINUE?Y

по данной команде файл FILE.MAC удаляется с диска RK0:.

5. DELETE/LOG/BEFORE:20:FEB:84 DY0:*.SAV
FILES DELETED:
DY0:A.SAV
DY0:B.SAV

по указанной команде файлы с типом .SAV удаляются с тома DY0:, затем печатается список удаленных файлов.

6. DELETE/INFORMATION DY0:(FIL1, FIL2, FIL3).TXT
?PIP—I—FILE NOT FOUND DY0:FIL2.TXT

по данной команде файлы FIL1.TXT и FIL3.TXT удаляются с тома DY0.; печатается также информационное сообщение — файл FIL2.TXT не найден.

4.15 Команда DIBOL. По команде DIBOL вызывается ДИБОЛ для трансляции указанных файлов. Формат команды:

DIB[OL]/прк] спф[,...]

где /прк — один или несколько переключателей;

спф — спецификация файла, который будет транслироваться.

Таблица 14

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ DIBOL

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с /LIST или /OBJECT для резервирования области памяти на томе для выходного файла. N — число блоков области памяти от 1 до 65535 для размещения файла. Если N = -1, на томе для выходного файла отводится наибольшая свободная область памяти
/ALPHABETIZE	Располагает по латинскому алфавиту таблицу символов и таблицу меток
/BUFFERING	Указывает, что будет использоваться простая буферизация при вводе — выводе. Обычно используется двойная буферизация
/CROSSREFERENCE	Создает поле таблицы перекрестных ссылок в листинге (максимум 4 секции), т. к. по умолчанию листинг не печатается, надо указать /LIST для получения распечатки таблицы перекрестных ссылок
/LINENUMBERS	Разрешает вывод номеров внутренней последовательности в объектный код
/NOLINENUMBERS	Запрещает вывод номеров внутренней последовательности в объектный код

1	2
/LIST[:спф]	Используется для получения листинга при трансляции; спф — спецификация файла листинга. Если /LIST следует за именем команды, то листинг получается на LP::; если за спецификацией файла, то файл листинга записывается на DK:
/OBJECT[:спф]	Используется для получения объектной программы; спф — спецификация объектного файла
/NOOBJECT	Запрещает создание объектного файла
/ONDEBUG	Включает таблицу символов в объектный файл
/PAGE:N	Определяет количество строк в странице листинга; N — десятичное число от 1 до 32768. По умолчанию N=66
/TABLES	Включает в листинг таблицу символов и таблицу имен, используется только с /LIST
/WARNINGS	Разрешает вывод предупреждающих сообщений об ошибке
/NOWARNINGS	Запрещает вывод предупреждающих сообщений об ошибке

По умолчанию тип файла листинга — .LST, тип объектного файла — .OBJ, исходного файла — .DBL.

Для независимой трансляции нескольких исходных файлов спецификации файлов разделяются запятыми. Знак плюс (+) между спецификациями используется для трансляции нескольких исходных файлов в один объектный файл с именем первого входного файла и типом .OBJ. Однако, пользователь может указать любое имя для объектного файла по /OBJECT[:спф].

Переключатели команды /DIBOL (/LIST и /OBJECT) имеют различное назначение в зависимости от занимаемого места в строке команды. Переключатели, следующие за именем команды, действуют на всю строку. Если переключатели

следуют за спецификацией файла, то они действуют только на этот файл (или группу файлов, соединенных знаками плюс).

Команда DIBOL может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается:

FILES?

Пользователь должен ввести с терминала спецификацию исходного файла и возврат каретки.

Примеры:

1. DIBOL/LIST MYPROG

по данной команде транслируется файл MYPROG.DBL, создается файл DK.MYPROG.OBJ и выводится листинг на LP:.

2. DIBOL MYPROG/LIST

по данной команде транслируется файл MYPROG.DBL и создаются файлы DK:MYPROG.OBJ и DK:MYPROG.LST.

3. DIBOL/LIST:TT: A

по данной команде листинг будет печататься на TT:.

4. DIBOL/LIST:RK3: A

по данной команде создается RK3:A.LST и DK:A.OBJ.

5. DIBOL/LIST:FILE1.OUT A+B/OBJECT

по данной команде файлы A и B транслируются вместе и создаются файлы — DK:FILE1.OUT (файл листинга) и DK:B.OBJ.

6. DIBOL A+B, D+C/NOBJECT/LIST

по данной команде транслируются:

— A.DBL и B.DBL вместе, создавая A.OBJ

— D.DBL и C. DBL вместе, создавая C.LST

4.16. Команда DIFFERENCES. По команде DIFFERENCES происходит сравнение двух файлов и различия между ними записываются в виде файла на указанное устройство или выводятся на печать. Формат команды:

DIF[FERENCES] [/прк] спф1, спф2

где /прк — один или несколько переключателей;

спф1 и спф2 — спецификации сравниваемых файлов.

Таблица 15

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ DIFFERENCES

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с переключателем /OUTPUT, /SLP и /SIPP для резервирования области памяти на томе для файла листинга. Аргумент, N, представляет собой число блоков области памяти для размещения файла; N — целое число от 1 до 65535 (если N = —1, на томе для файла листинга отводится наибольшая свободная область памяти)
/ALWAYS	Используется с /BINARY,/OUTPUT:спф или /SIPP:спф для создания файла листинга независимо от того, существуют ли различия между сравниваемыми файлами. Если /ALWAYS в строке команды следует за спецификацией выходного файла, то он действует на этот файл, если в конце строки, то /ALWAYS действует на все выходные файлы
/AUDITTRAIL	Используется с /SLP для указания признака модификации, признак модификации — строка символов, которая появляется справа перед измененной строкой в файле, создаваемом по /SLP (описание этого переключателя смотри ниже). По умолчанию программа SLP использует признак модификации: ;**NEW** Программа SLP подробно описана в документе [4]. После ввода /AUDIT-TRAIL, система печатает на терминале: AUDIT TRAIL? Пользователь должен ввести строку, не более 12 любых символов КОИ-7

1	2
/BINARY	(кроме /), которая будет использоваться как признак модификации Сравнивает двоичные файлы (типа: .LDA, REL, .SAV и .OBJ) и перечисляет различия между ними. По этому переключателю система печатает на терминале отличия двух двоичных файлов в следующем формате: <pre>BBBBBB 000/FFFFFF SSSSSS XXXXXX,</pre> BBBBBB — номер блока (восьмеричный), содержащего отличия; 000 — смещение (восьмеричное) внутри блока BBBBBB; FFFFFF — значение в первом сравниваемом файле; SSSSSS — значение во втором сравниваемом файле; XXXXXX — результат логической операции «исключающее или» для значений FFFFFFF и SSSSSS
/BLANKLINES	Включает пустые строки в сравнение файлов. Обычно система не рассматривает пустые строки при сравнении
/BYTES	Используется с /BINARY для печати отличий по байтам
/CHANGEVAR	Создает выходной файл, содержащий второй сравниваемый файл, в котором измененные строки помечены вертикальной чертой
/COMMENTS	Включает в сравнение файлов комментарии сравниваемых файлов
/NOCOMMENTS	Исключает из сравнения файлов комментарии. Переключатель рекомендуется использовать для сравнения двух

1	2
/DEVICE	исходных программ разного формата. Используется с /BINARY для сравнения томов, начиная с 0-го блока. При сравнении томов разной емкости, система выдает сообщение: ?BINCOM—W—DEVICE IS LONGER DEV.: при этом, содержимое тома большей емкости сравнивается до блока, номер которого равен номеру последнего блока тома меньшей емкости, включительно. Если отличия найдены, система выдает сообщение: ?BINCOM—W—DEVICES ARE DIFFERENT
/END[:N]	Используется с /BINARY и указывает номер последнего сравниваемого блока в файле; N — номер последнего блока (восьмеричный). Если N не указан, система предполагает последний блок файла
/FORMFEED	Включает в листинг символ перевод формата. Обычно система включает в сравнение перевод формата, но из листинга его исключает
/MATCH:N	Указывает, по сколько строк будет происходить сравнение файлов. Аргумент, N, — целое число от 1 до 200. По умолчанию N=3. Недопустим с /BINARY
* /OUTPUT :спф	Определяет имя тома и спецификацию файла листинга. По умолчанию тип файла листинга — .DIF
* /PRINTER /QUIET	Печатает листинг различий на LP: Используется с /BINARY для запрещения печати отличий на терминале, система печатает только ?BINCOM—W—FILES ARE DIFFERENT, если файлы различны

1	2
/SIPP:спф	Используется с /BINARY для создания файла, который можно использовать как входной файл для программы SIPP. Программа SIPP описана подробно в документе [4]. спф — спецификация создаваемого файла. Файл, создаваемый по /SIPP, посредством программы SIPP модифицирует второй сравниваемый файл до соответствия с первым сравниваемым файлом. Если спф не указана, то. файл выводится на ТТ:
/SLP[:спф]	Используется для создания файла, который посредством программы SLP модифицирует второй сравниваемый файл до соответствия с первым сравниваемым файлом. Если с /SLP не указана спецификация создаваемого файла (:спф), файл, создаваемый по /SLP, печатается на терминале
<u>/SPACES</u>	Включает в сравнение пробелы и табуляции
/NOSPACES	Исключает пробелы и табуляции из сравнения файлов
/START[:N]	Используется с /BINARY для указания номера начального сравниваемого блока в файле, N — восьмеричный номер блока. Если N не указан, система предполагает первый блок файла
<u>/TERMINAL</u> *	Печатает листинг различий на ТТ
<u>/TRIM</u>	Используется с /SLP для исключения из сравнения табуляций и пробелов, которые появляются в конце строк
/NOTRIM	Используется с /SLP для включения в сравнение табуляций и пробелов, которые появляются в конце строк

По умолчанию приняты следующие типы файлов: .MAC — для входного файла, .DIF — для выходного файла.

Команда DIFFERENCES задается с терминала в виде полной строки или краткой. Если задана краткая строка команды, на терминале печатается:

FILE 1?

FILE 2?

Пользователь должен ввести имя первого сравниваемого файла и возврат каретки, а затем — второго.

Спецификации сравниваемых файлов могут содержать конструкцию звездочки (*) или процента (%). Тогда перед печатью различий, система печатает спецификации сравниваемых файлов.

1. Если надо сравнить один файл с несколькими другими файлами, то конструкция звездочки (*) или процента (%) указывается в одной спецификации сравниваемых файлов.

2. Если надо сравнить несколько разных пар файлов, конструкция звездочки (*) или процента (%) используется в обеих спецификации сравниваемых файлов.

Примеры:

1. .DIF/SLP/OUTPUT:PATCH.COM MKLIB.BAK, MKLIB.MAC

по данной команде сравниваются файлы MKLIB.BAK и MKLIB.MAC и создается файл PATCH.COM.

2. .DIF/BINARY/SIPP:PATCH.COM D1.BAK, D1.SAV

по данной команде сравниваются двоичные файлы D1.BAK и D1.SAV и создается файл PATCH.COM.

3. .DIF/MATCH:1/OUTPUT:T.DIF DY0:T1.MAC, DY1:T2.*

По данной команде файл DY0:T1.MAC сравнивается с файлами на DY1: любого типа, имя которых T2. Все найденные отличия выводятся на DK: файлом T.DIF.

4. .DIFFERENCES/BINARY DY0:PROG1.*, DY1:PROG2.*

По данной команде сравниваются файлы PROG1 тома DY0: и PROG2 тома DY1: одинаковых типов в следующем порядке: сначала система определяет тип файла PROG1 на томе DY0:, затем ищет на DY1: файл PROG2 с типом, что и файл PROG1. Если такой файл существует — сравнивает найденные файлы. Если — нет, сравнение не выполняется. Затем на DY0: система ищет файл PROG1 с другим типом. И весь поиск повторяется.

4.17. Команда DIRECTORY позволяет распечатать требуемую информацию о томе устройства, файле или группе файлов.

Формат команды:
 DIR[ECTORY] [/прк][спф[/BEGIN]]
 где /прк — один или несколько переключателей;
 спф — имя тома или спецификация файла или группы
 файлов.

Таблица 16

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ DIRECTORY

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с /OUTPUT для резервирования области памяти на устройстве для файла листинга. Аргумент, N, представляет собой число блоков области для размещения файла; N — целое число от 1 до 65535; N может быть равно также — 1 (в этом случае на томе для файла листинга отводится наибольшая свободная область памяти)
/ALPHABETIZE ***	Распечатывает справочник указанного тома устройства в алфавитном порядке по имени файла и его типу (аналогично /ORDER:NAME)
/BACKUP	Печатает справочную информацию об указанном томе, который создан по команде BACKUP (т. е. содержит часть информации тома большой емкости). Листинг справочника включает: системную дату и порядковый номер указанного тома в наборе томов, созданных по BACKUP; затем перечисляется весь набор
/BADBLOCKS	Печатает абсолютные номера блоков тома устройства, чтение которых приводит к аппаратной ошибке
/BEFORE[дата] **	Включает в листинг справочника только файлы созданные до указанной даты

1	2
/BEGIN	Распечатывает справочник указанного тома, начиная с указанного файла и включая все файлы, следующие после него в справочнике. Имена файлов печатаются в листинге в том порядке, в каком файлы хранятся на томе устройства. Переключатель в строке команды следует за спецификацией файла
/BLOCKS	Распечатывает справочник указанного тома, включая номера начальных блоков (десятичные числа) всех файлов устройства
/BRIEF	Распечатывает в пять колонок краткий справочник (только имена и типы файлов) указанного тома
/COLUMNS:N	Распечатывает справочник в N колонок; N — целое от 1 до 9. Значение по умолчанию для обычных справочников — 2 колонки, для кратких — 5 колонок
/DATE[дата] **	Включает в листинг справочника только файлы с указанной датой создания
/DELETED	Включает в листинг справочника файлы, которые были удалены, но справочная информация, которых не была разрушена. Такой листинг содержит имя файла, тип, размер, дату создания и номер начального блока (десятичные числа). После команды DIRECTORY/DELETED можно использовать команду CREATE или программу DUP для восстановления файла (см. п. 4.10)
/END:N	Используется с переключателями /START:N и /BADBLOCKS и указывает номер последнего сканируемого блока тома. Если N не указано, сис-

1	2
<u>/EXCLUDE</u>	тема сканирует до последнего блока тома включительно
<u>/FAST</u>	Распечатывает справочник тома устройства, исключая указанные файлы
<u>/FILES</u>	Действие переключателя /FAST эквивалентно действию переключателя /BRIEF
<u>/FREE</u>	Используется с /BADBLOCKS для печати имен файлов, содержащих плохие блоки
<u>/FULL</u>	Распечатывает справочник неиспользуемых областей и их размеры
<u>/NEWFILES</u> **	Распечатывает весь справочник, включая неиспользуемые области и их размеры в блоках (десятичные числа)
<u>/OCTAL</u>	Включает в листинг справочника только файлы, имеющие текущую дату создания
<u>/ORDER[:XXX]</u> ***	В листинге справочника указанного тома устройства размеры файлов в блоках (и номера начальных блоков, если используется /BLOCKS) распечатываются в восьмеричном виде
<u>/OUTPUT:спф</u> *	Распечатывает справочник указанного тома устройства в определенном порядке; XXX — параметр, от которого зависит порядок печати листинга (параметры, XXX, описаны в табл. 17)
<u>/POSITION</u>	Используется для вывода файла листинга с именем, указанным в спецификации файла (спф), на указанный том устройства. Обычно, листинг справочника печатается на терминале. Если не указан тип файла листинга, система присваивает .DIR Включает в листинг справочника магнитной ленты последовательные номера файлов

1	2
/PRINTER *	Распечатывает справочник тома устройства на построочно-печатающем устройстве, устройство вывода листинга справочника по умолчанию — терминал. Переключатель /PRINTER не использует программу QUEUE
/PROTECTION	Печатает справочную информацию только о файлах защищенных от удаления
/NOPROTECTION	Печатает справочную информацию только о файлах, которые не защищены от удаления
/REVERSE	Распечатывает справочник в обратном порядке по отношению к упорядоченному, определенному переключателем /ALPHABETIZE, /ORDER или /SORT
/SINCE[дата] **	Включает в листинг справочника файлы, созданные в течение или после указанной даты
/SORT[:XXX] ***	Действие переключателя /SORT аналогично действию переключателя ORDER
/START[:N]	Используется с /BADBLOCKS и указывает номер первого сканируемого блока тома; N — десятичный номер блока. Если N не указано, система сканирует с первого блока тома. Если не указан /END:N, система сканирует том до последнего блока включительно
/SUMMARY	Распечатывает информационный справочник указанного тома устройства, т. е. количество файлов в каждом сегменте и число используемых сегментов
/TERMINAL *	Печатает справочную информацию на терминале

1	2
/VOLUMEID[:ONLY]	Печатает идентификацию тома указанного устройства, имя владельца и листинг справочника. Если указан аргумент ONLY, система печатает только идентификацию тома и имя владельца
/WAIT	Используется с /BADBLOCKS, если конфигурация вычислительной системы содержит один привод дисков. После подачи переключателя /WAIT, система инициирует указанную операцию, затем приостанавливается и печатает:
	MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя тома, справочную информацию которого надо получить. Пользователь должен установить том и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Если подать N (или любую строку, начинающуюся с N) и возврат каретки или два раза CY/C, операция прерывается и управление возвращается монитору. Любой другой ответ вызывает повторную печать сообщений. После завершения операции на терминале печатается:
	MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? Пользователь должен заменить на приводе входной том на системный и напечатать Y и возврат каретки. Для использования /WAIT, на системном томе должны присутствовать программы DUP

Таблица 17

КОДЫ ОРГАНИЗАЦИИ СПРАВОЧНИКА

Параметр (XXX)	Функция
1	2
DATE	Организует справочник по датам создания файлов. Файлы, имеющие одну и ту же дату создания, сортируются в порядке латинского алфавита по имени и типу файла
NAME	Организует справочник в порядке латинского алфавита по именам файлов. Файлы, имеющие одно и то же имя, сортируются по типу файла. (Имеет тот же результат, что и переключатель /ALPHABETIZE)
POSITION	Организует справочник файлов в порядке их расположения на устройстве. В указанном порядке печатается справочник также по /ORDER без параметра
SIZE	Производит организацию справочника, основанную на длине файлов в блоках. Файлы, имеющие одну и ту же длину, будут расположены в порядке латинского алфавита по имени и типу файлов
TYPE	Организует справочник в порядке латинского алфавита по типу файлов. Файлы с одним и тем же типом сортируются по имени в порядке латинского алфавита

В строке команды можно указать не более шести спецификаций файлов.

Спецификации файлов в строке команд разделяются запятыми. Можно получить также справочную информацию о группе файлов путем использования конструкции * и % в спецификации файла.

Обычно по команде DIRECTORY листинг справочника печатается на терминале в две колонки. Читать такие листинги необходимо слева направо, всю строку сразу. Однако, листинги справочника, полученные с помощью переключателей /ALPHABETIZE, /ORDER и /SORT, исключаются из этого правила. Эти листинги необходимо читать следующим образом: сначала левую колонку сверху вниз, затем аналогично правую.

Если в строке команды не указана спецификация файла или имя тома, распечатывается справочник устройства DK:. Если указано только имя тома, печатается справочная информация всех файлов указанного тома. Если в строке команды указано имя файла, печатается информация только об этом файле.

Некоторые переключатели допускают в качестве своего аргумента дату. Синтаксис, определяющий дату, следующий:

[:дд][:мм][:гг]

где дд — представляет собой день (десятичное число от 1 до 31);

мм — месяц (первые три буквы названия месяца);

гг — год (десятичное число от 82 до 99).

Значение даты по умолчанию соответствует текущей системной дате. Если опущена любая часть даты (дд, мм, гг), система использует соответствующую часть текущей системной даты. Например, если указать только :82 и текущая системная дата 4 MAY 1984, система использует дату 4:MAY:82. Если текущая системная дата не установлена, она считается нулевой.

Буква «Р» за размером файла в записи справочника для данного файла указывает, что файл защищен от удаления.

Примеры:

1. DIRECTORY/BADBLOCKS DX1:

?DUP—I—NO BAD BLOCKS FOUND

по данной команде печатаются абсолютные номера плохих блоков. Если том устройства не имеет плохих блоков, печатается соответствующее информационное сообщение.

2. DIRECTORY DX0:VKMAC.MAC/BEGIN

по данной команде печатается справочник тома DX0:, начиная с файла VKMAC.MAC.

3. DIRECTORY/BLOCKS DX0:

По данной команде печатается справочник тома DX0:, включая номера начальных блоков файлов.

4. DIRECTORY/BRIEF DX0:

по данной команде печатается в пять колонок краткий справочник тома DX0: (только имена и типы файлов).

5. .DIRECTORY/DATE:8:APR:83 DX0:

по данной команде в справочник включаются файлы тома DX0:, созданные 8 апреля 1983 года.

6. .DIRECTORY/DELETED DK1:

по данной команде печатается справочная информация о файлах, которые были удалены с тома DK1:

7. .DIRECTORY/EXCLUDE DX0:(*.SAV,*.SYS)

по данной команде печатается справочная информация о всех файлах тома DX0:, кроме файлов с типом .SAV и .SYS.

8. .DIRECTORY/FREE

по данной команде печатается справочная информация о неиспользуемых областях тома DK: и их размерах.

9. .DIRECTORY/FULL DK0:

по данной команде печатается весь справочник, включая неиспользуемые области и их размеры в блоках (десятичные числа).

10. .DIRECTORY/ORDER:DATE DX0:

по данной команде печатается справочник тома DX0:, организованный по датам создания файла.

11. .DIRECTORY/ORDER:DATE/REVERSE DX0:

по данной команде печатается справочник тома DX0: в обратном хронологическом порядке.

12. .DIRECTORY/WAIT/BADBLOCKS DX0:

MOUNT INPUT VOLUME IN DX0: ;CONTINUE?Y

?DUP—I—NO BAD BLOCKS DETECTED DX0:

MOUNT SYSTEM VOLUMED IN DX0: ;CONTINUE?Y

по данной команде выполняется сканирование диска DX0: на плохие блоки, если конфигурация вычислительной системы имеет лишь один привод.

13. .DIRECTORY/BACKUP DY0:

23—JAN—83

VOLUME 3 OF 4

VOLUME

FILENAME BLOCKS DATE

V1 CAFIL.BUP 980 23—JAN—84

V2 CAFIL.BUP 980 23—JAN—84

V3 CAFIL.BUP 980 23—JAN—84

V4 CAFIL.BUP 400 23—JAN—84

1 FILE, 980 BLOCKS

0 FREE BLOCKS

14. .DIRECTORY/BACKUP MT1:

23—JAN—83

VOLUME

FILENAME BLOCKS DATE

V1

DL1.BUP

20450

23—JAN—83

4.18. Команда DISMOUNT отменяет соответствие между файлом и логическим диском. Формат команды:

DIS[MOUNT] логдиск

где логдиск — имя логического диска (LDN:, N — целое число от 0 до 7).

Если логическому диску было назначено логическое имя устройства, то отменить соответствие логического диска можно, указывая в качестве логдиск назначенное логическое имя устройства.

Команда DISMOUNT задается с терминала в виде полной строки или краткой. Если задана краткая строка команды, на терминале печатается:

DEVICE?

Следует напечатать имя логического диска или логическое имя устройства.

Примеры:

1. .DISMOUNT LD3:

по данной команде отменяется соответствие логического диска (LD3:) файлу.

2. .ASSIGN LD3: DAT

.DISMOUNT DAT

данный пример имеет тот же результат, что и первый.

4.19. Команда DUMP предназначена для вывода на терминал или на построчно-печатающее устройство, а также для записи на периферийное ЗУ содержимого всего или части файла (тома) в восьмеричных кодах, а также символах КОИ-7 и RADIX-50. Команду DUMP удобно использовать для проверки справочников и файлов, содержащих двоичные данные. Формат команды:

DU[MP][/прк] спф

где спф — спецификация файла или имя тома, содержимое которого надо вывести или записать на периферийное устройство;

/прк — один или несколько переключателей.

Таблица 18

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ DUMP

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с переключателем /OUTPUT для резервирования области памяти на томе для файла листинга. Аргумент, N, представляет собой число блоков области для размещения файла; N — целое число от 1 до 65535; N может быть равно также —1 (в этом случае на томе для файла листинга отводится наибольшая свободная область памяти)
<u>/ASCII</u>	Печатает информацию в символах КОИ-7. Точка означает символ не в коде КОИ-7
/NOASCII	Запрещает печать символов КОИ-7
/BYTES	Печатает информацию по байтам
/END:N	Используется для определения номера конечного блока, предназначенного для вывода. Система распечатывает содержимое указанного тома устройства или файла с нулевого блока (кроме случая, когда используется /START) и до блока, номер которого указан с переключателем /END
/IGNORE	Используется для игнорирования ошибок, которые возникают при выполнении операции
/ONLY:N	Используется для печати N-ого блока
/OUTPUT :спф*	Используется для вывода файла листинга с именем, указанным в спецификации файла, на указанный том устройства. Обычно листинг печатается на построчно-печатающем устройстве. Если не указан тип файла в спе-

1	2
* <u>/PRINTER</u> /RAD50 /START:N * <u>/TERMINAL</u> <u>/WORDS</u> -----	циффикации файла листинга, система использует .DMP Используется для вывода листинга на построчно-печатающее устройство Печатает символы в кодах RADIX-50 Используется для определения номера начального блока, предназначенного для вывода. Система распечатывает содержимое тома или файла, начиная с N-ого блока и до последнего блока тома или файла (если не используется /END) Используется для вывода листинга на терминал Печатает информацию по словам

Если в строке команды не указан тип выходного файла, система использует .DMP.

Команда DUMP задается с терминала в виде полной строки или краткой. Если задана краткая строка команды, на терминале печатается:

DEVICE OR FILE?

Пользователь должен напечатать на терминале спецификацию файла или тома, содержимое которого надо вывести или записать на периферийное устройство, и возврат каретки.

Некоторые переключатели (/ONLY, /START и /END) используют в качестве аргумента (N) номер блока (восьмеричный). Номер первого блока тома или файла 0. Точка, следующая за номером блока, указывает, что этот номер — десятичное число.

Если выводится файл, указанный номер блока — номер относительно начала этого файла. Если выводятся данные тома, номера блоков являются абсолютными (физическими) номерами блоков на этом томе.

4.20. Команда E. По команде E на терминале распечатывается содержимое указанных ячеек памяти в восьмеричном коде.

Формат команды: **E** **адр1**{—**адр2**]
 где **адр1**(**адр2**) — адрес(восьмеричный), который в сумме с базой перемещения дает фактический адрес первой (последней) ячейки, содержимое которой надо проверить.

Если указать нечетный адрес в строке команды, система уменьшает адрес на единицу, т. е. делает его четным. Если в строке команды указать два адреса (**адр1** — **адр2**), система распечатывает содержимое ячеек, начиная с **адр1** до **адр2** включительно. Адрес **адр2** всегда должен быть больше **адр1**. Если в строке команды не указан адрес, система распечатывает содержимое ячейки 0.

Нельзя распечатать ячейку, расположенную вне фоновой области.

Примеры:

1. **E 1000**

по данной команде печатается содержимое ячейки 1000.

2. **B 1000**

E 1001—1005

по данным командам устанавливается база перемещения на 1000 и распечатывается содержимое ячеек 2000—2004.

4.21. Команда **EDIT**. По команде **EDIT** в оперативную память вызывается программа редактор текста.

Формат команды: **EDIT**[/**прк**] **спф**[/**ALLOCATE:N**]

где /**прк** — один или несколько переключателей;

спф — спецификация редактируемого файла.

Т а б л и ц а 19

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ EDIT

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с переключателем /OUTPUT или после спецификации файла для резервирования области памяти на томе для выходного файла. Аргумент, N, представляет собой число блоков области для размещения файла, N целое число от 1 до 65535; N может быть равно также —1 (в этом случае на томе для файла отво-

1	2
/CREATE * /EDIT /INSPECT * /K13 /OUTPUT:спф *	дится наибольшая свободная область памяти) Используется для создания нового файла можно также создать новый файл, используя команду редактора текста EW. Создание файла с помощью экранного редактора текста описано в документе [11] Вызывает редактор текста EDIT Открывает файл для чтения. По этому переключателю не создаются новые выходные файлы. Открыть входной файл можно также с помощью команды редактора текста ER Вызывает экранный редактор текста. Используется только при работе терминала 15ИЭ—00—013. Подробное описание экранного редактора текста см. в документе [11] Открывает файл для вывода, в который будет записан откорректированный текст, входной файл остается неизменным. Открыть файл для вывода можно также с помощью команды редактора текста EW

Команда EDIT задается с терминала в виде полной строки или краткой. Если задана краткая строка команды, на терминале печатается:

FILE?

пользователь должен напечатать на терминале спецификацию файла и возврат каретки. На терминале печатается звездочка (*), т. е. редактор текста находится в командном режиме, и пользователь может подавать команды редактору текста. Команды редактора текста описаны в документе [5].

Если в строке команды не указано ни одного переключателя, то на указанном файле выполняется команда редактора текста EW. Тогда редактор текста по окончании коррек-

тирования, если будет подана одна из следующих команд: EX, EF или EB, изменит спецификацию первоначального файла, присваивая ему тип .BAK.

Если нужного для редактирования файла нет на томе, печатается сообщение об ошибке. Если спецификация редактируемого файла совпадает со спецификацией выходного файла, печатается предупреждающее сообщение. После печати сообщения об ошибке можно подать любую другую команду редактору текста или подать команду СУ/С, которая должна заканчиваться двумя СУ/С, т. е. управление возвращается монитору.

Примеры:

1. .EDIT/CREATE NEWFILE.TXT

по данной команде открывается файл для вывода под именем NEWFILE.TXT на DK:. Печать звездочки на термине означает, что редактор текста находится в командном режиме.

4.22. Команда EXECUTE. По команде EXECUTE вызываются один или более языковых трансляторов (АССЕМБЛЕР, ДИБОЛ или ФОРТРАН) для трансляции указанных файлов, объединяются объектные модули и запускается программа. Формат команды:

EX[ECUTE][/прк] спф[, ... спф][/прк]

где спф — спецификация транслируемого файла;

/прк — один или несколько переключателей.

Т а б л и ц а 20

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ EXECUTE

Переключатель	Функция
1	2
Переключатели, общие для всех трансляторов: /ALLOCATE:N	Используется с переключателями /EXECUTE, /LIST, /MAP или /OBJECT для резервирования области памяти на томе для выходного файла. Аргумент, N, представляет собой число блоков области для размещения фай-

1	2
/BOTTOM:N /DEBUG[:спф] /DUPLICATE /EXECUTE[:спф] ----- /GLOBAL	ла; N — целое число от 1 до 65535. N может быть также —1 (в этом случае на том же для выходного файла отводится наибольшая свободная область памяти) Задаёт значение (N) младшего адреса перемещаемого кода в модуле загрузки. Аргумент, N, представляет собой шестизначное четное восьмеричное число без знака. Если этот переключатель не используется, система загружает модуль с адреса 1000 Объединяет отладчик с отлаживаемой программой (неавтономный метод отладки описан в документе [6]) Записывает копию библиотечного модуля в каждый оверлейный сегмент, из которого есть обращение к данному библиотечному модулю. Переключатель /DUPLICATE используется для уменьшения размера корневого сегмента. После ввода данного переключателя, система выдает сообщение: DUPLICATE SYMBOL? Пользователю следует подать все глобальные имена, модули которых надо записать из библиотеки объектных модулей в оверлейные сегменты, и возврат каретки Используется для определения имени файла или тома для хранения файла с типом .SAV Используется с /MAP для создания в карте загрузки секции перекрестных ссылок глобальных имен. Глобальные имена перечислены по алфавиту. Модули, которые содержат обращения к глобальным именам или определение глобальных имен, перечисляются за

1	2
/LINKLIBRARY :спф /LIST[:спф] /MAP[:спф] /OBJECT[:спф] ----- /PROMPT /RUN --- /NORUN	глобальными именами также по алфавиту. Знак номера (#) после имени модуля указывает, что глобальное имя определено в данном модуле. Знак плюс (+) после имени модуля указывает, что модуль взят из библиотеки Включает указанный библиотечный файл в операцию редактирования. Следует повторить переключатель, если надо указать более чем один файл библиотеки. В строке команды следует за спецификацией библиотечного файла Используется для получения листинга при трансляции; спф — спецификация файла листинга. Если спецификация опущена и /LIST следует за именем команды, то листинг печатается на LP: Используется для получения карты загрузки после операции редактирования; спф — спецификация карты загрузки Используется для определения имени файла устройства для объектного файла Указывает, что будут вводиться дополнительные входные строки редактору связей. Система продолжает воспринимать все строки как входные для редактора связей пока не встретит символы // Запускает программу, если не было ошибок при трансляции и редактировании Запрещает запуск программы. Выполняется только трансляция и редактирование

1	2
/WIDE	Формирует карту загрузки в шесть пар колонок, вместо трех
Переключатели для трансляции с языка ДИБОЛ:	
/ALPHABETIZE	Располагает по алфавиту записи в листинге таблицы символов
/BUFFERING	Указывает, что будет использоваться простая буферизация при вводе-выводе. Обычно используется двойная буферизация
/CROSSREFERENCE	Создает таблицу перекрестных ссылок в листинге, т. к. система не печатает листинг по умолчанию, надо указать /LIST для получения распечатки таблицы перекрестных ссылок
/DIBOL	Вызывает ДИБОЛ для трансляции указанных файлов
/LINENUMBERS	Разрешает вывод номеров внутренней последовательности в объектный код
/NOLINENUMBERS	Запрещает вывод номеров внутренней последовательности в объектный код
/LOG	Создает список сообщений об ошибках, полученных при трансляции
/ONDEBUG	Включает таблицу символов в объектный код
/PAGE:N	Определяет количество строк в странице листинга; N — любое десятичное число от 1 до 32768. По умолчанию N=66
/TABLES	Включает в листинг таблицу символов и таблицу имен. Используется с переключателем /LIST
/WARNINGS	Разрешает вывод предупреждающих сообщений об ошибках
/NOWARNINGS	Запрещает вывод предупреждающих сообщений об ошибках

1	2
Переключатели для трансляции с языка ФОРТРАН: /CODE:XXX	Используется для получения объектного кода определенного вида (XXX) исходя из конкретной аппаратной конфигурации. Аргумент XXX — трехбуквенное сокращение вида кода (допустимые значения — EIS, FIS и THR). Более подробное описание видов кода программы и их функций см. в документе [8]
/DIAGNOSE	Разрешает вывод расширения поля генерированного (поточного) кода листинга для определения ошибок трансляции и их исправления
/EXTEND	Разрешает расширение строк исходной программы до 80 символов
/FORTRAN	Вызывает ФОРТРАН для трансляции указанных файлов
/HEADER	Разрешает вывод поля режима работы транслятора
/I4	Резервирует два слова памяти для целых переменных; по умолчанию резервируется одно слово памяти
/LINENUMBERS	Разрешает вывод номеров внутренней последовательности в объектный код
----- /NOLINENUMBERS	Запрещает вывод номеров внутренней последовательности в объектный код
/ONDEBUG	Разрешает трансляцию строк отладки (строка отладки имеет букву D в 1-ой колонке). По умолчанию строки отладки рассматриваются как комментарии
/RECORD:N	Определяет максимальный размер записи в байтах для операторов форматного ввода-вывода ($4 < N < 4095$). По умолчанию $N = 132$
/SHOW:N	Управляет печатью основных полей листинга N — код, определяющий

1	2
/STATISTICS /SWAP ----- /NOSWAP /UNITS:N /VECTORS ----- /NOVECTORS /WARNINGS /NOWARNINGS ----- Переключатели для трансляции с языка АСЕМБЛЕР: /CROSSREFERENCE [:тип[...:тип]]	формат листинга в табл. 9 описаны аргументы и их значения Разрешает вывод поля статистики листинга Разрешает свопинг программы USR во время выполнения программы пользователя Запрещает свопинг программы USR во время выполнения программы пользователя Определяет максимальное число логических устройств ($1 \leq N \leq 16$), которые могут быть открыты одновременно. По умолчанию $N=6$ Разрешает векторизацию многомерных массивов Запрещает векторизацию многомерных массивов Разрешает вывод предупреждающих сообщений об ошибках. По умолчанию — запрещается Запрещает вывод предупреждающих сообщений об ошибках Создает определенное поле таблицы перекрестных ссылок в листинге. Аргумент, тип, представляет собой односимвольный код, указывающий, какие поля таблицы перекрестных ссылок должны быть включены транслятором в листинг (см. табл. 8). Т. к. система не печатает листинг по умолчанию, надо указать /LIST для получения распечатки таблицы перекрестных ссылок

1	2
/DISABLE:влч [...:влч]	Используется для определения директивы .DSABL (см. табл. 7, где описаны аргументы, влч, и их значения)
/ENABLE:влч [...:влч]	Используется для определения директивы .ENABL (см. табл. 7, где описаны аргументы, влч, и их значения)
/LIBRARY	Используется для определения файла МАКРОбиблиотеки. Перед обращением к системной макробиблиотеке SYSMAC.SML транслятор просматривает прежде все макробиблиотеки пользователя. В строке команды следует за спецификацией файла макроопределений
/MACRO	Вызывает АССЕМБЛЕР для трансляции указанных файлов
/SHOW[:влч]	Используется для определения директивы .LIST (см. табл. 9, где описаны аргументы, влч, и их значения)
/NOSHOW[:влч]	Используется для определения директивы .NLIST (см. табл. 9, где описаны аргументы, влч, и их значения)

Существует три способа определения в строке команды нужного языкового транслятора. Для этого пользователь должен указать:

1. Переключатель (/MACRO, /FORTRAN или /DIBOL)
2. Тип файла (.MAC, .FOR или .DBL)
3. Если в строке команды не указан ни переключатель, ни тип файла, система ищет в справочнике файл с указанным именем и типом .MAC и затем с типом .DBL. Если нужный файл в справочнике существует, вызывается соответствующий транслятор. Если — нет, система предполагает, что файл имеет тип .FOR и вызывает транслятор ФОРТРАН.

По умолчанию приняты следующие типы файлов:

.LST — для файлов листинга, .MAP — для файлов карт загрузки, .OBJ — для объектных файлов, .SAV — для файлов формата отображения памяти, .MAC — для файлов на языке АССЕМБЛЕР, .FOR — для файлов на языке ФОРТРАН, .DBL — для файлов на языке ДИБОЛ.

Знак плюс (+) между спецификациями используется для трансляции нескольких исходных файлов (максимально 6) в один объектный файл с именем первого входного файла и типом .OBJ.

Для независимой трансляции нескольких исходных файлов спецификации файлов разделяются запятыми.

Затем система объединяет все объектные файлы в один с именем первого объектного файла и типом .SAV.

Команда EXECUTE может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается:

FILES?

Пользователь должен ввести с терминала спецификации исходных файлов и возврат каретки.

Переключатели команды EXECUTE (/MAP, /LIST, /EXECUTE и /OBJECT) имеют различное назначение в зависимости от занимаемого места в строке команды. Переключатели, следующие за именем команды, действуют на всю строку. Если переключатели следуют за спецификацией файла, то они действуют только на этот файл (или группу файлов, спецификации которых в строке команды соединены знаками плюс (+)).

Примеры:

1. EXECUTE MYPROG.MAC

EXECUTE/EXECUTE MYPROG.MAC

т. к. команда EXECUTE создает выполняемый файл (файл с типом .SAV) по умолчанию, приведенные команды имеют одинаковый результат: создаются файлы MYPROG.OBJ и MYPROG.SAV.

2. EXECUTE/MACRO/EXECUTE:RK1: PROG1, PROG2
по данной команде создаются файлы: PROG1.OBJ, PROG2.OBJ на DK: и PROG1.SAV на RK1:.

3. EXECUTE/MACRO KMN1, KMN2, MYPROG/EXECUTE
по данной команде создаются файлы: KMN1.OBJ, KMN2.OBJ, MYPROG.OBJ и один выполняемый файл MYPROG.SAV на DK:.

4. EXECUTE/LIST:TT: A.MAC

по данной команде транслируется файл A.MAC и создается A.OBJ; на TT: печатается листинг; затем создается файл A.SAV, который загружается в память и запускается.

5. EXECUTE/NORUN/MACRO/LIST:FILE1.OUT A+B

по данной команде:

— транслируются файлы A.MAC и B.MAC вместе, создавая A.OBJ, файл листинга FILE1.OUT на DK;

— с помощью редактора связей из A.OBJ (используя, если необходимо, SYSLIB.OBJ) создается A.SAV.

Нельзя задавать строку команды подобную следующей:

.EXECUTE/MACRO/LIST:FILE2 A.MAC, B.MAC

6. .EXECUTE/MACRO A+B/LIST:RK3:

.EXECUTE/MACRO/LIST:RK3: A+B

по первой команде создаются файлы A.OBJ и A.SAV на DK: и B.LST на RK3:.

По второй команде — A.OBJ и A.SAV на DK:, A.LST на RK3:.

7. .EXECUTE/MACRO A/LIST:B

.EXECUTE/MACRO/LIST:B A

две данные команды имеют одинаковый результат: A.OBJ, B.LST и A.SAV.

8. .EXECUTE/GLOBAL/MAP:DY1: MYPROG

по данной команде файл MYPROG.MAC транслируется, обрабатывается редактором связей; создается файл карты загрузки, MYPROG.MAP, который содержит секцию перекрестных ссылок.

9. .EXECUTE/PROMPT COSINE

*TAN/O:1

*COS1/O:1

*SIN3/O:2

*LML3/O:2//

данный пример показывает использование переключателя /PROMPT: транслируется файл COSINE.MAC, затем объединяются модули COSINE.OBJ (корневой сегмент) и оверлейные сегменты — TAN.OBJ, COS1.OBJ, SIN3.OBJ, LML3.OBJ.

Переключатель /PROMPT обеспечивает удобный путь создания программы оверлейной структуры из косвенного файла.

Пусть косвенный файл LCP.COM состоит из следующих строк:

A/PROMPT

SUB1/O:1

SUB2/O:1

SUB3, SUB4/O:1

//

по следующей команде создается файл DK:A.SAV и карта загрузки на LP:.

.EXECUTE/MAP @LCP

4.23. Команда FORMAT. По команде FORMAT выполняется форматирование и/или проверка кассетных и гибких дисков.

Формат команды: FORMAT[/прк] уст

где /прк — один или несколько переключателей;

уст — имя тома, для которого надо выполнить форматирование и/или проверку.

Таблица 21

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ FORMAT

Переключатель	Функция
1	2
/PATTERN[:влч]	Используется с переключателем /VERIFY[:ONLY] для определения эталонов проверки (комбинаций 16-ти битов) тома, аргумент влч, — восьмеричное целое число от 0 до 377, определяющее, какой эталон проверки пользователь хочет использовать. Во время операции система печатает на ТТ:, какие эталоны используются. Если влч не указан, система использует эталон с номером 8. Подробное описание эталонов проверки см. в документе [4]. После завершения проверки на терминале печатается: PATTERN #X ----- NNNNNN где X — номер эталона; NNNNNN — номер найденного плохого блока Запрашивает у пользователя подтверждения для выполнения команды FORMAT. Следует подать: — Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции; — NO (или любую строку, начинающуюся с N) и возврат каретки для запрещения операции
/QUERY -----	

1	2
/NOQUERY	Запрещает на терминале печать опрашивающих сообщений перед выполнением команды
/SINGLEDENSITY	Устанавливает одинарную плотность записи при форматировании гибких дисков DY:
/VERIFY[:ONLY]	Форматирует и проверяет том. Если используется :ONLY, выполняется только проверка, без форматирования
/WAIT	Используется в системах с одним приводом. После подачи переключателя /WAIT, система приостанавливает выполнение операции и печатает: MOUNT INPUT VOLUME IN <УСТ>; CONTINUE? следует установить нужный том и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки для продолжения операции. Если напечатать N (или любую строку, начинающуюся с N) и возврат каретки или два раза CY/C, операция не выполняется, и монитор печатает точку. Любой другой ответ вызывает повторную печать сообщения. После завершения операции форматирования система снова печатает: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? следует установить системный том и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ вызывает повторную печать сообщения

Под форматированием диска следует понимать процесс занесения адресной информации на сектор диска с установлением одинарной (для RK: и DY:) или двойной (для DY:) плотности записи.

Команда **FORMAT** используется в следующих случаях:

— когда пользователь имеет новый диск **RK**;

— когда надо изменить одинарную плотность записи на двойную (для **DY**;) или наоборот;

— когда надо уничтожить плохие блоки (хотя **FORMAT** не гарантирует уничтожение всех плохих блоков, но может значительно уменьшить их число).

Под проверкой тома понимается запись 16-ти битового эталона на каждый блок тома и последующее его чтение. Если запись и чтение эталона на блок невозможны, система сообщает о плохом блоке.

Примечание. Гибкие диски можно форматировать только на устройстве **DY**:. При попытке форматирования гибких дисков с устройства **DX**:, система печатает сообщение об ошибке.

При попытке форматирования тома в то время, когда загружено основное задание, система выдает сообщение:

BACKGROUND LOADED

<UCT>/FORMAT— ARE YOU SURE?

Следует подать **Y** (или любую строку, начинающуюся с **Y**) и возврат каретки для выполнения форматирования; **N** (или любую строку, начинающуюся с **N**) и возврат каретки или два раза **CU/C** для запрещения форматирования. После любого другого ответа сообщение печатается повторно.

Примечание. Хотя можно форматировать том, который используется основным заданием, делать это не рекомендуется, т. к. основное задание или система могут быть разрушены.

При попытке форматирования тома, который содержит защищенные файлы, система выдает сообщение:

VOLUME CONTAINS PROTECTED FILES;ARE YOU SURE?

следует подать **Y** (или любую строку, начинающуюся с **Y**) и возврат каретки для выполнения форматирования; **N** (или любую строку, начинающуюся с **N**) и возврат каретки или два раза **CU/C** для запрещения операции. После любого другого ответа сообщение печатается повторно.

Примеры:

1. **FORMAT/VERIFY/PATTERN:25 RK0:**

RK0:/FORMAT—ARE YOU SURE?Y

?FORMAT—I—FORMATING COMPLETE

PATTERN #5

PATTERN #3

PATTERN #1

?FORMAT—I—VERIFICATION COMPLETE

по данной команде выполняется форматирование и проверка диска RK0:, эталоны для проверки определяются числом 25.

2. .FORMAT/VERIFY:ONLY DX0:

DX0:/VERIFY—ARE YOU SURE?Y

PATTERN #8

?FORMAT—I—VERIFICATION COMPLETE

по данной команде выполняется только проверка тома DX0:, без форматирования.

3. .FORMAT/WAIT DY0:

DY0:/FORMAT—ARE YOU SURE?Y

MOUNT INPUT VOLUME IN <УСТ>;CONTINUE?Y

?FORMAT—I—FORMATTING COMPLETE

MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE(Y)?Y

по данной команде выполняется форматирование диска DY0: при наличии одного привода в системе.

4.24. Команда FORTRAN. По команде FORTRAN вызывается ФОРТРАН для трансляции указанных файлов.

Формат команды:

FORT[TRAN][/прк] спф[,...]

где /прк — один или несколько переключателей;

спф — спецификация файла, который будет транслироваться.

Т а б л и ц а 22

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ FORTRAN

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с /LIST или /OBJECT для резервирования области памяти на томе для выходного файла. Аргумент, N, указывает число блоков области памяти для размещения файла; N — целое число от 1 до 65535; может быть равно также —1 (в этом случае на томе для выходного файла

1	2
/CODE:XXX /DIAGNOSE /EXTEND /HEADER /I4 /LINENUMBERS ----- /NOLINENUMBERS /LIST[:спф] /OBJECT[:спф] ----- /NOOBJECT /ONDEBUG	отводится наибольшая свободная область памяти) Используется для получения объектного кода определенного вида (XXX) исходя из конкретной аппаратной конфигурации. Аргумент XXX — трехбуквенное сокращение вида кода (допустимые значения — EIS, FIS и THR) Разрешает вывод расширения поля генерированного (поточного) кода листинга для определения ошибок трансляции и их исправления Разрешает расширение строк исходной программы до 80 символов Разрешает вывод поля режима работы транслятора Резервирует два слова памяти для целых переменных. По умолчанию резервируется одно слово памяти Разрешает вывод номеров внутренней последовательности в объектный код Запрещает вывод номеров внутренней последовательности в объектный код Используется для получения листинга при трансляции; спф — спецификация файла листинга. Если спф опущена и /LIST следует за именем команды, то листинг печатается на LP: Используется для получения объектной программы; спф — спецификация объектного файла Запрещает создание объектного файла Разрешает трансляцию строк отладки (строка отладки имеет букву D в 1-ой колонке). По умолчанию строки отладки рассматриваются как комментарии

1	2
/RECORD:N	Определяет максимальный размер записи в байтах для операторов форматного ввода—вывода ($4 < N < 4095$). По умолчанию $N=132$
/SHOW:N	Управляет печатью основных полей листинга. Аргумент N — код, определяющий формат листинга (см. табл. 9)
/STATISTICS	Разрешает вывод поля статистики листинга
/SWAP -----	Разрешает свопинг программы <code>USR</code> во время выполнения программы пользователя
/NOSWAP	Запрещает свопинг программы <code>USR</code> во время выполнения программы пользователя
/UNITS:N	Определяет максимальное число логических устройств ($1 \leq N \leq 16$), которые могут быть открыты одновременно по умолчанию $N=6$
/VECTORS -----	Разрешает векторизацию многомерных массивов
/NOVECTORS	Запрещает векторизацию многомерных массивов
/WARNINGS	Разрешает вывод предупреждающих сообщений об ошибках
/NOWARNINGS -----	Запрещает вывод предупреждающих сообщений об ошибках

По умолчанию тип файла листинга — `.LST`, тип объектного файла — `.OBJ`, исходного файла — `.FOR`.

Для независимой трансляции нескольких исходных файлов спецификации файлов разделяются запятыми. Знак плюс (+) между спецификациями используется для трансляции нескольких исходных файлов в один объектный файл с именем первого входного файла и типом `.OBJ`. Однако, пользователь может указать любое имя для объектного файла по `/OBJECT[:сф]`.

Переключатели команды `FORTRAN` (`/LIST` и `/OBJECT`) имеют различное назначение в зависимости от занимаемого

места в строке команды. Переключатели, следующие за именем команды, действуют на всю строку. Если переключатели следуют за спецификацией файла, то они действуют только на этот файл (или группу файлов, соединенных знаками плюс).

Команда FORTRAN может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается:

FILES?

Пользователь должен ввести с терминала спецификацию исходного файла и возврат каретки.

Примеры:

1. FORTRAN/LIST MYPROG

по данной команде транслируется файл MYPROG.FOR, создается файл MYPROG.OBJ и на LP: печатается листинг.

2. FORTRAN MYPROG/LIST

по данной команде транслируется файл MYPROG.FOR и создаются файлы MYPROG.OBJ и MYPROG.LST.

3. FORTRAN/LIST:TT: A

по данной команде транслируется A.FOR, создается A.OBJ и листинг печатается на терминале.

4. FORTRAN/LIST:RK3: A

по данной команде транслируется A.FOR, создаются DK:A.OBJ и RK3:A.LST.

5. FORTRAN A+B/LIST:RK3:

по этой команде файлы A.FOR и B.FOR транслируются вместе и создаются файлы DK:A.OBJ и RK3:B.LST.

6. FORTRAN/LIST:FILE1.OUT A+B

по этой команде файлы A.FOR и B.FOR транслируются вместе и создаются файлы A.OBJ и FILE1.OUT (файл листинга) на DK:.

7. FORTRAN A/LIST:B

.FORTRAN/LIST:B A

эти две команды имеют одинаковый результат — B.LST и A.OBJ на DK:.

8. FORTRAN A/LIST, B

по этой команде создаются файлы A.OBJ, A.LST и B.OBJ.

4.25. Команда FRUN

По команде FRUN система загружает и запускает основную программу.

Формат команды:

FR[UN] спф[/прк]

где спф — спецификация файла, предназначенного для загрузки;

/прк — один или несколько переключателей.

Таблица 23

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ FRUN

Переключатель	Функция
1	2
/BUFFER:N	Резервирует дополнительно «N» слов сверх объема программы. По следующей формуле вычисляется объем памяти необходимый для выполнения программы на языке ФОРТРАН как основной программы. $N = [1/2[504 + (35 * N) + (R - 136) + (A * 512)]]$ где A — число файлов, открытых одновременно N — максимальное число (десятичное) каналов одновременно открытых. Устанавливается по команде /FORTRAN UNITS. По умолчанию N=6 R — максимальный размер записи. Устанавливается по команде FORTRAN/RECORD. По умолчанию N=6 Данная формула модифицируется с помощью определенных функций SYSLIB. Функция IQSET увеличивает N на дополнительную область для элементов очереди (QCOUNT): $N = [1/2[504 + (35 * N) + (R - 136) + A * 512]] + [10 * QCOUNT]$ функция ICDFN увеличивает N на дополнительную область для числа каналов (NUM):

1	2
/PAUSE	$N = [1/2[504 + (35 * N) + (R - 136) + A * 512]] + [6 * NUM]$ функция INTSET увеличивает N на дополнительную область для числа вызовов INTSET программой: $N = [1/2[504 + (35 * N) + (R - 136) + A * 512]] + [25 * INTSET]$ любая функция, включая INTSET, которая затрагивает подпрограммы завершения, увеличивает N на $[64 + R/2]$ слов, если не использовался /RECORD при трансляции. $N = [1/2[504 + (35 * N) + (R - 136) + A * 512]] + [64 + R/2]$ если по переключателю /BUFFER не отводится достаточно основной области для начального вызова подпрограммы завершения, на терминале печатается: ?ERR 0, MON—FORTRAN ERROR CALL Это сообщение также появляется, если недостаточно свободной памяти для фонового задания, или если две подпрограммы завершения монитора SJ пытаются выполняться одновременно. В последнем случае задание прерывается — пользователь должен использовать монитор FB для одновременного выполнения нескольких подпрограмм завершения Используется для целей отладки. В ответ на строку команды с этим переключателем, система загружает файл, печатает адрес загрузки, но не осуществляет запуск программы. Выполнение основного задания можно начать по команде RESUME

1	2
/TERMINAL:N	Назначает терминал с логическим номером «N» основному заданию. Если /TERMINAL:N не используется, то системный терминал совместно используется как основным, так и фоновым заданием
/NAME:имя	Назначает логическое имя (имя) основному заданию (переключатель допустим лишь для мониторов, поддерживающих системные задания)

Если при подаче команды FRUN активно другое основное задание, на терминале печатается сообщение об ошибке. В каждый момент времени может выполняться только одно основное задание. Если выполнение основного задания, занимающего некоторую область памяти, завершилось, система использует эту область памяти для другого задания. Команда FRUN используется для запуска виртуального основного задания и виртуального файла типа .SAV в основной области под управлением монитора ХМ.

Пример:

```
.FRUN DEMOSP/P
LOADED AT 127276
.RESUME
```

в данном примере загружается программа DEMOSP.REL, печатается адрес загрузки и затем по команде RESUME программа запускается.

4.26. Команда GET. По команде GET файл формата отображения памяти загружается в память.

Формат команды: GE[T] спф
где спф — спецификация загружаемого файла. Тип файла по умолчанию — .SAV.

Магнитная лента является недопустимым устройством для команды GET. Команда GET используется только для фоновых заданий. Нельзя по этой команде загружать в память программу, которая использует виртуальную память.

Команда GET обычно используется для загрузки программы в память при ее модификации и/или отладке. Команда GET может также использоваться совместно с командой

ми B, E, D и START для проверки изменений. По команде SAVE можно сделать эти изменения постоянными.

Пример:

.GET DEMOSP

.GET ODT

.START

по данным командам:

— загружается в память программа DEMOSP.SAV;

— загружается в память отладчик;

— начинается выполнение с точки входа ODT.SAV.

Нельзя использовать команду GET для загрузки оверлейных сегментов программы; по этой команде можно загрузить только корневой сегмент.

Если при использовании нескольких команд GET программы в памяти накладываются одна на другую, сохраняется лишь последняя. Если указанный в строке команды файл расположен не на системном устройстве, система автоматически загружает драйвер этого устройства.

4.27. Команда INITIALIZE. Команда INITIALIZE используется для очистки и инициализации справочника тома.

Формат команды: INI[TIALIZE]/прк] уст

где /прк — один или несколько переключателей;

уст — имя тома, инициализацию которого надо выполнить.

Таблица 24

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ INITIALIZE

Переключатель	Функция
1	2
/BACKUP	Используется для инициализации тома таким образом, что он будет использован в качестве выходного тома по команде BACKUP и для поиска плохих блоков. Выходные томы для команды BACKUP должны быть инициализированы по INIT/BACKUP (кроме магнитной ленты, т. к. система инициализирует магнитную ленту при выполнении команды BACKUP)

1	2
/BADBLOCKS	Используется для поиска на томе плохих блоков и записи файлов с типом .BAD на их место. После завершения поиска и инициализации тома справочник содержит справочную информацию только файлов с плохими блоками. Эта операция гарантирует, что система не будет использовать плохие блоки во время текущих операций. Если система находит плохой блок в загрузчике системы или справочнике тома, на терминале печатается сообщение об ошибке, означая, что устройство не пригодно к использованию. Если используется /BADBLOCKS.RET, система сохраняет файлы тома с типом .BAD и дает им имя FILE.BAD, не выполняя поиска плохих блоков
/FILE:спф	Используется для инициализации тома магнитной ленты и создания дистрибутивной ленты; спф — уст:мвоот. Вот, где уст — имя тома.
<u>/QUERY</u>	Запрашивает у пользователя подтверждения для выполнения команды INITIALIZE. Пользователю следует подать: — Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции; — NO (или любую другую строку, начинающуюся с N,) и возврат каретки для запрещения операции
/NOQUERY	Запрещает печать на терминале сообщений перед выполнением команды
/RESTORE	Восстанавливает справочник и файлы, которые находились на томе до инициализации, если после инициализации на этот том не было записи.

1	2
	/RESTORE не восстанавливает блоки загрузчика системы (используйте для этого COPY/BOOT)
/SEGMENT:N	Используется для инициализации тома диска и задания числа сегментов справочника. Число сегментов справочника определяет, в частности, число файлов, которые могут храниться на томе. Сегмент справочника содержит справочную информацию максимум о 72 файлах.
/VOLUMEID[:ONLY]	N — целое от 1 до 31. Используется для записи идентификации тома. Идентификация состоит из идентификатора тома (до 12 символов для диска, до 6 символов для магнитной ленты) и имени владельца (соотв. До 12 и 10 символов).
/WAIT	Переключатель /VOLUMEID:ONLY записывает новую идентификацию тома без повторной инициализации тома Используется в вычислительных системах с одним приводом. После подачи строки команды с /WAIT система инициализирует выполнение команды, но затем приостанавливается и печатает:
	MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где уст — имя устройства для тома, который будет инициализирован. Следует установить том, который будет инициализирован, и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции. Для запрещения операции надо подать N (или любую строку, начинающуюся с N) и возврат каретки или два раза CУ/С. Любой другой

1	2
	Ответ вызовет повторную печать сообщения. По завершении инициализации на терминале печатается: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? Следует заменить инициализированный том системным и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ вызывает повторную печать сообщения. Для использования /WAIT на системном томе должен присутствовать файл DUP.SAV

Инициализировать неиспользуемый ранее том — значит подготовить его для использования в операционной системе ФОДОС. Для магнитной ленты — записать метку тома VOL1 и метку LEOT (логический конец ленты); для диска — записать идентификацию тома и организовать на нем справочник.

Операция инициализации должна быть первой операцией, которую необходимо выполнить на новом томе после получения его от изготовителя.

Инициализация использованного ранее тома сводится к обнулению справочника этого тома.

Если в строке команды не указано имя тома, на терминале печатается:

DEVICE?

Пользователь должен ввести имя тома и возврат каретки.

Примеры:

1. .INI/BADBLOCKS RK1:

по данной команде выполняется инициализация тома RK1: и поиск плохих блоков.

2. .INI/FILE:MBOOT. BOT MT0:

по данной команде выполняется инициализация тома MT0: и запись файла MBOOT.BOT на MT0:.

3. .INITIALIZE/VOLUMEID RK1:

по данной команде происходит инициализация тома RK1:
и на терминале печатается:

VOL ID?

OWNER NAME?

Пользователь должен напечатать на терминале:

— идентификатор тома и возврат каретки;

— имя владельца и возврат каретки.

4. INITIALIZE/BADBLOCKS DY0:

DY0:/INITIALIZE;ARE YOU SURE?Y

BLOCK

TYPE

000120 80.

HARD

000471 313.

HARD

000521 337.

HARD

?DUP—W—BAD BLOCKS DETECTED 3.

В данном примере выполняется инициализация тома DY0: и поиск на нем плохих блоков. Система печатает адрес плохих ячеек (восьмеричные и десятичные) и тип плохого блока (HARD или SOFT).

4.28. Команда INSTALL. По команде INSTALL указанное устройство включается в систему.

Формат команды:

INS[TALL] уст[. . . уст]

где уст — имя устройства, которое должно быть включено в систему.

Команда INSTALL не допускает использование переключателей. Она позволяет вводить в системные таблицы имена устройств, которые первоначально отсутствовали в системе. (Существование драйвера, вводимого в систему устройства должно быть отмечено в соответствующей таблице монитора). Использование данной команды не изменяет файл монитора, но модифицирует таблицы монитора, находящиеся в данный момент в памяти.

Команда INSTALL может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается:

DEVICE?

Пользователь должен ввести с терминала имя устройства. Нельзя использовать в качестве имени устройства SY, DK или BA.

Если в таблицах монитора нет места для нового устройства (это можно определить с помощью команды SHOW), можно удалить из системы имеющееся устройство (по коман-

де REMOVE) перед включением в систему нового устройства.

Пример:

.INSTALL LP:

по данной команде построчно-печатающее устройство будет включено в систему.

4.29. Команда LIBRARY предназначена для создания, корректировки и распечатки каталогов библиотечных файлов.

Формат команды:

LIB[RARY] [/прк] спфбибл[спф[, ... спф] [/прк]]

где /прк — один или несколько переключателей;

спфбибл — спецификация библиотечного файла;

спф — спецификация входного файла.

Таблица 25

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ LIBRARY

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с переключателями /LIST или /OBJECT для резервирования области памяти на томе для выходного файла. Аргумент, N, указывает число блоков области памяти для размещения файла; N — целое число от 1 до 65535, может быть равно также — 1 (в этом случае на томе для выходного файла отводится наибольшая свободная область памяти)
/CREATE	Используется для создания объектной библиотеки. В строке команды указываются имя библиотечного файла и спецификации файлов, модули которых должны быть включены в библиотеку
/DELETE	Используется для удаления объектных модулей и всех связанных с ними глобальных имен из каталога файла библиотеки, т. к. объектные модули сами не удаляются, а только уда-

1	2
<code>/EXTRACT</code>	ляются их имена из каталога, модули и все глобальные имена, ранее удаленные, восстанавливаются при корректировке библиотеки Используется для исключения объектного модуля из библиотеки и хранения его на томе с тем же именем и типом .OBJ. Нельзя объединять этот переключатель с любым другим переключателем
<code><u>/INSERT</u></code>	Используется для включения объектного модуля в существующую библиотеку. Хотя можно ввести в библиотеку два или большее количество объектных модулей, имеющих одинаковое имя, это делать не рекомендуется, вследствие трудностей, возникающих при перемещении и корректировке модулей. Если в строке команды не указан никакой переключатель, система подразумевает <code>/INSERT</code>
<code>/LIST[:снф]</code>	Используется для исключения объекта каталога объектной библиотеки. Можно использовать этот переключатель с любыми другими переключателями (кроме <code>/MACRO</code>) для получения каталога объектной библиотеки после создания или корректировки библиотеки. Нельзя получить листинг каталога макробιβотеки
<code>/MACRO[:N]</code>	Используется для создания макробιβотеки. Аргумент N — размер (в блоках) каталога библиотеки. Это единственная допустимая операция для макробιβотек, т. е. можно только создать макробιβотеку, но нельзя ее распечатать или корректировать.

1	2
/OBJECT[:спф] /NOOBJECT /PROMPT /REMOVE /REPLACE /UPDATE	Для корректировки макробιβлиотеки используется редактор текста, затем полученный после редактирования файл повторно обрабатывается командой LIBRARY/MACRO Используется для создания файлов объектных библиотек Используется для запрещения создания новой объектной библиотек Указывает, что спецификации входных файлов будут продолжены на следующую строку. Этот переключатель допустим всеми другими переключателями кроме /EXTRACT. Символы // должны завершать список спецификаций входных файлов Стирает указанные глобальные имена из каталога библиотечного файла, т. к. глобальные имена стираются только из каталога (а не из самого объектного модуля), все глобальные имена, стерты ранее, восстанавливаются при корректировке библиотек В строке команды следует за спецификацией входного файла и используется для замены модулей в объектной библиотеке на модули с тем же именем, содержащиеся в указанных файлах. Если старый модуль не существует с тем же самым именем, что и указанный входной модуль, или, если переключатель /REPLACE следует за спецификацией библиотечного файла, то система печатает сообщение об ошибке и команда не выполняется Объединяет функции переключателей /INSERT и /REPLACE. Указывается после спецификации каждого файла,

1	2
	к которому применяется. Если имена модулей входного файла совпадают с именами библиотечных модулей, то система заменяет библиотечные модули модулями входного файла. Если же в библиотеке нет модулей с именами модулей входного файла, система включает в библиотеку модули входного файла

Спецификации входных файлов в строке команды разделяются запятыми. По умолчанию приняты следующие типы файлов: `.LST` — для файла листинга каталога библиотеки, `.OBJ` — для библиотек объектных модулей и входных объектных файлов, `.MAC` — для библиотек макроопределений и входных файлов на языке АССЕМБЛЕР.

Если в строке команды не указано имя библиотечного файла, на терминале печатается:

LIBRARY?

Если с переключателем `/CREATE`, `/INSERT` или `/MACRO` не указано имя входного файла, на ТТ: печатается:

FILES?

Если с переключателем `/EXTRACT` не указано имя входного файла, на терминале печатается:

FILE?

Другие переключатели не вызывают печати `FILES?` или `FILE?`. В ответ на эти сообщения пользователь должен напечатать на терминале имя библиотечного файла или имя входного файла и возврат каретки.

Команда `LIBRARY` выполняет все перечисленные выше функции на объектных библиотечных файлах. Она может создавать также файлы макробиблиотеки. Макробиблиотека — файл произвольного доступа (файл, который имеет справочник), содержащий один или более модулей одного типа. Объектная библиотека — файл, который содержит заголовков, каталог и один или более объектных модулей. Объектные модули в библиотеке — это подпрограммы, которые используются более чем одной программой или собраны вместе для удобства. Примером объектной библиотеки является

системная библиотека SYSLIB.OBJ, используемая редактором связей; SYSMAC.SML — пример макробιβотеки.

Объектные модули выбираются из библиотечного файла с помощью вызовов или обращений к глобальным именам объектного модуля; затем редактор связей редактирует объектные модули и программу пользователя, получая в результате один готовый к выполнению модуль. Каждый входной файл для объектной библиотеки содержит один или более объектных модулей и хранится на томе с указанным именем и типом. Как только объектный модуль вводится в библиотечный файл, обращение к нему будет происходить не с помощью имени файла, частью которого был этот модуль, а с помощью индивидуального имени модуля.

Если в строке команды с переключателем /DELETE не указаны имена удаляемых модулей, на терминале печатается:

MODULE NAME?

Перед исключением глобального имени из каталога библиотеки (по /REMOVE) и объектного модуля (по /EXTRACT), на терминале печатается:

GLOBAL?

Пользователь должен напечатать удаляемые глобальные имена. После каждого имени подается возврат каретки. Для завершения ввода списка имен надо подать строку, состоящую только из возврата каретки. Ответ пользователя на MODULE NAME? аналогичен, но вместо глобального имени надо напечатать имя объектного модуля.

Примеры:

1. .LIB/CREATE NEWLIB FIRST,SECOND

по данной команде создается объектная библиотека NEWLIB.OBJ, содержащая модули файлов FIRST.OBJ и SECOND.OBJ.

2. .LIB MATHLB MFUNCT/REPLACE

по данной команде происходит замена тех модулей в библиотеке MATHLB.OBJ, имена которых совпадают с именами модулей файла MFUNCT.OBJ.

3. .LIB/INSERT/OBJECT:NEWLIB OLDLIB THIRD,FOURTH

по данной команде создается библиотека NEWLIB.OBJ, которая содержит библиотеку OLDLIB.OBJ и модули файлов THIRD.OBJ и FOURTH.OBJ.

4. .LIB/MACRO/PROMPT MACLIB A, B, C, D

*E, F, G

*//

по данной команде создается макробиблиотека из семи входных файлов.

В табл. 26 показано, в какой последовательности выполняются переключатели команды LIBRARY.

Таблица 26

Переключатель	Подсказка
1	2
/CREATE /DELETE /REMOVE /UPDATE /REPLACE /INSERT /LIST	MODULE NAME? GLOBAL?

Следующий пример показывает использование нескольких переключателей.

.LIBRARY/LIST:TT:/REMOVE/INSERT NEWLIB

LIB2/REPLACE, LIB3

GLOBAL? SQRT

GLOBAL?

FODOS LIBRARIAN B03.00 FRI 15—JUL—83 00:08:37

NEWLIB FRI 15—JUL—83 00:08:35

MODULE GLOBALS GLOBALS GLOBALS

COS SIN

DATAN DATAN2

ATAN ATAN2

DCOS DSIN

данная команда выполняется в следующем порядке:

— удаляется глобальное имя SQRT из каталога NEWLIB;

— заменяет в NEWLIB все модули, имена которых совпадают с именами модулей в файле LIB2, на новые модули с теми же именами из файла LIB2.OBJ;

- включает в библиотеку модули файла LIB3;
- печатает на терминале каталог библиотеки NEWLIB.OBJ.

4.30. Команда LINK. По команде LINK объектные модули преобразуются в формат пригодный для загрузки и выполнения.

Формат команды: LIN[K] [/прк] спф[...спф]

где /прк — один или несколько переключателей;

спф — спецификация объектного редактируемого модуля.

Таблица 27

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ LINK

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с переключателями /MAP и /EXECUTE для резервирования области памяти на томе для входного файла. Аргумент, N, указывает число блоков области памяти для размещения файла; N — целое число от 1 до 65535, может быть равно также —1 (в этом случае на томе для выходного файла отводится наибольшая свободная область памяти). Переключатель /EXECUTE допустим с /ALLOCATE:N, только если создаются файлы типа .REL или .LDA
/ALPHABETIZE	Печатает по алфавиту глобальные имена из карты загрузки
/BITMAP	Создает карту использования памяти
/NOBITMAP	Запрещает создание карты использования памяти
/BOTTOM:N*	Указывает младший адрес перемещаемого кода модуля загрузки. Аргумент, N, — четное восьмеричное шестизначное число без знака. Если не используется этот переключатель, то при загрузке модуля нижний адрес его будет равен 1000

1	2
/BOUNDARY:N	Используется для запуска определенной программной секции с конкретного адреса.
	Система увеличивает значение адреса загрузки указываемой программной секции до величины, кратной N. Аргумент, N, должен быть степенью 2. После ввода полной строки команды LINK система печатает:
	BOUNDARY SECTION? Пользователь должен ввести имя соответствующей программной секции и возврат каретки
/DEBUG[:спф]	Используется для связывания программы отладчик с программой пользователя для отладки последней (не-автономный метод отладки описан в документе [6])
/DUPLICATE	Записывает копию библиотечного модуля в каждый оверлейный сегмент, из которого есть обращение к данному библиотечному модулю. Переключатель /DUPLICATE используется для уменьшения размера корневого сегмента. После ввода данного переключателя система выдает сообщение:
	BOUNDARY SECTION? Пользователю следует подать все глобальные имена, модули которых надо записать из библиотеки объектных модулей в оверлейные сегменты, и возврат каретки
/EXECUTE[:спф] -----	Указывает имя файла или устройство для файла формата отображения памяти (.SAV)
/NOEXECUTE	Запрещает создание файла формата .SAV

1	2
/EXTEND:N /FILL:N* /FOREGROUND[:N]* /GLOBAL	Увеличивает размер указываемой программной секции до N — байтов. Размер полученной программной секции будет больше или равен N в зависимости от того, сколько места фактически требует объектная программа. После ввода полной строки команды LINK система печатает: EXTEND SECTION? Пользователь должен ввести имя соответствующей программной секции и возврат каретки Используется для обнуления неиспользуемых ячеек модуля загрузки и записи значения N в эти ячейки Формирует загрузочный модуль в перемещаемом формате (.REL) для использования этого модуля в качестве основного задания под управлением монитора FB или XM. Нельзя использовать файлы с типом .REL с монитором SJ. Аргумент, N — размер стека в байтах. Если аргумент напечатан с точкой (N.), то N — десятичное число; если точка не указана, N — восьмеричное число. По умолчанию N=128.. Можно использовать /FOREGROUND[:N] с /XM, чтобы связать привилегированное основное задание с виртуальными оверлейными сегментами Используется с /MAP для создания в карте загрузки секции перекрестных ссылок глобальных имен. Глобальные имена перечислены по алфавиту. Модули, в которых выполняется обращение к глобальным именам или определение глобальных имен, перечисляются за глобальными именами так-

1	2
/INCLUDE /LDA * /LIMIT:N /LINKLIBRARY:спф	же по алфавиту. Знак номера (#) после имени модуля указывает, что глобальное имя определено в данном модуле. Знак плюс (+) после имени модуля указывает, что модуль взят из библиотеки Включает в загрузочный модуль указываемые модули из библиотек. После ввода полной строки команды /LINK на терминале печатается: LIBRARY SEARCH? Пользователь должен ввести с терминала глобальные имена, которые будут включены в загрузочный модуль. После каждого глобального имени необходимо печатать возврат каретки. Печать только возврата каретки завершает список глобальных имен. Формирует загрузочный модуль в абсолютном двоичном формате (.LDA). Устройством вывода файла такого формата может быть любое устройство системы файловой структуры, а также PC:. Переключатель /LDA формирует файл, который может быть загружен в память с помощью абсолютного загрузчика Используется с /XM и указывает, что количество памяти, зарезервированной по программному запросу .SETTOP, не должно превышать N (восьмеричное число) K слов. Если /LIMIT не используется, по запросу .SETTOP резервируется до 32K слов памяти, если имеется в наличии объем памяти меньше 32K слов, то весь имеющийся объем Включает указанный библиотечный файл в операцию редактирования в

1	2
	виде объектного библиотечного модуля. Этот переключатель не является необходимым, т. к. система автоматически распознает библиотечные файлы; он предусмотрен для совместимости с командой EXECUTE
/MAP[:спф]	Создает файл карты загрузки. По умолчанию карта загрузки печатается на LP:
/PROMPT	Указывает, что будут вводиться дополнительные входные строки редактору связей. Система продолжает воспринимать все строки как входные для редактора связей пока не встретит символы //
/ROUND:N	Увеличивает размер указываемой программной секции корневого сегмента так, чтобы размер корневого сегмента (в байтах) увеличился до величины кратной N. Аргумент, N, должен быть степенью 2. После ввода полной строки команды LINK система печатает: ROUND SECTION?
/RUN *	Пользователь должен напечатать имя соответствующей программной секции и возврат каретки Запускает .SAV файл. Этот переключатель допустим только для фоновых заданий
/SLOWLY	Разрешает системе использовать максимально возможный объем памяти для таблицы имен. Данный переключатель используется, если попытка редактирования не удалась вследствие переполнения таблицы имен
/STACK [:N] *	Устанавливает начальное значение стека равным значению указываемого глобального имени или значению N.

1	2
	При выполнении программы указатель стека автоматически становится равным содержимому 42 ячейки. Аргумент, N, — четное шестизначное восьмеричное число без знака. После ввода строки команды без указания значения N система печатает: STACK SYMBOL? В ответ пользователь должен напечатать глобальное имя, значение которого является адресом стека, и возврат каретки. Если указано несуществующее имя, система печатает сообщение об ошибке, и адрес стека устанавливается на 1000 (для файлов формата отображения памяти) или на младший адрес, если использовался /BOTTOM /SYMBOLTABLE [:спф] Создает файл таблицы глобальных имен; спф — спецификация создаваемого файла. Если спф не указана, система использует имя первого входного файла и тип .STB в качестве имени и типа файла /TOP:N Указывает старший адрес перемещаемого кода модуля загрузки. Аргумент, N, — четное восьмеричное число без знака /TRANSFER[:N] Указывает точку входа редактируемой программы. В качестве точки входа используется значение указываемого глобального имени или значение N. Аргумент, N, — четное шестизначное восьмеричное число без знака. После ввода строки команды без указания значения N система печатает: TRANSFER SYMBOL?

1	2
/WIDE	В ответ пользователь должен напечатать глобальное имя, значение которого является точкой входа и возврат каретки. Если указано несуществующее имя, печатается сообщение об ошибке и адрес точки входа устанавливается на 1. Если указан нечетный адрес точки входа, программа не запускается после загрузки и управление возвращается монитору
/XM	Используется с /MAP и формирует широкоформатную карту загрузки. Обычно в листинге карты загрузки глобальные имена печатаются в три колонки. По переключателю /WIDE листинг карты загрузки печатается в шесть колонок Разрешает использование программных запросов .SETTOP и .LIMIT монитора XM. Описание данных запросов см. в документе [9]

Операционная система ФОДОС позволяет отдельно транслировать программу пользователя и каждую из подпрограмм без присвоения абсолютного адреса загрузки во время трансляции. Редактор связей обрабатывает объектные модули и формирует загрузочный модуль, выполняя при этом:

- присвоение абсолютных адресов;
- связывание модулей посредством сопоставления глобальных имен, определенных в одном модуле, но обращение к которым есть в другом модуле;
- создает блок управления, используемый при загрузке программы;
- если указано (с помощью переключателя /PROMPT), формирует загрузочный модуль оверлейной структуры;
- включает в загрузочный модуль требуемые модули из системной библиотеки (SYSLIB.OBJ) и библиотек, указанных пользователем;

— создает карту загрузки.

По умолчанию приняты следующие типы файлов:

.SAV — для модуля загрузки; .REL — после переключателя /FOREGROUND и .LDA после /LDA; .MAP — для файла карты загрузки; .OBJ — для объектного модуля.

В строке команды спецификации файлов разделяются запятыми. Система создает загрузочный модуль с именем первого входного модуля.

Переключатели команды LINK (/EXECUTE и /MAP) имеют различное назначение в зависимости от занимаемого места в строке команды. Переключатели, следующие за именем команды, действуют на всю строку. Если переключатели следуют за спецификацией файла, то они действуют только на этот файл (или группу файлов, отделенных знаками плюс).

Примеры:

1. LINK MYPROG

.LINK/EXECUTE MYPROG

обе эти команды имеют одинаковый результат: при редактировании файла MYPROG.OBJ создается файл MYPROG.SAV на DK:.

2. LINK/MAP:TT: MYPROG

.LINK/MAP:RK3: MYPROG

по первой команде создается файл MYPROG.SAV на DK: и карта загрузки печатается на TT:. По второй команде создаются файлы MYPROG.SAV на DK: и MYPROG.MAP на RK3:.

3. LINK PROG1, PROG2/EXECUTE/MAP

по данной команде при редактировании создаются файлы PROG2.SAV и PROG2.MAP на DK:.

4. LINK/MAP:MAP.OUT PROG1, PROG2

по данной команде создаются PROG1.SAV и MAP.OUT на DK:.

4.31. Команда LOAD загружает указанные драйверы в оперативную память (драйверы становятся резидентными).
Формат команды:

LO[AD] уст[=типзд][, ... уст[=типзд]]

где уст — имя устройства, драйвер которого должен быть сделан резидентным;

типзд — тип задания, В или F, т. е. драйвер устройства может использоваться фоновым или основным заданием.

Аргумент, типзд, недопустим с монитором SJ. Любой драйвер, который используется основным заданием, должен быть предварительно загружен в оперативную память. Устройство

может быть занято исключительно основным, фоновым или системным заданием. (Если выполняется программа BATCH, то она считается фоновым заданием для мониторов FB и XM). Различные тома устройства одного типа могут быть заняты различными заданиями. Так, например, RK1: может принадлежать фоновому заданию в то время, как RK2: — основному. Если принадлежность устройства определенному заданию не указана, его может использовать любое задание. Если по команде LOAD загружается драйвер устройства нефайловой структуры и это устройство назначается заданием определенного типа, то все тома этого устройства становятся назначенными заданиям того же типа. Для изменения принадлежности устройства (или только тома) необходимо использовать следующую команду LOAD, т. е. нет необходимости делать драйверы нерезидентными по команде UNLOAD. Однако, если прерывается выполнение операций, связанных с магнитной лентой, необходимо сделать драйвер устройства нерезидентным (по команде UNLOAD), а затем повторно загрузить драйвер устройства (по команде LOAD). Нельзя изменить принадлежность системного тома и любая попытка сделать это игнорируется. В команде LOAD допускается использовать логические имена устройств.

Примеры:

1. LOAD DX:, MT:=B, LP:=F

по данной команде в оперативную память загружаются драйверы гибкого диска, построочно-печатающего устройства и магнитной ленты. После этой команды гибкий диск будет использоваться как фоновым, так и основным заданием; LP: — основным заданием; MT: — фоновым.

2. ASSIGN RK1: XY

.LOAD XY:=F

по первой команде тому RK1: присваивается логическое имя XY; по второй — том RK1: назначается основному заданию и драйвер устройства RK: загружается в память.

4.32. Команда MACRO. По команде MACRO вызывается АССЕМБЛЕР для трансляции одного или более исходных файлов.

Формат команды:

M[ACRO][/прк] спф[, ... спф][/прк]

где /прк — один или несколько переключателей;

спф — спецификация файла, который будет транслироваться.

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ MASRO

Переключатель	Функция
1	2
/ALLOCATE:N	Используется с /LIST или /OBJECT для резервирования области памяти на томе для выходного файла. Аргумент, N, указывает число блоков области памяти для размещения файла; N — целое число от 1 до 65535, может быть равно также — 1 (в этом случае на томе для выходного файла отводится наибольшая свободная область памяти)
/CROSSREFERENCE [:тип[...:тип]]	Создает определенное поле таблицы перекрестных ссылок в листинге. Аргумент, тип, представляет собой однознаковый код, указывающий, какие поля таблицы перекрестных ссылок должны быть включены транслятором в листинг (см. табл. 8), т. к. система не печатает листинг по умолчанию, надо указать /LIST для получения распечатки таблицы перекрестных ссылок
/DISABLE:влч [...:влч]	Используется для определения директивы .DSABLE (см. табл. 7)
/ENABLE:влч [...:влч]	Используется для определения директивы .ENABLE (см. табл. 7)
/LIBRARY	Используется для определения файла макробιβлиотеки перед обращением к системной макробιβлиотеке SYSMAC.SML, транслятор просматривает прежде все макробιβлиотеки пользователя. В строке команды следует за спецификацией файла макроопределений

1	2
/LIST[:спф]	Используется для получения листинга при трансляции; спф — спецификация файла листинга. Если спецификация и /LIST следует за именем команды, то листинг печатается на LP:
/OBJECT[:спф]	Используется для получения объектной программы; спф — спецификация объектного файла
/NOOBJECT	Запрещает создание объектного файла
/SHOW:влч	Уточняет использование директивы .LIST (см. табл. 9)
/NOSHOW:влч	Уточняет использование директивы .NLIST (см. табл. 9)

По умолчанию тип файла листинга — .LST, тип объектного файла — .OBJ, исходного файла — .MAC.

Для независимой трансляции нескольких исходных файлов спецификации файлов разделяются запятыми. Знак плюс (+) между спецификациями используется для трансляции нескольких исходных файлов в один объектный файл с именем первого входного файла и типом .OBJ. Однако, пользователь может указать любое имя для объектного файла по /OBJECT[:спф].

Переключатели команды MACRO (/LIST и /OBJECT) имеют различное назначение в зависимости от занимаемого места в строке команды. Переключатели, следующие за именем команды, действуют на всю строку. Если переключатели следуют за спецификацией файла, то они действуют только на этот файл (или группу файлов, отделенных знаками плюс).

Команда MACRO может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается:

FILES?

Пользователь должен ввести с терминала спецификацию исходного файла и возврат каретки.

Примеры:

1. `.MACRO MYLIB1/LIBRARY+A+MYLIB2/LIBRARY+B`
по данной команде при трансляции файла `A.MAC`, `АС-СЕМБЛЕР` просматривает сначала библиотеку `MYLIB1.MAC`, затем `SYSMAC.SML`, удовлетворяя аргументы директивы `.MCALL`. При трансляции `B.MAC` `АССЕМБЛЕР` просматривает библиотеку `MYLIB2.MAC`, затем `MYLIB1.MAC` и `SYSMAC.SML` в указанном порядке. В результате создается файл `A.OBJ`.

2. `.MACRO/LIST:TT: A`

`.MACRO/LIST:RK3: A`

по первой команде создается файл `A.OBJ` на `DK`; и листинг будет печататься на `TT`:. По второй команде создаются файлы `A.OBJ` на `DK`; и `A.LST` на `RK3`..

3. `.MACRO/LIST:FILE1.OUT A+B`

по данной команде `A.MAC` и `B.MAC` транслируются вместе и создается `A.OBJ` и `FILE1.OUT` на `DK`..

Нельзя задавать строку команды подобную следующей:

`.MACRO/LIST:FILE2 A, B`

4. `.MACRO A/LIST:B`

`.MACRO/LIST:B A`

после выполнения обеих команд создаются файлы `A.OBJ` и `B.LST` на `DK`..

5. `.MACRO A+B/OBJECT`

по этой команде `A.MAC` и `B.MAC` транслируются вместе и создается файл `B.OBJ` на `DK`..

4.33. Команда `MOUNT`. По команде `MOUNT` логическому диску ставится в соответствие файл.

Формат команды:

`MO[UNT]/прк логдиск спф [лимуст]`

где `/прк` — переключатель команды `MOUNT`;

логдиск — имя логического диска (`LDN`:, `N` — целое от 0 до 7) или логическое имя устройства, которое было назначено логическому диску;

спф — спецификация файла, который будет использоваться как логический диск (по умолчанию тип файла `.DSK`);

лимуст — логическое имя устройства, которое будет назначено логическому диску.

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ MOUNT

Переключатель	Функция
1	2
/WRITE	Разрешает запись на логический диск. После подачи /WRITE на логический диск разрешены чтение и запись
/NOWRITE	Запрещает запись на логический диск, т. е. разрешено только чтение на указанном логическом диске

Команда MOUNT может быть задана в виде полной строки или краткой. После ввода краткой строки система выдает сообщения:

DEVICE?

FILE?

Пользователь должен ввести с терминала имя логического диска (спецификацию файла и логическое имя устройства) и возврат каретки.

По команде SET LD CLEAN (п. 4.44) можно проверить и исправить соответствие логического диска и файла.

Примеры:

1. .MOUNT/NOWRITE LD5: DY0:DATA

по данной команде файл DATA.DSK на DY0: ставится в соответствие логическому диску LD5:. Для LD5: разрешено только чтение.

2. .ASSIGN LD5: OUT

.MOUNT

DEVICE?OUT

FILE?DY0:DATA.TST

в данном примере логическому диску LD5: назначается логическое имя OUT; логическому диску LD5: ставится в соответствие файл DY0:DATA.DSK и назначается логическое имя TST.

4.34. Команда PRINT предназначена для печати на порочно-печатающем устройстве содержимого одного или более файлов.

Формат команды:
 PRI[NT[/прк] спф[, ... спф]
 где спф — спецификация распечатываемого входного файла;
 /прк — один или несколько переключателей.

Т а б л и ц а 30

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ PRINT

Переключатель	Функция
1	2
/BEFORE[дата] *	Печатает все файлы, созданные до указанной даты. Если аргумент дата не указан, система использует системную дату
/COPIES:N	Делает N копий файла на LP.; N — десятичное число от 2 до 32 (N=1 по умолчанию)
/DATA[дата] *	Печатает файлы с указанной датой создания. Если аргумент дата не указан, система использует системную дату
/DELETE	Удаляет указанный файл на томе после печати его содержимого на LP.. Перед удалением файла система не спрашивает пользователя о включении в указанную операцию файла
/FLAGPAGE:N	Печатает заголовок перед текстом файла: имя распечатываемого файла (заглавными буквами), а также имя задания, дату и время вывода задания, номер копии и число копий, спецификацию входного файла. Переключатель /FLAGPAGE:N допустим, только если выполняется QUEUE. Аргумент N указывает, сколько раз будет печататься заголовок. Если в строке команды указаны несколько файлов, то заголовок будет печататься перед каждым файлом. Заголовок также бу-

1	2
	дет печататься по умолчанию после ввода командной строки с переключателем /P программы QUEMAN. (см. [4])
/NOFLAGPAGE	Запрещает печать заголовка перед печатью файла. Используется только, если выполняется QUEUE и если ранее был установлен переключатель /P в командной строке программы QUEMAN
/INFORMATION	Используется, если пользователь не уверен есть ли на томе указанные в строке команды файлы. Если указанный файл не найден — печатается информационное сообщение об ошибке, но выполнение операции не прекращается — печатаются все найденные файлы
/LOG	Печатает на терминале список файлов, которые были включены в текущую операцию. Обычно система печатает список файлов, если спецификация файла в строке команды содержит символы % и *. Если указать /QUERY, то система будет опрашивать пользователя о включении в указанную операцию каждого файла, но не будет печатать список всех файлов, участвовавших в этой операции. Чтобы происходил опрос пользователя и печать списка, в строке команды должны присутствовать переключатели /LOG/QUERY
/NOLOG	Запрещает печать списка файлов, которые были распечатаны по текущей команде. Этот переключатель используется для запрещения печати списка файлов, если в строке команды спецификация файла содержала симво-

1	2
/NAME:[уст:] имзад /NEWFILES* /PRINTER /PROMPT /QUERY	лы % и *. Переключатель /NOLOG недопустим, если выполняется QUEUE. Указывает имя задания для файлов, которые пользователь хочет распечатать. Допустим, если выполняется QUEUE. Аргумент, имязд, представляет собой имя задания (не более 6 буквенно-цифровых символов). Если не указан /NAME, система использует имя первого выходного файла в качестве имени задания. Если указано устройство, уст, то перечисленные в строке команды файлы будут выводиться на это устройство одним файлом с именем имязд и типом JOB Печатает содержимое файлов, имеющих текущую дату создания Печатает файл на построчно-печатающем устройстве, игнорируя вывод на LP: по QUEUE. Используется, если выполняется QUEUE Указывает, что строка команды будет продолжена на следующие строки. Допустим, если выполняется QUEUE, символы // завершают строку команды Опрашивает пользователя о включении в указанную операцию каждого файла. Особенно полезен в командах, у которых спецификации файлов содержат символы % и *. Для выполнения операции надо напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Система интерпретирует другой ответ как NO и не выполняет указанную операцию

1	2
/SINCE[дата] * /WAIT	Печатает файлы, созданные в течение или после указанной даты. Если аргумент дата не указан, используется системная дата Используется, если конфигурация вычислительной системы содержит один привод. Недопустим, если выполняется QUEUE. После ввода строки команды с /WAIT система инициирует выполнение команды, затем приостанавливается и печатает: MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя устройства для входного тома. Следует установить входной том и напечатать: — Y (или любую строку, начинающуюся с Y) возврат каретки для выполнения операции; — N (или любую строку, начинающуюся с N) или два раза CY/C для прекращения выполнения операции и печати монитором точки; — любой другой ответ вызывает повторную печать сообщения. Затем система выдает сообщение: MOUNT OUTPUT VOLUME IN <LP0:>;CONTINUE? т. е. пользователь должен проверить готовность LP: и напечатать Y и возврат каретки. После завершения операции, на TT: печатается: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE?

1	2
	Пользователь должен заменить входной том системным и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ вызывает повторную печать сообщения

Спецификации входных файлов (допустимо указывать не более 6) отделяются запятыми в строке команды. Система печатает содержимое файлов в том порядке, в каком они указаны в строке команды. Допустимо использовать символы * и % в спецификации файла. В этом случае система печатает файлы в том порядке, в каком они записаны на томе. Тип входного файла по умолчанию — .LST.

Команда PRINT может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается:

FILES?

Пользователь должен ввести с терминала спецификацию входного файла и возврат каретки.

Некоторые переключатели допускают в качестве своего аргумента дату. Синтаксис, определяющий дату, следующий:

[:дд] [:ммм] [:гг]

где дд — день (десятичное число от 1 до 31);

ммм — месяц (первые три буквы названия месяца);

гг — год (десятичное число от 82 до 99).

Значение даты по умолчанию соответствует текущей системной дате. Если опущена любая часть даты (дд, ммм, гг), система использует соответствующую часть текущей системной даты.

Например, если указать только :82 и текущая системная дата 4 MAY 1983, система использует 4:MAY:82. Если текущая системная дата не установлена, она считается нулевой.

Примеры:

1. .PRINT/COPIES:3 REPORT

по данной команде на LP: печатается три копии файла REPORT.LST.

2. .PRINT/LOG/DELETE REPORT

по данной команде на LP: печатается файл REPORT.LST, затем этот файл удаляется с DK: и на терминале печатается список файлов, участвовавших в этой операции.

3. .PRINT/NEWFILES *.LST

по данной команде на LP: печатаются все файлы, имеющие текущую дату создания и тип .LST, после чего на терминале будет напечатан список этих файлов.

4. .PRINT/NAME:DX1:JOB5 FILE1, FILE2, FILE3

по данной команде задание JOB5, состоящее из файлов FILE1.LST, FILE2.LST и FILE3.LST, выводится файлом JOB5.JOB на DX1:.

5. .PRINT/WAIT RK0:ERREX.MAC

MOUNT INPUT VOLUME IN RK0;;CONTINUE?Y

MOUNT OUTPUT VOLUME IN LP0;;CONTINUE?Y

MOUNT SYSTEM VOLUME IN RK0;;CONTINUE?Y

по данной команде после установки входного тома и проверки готовности LP0: система печатает указанный файл, после чего требует установить системный том.

4.35. Команда PROTECT. По команде PROTECT устанавливается защита от удаления файлу, т. е. файл не может быть удален, пока не будет отменена защита файла (по команде UNPROTECT, п. 4.53).

Формат команды:

PRO[TECT][/прк] спф[, . . . , спф]

где /прк — один или несколько переключателей команды PROTECT;

спф — спецификация файла, для которого устанавливается защита.

Таблица 31

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ PROTECT

Переключатель	Функция
1	2
/BEFORE[дата]*	Устанавливает защиту файлам, созданным до указанной даты. Если аргумент дата не указан, используется текущая системная дата

1	2
/DATE[дата] [*]	Устанавливает защиту файлам с указанной датой создания, если аргумент дата не указан, используется текущая системная дата
/EXCLUDE /INFORMATION	Устанавливает защиту всем файлам тома, кроме указанных Используется, если пользователь не уверен, есть ли на томе указанные в строке команды файлы. Если указанный файл не найден — печатается информационное сообщение об ошибке, но выполнение операции не прекращается — устанавливается защита для найденных файлов
/LOG	Печатает на терминале список файлов, для которых установлена защита по текущей команде. Данный список печатается по умолчанию, если спецификация файла содержит конструкцию * и %. Если указан /LOG, система не опрашивает пользователя о включении каждого файла в операцию
/NOLOG	Запрещает печать на терминале списка файлов, для которых установлена защита по текущей команде
/NEWFILES [*]	Устанавливает защиту для файлов, имеющих текущую дату создания
/QUERY	Опрашивает пользователя о включении в указанную операцию каждого файла. Этот переключатель полезен, если спецификация файла содержит конструкцию * и %. Пользователь должен напечатать на терминале: — Y (или любую другую строку, начинающуюся с Y) и возврат каретки для выполнения операции;

1	2
	— NO (или любой другой ответ) и возврат каретки, если файл исключается из операции
/SETDATE[дата]	Присваивает указанную дату файлам, для которых установлена защита по текущей команде. Если аргумент дата не указан, используется текущая системная дата. Если системная дата не установлена, используется нулевая дата. Обычно, при выполнении операции система сохраняет дату создания файла
/SINCE[дата] *	Устанавливает защиту для файлов, созданных в течение или после указанной даты.
/SYSTEM	Если аргумент дата не указан, используется текущая системная дата. Устанавливает защиту для системных файлов (.SYS) без этого переключателя системные файлы исключаются из операции и печать сообщений напоминает об этом
/WAIT	Используется, если конфигурация вычислительной системы содержит один привод. После подачи строки команды с переключателем /WAIT, система инициирует выполнение команды, но затем приостанавливается и выдает сообщение: MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя тома устройства. Следует установить том, для файлов которого надо установить защиту, и напечатать:

1	2
	— Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции; — N (или любую строку, начинающуюся с N) и возврат каретки или два раза СУ/С для прекращения операции и передачи управления монитору; — любой другой ответ вызывает повторную печать сообщения. После завершения операции система выдает сообщение: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? Следует установить системный том и подать Y или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ вызывает повторную печать сообщения

В строке команды можно указать не более шести спецификаций файлов. Спецификации файлов в строке команды разделяются запятыми. Спецификация файла может содержать конструкцию * и %.

Команда PROTECT задается с терминала в виде полной строки или краткой. Если задана краткая строка команды, на терминале печатается:

FILES?

Пользователь должен напечатать на терминале спецификацию файла, для которого будет установлена защита и возврат каретки.

Некоторые переключатели допускают в качестве своего аргумента дату. Синтаксис, определяющий дату, следующий:

[:дд][:ммм][:гг]

где дд — день (десятичное число от 1 до 31);

ммм — месяц (первые три буквы названия месяца);

гг — год (десятичное число от 82 до 99).

Значение даты по умолчанию соответствует текущей системной дате. Если опущена любая часть даты (дд, ммм, гг), система использует соответствующую часть текущей системной даты. Например, если указать только ::82 и текущая системная дата 4 MAY 1983, система использует 4:MAY:82. Если текущая системная дата не установлена, она считается нулевой.

Примеры:

1. .PROTECT/BEFORE:20:MAR:83*.MAC
FILES PROTECTED:
DK:A.MAC
DK:B.MAC
DK:C.MAC

по данной команде для всех файлов с типом .MAC, которые были созданы до 20 марта 1983 года, устанавливается защита.

2. .PROTECT/EXCLUDE DY0:*.SAV
?PIP—W—NO.SYS ACTION
FILES PROTECTED:
DY0:ABC.OLD
DY0:AAF.OLD
DY0:COMB.
DY0:MERGE.OLD

по данной команде устанавливается защита для всех файлов тома DY0:, кроме файлов с типами .SAV и .SYS.

3. .PROTECT/INFORMATION DY0: (FIL1, FIL2, FIL3).TXT
?PIP—I—FILE NOT FOUND DY0:FIL2.TXT

по данной команде устанавливается защита для файлов DY0:FIL1.TXT и DY0:FIL3.TXT. Файл FIL2.TXT на DY0: не найден.

4. .PROTECT/SYSTEM DY0:MM.*
FILES PROTECTED:
DY0:MM.MAC
DY0:MM.OBJ
DY0:MM.SAV
DY0:MM.SYS

по данной команде устанавливается защита для всех файлов с именем MM и любым типом, в том числе с типом .SYS.

5. .PROTECT/WAIT DY0:FILE.MAC
MOUNT INPUT VOLUME IN DY0;;CONTINUE?Y
MOUNT SYSTEM VOLUME IN DY0;;CONTINUE?Y

данный пример показывает использование переключателя /WAIT, устанавливается защита для файла DY0:FILE.MAC.

4.36. Команда R. По команде R указанный файл формата отображения памяти загружается с системного тома в оперативную память, и инициируется его выполнение.

Формат команды: R спф
где спф — спецификация загружаемого файла.

По умолчанию тип файла — .SAV, устройство — SY:. Команда R подобна команде RUN, но указанный в строке команды файл должен храниться на системном томе. Команда R используется только с фоновым заданием.

Команда R — единственная команда командного языка системы, по которой запускается виртуальное фоновое задание под управлением монитора ХМ. По команде R выполняется разбиение виртуальной памяти для задания, создание блока определений области 0 и окна 0 и устанавливаются регистры отображения пользователя (см. [7]).

Пример:

.R PROG

по данной команде в оперативную память загружается файл PROG.SAV с тома SY: и система инициирует его выполнение.

4.37. Команда REENTER. По команде REENTER инициируется выполнение программы с адреса повторного пуска (пусковой адрес минус два).

Формат команды: REE[ENTER]

Формат команды REENTER не допускает использование переключателей или аргументов. Команда REENTER не очищает и не восстанавливает какие-либо области памяти. Эта команда рекомендуется для повторного запуска программы без повторной загрузки. Команда REENTER используется для возврата к системной программе или любой программе после ее окончания или после того, как выполнение программы было прервано с помощью двух нажатий CY/C.

Если команда REENTER не допустима для программы, на терминале печатается сообщение:

?KMON—F—ILLEGAL COMMAND

Запустить программу можно используя команду R или RUN. Если действует установка SET EXIT NOSWAP, инициирование программы с адреса повторного пуска невозможно.

Пример:

.R DIR

LP:=DK:.*

^C

^C

.REENTER

*

В данном примере программа DIR печатает на LP: справочник DK:. Два нажатия СУ/С прерывают печать листинга и управление возвращается монитору. Команда REENTER запускает DIR с адреса повторного пуска. В примере команда REENTER не продолжает печать листинга с прерванного места, но управление возвращается программе DIR.

4.38. Команда REMOVE. По команде REMOVE указанное устройство удаляется из системы.

Формат команды: REM[OVE] уст[, ... уст]

где уст — имя устройства, предназначенного для удаления из системы. Недопустимые имена — SY:, BA:, MQ: и TT:.

Команда REMOVE не допускает использование переключателей. Команда REMOVE может быть задана в виде полной или краткой строки. После ввода краткой строки на терминале печатается:

DEVICE?

Пользователь должен ввести с терминала имя устройства и возврат каретки.

Использование команды REMOVE не изменяет файл монитора на диске, а лишь модифицирует таблицы монитора, находящегося в памяти в текущий момент.

Это позволяет изменять конфигурацию системы. Повторная загрузка в память файла монитора восстановит стандартную конфигурацию системы.

После команды REMOVE можно использовать команду INSTALL для включения в систему нового устройства. С помощью команды SHOW можно распечатать на терминале список устройств, которые в данный момент обслуживаются системой.

Пример:

.REMOVE LP:,MT:

по данной команде из системы удаляются построчно-печатающее устройство и магнитная лента.

4.39. Команда RENAME. По команде RENAME существующим файлам присваиваются новые имена.

Формат команды:
REN[AME] [/прк] входспф[... входспф] выходспф
 где /прк — один или несколько переключателей;
 входспф — спецификация файла, который должен быть
 переименован; выходспф — новая спецификация файла.

Т а б л и ц а 32

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ **RENAME**

Переключатель	Функция
1	2
/BEFORE[дата]	Переименовывает файлы, созданные до указанной даты. Если аргумент дата не указан, используется текущая системная дата
/DATA[дата]	Переименовывает файлы с указанной датой создания. Если аргумент дата не указан, используется текущая системная дата
/INFORMATION	Используется, если пользователь не уверен, есть ли на томе указанные в строке команды файлы. Если указанный файл не найден — печатается информационное сообщение об ошибке, но выполнение операции не прекращается — переименовываются найденные файлы
/LOG	Печатает на терминале список файлов, переименованных текущей операцией. Обычно, система печатает список, если спецификация файла содержит символы * и %. Если в строке команды указать переключатель /QUERY , система будет опрашивать пользователя о включении в указанную операцию каждого указанного файла, но не будет печатать список всех файлов, участвовавших в этой операции. Чтобы происходил опрос пользователя и печать списка, в стро-

1	2
/NOQUERY	ке команды должны присутствовать переключатели /LOG/QUERY
/NEWFILES	Запрещает печать списка файлов, переименованных текущей операцией
/PROTECTION	Переименовывает только файлы с текущей датой создания
/NOPROTECTION /QUERY	Устанавливает защиту файла от удаления, т. е. файл не может быть случайно удален
	Отменяет защиту файла от удаления Опрашивает пользователя о включении в указанную операцию каждого указанного файла. Для включения в операцию указанного файла и начала выполнения операции пользователь должен напечатать на терминале Y (или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ система интерпретирует как NO, и операция не выполняется. Рекомендуется использовать этот переключатель, если спецификация файла в строке команды содержит символы * и %
/REPLACE	Если на томе вывода уже существует файл с именем выходного файла, система при переименовании файла стирает его дубликат.
/NOREPLACE	Не переименовывает файл, если на томе вывода уже существует файл с тем же самым именем
/SETDATE[дата]	Если указан аргумент дата, присваивает указанную дату всем переименованным файлам. Если аргумент не указан, система сохраняет переименованному файлу дату создания файла
/SINCE[дата]	Переименовывает все файлы на указанном томе, созданные в течение или после указанной даты

1	2
/SYSTEM /WAIT	Переименовывает системные файлы. Без этого переключателя системные файлы исключаются из операций Используется, если конфигурация вычислительной системы содержит один привод. После ввода строки команды с /WAIT система инициирует выполнение команды, затем приостанавливается и выдает сообщение: MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя устройства для входного тома. Пользователю следует установить том, файлы которого надо переименовать, и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции. Для прекращения операции следует подать N (или любую строку, начинающуюся с N) и возврат каретки или два раза CU/C. После завершения операции система печатает: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? Пользователь должен заменить входной том на системный, напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ вызывает повторную печать сообщения. При использовании /WAIT, на системном томе должна находиться программа PIP.SAV

Допустимое число входных файлов — 6, выходных — 1. Имя тома должно быть одинаковым во входной и выходной спецификации, т. е. нельзя переименовать файл с одного тома на другой.

Для переименования файла, содержащего плохие блоки, надо точно указать имя и тип файла. Обычно нет необходимости работать с такими файлами или переименовывать файлы с типом .BAD.

Некоторые переключатели допускают в качестве своего аргумента дату. Синтаксис, определяющий дату, следующий:

[дд:][:ммм][:гг]

где дд — день (десятичное число от 1 до 31);

ммм — месяц (первые три буквы названия месяца);

гг — год (десятичное число от 82 до 99).

Значение даты по умолчанию соответствует текущей системной дате. Если опущена любая часть даты (дд,ммм,гг), система использует соответствующую часть текущей системной даты. Например, если указать только : :82 и текущая системная дата 4 MAY 1983, система использует 4:MAY:82. Если текущая системная дата не установлена, она считается нулевой.

Примеры:

1. `.RENAME DX0:A% % .MAC DX0:* .FOR`

по данной команде после переименования на терминале печатается список файлов, переименованных в данной операции.

2. `.RENAME/QUERY DX0: (PIP1.SAV PIP.SAV)`

по данной команде перед переименованием система спрашивает пользователя о включении в операцию файла PIP1.SAV на DX0:.

3. `.RENAME/SETDATE DX0:(* .FOR * .OLD)`

по данной команде система переименовывает файлы и присваивает им текущую дату. Затем на терминале печатается список переименованных файлов.

4. `.RENAME/SYSTEM DX0:MM.SYS DX0:MX.SYS`

по данной команде происходит переименование файла MM.SYS на MX.SYS.

5. `.RENAME/BEFORE:21:APR:83 DY0:* .MAC`

`DY0:* .BAK`

по данной команде переименовываются все файлы на DY0: с типом .MAC, созданные до 21 апреля 1983 года.

4.40. Команда RESET. По команде RESET восстанавливаются определенные таблицы системы и очищается фоновая область.

Формат команды: `RESE[T]`

Команда RESET не допускает использование переключателей и аргументов. Команда RESET используется для очист-

ки каналов ввода-вывода, области пользователя и делает нерезидентными драйверы всех устройств, которые не были сделаны резидентными по команде LOAD.

Команда RESET используется также для запрещения действия C/D, для очистки ячеек 40—53 и восстановления указателя стека клавиатурного монитора. Команда RESET используется, если надо привести в исходное состояние файл монитора или аннулировать результаты команд GET. Команда RESET не действует на основное или системное задание.

Пример:

.RESET

.R PROG

в данном примере перед выполнением программы используется команда RESET.

4.41. Команда RESUME. По команде RESUME возобновляется выполнение основного или системного задания с точки, где было прервано выполнение задания командой SUSPEND.

Формат команды: .RESU[ME][имязд]

где имязд — имя системного или основного задания. Указывается, если монитор поддерживает системное задание. Допускается использовать логические имена заданий.

Команда RESUME не допускает использование переключателей. По команде RESUME основное задание вводит подпрограммы завершения, запланированные после прекращения выполнения задания. Команда RESUME допустима только с монитором FB или XM.

Команда RESUME используется также для запуска основного (или системного) задания, которое было загружено в оперативную память по команде FRUN/PAUSE (см. п. 4.25).

Примеры:

1. .RESUME

по данной команде возобновляется выполнение основного задания, которое в настоящий момент приостановлено.

2. .RESUME QUEUE

по данной команде возобновляется выполнение системного задания, QUEUE.SYS, которое было приостановлено.

4.42. По команде RUN файл формата отображения памяти загружается в память и инициируется его выполнение.

Формат команды: RU[N] спф[входспф выходспф]

где спф — спецификация загружаемого файла;

входспф — список спецификаций входных файлов для выполняемой программы;

выходспф — список спецификаций выходных файлов для выполняемой программы.

Допустимый тип для загружаемого файла — .SAV, устройство по умолчанию — DK:. По команде RUN автоматически загружается драйвер указанного устройства, если он (драйвер) не был резидентным ранее. По команде RUN выполняются только программы, подготовленные к выполнению в качестве фоновых заданий. Нельзя использовать команду RUN с заданием, которое требует виртуальной памяти и выполняется под управлением монитора ХМ (загружается и запускается фоновое виртуальное задание по команде R).

Если в строке команды указаны входспф и выходспф, то программе пользователя или системной программе фактически задается входная строка. Система автоматически преобразует указанные списки спецификаций в формат, допустимый интерпретатором командной строки.

Примеры:

1. .RUN DK1:PROG.SAV

по данной команде загружается и запускается файл PROG.SAV с тома DK1:.

2. .RUN DIR DX1:*. * LP:/E

по данной команде справочник тома DX1: печатается на LP:. Данная команда КЯС эквивалентна следующему набору командных строк:

.RUN DIR

LP:/E=DX1: *

*^C

Если в строке команды указаны спецификации входных (выходных) файлов, то по окончании выполнения программы управление возвращается монитору.

4.43. Команда SAVE. По команде SAVE содержимое области оперативной памяти записывается в виде файла в формате отображения памяти на указанное устройство произвольного доступа.

Формат команды: SA[VE] спф[прм]

где спф — спецификация файла, который будет записан на том;

прм — параметры, представляющие собой адреса ячеек, содержимое которых будет записано в виде файла на том, имеют следующий формат:

адр(1)[—адр(2)][,адр(3)[...—адр(N)]]

где адр — восьмеричное число, указывающее, какой блок ячеек памяти должен быть записан на том. Каждый последующий адрес в строке команды должен быть больше предыдущего.

По умолчанию тип файла — .SAV. В операционной системе ФОДОС содержимое области памяти записывается на том по блокам (256. слов), начиная с адреса кратного 256-и. Если указанные адреса составляют блок размером менее 256 слов, происходит добавление адресов до размера 256 слов. Система может записывать на том содержимое памяти, начиная с нулевого адреса до старшего, указанного в строке команды, или до старшего адреса программы, указанного в ячейке 50.

По команде SAVE на том записывается только корневой сегмент программы оверлейной структуры. Команду SAVE нельзя использовать для записи на том основного или виртуального заданий.

Пример:

.SAVE FILE1 10000—11000, 14000—14100

по данной команде содержимое ячеек 10000—10777 и 14000—14777 записывается на том в виде файла FILE1.SAV.

4.44. Команда SET позволяет изменить параметры драйвера устройства и конфигурации системы.

Формат команды:

SET фимуст знач или SET элконфгр знач

где фимуст — физическое имя устройства, драйвер которого надо модифицировать;

знач — допустимые значения параметров драйвера и конфигурации системы;

элконфгр — параметр конфигурации системы, который надо модифицировать.

В табл. 33 перечисляются параметры драйверов и конфигурации системы и допустимые значения этих параметров.

Т а б л и ц а 33

Фимуст/ элконфгр	Значение	Действие
1	2	3
NN:	CSR=K	Модифицирует непосредственно в драйвере устройства адрес регистра состояния (уста-

1	2	3
NN:	CSR2=K	навливая его равным K) для первого контроллера. Допустимый диапазон для K от 160000 до 177570. Переменная NN представляет собой постоянное имя одного из перечисленных ниже устройств. Установка регистров состояний допустима для следующих устройств: RK, DX, DY:, DU:, DD: Модифицирует непосредственно в драйвере устройства адрес регистра состояния (устанавливая его равным K) для второго контроллера. Данная установка допустима для перечисленных выше устройств, кроме RK:, если при генерации системы создается драйвер, обслуживающий два контроллера
NN:	RETRY=K	Определяет число попыток (K) выполнить операцию, если во время этой операции имела место ошибка (при этом должен выполняться регистратор ошибок). Аргумент K — целое число 1 до 8. По умолчанию K=8
NN:	<u>SUCCESS</u>	После данной установки регистратор ошибок будет выполнять регистрацию как успешных операций ввода-вывода, так и ошибок
NN:	NOSUCCESS	После данной установки регистратор ошибок будет выполнять регистрацию только ошибок

1	2	3
NN:	VECTOR=K	Модифицирует, непосредственно в драйвере, адрес вектора (устанавливая его равным K) для первого контроллера. Аргумент K — целое восьмеричное число от 100 до 474
NN:	VEC2=K	Модифицирует непосредственно в драйвере адрес вектора (устанавливая его равным K) для второго контроллера. Данная установка допустима, если при генерации системы создается драйвер, обслуживающий два контроллера
DXN:	WRITE	Разрешает запись для привода N устройства DX:. Система возвращается в это состояние после повторной загрузки системы, если DXN: не является системным устройством при подаче команды. По умолчанию N=0
DXN:	NOWRITE	Запрещает запись для N-ого привода устройства DX:. Система возвращается в это состояние после повторной загрузки системы, если DXN: не является системным устройством при подаче команды. По умолчанию N=0
DYN:	WRITE	Разрешает запись для привода N устройства DY:. Система возвращается в это состояние после повторной загрузки, если DYN: не является системным устройством при подаче команды. По умолчанию N=0
DYN:	NOWRITE	Запрещает запись для N-ого привода устройства DY:. Сис-

1	2	3
EDIT	EDIT	тема возвращается в это состояние после повторной загрузки, если DYN: не является системным устройством при подаче команды. По умолчанию $N=0$ Вызывает редактор текста EDIT по команде КЯС EDIT. Система возвращается в это состояние после повторной загрузки
EDIT	K52	Вызывает экранный редактор текста K13; допустим, если используется терминал 15ИЭ—00—013
EL:	LOG	Используется, если запущен регистратор ошибок под управлением монитора одного задания. После этой установки запускается регистратор ошибок, если загружен драйвер EL, и начинается регистрация ошибок во внутренний буфер драйвера EL
EL:	NOLOG	Используется, если запущен регистратор ошибок под управлением монитора одного задания для прекращения действия регистратора ошибок. Система возвращается в это состояние после повторной загрузки
EL:	PURGE	Используется, если запущен регистратор ошибок под управлением монитора одного задания, для удаления содержимого внутреннего буфера регистратора ошибок. Данная

1	2	3
ERROR	ERROR -----	установка допустима после команды SET EL: LOG Прекращает обработку косвенного файла и команд КЯС, включающих сложные операции, если произошла ошибка или неустраняемая ошибка при выполнении команды (например, EXECUTE, которая объединяет операции транслирования, редактирования и выполнения), т. е. произошла ошибка, которая имеет признак —Е—, —F— или —U— в сообщении об ошибке Примером ошибки является неопределенный символ при трансляции. Пример неустраняемой ошибки — попытка записи на устройство, если это устройство запрещено для записи. Если произошла одна из этих ошибок, выполнение косвенного файла или команды КЯС прерывается, и управление передается монитору. Система возвращается в это состояние после повторной загрузки
ERROR	FATAL	Вызывает предусмотренное прерывание косвенного файла. или команд КЯС при появлении неустраняемой ошибки, которая имеет признак —Е—, —F— или —U—
ERROR	NONE	Позволяет продолжать выполнение косвенных файлов и команд КЯС, даже если при

1	2	3
ERROR	SEVERE	их выполнении происходят ошибки. Большинство неустраняемых ошибок, которые имеют признак —U— в сообщении об ошибках, однако, вызывают предусмотренное прерывание косвенного файла или команд КЯС Вызывает предусмотренное прерывание косвенного файла или команд КЯС при появлении неустраняемой ошибки, которая имеет признак —F— или —U— в сообщении об ошибке
ERROR	WARNING	Вызывает предусмотренное прерывание косвенного файла или команд монитора или неустраняемой ошибки, которые имеют признак —W—, —E—, —F— или —U— в сообщении об ошибке
EXIT	SWAP ----	Записывает часть программы, которая хранилась в SWAP.SYS, обратно в память после завершения выполнения программы
EXIT	NOSWAP	Запрещает запись части программы, которая хранилась в SWAP.SYS, обратно в память после завершения выполнения программы. Во время действия данной команды программа не может быть запущена с адреса повторного пуска
KMON	IND	Файл, указанный в строке команды Эспф, будет выполняться как управляющий файл.

1	2	3
KMON	NOIND -- --	Файл, указанный в строке команды X Эспф, будет выполняться как косвенный командный файл Файл, указанный в строке команды Эспф, будет выполняться как косвенный командный файл. При попытке выполнить управляющий файл происходит ошибка
LD	CLEAN	Проверяет и исправляет, если необходимо, соответствие текущего логического диска определенному файлу. Эта команда используется после изменения положения файлов на томе или после удаления тома с устройства. Если файл логического диска меняет свое положение, пользователь может по-прежнему использовать логический диск. Если удаляется файл логического диска или удаляется том, содержащий файл логического диска, отменяется соответствие файлу логического диска (во втором случае отменяется соответствие временно). Чтобы повторно установить соответствие при повторной установке тома, следует подать команду SET LD CLEAN. Команды KЯC SQUEEZE и BOOT автоматически выполняют команду SET LD CLEAN
LDN:	WRITE	Используется при установке тома для разрешения чтения и записи на логический диск

1	2	3
LDN:	NOWRITE	с номером N. Аргумент N — целое число от 0 до 7 Используется при установке тома для запрещения записи на логический диск с номером N, т. е. на логическом диске разрешено только чтение. Аргумент N — целое число от 0 до 7
LP:	CR	Разрешает передачу символа возврат каретки на LP:
LP:	<u>NOCR</u>	Запрещает передачу символа возврат каретки на LP:. Это состояние существенно увеличивает скорость печати на LP:, т. к. после печати каждой строки на LP: выполняется только перевод строки
LP:	CSR=N	Модифицирует непосредственно в драйвере LP: адрес регистра состояния (устанавливая его равным N) для контроллера LP:. Значение N — четный восьмеричный адрес не меньше 160000
LP:	<u>FORM</u>	Выполняет аппаратный перевод формата
LP:	NOFORM	Имитирует код перевод формата несколькими кодами перевода строки. Используется вместе с установкой LENGTH=N
LP:	FORM0	Выполняет перевод формата перед печатью нулевого блока
LP:	<u>NOFORM0</u>	Запрещает режим FORM0
LP:	<u>HANG</u>	Ожидает, когда будет установлена готовность LP:, если LP: не готово для печати или про-

1	2	3
LP:	NOHANG	изошел сброс готовности во время печати. Если ожидается вывод на LP:, а система работает вхолостую, следует проверить наличие бумаги и готовность LP:
LP:	LC	Если не установлена готовность LP:, немедленно вырабатывается ошибка
LP:	NOLC	Разрешает посылать коды нижнего регистра на LP:. Используется, если LP: имеет набор символов нижнего регистра
LP:	-----	Преобразует коды нижнего регистра в коды верхнего регистра перед печатью
LP:	LENGTH=N	Указывает число строк в странице (по умолчанию N=66). Используется вместе с NOFORM и SKIP=N
LP:	SKIP=M	Выполняет M кодов перевода строки между страницами так, чтобы сумма M и числа строк с информацией в странице была равна N ($0 \leq M \leq N$). Если M=0 страницы не разделяются кодами перевода строки
LP:	TAB	Посылает код табуляции на LP:
LP:	NOTAB	Имитирует код TAB несколькими кодами <ПР>
LP:	-----	Модифицирует непосредственно в драйвере адрес вектора (устанавливая его равным N) для контроллера LP:. Аргумент N — целое восьмеричное число меньше 500
LP:	VECTOR=N	

1	2	3
LP:	WIDTH=N	Устанавливает длину строки, N, для LP: аргумент, N, — любое целое число от 30 до 255 включительно. Система игнорирует все коды, порядковый номер которых в строке больше N
MT:	ODDPAR	Устанавливает паритет по нечетности (проверку на четность). Рекомендуется именно этот режим
MT:	NOODDPAR	Устанавливает паритет на четность
SL:	ON	Загружает и дает возможность работать с SL
SL:	OFF	Разгружает и делает невозможным работу с SL
SL:	SYSGEN	Преобразует SL.SYS (под управлением монитора SJ и FB) или SLX.SYS (под управлением монитора XM) для соответствия его текущему монитору
SL:	TTYIN	Позволяет редактировать ответы на опрашивающие сообщения. Когда действует эта установка, подсказки печатаются на одной строке, ответы на следующей
SL:	<u>NOTTYIN</u>	Запрещает редактировать ответы на опрашивающие сообщения. Когда действует эта установка, подсказки и ответы на них печатаются на одной строке
SL:	VT13	Сообщает тип используемого терминала. Используется для работы на терминале 15ИЭ—00—013.

1	2	3
SL:	WIDTH=N	Устанавливает длину строки терминала, N — максимальное число символов строки. Максимальная допустимая длина строки терминала равна $N-K-1$ где K — длина строки подсказки, куда входит точка монитора (.). Например, если подается команда SET SL: WIDTH=50 и подсказка — точка монитора, то максимальная длина равна $50-1-1=48$ символов. По умолчанию N=79
TT:	CONSOL=N	Устанавливает в качестве системного терминала (консоль) с логическим номером N. По умолчанию N=0. Терминал, чей логический номер указывается, не должен обслуживать основное или системное задание. Система возвращается в это состояние после повторной загрузки
TT:	CRLF	Система выполняет возврат каретки/перевод строки на терминале после печати допустимого числа символов. Изменить длину строки можно с помощью команды SET TT: WIDTH=N. Данная установка недопустима для монитора SJ, не имеющего мультитерминальной поддержки. Система возвращается в это состояние после повторной загрузки
TT:	NOCRLF	Система не выполняет возврат каретки/перевод строки на тер-

1	2	3
TT:	FB	минале после ввода допустимого числа символов. Данная установка недопустима для монитора SJ, не имеющего мультитерминальной поддержки. Рассматривает СУ/В и СУ/Ф (и СУ/Х в мониторах, которые поддерживают системное задание) как коды управления основной и фоновой программой и не передает их в программу пользователя. Данная установка недопустима для монитора SJ. Система возвращается в это состояние после повторной загрузки.
TT:	NOFB	Система отменяет действие управляющих кодов СУ/В и СУ/Ф (и СУ/Х в мониторах, которые поддерживают системное задание). Данная установка недопустима для монитора SJ.
TT:	FORM	Выполняет аппаратный перевод формата. Данная установка недопустима для монитора SJ, не имеющего мультитерминальной поддержки.
TT:	NOFORM	Имитирует перевод формата посредством выполнения восьми символов перевода строки. Эта установка недопустима для монитора SJ, не имеющего мультитерминальной поддержки. Система возвращается в это состояние после повторной загрузки.

1	2	3
TT:	<u>PAGE</u>	Рассматривает коды управления CU/S , CU/Q как коды, приостанавливающий и возобновляющий вывод на терминал. Система возвращается в это состояние после повторной загрузки. Данная установка недопустима для монитора SJ, не имеющего мультитерминальной поддержки
TT:	NOPAGE	Отменяет действие управляющих кодов CU/S и CU/Q . Данная установка недопустима для монитора SJ, не имеющего мультитерминальной поддержки. Система возвращается в это состояние после повторной загрузки
TT:	QUIET	Предотвращает печать строк косвенного файла. Система возвращается в это состояние после повторной загрузки
TT:	<u>NOQUIET</u>	Происходит печать строк косвенного файла. Система возвращается в это состояние после повторной загрузки
TT:	SCOPE	Нажатие клавиши <ЗБ> отражается на терминале как <ПР> стертый знак <ПР> Система возвращается в это состояние после повторной загрузки
TT:	<u>NOSCOPE</u>	Нажатие клавиши <ЗБ> отражается на терминале как /стертый знак/.

1	2	3
TT:	WIDTH=N	Система возвращается в это состояние после повторной загрузки Устанавливает длину строки терминала равной N. Аргумент, N,— целое число от 30. До 255. Первоначально N=80. Эта установка недопустима для монитора SJ, не имеющего мультитерминальной поддержки. После повторной загрузки N=80
VM	BASE=NNNNNN	Определяет начальный адрес 0-ого блока виртуального диска (адрес базы). Т. к. адрес базы — 22-х разрядный адрес — представляется 16-ти разрядным числом, младшие шесть разрядов (0—5) всегда равны 0. По умолчанию NNNNNN=1600 под управлением мониторов SJ или FB 10000 и под управлением монитора ХМ
WILD	EXPLICIT	Заставляет систему воспринимать спецификацию файла в том виде, в каком она была напечатана пользователем. Система не заменяет пропущенные части спецификации файла символом *. Использование символов * и % описано в п. 2.2
WILD	IMPLICIT	Система воспринимает пропущенные части спецификации файла как символ * (см. табл. 2), где объясняется, как система интерпретирует команды с пропущенными частями

1	2	3
USR	SWAP ---	спецификации файла). Это режим, в который система возвращается после повторной загрузки Система допускает процесс свопинга для USR. Эта установка недопустима для монитора ХМ. Система возвращается в это состояние после повторной загрузки
USR	NOSWAP	Не допускает процесс свопинга для USR. Эта установка недопустима для монитора ХМ

Примечание. Все команды, изменяющие параметры драйвера LP:, имеют аналогичное действие и на драйвер устройства SL:.

Чтобы указать несколько значений параметров в строке команды, надо отделить значения запятыми. За исключением команд SET TT:, SET USR и SET элконфгр, по команде SET система модифицирует драйвер SY:фимуст.SYS таким образом, что изменения (не все, однако) остаются в силе даже после перезагрузки драйвера. Для команд SET TT:, SET USR, SET элконфгр изменения действуют до первой перезагрузки драйвера. Если драйвер уже загружен в память перед использованием команды SET, надо разгрузить этот драйвер (кроме драйвера для TT:). Символ двоеточие (:) является необязательным для имени устройства.

Примеры:

1. .SET LP: LC

данная команда позволяет посылать символы нижнего регистра на LP:.

2. .SET WILD IMPLICIT

после данной команды система воспринимает пропущенные части спецификации файла как символ *.

4.45. По команде SHOW на терминале печатается требуемая информация о ФОДОС.

Формат команды: SH[OW] [типинф[/прк]]
 где **типинф** — параметр, указывающий, какую информацию надо распечатать (табл. 34);
 /прк — один или несколько переключателей параметра ERRORS.

Таблица 34

ПАРАМЕТРЫ КОМАНДЫ SHOW

Параметр (типинф)	Функция
1	2
ALL	Объединяет действие параметров CONFIGURATION, DEVICES, все назначения устройств (действие команды SHOW без параметров), JOBS, TERMINALS, MEMORY и SUBSET в указанном порядке
CONFIGURATION	Печатает на терминале номер версии монитора, установки команды SET, элементы аппаратной конфигурации и общий объем памяти системы
DEVICES[:DD]	Выдает имена устройств, их состояние, адреса регистров состояний и векторов. Используя аргумент DD, можно получить информацию только об указанном устройстве
ERRORS	Вызывает одну из трех программ пакета регистрации ошибок (ERRROUT [см. 4]), которая выполняется как фоновое задание, для создания и вывода сообщений о вводе-выводе и статистике об ошибках на GT:, LP: или на любой том файлом, допустим, если используемый монитор поддерживает ERRROUT. Переключатели, используемые с ERRORS: ERRORS — печатает сообщение о каждой операции ввода-вывода, а также о каждой ошибке ввода-вывода, четности памяти и кэш памяти.

ERRORS/ALL — действие этой команды такое же как у команды SHOW ERRORS<BK>.

ERRORS/FILE:спф — печатает указанный файл, содержащий сообщения об операциях ввода-вывода и об ошибках. Файл должен иметь формат .DAT по умолчанию понимается файл ERRLOG.DAT.

ERRORS/FROM:[дата] — печатает сообщения об операциях ввода-вывода и об ошибках, которые произошли после указанной даты.

Аргумент, дата, имеет формат — DD:MMM:YY,

где DD — 2 цифры числа (десятичные);

MMM — первые три символа имени месяца;

YY — последние две цифры года.

ERRORS/TO[:дата] — печатает сообщения об операциях ввода-вывода и об ошибках, которые произошли до указанной даты.

ERRORS/OUTPUT:спф — выводит все сообщения в виде файла на указанное устройство.

ERRORS/PRINTER — печатает сообщения на LP:.

ERRORS/SUMMARY — печатает итоговое сообщение об ошибках (сообщения об успешных передачах ввода-вывода не печатаются)

ERRORS/TERMINAL — печатает сообщения на TT:

Печатает следующую информацию о заданиях, которые загружены в текущий момент:

— номер и имя задания;

JOBS

1	2
	— номер терминала, закрепленного за данным заданием (для немультитерминального монитора — 0); — уровень приоритета задания; — состояние задания (RUN, SUSPEND и DONE) где RUN — задание выполняется; SUSPEND — выполнение задания приостановлено; DONE — завершено, но еще не разгружено; — верхний и нижний пределы памяти, используемые заданием; — начальный адрес смешанной области памяти задания Печатает список организации оперативной памяти: адрес загрузки заданий, драйверов, KMON, USR и количество слов памяти, занятое каждым из них. Адреса памяти воспроизводятся как восьмеричные числа. Если команда SHOW MEMORY подается под управлением монитора ХМ, список организации памяти печатается двумя секциями: первая — для расширенной памяти, вторая — для памяти ядра (до 32K слов) Печатает содержимое очереди: имя устройства, имя задания, имя и тип входных файлов, состояние задания (Q — ожидает вывода, P — выводится в настоящий момент), число копий вывода каждого задания Печатает информацию о логическом диске: имя логического диска и имя соответствующего файла, размер в блоках (десятичное число) логического диска. Если за информацией о ло-
MEMORY	
QUEUE	
SUBSET	

1	2
TERMINAL	гическом диске следует звездочка (*), значит соответствующий файл не существует на установленном томе; если следует знак номер (#), значит драйвер устройства не загружен. Если LD.SYS не установлен в системе, система печатает: драйвер LD не доступен если по команде MOUNT логическому диску не был поставлен в соответствие файл, система печатает: устройство LD не выделено Если монитор включает мультитерминальную поддержку на терминале печатается следующая информация в виде таблицы: — номер терминала: 0—15; — принадлежность терминала: основному, фоновому, системному заданию; — тип терминала: локальный, удаленный, консоль, S — консоль (совместно используемая фоновым, основным или системным заданиями); — длина строки: до 255 символов; — установки команды SET для : TAB, CRLF, FORM, SCOPE; — скорость печати: в бодах, N/A, если постоянна

Информация, печатаемая по команде SHOW, включает некоторые данные аппаратной конфигурации, номер версии монитора, общий объем памяти системы и организация памяти, физические и логические имена устройств, характеристики терминалов для мультитерминальной системы, информацию о логических дисках и состоянии драйверов устройств. Если монитор поддерживает ERRORT и QUEUE, то по команде SHOW можно распечатать статистику ошибок и очередь файлов, ожидающих вывода на определенное устройство.

Если команда SHOW не содержит переключателей, то на терминале печатается список имен всех устройств, доступных монитору, который работает в текущий момент. В этом списке происходят изменения после выполнения команд INSTALL и REMOVE, т. е. после ввода или удаления из системы каких-либо устройств. Листинг, полученный по команде SHOW, включает дополнительную информацию об определенных устройствах.

Информационные сообщения и их значения следующие:

(RESORC) или==RESORC	указывает, что устройство или привод назначен фоновому заданию (для мониторов FB или XM).
(F) или=F	указывает, что устройство или привод назначены основному заданию (для мониторов FB и XM, которые не поддерживают системное задание).
(Имя задания) или=имя задания	указывает, что устройство или том назначены системному или основному заданию (для мониторов FB и XM, которые поддерживают системное задание), где имя задания — имя системного или основного задания.
(Загружен)	указывает, что драйвер устройства был загружен в память по команде LOAD.
(Резидентный)	указывает, что драйвер устройства включен в резидентную часть монитора (RMON).
=лог.Имя (1), =лог.Имя (1), ... =лог.Имя (N)	указывает, что указанное логическое имя было присвоено устройству или тому по команде ASSIGN.
XX свободных мест	указывает, сколько (XX) ячеек таблиц монитора не используются.

Примеры:

1. В данном примере по команде SHOW печатается список всех устройств системы, обслуживаемых монитором, который поддерживает системное задание.

SHOW

TT (резидентный)

RK (резидентный)

RK1=SY,DK,OBJ,SRC,BIN

RK2=LST,MAP

DX (загружен)
DX0: (MYPROG)
DX1: (RESORC)
LP: (загружено=QUEUE)
MT
5 свободных мест

Из данного листинга видно, что TT.SYS и RK.SYS — резидентные драйверы. Система обслуживает также устройства: DX, LP и MT. В таблицах монитора есть 5 свободных ячеек для включения в систему пяти устройств. По команде ASSIGN были введены следующие логические имена: для RK1: — SY, DK, OBJ, SRC, BIN; для RK2: — LST и MAP. По команде LOAD в память был загружен драйвер DX.SYS; том DX0: используется основным заданием MYPROG; DX1: — фоновым заданием. Драйвер LP.SYS также загружен по команде LOAD и используется системным заданием QUEUE.

2. SHOW CONFIGURATION

ФОДОС V03.00

Загрузка с RK0:FMONXM

команда: SET USR SWAP

команда: SET EXIT SWAP

команда: SET KMON IND

команда: SET TT QUIET

команда: SET ERROR ERROR

глубина вложения 3

процессор MC 1601

64KB памяти

расширенная арифметика (PA)

проверка четности памяти

мультитерминальная поддержка

в данном листинге указаны:

— номер версии монитора;

— устройство, с какого была загружена система;

— установки команды SET (SWAP для USR, SWAP для EXIT, IND для KMON, NOQUIET для TT:, ERROR для ERROR);

— уровень вложения косвенного файла равен 3;

— тип процессора MC 01601;

— 64К байт памяти;

— расширенная арифметика (PA);

— проверка четности памяти;

— мультитерминальная поддержка текущего монитора.

3. .SHOW DEVICES

DEVICE	состояние	CSR	VECTOR
DX	установлен	177170	264
RK	резидентный	177400	220
LP	установлен	177514	200
NL	установлен	000000	000
DY	122620	177170	264
MT	установлен	172520	224

по данной команде печатаются имена устройств, адреса регистров состояний и адреса векторов этих устройств и загружен или нет драйвер в текущий момент.

4. .SHOW MEMORY

расширенная память

адрес	модуль	слова
01000000	VM	393216.
00160000	...	102 400.

Ядро памяти

адрес	модуль	слова
157350	RK	140.
124144	RMON	6970.
122612	DY	365.
111610	USR	2305.

В данном примере печатается список организации памяти, т. е. адреса загрузки драйверов, RMON, USR и объем памяти, занимаемый каждым из них.

5. .SHOW QUEUE

DEVICE	JOB	STATUS	COPIES	FILES
LP0:	LAB2	P	1	PASS3.LST
			2	PASS4.LST
			2	PASS5. LST
LP0:	HODG	Q	3	MESMAN.DOC

В данном примере печатается содержимое очереди. Задание LAB2 в настоящий момент выводится на LP0.; HODG — ожидает вывода.

6. .SHOW SUBSET

LD0—DY:DISK.LST[4000.]
LD2 — DY1:DISK.SRC[1200.]

Из данного примера логическому диску LD0 соответствует файл DY:DISK.LST (размер файла 4000. блоков); логиче-

скому диску LD2 соответствует файл DY1.DISK.SRC (размер файла 1200.).

4.46. По команде SQUEEZE все неиспользуемые блоки указанного тома объединяются в одну область, а также объединяются записи справочника, т. е. том сжимается.

Формат команды: SQ[UEEZE][/прк] уст

где /прк — один или несколько переключателей;

уст — имя тома, который должен быть сжат.

Таблица 35

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ SQUEEZE

Переключатель	Функция
1	2
/OUTPUT:уст	Сжимает диск на другой диск, где уст — устройство вывода. Эта операция оставляет диск ввода неизменным. В качестве диска вывода может использоваться только инициализированный диск (команда INITIALIZE используется для этого). Если диск вывода не был инициализирован, система печатает сообщение об ошибке, и команда не выполняется. /OUTPUT не копирует блоки загрузчика системы (команда COPY/BOOT используется для этого) и не печатает на терминале опрашивающие сообщения перед выполнением команды
/QUERY	Перед выполнением команды система опрашивает пользователя, надо ли выполнять указанную операцию. Следует подать Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции. Использование сразу двух переключателей /OUTPUT:уст и /QUERY недопустимо
/NOQUERY	Запрещает печать на терминале опрашивающих сообщений перед выполнением команды

1	2
/WAIT	Используется, если конфигурация вашей вычислительной системы содержит один привод. После подачи переключателя /WAIT, система инициирует выполнение команды, но затем приостанавливается и печатает: MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя тома устройства, который надо сжать. Пользователь должен установить нужный том и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции. Для прекращения выполнения операции следует подать N (или любую строку, начинающуюся с N) или два раза CY/C. После завершения операции система печатает: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? Следует заменить том, который сжимается, системным и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ вызывает повторную печать последнего сообщения

При выполнении операции сжатия система передвигает все файлы тома в начало, получая одну неиспользованную область после всех файлов. По команде SQUEEZE не изменяется содержимое блоков загрузчика системы.

В операцию сжатия не включаются файлы с типом .BAD. Это позволяет избежать повторного использования плохих блоков диска. Система передвигает файлы тома до и после .BAD файла до тех пор, пока объем памяти между последним передвинутым файлом и файлом .BAD будет меньше очередного передвигаемого файла. После завершения операции сжатия на системном томе, система автоматически повторно загружается.

Операция сжатия недопустима во время выполнения основного задания.

Пример:

.SQUEEZE/WAIT RK0:

RK0:/SQUEEZE;ARE YOU SURE?Y

MOUNT INPUT VOLUME IN RK0:;CONTINUE?Y

MOUNT SYSTEM VOLUME IN RK0:;CONTINUE?Y

по данной команде сжимается диск RK0:.

4.47. По команде SRUN инициируется выполнение системного задания.

Формат команды:

SRUN спф[/прк]

где спф — спецификация файла системного задания;

/прк — один или несколько переключателей команды SRUN.

Таблица 36

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ SRUN

Переключатель	Функция
1	2
/BUFFER:N	Резервирует дополнительно «N» слов сверх объема программы. Используется для выполнения программы на языке ФОРТРАН как системного задания. Назначает уровень приоритета заданию ($1 \leq N \leq 6$). При попытке назначить одинаковый приоритет двум заданиям на терминале печатается сообщение об ошибке. Если N не указано, заданию назначается самый высокий неназначенный уровень приоритета.
/LEVE:N	
/PAUSE	
	Печатает адрес загрузки и затем приостанавливает выполнение системного задания. Во время приостановки пользователь может проверить или модифицировать программу посредством ODT. Для

1	2
/TERMINAL:N	возобновления выполнения задания используется команда RESUME Назначает системному заданию терминал с логическим номером N. Используется, если система имеет мультитерминальную поддержку

Одновременно с основным и фоновым заданиями могут выполняться до шести системных заданий. При попытке запустить системное задание, которое уже выполняется, на терминале печатается сообщение об ошибке. Если в спецификации файла не указан тип, система предполагает — .REL.

Под управлением монитора XM команда SRUN инициирует выполнение виртуального файла с типом .SAV.

Пример:

```
.SRUN MFUNCT/PAUSE
```

```
LOADED AT 126556
```

```
.RESUME MFUNCT
```

4.48. По команде START инициируется выполнение текущей программы (загруженной по команде GET) с указанным адреса.

Формат команды:

ST[ART] адр

где адр — восьмеричный 16-разрядный адрес. Если адрес опущен или равен 0, система использует пусковой адрес из ячейки 40.

Если указанный адрес не существует или недопустим, происходит прерывание по 4-му вектору. Команда START допустима только для фоновых заданий.

Примеры:

```
1. .GET PROG
```

```
.START
```

По данным командам система загружает файл PROG.SAV в оперативную память и инициирует его выполнение.

```
2. .GET PROG
```

```
.GET ODT
```

```
.START
```

По данным командам в оперативную память загружаются PROG.SAV и ODT.SAV и инициируется их выполнение с пускового адреса ODT.

4.49. По команде SUSPEND система приостанавливает выполнение основного или фонового задания.

Формат команды: SUS[PEND][имязд]

где имязд — имя основного или системного задания.

Если монитор поддерживает системное задание, пользователь может указать имя системного или основного задания, выполнение которого надо приостановить.

Если монитор не поддерживает системное задание, то в команде не допускается использование аргументов. Команда SUSPEND недопустима для монитора SJ. Система позволяет завершить операции ввода-вывода основного задания, если во время подачи команды SUSPEND эти операции выполнялись. Однако система отложит выполнение новых запросов ввода-вывода и подпрограмм завершения. Продолжить выполнение основного задания можно по команде RESUME.

Примеры:

1. .SUSPEND

по этой команде приостанавливается выполнение основного задания, которое выполняется в настоящий момент под управлением монитора, не поддерживающего системное задание.

2. .SUSPEND QUEUE

по этой команде приостанавливается выполнение системного задания QUEUE.

4.50. Команда TIME. По команде TIME система устанавливает время суток или печатает текущее время суток.

Формат команды: TI[ME][чч:мин:сек]

где чч — час (целое число от 0 до 23);

мин — минуты (целое число от 0 до 59);

сек — секунды (целое число от 0 до 59).

Система использует 24-часовой таймер. Рекомендуется устанавливать таймер на определенное время сразу после загрузки системы. Если пропущен один из аргументов, система использует ноль. Мониторы FB и XM автоматически сбрасывают таймер каждые сутки в полночь. Это верно и для монитора SJ, но сгенерированного с поддержкой таймера. Для печати текущего времени дня подается команда TIME без аргументов.

Примеры:

1. TIME 11:15

по данной команде таймер устанавливается на 11:15:00.

2. TIME

11:15:02

по этой команде система распечатывает текущее время дня.

4.51. Команда TYPE предназначена для печати на терминале содержимого одного или более файлов.

Формат команды: TYPE[/прк] спф[, ... спф]

где /прк — один или несколько переключателей;

спф — спецификация распечатываемого (входного) файла.

Таблица 37

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ TYPE

Переключатель	Функция
1	2
/BEFORE[дата]	Печатает на терминале все файлы, созданные до указанной даты
/COPIES:N	Делает N копий файла на ТТ.; N — целое число от 2 до 32 (N=1 по умолчанию)
/DATE[дата]	Печатает на терминале все файлы с указанной датой создания
/DELETE	Удаляет указанный файл на томе после печати его содержимого на ТТ.: Перед удалением файла система не опрашивает пользователя о включении в указанную операцию файла
/INFORMATION	Используется, если пользователь не уверен, есть ли на томе указанные в строке команды файлы. Если указанный файл не найден — печатается информационное сообщение об ошибке, но выполнение операции не прекращается — печатаются найденные файлы
/LOG	Печатает на терминале список файлов, которые были включены в текущую операцию. Обычно система печатает список файлов, если в строке команды спецификация файла содержит символы % и *. Если указать /QUERY, то систе-

1	2
/NOLOG /NEWFILES /QUERY /SINCE[дата] /WAIT	ма будет опрашивать пользователя о включении в указанную операцию каждого файла, но не будет печатать список всех файлов, участвовавших в этой операции. Чтобы происходил опрос пользователя и печать списка, в строке команды должны присутствовать переключатели /LOG/QUERY. Запрещает печать списка файлов, которые были распечатаны по текущей команде. Этот переключатель используется для запрещения печати списка файлов, если в строке команды спецификация файла содержала символы % и *. Печатает содержимое файлов, имеющих текущую дату создания Опрашивает пользователя о включении в указанную операцию каждого файла. Особенно полезен в командах, у которых спецификации файлов содержат символы % и *. Для начала выполнения операции следует напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки. Система интерпретирует другой ответ как NO и не выполняет указанную операцию Печатает на терминале все файлы, созданные в течение или после указанной даты Используется, если конфигурация вычислительной системы содержит один привод. После подачи переключателя /WAIT, система иницирует выполнение команды, но затем приостанавливается и печатает:

1	2
	MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя устройства, куда устанавливается том. Пользователь должен установить нужный том и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции. Печать N (или любой строки, начинающейся с N) и возврат каретки или двух CY/C прерывает выполнение операции и управление возвращается монитору. Любой другой ответ вызывает повторную печать сообщения. После завершения печати файла система печатает: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? Следует заменить входной том системным и напечатать Y (или любую строку, начинающуюся с Y) и возврат каретки для завершения операции TYPE. Любой другой ответ вызывает повторную печать последнего сообщения

Спецификации входных файлов (допустимо указывать не более 6) отделяются запятыми в строке команды. Система печатает содержимое файлов в том порядке, в каком они указаны в строке команды. Допустимо использовать символы * и % в спецификации файла. В этом случае система печатает файлы в том порядке, в каком они записаны на томе. По умолчанию тип входного файла .LST.

Команда TYPE может быть задана в виде полной строки или краткой. После ввода краткой строки на терминале печатается: FILES? Необходимо ввести с терминала спецификацию входного файла и возврат каретки.

Некоторые переключатели допускают в качестве своего аргумента дату. Синтаксис, определяющий дату, следующий:
[:dd][:mmm][:rr]

где **дд** — день (десятичное число от 1 до 31);

ммм — месяц (первые три буквы названного месяца);

гг — год (десятичное число от 82 до 99).

Значение даты по умолчанию соответствует текущей системной дате. Если опущена любая часть даты (**дд,ммм,гг**) система использует соответствующую часть текущей системной даты. Например, если указать только **:83** и текущая системная дата **4 MAY 1984**, система использует **4:MAY:83**. Если текущая системная дата не установлена, она считается нулевой.

Примеры:

1. **.TYPE/COPIES:3 REPORT**

по данной команде на **TT**: печатается три копии файла **REPORT.LST**.

2. **.TYPE/LOG/DELETE REPORT**

по данной команде на **TT**: печатается файл **REPORT.LST**, затем этот файл удаляется с **DK**: и на терминале печатается имя удаленного файла.

3. **.TYPE/NEWFILES *.LST**

по данной команде на **TT**: печатаются все файлы, имеющие текущую дату создания и тип **.LST**; после чего на терминале будет напечатан список этих файлов.

4.52. Команда UNLOAD. По команде **UNLOAD** драйверы, ранее загруженные, удаляются из памяти, освобождая занятую ими память. По этой команде удаляются из памяти также завершенные основные или системные задания.

Формат команды: **UNLOAD уст[,...уст]**

или

UNLOAD имязд[,...имязд]

где **уст** — имя устройства, драйвер которого будет удален из памяти;

имязд — имя файла основного или системного задания.

Для мониторов, которые не поддерживают системное задание, двоеточие за именем устройства в строке команды **необязательно**. Для мониторов, которые поддерживают системные задания, двоеточие должно обязательно следовать за именем устройства в строке команды. Если двоеточие не указано, система предполагает, что подано имя задания.

Команда **UNLOAD** отменяет также принадлежность заданиям всех приводов указанных в команде устройств. Команда **UNLOAD** для драйвера системного устройства отменяет принадлежность заданиям всех приводов системного устройства, оставляя сам драйвер резидентным в памяти. Если бы-

ли прервана операция с МТ:, необходимо удалить из памяти драйвер этого устройства, а затем повторно его загрузить.

Нельзя использовать команду UNLOAD для устройства, один из приводов которого принадлежит выполняющемуся в текущий момент основному заданию. Можно удалить из памяти драйвер устройства, если ни один из приводов этого устройства не принадлежит выполняющемуся в данный момент основному заданию.

Команда UNLOAD также удаляет из памяти завершенное основное или системное задание и разрешает использовать область памяти, занимаемую этим заданием.

Примеры:

1. UNLOAD F

по этой команде удаляется из памяти основное задание, если его выполнение завершено.

2. UNLOAD QUEUE

по этой команде из памяти удаляется системное задание QUEUE.

3. UNLOAD RK:,LP:

по данной команде отменяется принадлежность заданиям всех приводов RK:, но т. к. RK: — системное устройство, драйвер RK.SYS остается резидентным в памяти. По данной команде удаляется из памяти также драйвер LP.SYS.

4.53. Команда UNPROTECT. По данной команде отменяется защита от удаления указанного файла, т. е. файл может быть удален после указанной команды.

Формат команды:

UNP[ROTECT][/прк] спф[, ..., спф]

где /прк — один или несколько переключателей команды UNPROTECT;

спф — спецификация файла, для которого отменяется защита от удаления.

Таблица 38

ПЕРЕКЛЮЧАТЕЛИ КОМАНДЫ UNPROTECT

Переключатель	Функция
1	2
/BEFORE[дата] *	Отменяет защиту файлам, созданным до указанной даты. Если аргумент, да-

1	2
/DATE[дата] * /EXCLUDE /INFORMATION /LOG /NOLOG /NEWFILES * /QUERY	та, не указан, используется системная текущая дата Отменяет защиту файлам с указанной датой создания Отменяет защиту всем файлам тома, кроме указанных Используется, если пользователь не уверен, есть ли на томе указанные в строке команды файлы. Если указанный файл не найден — печатается информационное сообщение об ошибке, но выполнение операции не прекращается — отменяется защита для найденных файлов Печатает на терминале список файлов, для которых отменена защита по текущей команде. Данный список печатается по умолчанию, если спецификация файла содержит конструкцию * и %. Если указан /LOG, система не опрашивает пользователя о включении каждого файла в операцию Запрещает печать на терминале списка файлов, для которых отменена защита по текущей команде Отменяет защиту файлам, имеющим текущую дату создания Опрашивает пользователя о включении в указанную операцию каждого файла. Этот переключатель полезен, если спецификация файла содержит конструкцию * и %. Пользователь должен напечатать на терминале: — Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции;

1	2
	— NO (или любой другой ответ) и возврат каретки, если файл исключается из операции /SETDATE[дата] * Присваивает указанную дату всем файлам, для которых отменяется защита от удаления. Обычно, при выполнении операции система сохраняет дату создания файла /SINCE[дата] * Отменяет защиту для файлов, созданных в течение или после указанной даты /SYSTEM Отменяет защиту для системных файлов (.SYS). Без этого переключателя системные файлы исключаются из операции и печать сообщений напоминает об этом /WAIT Используется, если конфигурация вычислительной системы содержит один привод. После подачи строки команды с переключателем /WAIT, система инициирует выполнение команды, но затем приостанавливается и печатает: MOUNT INPUT VOLUME IN <УСТ>;CONTINUE? где <УСТ> — имя тома устройства. Следует установить том, для файлов которого надо отменить защиту, и напечатать: — Y (или любую строку, начинающуюся с Y) и возврат каретки для выполнения операции; — N (или любую строку, начинающуюся с N) и возврат каретки или два раза CУ/С для прерывания операции, и управление возвращается монитору;

1	2
	— любой другой ответ вызывает повторную печать сообщения. После завершения операции система печатает: MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? следует установить системный том и подать Y (или любую строку, начинающуюся с Y) и возврат каретки. Любой другой ответ вызывает повторную печать сообщения. Для использования /WAIT на системном томе должна присутствовать программа PIP.SAV

В строке команды можно указать не более шести спецификаций файлов. Спецификации файлов в строке команды разделяются запятыми. Спецификация файла может содержать конструкцию * и %.

Команда UNPROTECT задается с терминала в виде полной строки или краткой. Если задана краткая строка команды, на терминале печатается: FILES?

Пользователь должен напечатать на терминале спецификацию файла, для которого будет отменена защита от удаления, и возврат каретки.

Некоторые переключатели допускают в качестве своего аргумента дату. Синтаксис, определяющий дату, следующий:

[дд:][:ммм][:гг]

где дд — день (десятичное число от 1 до 31);

ммм — месяц (первые три буквы названия месяца);

гг — год (десятичное число от 82 до 99).

Значение даты по умолчанию соответствует текущей системной дате. Если опущена любая часть даты (дд,ммм,гг), система использует соответствующую часть текущей системной даты. Например, если указать только :82 и текущая системная дата 4 MAY 1983, система использует 4:MAY:82. Если текущая системная дата не установлена, она считается нулевой.

Примеры:

1. UNPROTECT/BEFORE:20:MAR:83 *.MAC
FILES UNPROTECTED;
DK:A.MAC
DK:B.MAC
DK:C.MAC

по данной команде отменяется защита для всех файлов с типом .MAC, которые были созданы до 20 марта 1983 года.

2. UNPROTECT/EXCLUDE DY0:*.SAV
?PIP—W—NO.SYS ACTION
FILES UNPROTECTED:
DX0:ABC.OLD
DX0:AAF.OLD
DX0:COMB.
DX0:MERGE.OLD

по данной команде отменяется защита для всех файлов тома DY0:, кроме файлов с типами .SAV и SYS.

3. UNPROTECT/INFORMATION DY0:
(FIL1,FIL2,FIL3).TXT
?PIP—I—FILE NOT FOUND DY0:FIL2.TXT

по данной команде отменяется защита для файлов DY0:FIL1.TXT и DY0:FIL3.TXT. Файл FIL2.TXT на DY0: не найден.

4. UNPROTECT/SYSTEM DY0:MM.*
FILES UNPROTECTED:
DY0:MM.MAC
DY0:MM.OBJ
DY0:MM.SAV
DY0:MM.SYS

по данной команде отменяется защита для всех файлов с именем MM и любым типом (и также с типом .SYS).

5. UNPROTECT/WAIT DY0:FILE.MAC
MOUNT INPUT VOLUME IN DY0:;CONTINUE?Y
DY0:FILE.MAC?Y
MOUNT SYSTEM VOLUME IN DY0:;CONTINUE?Y

данный пример показывает использование переключателя /WAIT, отменяется защита для файла DY0:FILE.MAC.

5. ОПИСАНИЕ ККЯС

Краткий командный язык системы (ККЯС) позволяет с помощью одной командной строки и загружать в память нужную программу, и подавать ей командную строку одной строкой команды.

Общий формат строки команд ККЯС:

.Имяпр входспф[...входспф[/прк] выхспф[...выхспф[/прк] или

.Имяпр выхспф[...выхспф[/прк]=входспф[...входспф[/прк]

где имяпр — имя загружаемой программы;

входспф — имя устройства, имя и тип входного файла.

Нельзя опускать ни имя, ни тип файла. Допустимо использовать конструкцию * и % в спецификации файла;

выхспф — имя устройства, имя и тип выходного файла.

Нельзя опускать ни тип, ни имя файла. Допустимо использовать конструкцию * и % в спецификации файла. В первом общем формате строки команды ККЯС выходспф — необязательный аргумент;

/прк — один или несколько переключателей загружаемой программы.

Примеры:

1. .PIP DY0:*MAC/I:12:JAN:83 DY1:*.*

По данной команде ККЯС все файлы с DY0: с типом .MAC, созданные в течение или после 12 января 1983 года, копируются на DY:.

2. .PIP DY1:*.*=DY0:*MAC/I:12:JAN:83

Данная команда имеет такой же результат, что и предыдущая.

6. ПРОЦЕССОР КОСВЕННЫХ УПРАВЛЯЮЩИХ ФАЙЛОВ (IND)

Программа процессор косвенных управляющих файлов или IND предназначена для создания и работы с косвенными управляющими файлами.

Косвенный управляющий файл — это объединенный набор команд КЯС или ККЯС и директив программы IND.

Косвенные управляющие файлы используются для выполнения команд, для создания и редактирования исходных файлов, а также выполнения логических проверок.

6.1. Создание косвенного управляющего файла

Косвенный управляющий файл состоит из нескольких командных строк, включающих метки, директивы программы IND, команды монитора и комментарии (рис. 1).

.SUBONE	IFNDF SYM .GOSUB SUBTWO	.;GET DEFINITION .;OF SYM
Метка	Директива IND	Комментарий

6.1.1. Метки. Метка ставится в начале строки для повторного обращения к этой строке. Максимальное число символов, из которых может состоять метка, — шесть. В начале строки перед меткой ставится точка, а за ней двоеточие.

Примеры:

1. .START: .ASK ANS DO YOU HAVE A LINEPRINTER?

2. .LI:

.SETT VT

Метки .START и .LI являются метками прямого доступа в отличие от метки указанной в следующем примере. Метка 200 не является меткой прямого доступа.

Пример.

.RETURN

.200

.ASK A DO YOU WAIT TO CONTINUE

.IFT A .GOSUB 100

При обработке косвенного управляющего файла программа IND заносит все имена и адреса ячеек меток прямого доступа в таблицу меток прямого доступа. Когда программа IND встречает обращение к метке, она прежде всего определяет тип метки и проверяет таблицу меток прямого доступа. Если программа IND находит метку в таблице, то осуществляется непосредственно переход к ячейке без поиска файла. Если метка не является меткой прямого доступа, то поиск начинается от текущей позиции до конца файла. Если метка не найдена, то программа IND ведет поиск от начала файла до текущей позиции.

В пределах файла можно определить до 20 меток прямого доступа. Если определено больше 20 меток прямого доступа, то вновь определенные метки заменяют уже определенные метки прямого доступа, начиная с первой определенной метки прямого доступа. После этого метки прямого доступа рассматриваются как обычные метки.

6.1.2. Директивы программы IND. Вторым элементом командной строки управляющего файла могут быть директивы IND, команды КЯС или команды ККЯС. Директивы и команды можно указывать как на одной строке, так и на не-

скольких. Некоторые директивы можно использовать с другими директивами на одной и той же командной строке. При обработке косвенного управляющего файла IND считывает его и интерпретирует каждую командную строку, как команду для передачи в KMON или как запрос для работы программы IND. Директивам IND предшествует точка. Команды КЯС отделяются от директив пробелом. Директивы IND можно использовать как одни на строке, так и с меткой или комментарием или с тем и другим вместе.

Пример:

.VIDEO: .SETT VT.;VIDEO TERMINAL AVAILABLE

С помощью директив IND можно определить метки, указать значения для имен: логического, числового, строкового, осуществлять доступ к файлам, разрешать и запрещать операционные режимы, наращивать или уменьшать значение числового имени. В табл. 39 приведено краткое описание директив программы IND.

Таблица 39

ДИРЕКТИВЫ ПРОГРАММЫ IND

Директива	Функция
1	2
.Метка:	Назначает имя строке косвенного управляющего файла для обращения к ней
.ASK	Устанавливает логическое имя на «истинно» или «ложно»
.ASKN	Присваивает значение числовому имени
.ASKS	Присваивает значение строковому имени
.DUMP	Воспроизводит таблицу имен
.ERASE	Стирает глобальные и локальные имена из таблицы имен
.PARSE	Разбивает строку на подстроки
.SETD/.SETO	Присваивает числовому имени десятичное (.SETD) или восьмеричное (.SETO) значение
.SETL	Определяет логическое имя

1	2
.SETN	Определяет числовое имя
.SETS	Определяет строковое имя
.SETT/SETF	Устанавливает логическое имя на «исти- но» (.SETT) или «ложно» (.SETF). Устанавливает или очищает разряды в значении числового имени
.TEST	Определяет тип имени и основание системы счисления, проверяет строку символов на коды ASCII и находит ее начальную позицию
.TESTDEVICE	Выдает информацию об указанном ус- тройстве, результат хранится в спе- циальном имени <EXSTRI>
.TESTFILE	Определяет, существует ли файл, и записывает результаты в специальные имена <FILSPC> и <FILERR>
.VOL	Идентификатору тома устройства при- сваивается указанное строковое имя
.CHAIN	Закрывает текущий управляющий файл и передает управление другому управляющему файлу
.CLOSE	Закрывает файл входных данных
.DATA	Посылает строку символов в файл выходных данных
.OPEN	Создает файл ввода данных
.OPENA	Открывает файл для редактирования
.OPENR	Открывает файл для чтения
.PURGE	Закрывает выходной файл
.READ	Читает строку символов ASCII
.BEGIN	Отмечает начало блока управляюще- го файла
.END	Отмечает конец блока управляющего файла
.EXIT	Завершает обработку текущего управ- ляющего файла или блока управ- ляющего файла и возвращает управле- ние предыдущему уровню
.GOSUB	Переходит к подпрограмме в преде- лах управляющего файла

1	2
.GOTO	Переходит к другой строке в управляющем файле
.ONERR	При обнаружении ошибки происходит переход на метку, указанную в директиве
.RETURN	Передаёт управление из подпрограммы в косвенный управляющий файл
.STOP	Останавливает обработку управляющего файла
.IF	Определяет, удовлетворяет ли имя одному из возможных условий
.IFDF/IFNDF	Проверяет, определено имя или нет
.IFENABLED/	Определяет, разрешен или запрещен
.IFDISABLED	операционный режим
.IFLOA/IFNLOA	Определяет, загружен ли драйвер устройства
.IFT/IFF	Проверяет значение, на которое установлено логическое имя или соответствие разрядов в значении числового имени разрядам, установленным в маске
.DELAY	Задерживает обработку файла на указанный период времени
.DISABLE	Запрещает указанный операционный режим
.ENABLE	Разрешает указанный операционный режим
.DEC	Уменьшает значение числового имени на единицу
.INC	Увеличивает значение числового имени на единицу

6.1.3. Команды. Командная строка, использующая команды КЯС или ККЯС, может содержать метки, но в ней нельзя использовать комментарии. Клавиатурный монитор будет воспринимать комментарий как неправильную команду. Команда должна заканчиваться на одной строке.

Пример:

RUN MACRO

PROG=PROGA

△C

Если используются опрашивающие команды (такие как DELETE) или команды, выдающие подсказку (такие как LINK/PROMPT), надо использовать переключатель /NOQUERY или ввести ответ на запрос с терминала.

Однако, если в управляющем файле используются директивы .ENABLE DATA или .DATA для создания и выполнения косвенного управляющего файла, то клавиатурный монитор может обрабатывать команды, введенные на нескольких строках.

Пример:

.IFT LOGSYM DUP DY1:* =DY0/I/Y

В этом примере копируется диск в режиме отображения.

К командам ККЯС применимы те же правила, что и к командам КЯС.

Примечание. Следующие команды КЯС не рекомендуется использовать в управляющих файлах, так как они могут привести к непредсказуемым результатам: BASE, DEPOSIT, EXAMINE, GET, R, REENTER, SAVE, START.

6.1.4. Комментарии. Последним полем командной строки управляющего файла является комментарий. Комментарий несет информацию о выполняющихся действиях, но никаких операций не выполняет. Имеется два типа комментариев:

— внешние;

— внутренние.

Внешние комментарии начинаются с точки с запятой (;) и во время работы программы IND выводятся на терминал. Внешние комментарии нельзя использовать на строке вместе с условными директивами. Внутренний комментарий начинается с точки и точки с запятой (.;). Внутренний комментарий никогда не выводится на терминал. В поле комментария можно использовать не больше 80 символов.

Пример:

DEVICE: .IFLOA DY0: .GOSUB COPY ;DY0: IZ LOADED GOTO
;COPY SUBROUTINE

6.2. Пуск программы.

Вызов косвенного управляющего файла с системного устройства осуществляется по команде:

R IND <BK> или IND <BK>

после того, как монитор напечатает на терминале точку.

После вызова программа IND печатает звездочку (*) и ожидает ввода командной строки. Если в это время подать

возврат каретки, то программа печатает номер своей версии.

6.2.1. Команды оператора

Режим работы программы IND задается введением с терминала командной строки следующего формата:

спф[/прк][прм],

где спф — спецификация косвенного управляющего файла (устройство, имя, тип);

прк — один или несколько переключателей (см. табл. 40);

прм — одно или несколько значений (до девяти), которые пользователь может передать в управляющий файл (см. п. 6.2.7).

Если косвенный управляющий файл вызывается с помощью командной строки

.RUN IND спф[/прк] <BK> или .IND спф[/прк] <BK>, то в этих случаях в командной строке нельзя использовать параметры. Если действует установка SET KMON IND, то командная строка имеет следующий формат:

@ спф[/прк][прм] <BK>

можно выполнять программу IND с терминала.

Пример:

.RUN IND TT.

*ASKS NAM WHAT IS YOUR NAME?

*WHAT IS YOUR NAME?[S] JON<BK>

Этот способ удобен для проверки результатов выполнения команд.

Таблица 40

ПЕРЕКЛЮЧАТЕЛИ ПРОГРАММЫ IND

Переключатель	Функция
1	2
/D	Стирает управляющий файл после его выполнения
/N	Игнорирует все команды монитора в косвенном управляющем файле
/Q	Запрещает воспроизведение команд монитора
/T	Воспроизводит каждую обработанную командную строку

6.2.2. Переключатель /D. По переключателю /D стирается косвенный управляющий файл после его выполнения. Переключатель /D действует, если управляющий файл заканчивается директивой .EXIT или достигает конца управляющего файла. Если программа IND встречает директиву .STOP или логический конец файла, указанный символом /, то файл не стирается. Операционный режим .DISABLE DELETE запрещает выполнение переключателя /D.

6.2.3. Переключатель /N. По этому переключателю программа IND обрабатывает все директивы управляющего файла, но игнорирует все команды монитора. Этот переключатель удобно использовать для проверки логического хода управляющего файла.

Если переключатель /N используется во время вызова косвенного управляющего файла, то команды монитора игнорируются до тех пор, пока не будет вызван предыдущий управляющий файл, или не встретятся директивы .ENABLE DCL или .ENABLE MCR, или не завершится обработка управляющего файла на всех уровнях.

6.2.4. Переключатель /Q. По переключателю /Q запрещается воспроизведение команд по мере их выполнения. Переключатель /Q действует даже в том случае, если управление передается вложенному управляющему файлу. Операционный режим .DISABLE QUIET запрещает выполнение переключателя /Q.

6.2.5. Переключатель /T. При использовании этого переключателя в командной строке на терминал выводятся все директивы и команды монитора по мере их обработки. Этот переключатель удобно использовать для проверки и отладки управляющего файла. Если задана трассировка, то печатается символ восклицательный знак (!) перед каждой строкой с директивой. По этому переключателю не воспроизводится символ точка с запятой (;) перед комментариями или командами монитора по мере их обработки. Если при обработке управляющего файла встретится переход на метку, то необработанные командные строки не воспроизводятся. Переключатель /T эквивалентен действию операционного режима .ENABLE TRACE и запрещается режимом, заданным .DISABLE TRACE.

6.2.6. Параметры. Параметры позволяют заранее указать содержимое определенных символов при обработке их программой IND. В командной строке можно указывать до девяти параметров. В том случае, когда в командной строке

указываются параметры, программа .IND записывает командную строку в таблицу локальных имен с именем .COMMAN. и далее записываются параметры с P0 по P9. Значение параметра может быть глобальным именем, локальным, логическим, числовым, строковым. Параметры отделяются друг от друга пробелами.

Пример:

.IND

*DX: FIL A B D

Указанная командная строка записывается в таблицу локальных имен. DX: FIL записывается в таблицу локальных имен с именем .COMMAN. P0 присваивается значение A, P1 — B, P2 — D. Остальные параметры имеют нулевые значения. Количество указанных параметров хранится в специальном имени <STRLEN>.

6.2.7. Вложенные косвенные управляющие файлы

Косвенный управляющий файл может состоять из нескольких управляющих файлов, которые называют вложенными. В операционной системе предусмотрено использование до трех уровней вложения. Для вызова вложенного управляющего файла из косвенного управляющего файла надо подать следующую командную строку:

⌘ спф[/прк] [прм]

где спф — спецификация вложенного управляющего файла, по умолчанию тип .COM;

прк — один или несколько переключателей (см. табл. 40);

прм — один или несколько параметров (до девяти), которые пользователю нужно ввести в управляющий файл. В приведенной выше строке нельзя использовать комментарий.

Если в управляющем файле указана директива .ENABLE GLOBAL, то все имена, указанные с помощью символа ⌘, считаются глобальными и признаются на всех уровнях вложения. Если директива .ENABLE GLOBAL не указана, то имена, используемые во вложенном управляющем файле, считаются локальными и признаются только на данном уровне вложения.

6.3. Вызов косвенных командных файлов из управляющих файлов. Вызов косвенного командного файла из косвенного управляющего файла осуществляется посредством ввода следующей командной строки: ⌘ ⌘ спф,

где спф — спецификация косвенного командного файла.

После выполнения косвенного командного файла управление возвращается косвенному управляющему файлу.

Пример:

⌘@TIT

Эта командная строка вызывает косвенный командный файл TIT.COM.

6.4. Специальные символы

В табл. 41 приведены специальные символы, использующиеся в управляющем файле.

Таблица 41

Символ	Функция
1	2
.	Используется для обозначения всех директив IND и меток, а также для определения основания системы счисления
#	Используется для указания номера файла в директивах .CLOSE, .DATA, .OPEN, .OPENA, OPENR и RURGЕ
@	Используется для вызова вложенного управляющего файла
⌘	Используется для определения глобальных имен
⌘@	Используется для вызова косвенного командного файла
/	Обозначает конец файла
+	Соединяет символы строки
;	Используется в командной строке для отделения внешнего комментария

6.5. Имена. В командной строке управляющего файла имена используются в виде переменных, с помощью которых пользователь может управлять выполнением файла. Имена бывают глобальными и локальными, логическими, числовыми, строковыми. Имя может состоять не более чем из шести буквенно-цифровых символов.

6.5.1. Локальные и глобальные имена. Все имена после директивы .ENABLE GLOBAL, перед которыми стоит символ ⌘, являются глобальными и признаются на всех уровнях вложения косвенного управляющего файла. Все остальные

ные имена определяются как локальные и признаются только в пределах блока или вложенного управляющего файла.

6.5.2. Логические имена представляют истинные или ложные значения. Можно определить или повторно определить логическое имя посредством использования директивы `.ASK`, `.SETL`, `.SETT` или `.SETF`.

6.5.3. Числовые имена используются для указания аргументов в командных строках для формирования арифметических выражений. Действительный диапазон для целого числа, представленного числовым именем от 0 до 65535 (десятичное) или 177777 (восьмеричное).

6.5.3.1. Определение системы счисления. Основание системы счисления указанного числового значения определяется операционным режимом `.OCTAL` или специальными символами «номер» (`#`), обозначающий восьмеричную систему и точка (`.`), обозначающая десятичные значения.

Если задан режим `.ENABLE OCTAL` и перед значением стоит символ «номер» (`#`), то значение интерпретируется и хранится в восьмеричной системе. Если в этом же режиме значение задано с помощью десятичной точки, то оно интерпретируется как десятичное, но хранится в восьмеричной системе.

В режиме `.DISABLE OCTAL` значения, заданные с помощью символа «номер» (`#`), интерпретируются как восьмеричные, но хранятся как их десятичный эквивалент. Значения, заданные в этом режиме с помощью десятичной точки, интерпретируются и хранятся в десятичной системе.

По умолчанию восьмеричные значения интерпретируются как восьмеричные, десятичные как десятичные.

Если восьмеричное значение задано с помощью десятичной точки, то оно будет интерпретироваться как десятичное. Если при записи директивы используются десятичные и восьмеричные значения, то все значения будут интерпретироваться и храниться в десятичной системе.

6.5.3.2. Цифровые выражения формируются с помощью операций, приведенных в табл. 42.

Таблица 42

Операция	Функция
1	2
+	Сложение
—	Вычитание
/	Деление
*	Умножение
!	Логическое ИЛИ
&	Логическое И
^	Логическое НЕ
EQ или =	Равно
NE или <>	Не равно
GE или >=	Больше или равно
LE или <=	Меньше или равно
GT или >	Больше
LT или <	Меньше

При делении программа IND усекает делимое до целого. Цифровые выражения вычисляются слева направо при отсутствии скобок. При наличии скобок сначала вычисляется выражение в скобках.

Примеры:

1. .SETN N1 2
.SETN N2 3
.SETN N3 N1+N2*4
2. .SETN N1 2
.SETN N2 3
.SETN N3 N1+(N2*4)

В первом примере числовому имени N3 присваивается значение 24 (восьмеричное).

Во втором примере числовому имени N3 присваивается значение 16 (восьмеричное).

6.5.4. Строковые имена используются для обозначения строки, состоящей из символов КОИ-7. Строка символов КОИ-7, которой присваивается строковое имя, должна быть заключена в кавычки и не превышать 132 символов.

Пример:

.SETS PROMPT «DO YOU HAVE ANY PROTECTED FILE»

Строковому имени PROMPT присваивается строка символов, заключенная в кавычки. IND позволяет с помощью строкового имени разбивать строки.

6.6. Операционные режимы. С помощью директив .ENABLE или .DISABLE можно соответственно задавать или запрещать операционный режим. Операционный режим может быть как локальным, так и глобальным. Операционные режимы приведены в табл. 43.

Таблица 43

Операционный режим	Умолчание	Область действия	Функция
1	2	3	4
DATA	Запрещен	Локальный	В режиме DATA программа IND посылает в выходной файл все строки, следующие за директивой .ENABLE DATA пока не встретится директива .DISABLE DATA или .CLOSE
DCL	Разрешен	Локальный	Отмена режима запрещает выполнение команд клавиатурного монитора
DELETE	Запрещен	Локальный	Если режим DELETE разрешен, управляющий файл удаляется после его выполнения
ESCAPE	Запрещен	Глобальный	В режиме ESCAPE разрешен ответ AP2 возврат каретки для директив: .ASK, .ASKN, .ASKS
GLOBAL	Запрещен	Глобальный	В этом режиме все имена, начинающиеся с символа

1	2	3	4
LOWERCASE	Разрешен	Глобальный	⌘, считаются глобальными на всех уровнях управляющего файла Разрешает вывод строки символами нижнего регистра
MCR	Разрешен	Локальный	Отмена режима запрещает выполнение команд монитора
OCTAL	Разрешен	Глобальный	В режиме OCTAL числовому имени присваивается значение в восьмеричной системе
PREFIX	Разрешен	Глобальный	Отмена режима запрещает печать символов звездочка (*) перед подсказкой и точка с запятой перед комментарием
QUIET	Запрещен	Локальный	Режим QUIET разрешает воспроизводить командные строки
SUBSTITUTION	Разрешен	Глобальный	В этом режиме разрешена замена строкового имени на присвоенное ему значение
SUFFIX	Разрешен	Глобальный	В этом режиме IND запрещает воспроизводить в подсказке параметры, указанные в директивах .ASK, .ASKN, .ASKS

1	2	3	4
TIMEOUT	Запрещен	Глобальный	Если режим TIMEOUT разрешен, IND признает значение счетчика ожидания готовности устройства, указанное в директивах .ASK, .ASKN, .ASKS
TRACE	Запрещен	Локальный	В режиме TRACE воспроизводится каждая обработанная командная строка управляющего файла

6.7. Специальные имена. Кроме ранее перечисленных символов IND имеет специальные имена, приведенные в табл. 44. IND устанавливает эти имена в соответствии с определенными характеристиками системы и ответами, поданными во время выполнения командного файла. Но нельзя их установить непосредственно. Специальные имена программы IND заключаются в угловые скобки.

Таблица 44

СПЕЦИАЛЬНЫЕ ИМЕНА

Специальное имя	Значения
1	2
<ALPHAN>	Устанавливается на «истинно», если строка символов, введенная в ответ на директиву .ASKS, содержит только буквенно-цифровые символы. Директивы .TEST и .TESTFILE могут устанавливать <ALPHAN> на «истинно» или «ложно». Если введена пустая строка, то это специальное

1	2
<ALTMODF> или <ESCAPE>	имя устанавливается на «истинно», а если пробел, то на «ложно». Если режим LOWERCASE запрещен, то ответ на запрос символами нижнего регистра устанавливает специальное имя <ALPHAN> на «ложно» Устанавливается на «истинно», если установлен режим ESCAPE, и ответ был подан как AP2 возврат каретки. В противном случае это специальное имя устанавливается на «ложно»
<DEFAULT>	Устанавливается на «истинно», если в ответ на подсказку директивы был подан возврат каретки
<EOF>	Устанавливается на «истинно», когда программа IND встречает конец управляющего файла. В противном случае специальное имя устанавливается на «ложно»
<FALSE>	Постоянно установлено на «ложно», может быть использовано для обозначения ответа в директиве .ASK
<MAPPED>	Устанавливается на «истинно», если IND выполняется под управлением монитора XM, в противном случае — «ложно»
<OCTAL>	Устанавливается на «истинно», если числовому имени присваивается значение в восьмеричной системе, и на «ложно» в противном случае
<RAD50>	Устанавливается на «истинно», если введена пустая строка или введенная строка содержит символы RAD50
<TIMEOUT>	Устанавливается на «истинно», если разрешен режим TIMEOUT и время счетчика ожидания готовности устройства, указанное в директивах .ASK, .ASKN, .ASKS истекло, в противном случае — «ложно»

1	2
<TRUE>	Постоянно установлено на «истинно», может быть использовано для обозначения ответа по умолчанию для директив .ASK, .ASKN, .ASKS
<ERROR>	Постоянно содержит значение 2 (восьмеричное), обозначающее ошибку
<EXSTAT>	Содержит переменную, устанавливаемую после завершения команды монитора. Переменная может принимать одно из следующих значений: 0 — предупреждение; 1 — успешное выполнение; 2 — ошибка; 4 — неустраняемая ошибка. Директива .EXIT может также изменять значение этого специального имени
<FILERR>	Содержит код, определяемый после выполнения файла. Коды, которые могут быть установлены: 1 — успешное выполнение; 372 — недостаточно места для требуемого драйвера устройства; 366 — конец файла; 363 — переполнение данных; 351 — устройство переполнено; 346 — нет такого файла; 343 — файл доступен для записи; 340 — ошибка чтения; 337 — ошибка записи; 333 — на данном канале файл недоступен; 327 — файл превышает заданную область распределения; 325 — недопустимый тип записи; 324 — файл доступен для чтения; 313 — файл уже открыт; 312 — недопустимое имя файла;

1	2
	244 — недействительное устройство или привод.
<SEVERE>	Постоянно имеет значение 4 (восьмеричное), которое указывает на несправивую ошибку
<SPACE>	Содержит значение свободной области памяти в байтах. Программа IND использует для записи символа минимум 10 (восьмеричное) байтов
<STRLEN>	Содержит длину строки символов, назначенную строковому имени, в восьмеричной системе
<SUCCES>	Постоянно имеет значение 1 (восьмеричное), указывающее на успешное выполнение операции
<SYMTYP>	Содержит код, обозначающий тип имени, проверенного директивой .TEST. Коды: 0 — логическое имя; 2 — числовое имя; 4 — строковое имя
<SYSTEM>	Содержит восьмеричный код, обозначающий операционную систему, в которой выполняется программа IND. Коды: 1 — ОС PB 2 — МИС PB 7 — ФОДОС (SJ монитор) 10 — ФОДОС (FB монитор). Если истинно <MAPPED>, то загружен ХМ монитор
<SYUNIT>	Содержит номер привода системного устройства
<WARNIN>	Постоянно содержит нулевое значение, обозначающее предупреждение.
<DATE>	Содержит текущую дату. Формат даты: DD—МММ—YY (день, месяц, год). Запись даты производится девятью символами. Если запись даты

1	2
<EXSTRI>	занимает меньше девяти символов, то пустые места заполняются пробелами. Если дата не указана, то содержимое специального имени <DATE> заполняется пробелами Содержит физическое имя устройства, размер устройства и атрибуты устройства, проверенные директивой .TESTDEVICE
<FILESPC>	Содержит полную спецификацию файла (устройство, имя файла, тип файла), проверенного директивой .TESTFILE
<MONNAM>	Указывает имя текущего монитора. Длина этой записи заранее определена (6 символов). Если запись занимает меньше шести символов, то пустые места заполняются пробелами
<SYSDISK>	Содержит имя системного устройства
<TIME>	Содержит значение текущего времени. Формат записи: HH:MM:SS (часы, минуты, секунды)

6.8. Замена. В режиме SUBSTITUTION осуществляется замена строкового имени на его значение. Строковое имя в этом случае должно заключаться в кавычки. Замена разрешается в любой строке управляющего файла. Операцию замены можно также использовать в поле комментария.

Пример:

.ENABLE SUBSTITUTION

.ASKS DEV ENTER INPUT DEVICE

ASSIGN 'DEV' INP

после обработки директивы .ASKS на консоль выводится подсказка:

*ENTER INPUT DEVICE[S]: DY

.ASSIGN DY INP

Строковому имени DEV присваивается ответ, поданный пользователем на подсказку т. е. DY. Далее, при использо-

вании в этом управляющем файле имени DEV в кавычках, оно будет каждый раз заменяться на DY. Если режим SUBSTITUTION запрещен, то программа IND будет просто передавать строку в KMON в том виде, в каком она появляется в управляющем файле (т. е. ASSIGN 'DEV' INP).

Для обозначения кавычек в тексте используется не один апостроф, а два.

Пример:

```
;THE SYMBOL'S VALUE
```

программа IND воспроизводит эту строку управляющего файла как:

```
;THE SYMBOL'S VALUE
```

6.9. Директивы программы IND

В следующих пунктах документа приводится подробное описание директив программы IND.

6.9.1. Директива метка используется для обозначения строки косвенного управляющего файла для повторного обращения к этой строке.

Формат директивы:

.Метка:

Для обозначения метки можно использовать любые буквенно-цифровые символы. Директива метка должна стоять в начале строки.

6.9.2. Логический конец файла (/). Символ / означает логический конец управляющего файла. Когда программа IND встречает символ /, обработка файла завершается и на консоле печатается сообщение:

@ <EOF>

Программа IND игнорирует любые символы, которые следуют за символом / в этой же самой строке. Можно использовать эту директиву в любой строке управляющего файла для быстрого завершения обработки файла.

Пример:

```
.ASK CONT DO YOU WISH TO CONTINUE
```

```
.IFT CONT .GOTO 100
```

```
/
```

.100:

В этом примере выполнение завершается, если логический символ CONT определяется как ложный.

6.9.3. Директива .ASK. По директиве .ASK значение указанного логического имени устанавливается на «истинно» или «ложно». Если при ответе на запрос директивы .ASK,

дан положительный ответ, т. е. YES, то логическое имя устанавливается на «истинно», в противном случае — на «ложно». До присвоения значения логическому имени программа IND производит запись в таблицу имен. Если логическое имя было ранее определено, то программа IND восстанавливает его значение. Если вместо логического имени вводится числовое имя или строковое, то IND выдает сообщение об ошибке.

Формат директивы:

.ASK [умл:вrm] лимим подсказка,

где умл — значение по умолчанию, которое используется, если в ответ на запрос директивы подан возврат каретки или по истечении указанного интервала времени ответ не дан. Значение «истинно» или «ложно» устанавливается по умолчанию также с помощью специальных имен <TRUE> и <FALSE>. Если используется ответ по умолчанию, то специальное имя <DEFAULT> устанавливается на «истинно»;

вrm — значение счетчика ожидания готовности устройства. Если по истечении указанного значения времени ответ не дан, то IND использует ответ по умолчанию и специальные имена <DEFAULT> и <TIMEOUT> устанавливаются на «истинно». Счетчик ожидания готовности устройства можно использовать, если разрешен операционный режим .ENABLE TIMEOUT, в конфигурацию монитора включен таймер и монитор осуществляет поддержку таймера. Если эти условия не выполняются, то значение счетчика ожидания готовности устройства игнорируется. Счетчик ожидания готовности устройства имеет следующий формат: NNU

где NN — число, указывающее интервал времени, в течение которого должен быть дан ответ. NN интерпретируется как восьмеричное число, если за ним не следует десятичная точка. Если в качестве NN указан недействительный параметр, то IND выдает сообщение об ошибке;

U — единица времени, где т — тики (1 тик равен 1/50 сек.), M — минуты, S — секунды, H — часы;

лимим — логическое имя, которое устанавливается на «истинно» или «ложно»;

подсказка — вопрос, который печатается на консоле после обработки директивы .ASK, и может включать до 80 символов.

Скобки для произвольных параметров обязательны, они являются частью синтаксиса. Параметры отделяются друг от друга двоеточием.

Пример:

`.ASK [:15.S]DONE ARE YOU FINISHED`

после обработки директивы `.ASK` программа `IND` печатает звездочку, подсказку и в скобках параметры.

`*ARE YOU FINISHED? [Y/N D:N T:15.S]`

Пользователь должен подать ответ на подсказку. `Y/N` указывает, что требуется ответ `YES` или `NO`, соответственно логическому имени присваивается истинное или ложное значение. `D:N` показывает, что по умолчанию устанавливается то же значение, что и при ответе `NO`. Если в ответ на подсказку подан любой другой символ, то программа `IND` повторяет вопрос. Параметры `T:15.S` показывают, что ответ должен быть дан в течение 15 секунд. При ответе `AP2` возврат каретки, где `AP2` — клавиша, представленная кодом 33, когда разрешен операционный режим `.ENABLE ESCAPE`, логическое имя устанавливается на «истинно», если оно ранее не было определено, и специальное имя `<ESCAPE>` устанавливается на «истинно». Значение ранее определенного логического имени остается неизменным. Если ответ `AP2` возврат каретки подается при запрещенном режиме `ESCAPE`, то `IND` выдает сообщение об ошибке. При ответе `CV/Z` программа `IND` завершает обработку файла и выдает следующее сообщение:

␣ EOF

6.9.4. Директива `.ASKN`. По директиве `.ASKN` устанавливается значение для числового имени. Если числовое имя не было ранее определено, то программа `IND` делает запись в таблице имен. Если числовое имя было ранее определено, то программа `IND` восстанавливает его значение (десятичное или восьмеричное в соответствии с заданным режимом).

Формат директивы:

`.ASKN [ниж:верх:умл:вrm]` числим подсказка

где `ниж:верх` — диапазон чисел, из которого выбирается ответ. По умолчанию диапазон может быть от 0 до 177777 (восьмеричное) или от 0 до 65535 (десятичное). Числа из диапазона можно также использовать в цифровых выражениях;

`умл` — указывает ответ по умолчанию, используется, когда в качестве ответа подается возврат каретки или в течение указанного интервала времени ответ не будет дан. Если ответ по умолчанию не указан, то выбирается самое нижнее значение из назначенного диапазона;

`вrm` — см. директиву `ASK`;

числим — числовое имя, которому будет присвоено значение ответа;

подсказка — вопрос, который воспроизводится на терминале после обработки директивы .ASKN. Подсказка может включать до 80 символов.

Пример:

.ENABLE OCTAL

.ASKN [0:10:3.:20.S] ERR NO. OF ERROR CODES TO USE

После обработки директивы .ASKN на консоль выдается следующая подсказка:

*NO. OF ERROR CODES TO USE [D R:0—8. D:3. T:20.S]

В этом примере разрешен восьмеричный режим. Но так как параметры даны как в восьмеричной системе, так и в десятичной, то все значения интерпретируются как десятичные после обработки директивы .ASKN. Обозначение R:0—8. показывает, что ответ должен быть выбран из диапазона от 1 до 8. Ответ можно указать как в десятичной системе, так и в восьмеричной. D:3 указывает значение ответа по умолчанию, т. е. числовому имени присваивается значение, равное 3, если в ответ на подсказку подан возврат каретки. Обозначение T:20.S, указывает, что ответ должен быть дан в течение 20 секунд. Если указанный интервал времени истек, а ответ не дан, то числовому имени присваивается значение по умолчанию, т. е. 3. Если разрешен режим .ENABLE ESCAPE и ответ подан в качестве AP2 <возврат каретки>, то специальное имя <ESCAPE> устанавливается на «истинно» и числовому имени присваивается значение нуль, при условии, что числовое имя не было ранее определено. При запрещенном режиме .ENABLE ESCAPE ответ AP2 <возврат каретки> устанавливает специальное имя <ESCAPE> на «ложно» и программа IND. выдает сообщение об ошибке. При ответе CY/Z программа IND завершает обработку косвенного управляющего файла и выдает сообщение: @EOF.

6.9.5. Директива .ASKS. По директиве .ASKS указанному строковому имени присваивается строка символов КОИ-7. Если строковое имя было ранее определено, то IND заносит это имя в таблицу имен. Если строковое имя не было определено, программа IND восстанавливает присвоенное ему значение. IND выдает сообщение об ошибке, если строковое имя было ранее определено как логическое или числовое.

Формат директивы:

.ASKS[ниж:верх:«умл»:Врм] стрим подсказка

где ниж:верх — диапазон чисел, с помощью которых опре-

деляется длина строки, присвоенной строковому имени. По умолчанию диапазон от 0 до 204 (восьмеричное) или от 0 до 132 (десятичное);

«умл» — указывает ответ по умолчанию, который используется, если в качестве ответа подать возврат каретки или истек заданный интервал времени. В качестве ответа по умолчанию можно указать строку символов КОИ-7, символ или выражение;

Врм — см. директиву .ASKS;

стрим — строковое имя, которому будет присвоено значение ответа;

подсказка — вопрос, который воспроизводится на терминале. Подсказка может включать не больше 80 символов.

Пример:

.ENABLE OCTAL

.ASKS [0:10«ФОДОС»:20.S] VALID TYPE YOUR VOLUME ID
после обработки директивы .ASKS на терминал выводится:

*TYPE YOUR VOLUME ID [S R:0—8 D:«ФОДОС» T:20.S]

В ответ на подсказку должна быть подана строка символов КОИ-7, о чем свидетельствует символ S, указанный в подсказке. Обозначение R:0—8. указывает, что ответ не должен превышать 8 символов. Обозначение D:«ФОДОС» показывает, что строковому имени присваивается значение «ФОДОС», если ответ подан в качестве возврата каретки. Если ответ по умолчанию не указан, то обозначение D: не воспроизводится. Обозначение T:20.S указывает на то, что ответ должен быть дан в течение 20 сек. Если разрешен операционный режим .ENABLE ESCAPE и дан ответ AP2 возврат каретки, то специальное имя <ESCAPE> устанавливается на «истинно», но строковому имени никакого значения не присваивается. Если этот операционный режим запрещен и подан ответ AP2 возврат каретки, то специальное имя <ESCAPE> устанавливается на «ложно», и программа IND выдает сообщение об ошибке. По команде CY/Z, IND завершает обработку косвенного управляющего файла и на консоль выводится сообщение: EOF.

6.9.6. Директива .BEGIN. Директивы .BEGIN и .END позволяют пользователю организовать управляющий файл из блоков. Такие файлы легче отлаживать и эксплуатировать. Локальные имена, определенные в блоке управляющего фай-

ла, действительны только внутри этого блока. Они стираются из таблицы имен, как только программа IND встречает директиву .END. Поэтому, если встречаются логические, числовые или строковые имена вне блока, то пользователь может снова определить эти же имена и даже с другим типом внутри блока. Директива .BEGIN отмечает начало блока. Все локальные имена, следующие за директивой, являются локальными для блока, а не для всего управляющего файла. Директива .ERASE LOCAL, указанная в блоке, стирает все локальные имена в пределах блока. В управляющем файле можно использовать до 127 блоков.

Формат директивы:

.BEGIN

каждой директиве .BEGIN должна соответствовать директива .END.

6.9.7. Директива .CHAIN. По директиве .CHAIN закрывается текущий косвенный управляющий файл и передается управление другому косвенному управляющему файлу.

Формат директивы:

.CHAIN спф[/прк...],

где спф — спецификация косвенного управляющего файла, которому надо передать управление;

прк — один или несколько переключателей (см. табл. 40).

Пример:

.CHAIN CC.MAC

В этом примере управление передается управляющему файлу DK:CC.MAC.

6.9.8. Директива .CLOSE. По директиве .CLOSE закрывается файл, открытый по одной из директив: .OPEN, .OPENA, .OPENR. Прежде чем передать управление клавиатурному монитору, пользователь должен закрыть файл. Директива .CLOSE запрещает режим данных.

Формат директивы:

.CLOSE [#N] [спф],

где N — номер управляющего файла от 0 до 3 (по умолчанию 0). Если разрешен режим замены, то можно вместо числа использовать любой другой символ, заключив его в кавычки;

спф — спецификация косвенного управляющего файла.

Если в директиве не указан символ номер (#), то программа IND выдает сообщение об ошибке.

Использование директивы .CLOSE после директивы .OPENR имеет то же действие, что и директива .PURGE.

6.9.9. Директива `.DATA`. По директиве `.DATA` делается запись в файле, предварительно открытом по директиве `.OPEN` или `.OPENA`.

Формат директивы:

`.DATA [#N]` текст,

где N — номер файла от 0 до 3 (по умолчанию 0). Если разрешен режим замены, то пользователь может вместо числа указать любой символ, заключив его в кавычки;

текст — данные, которые будут посланы в выходной файл. (Если разрешена замена, текст может быть строковым именем в кавычках). В виде текста в файл можно послать как пустые строки, так и символы.

Командная строка не должна превышать 132 символов.

Примеры.

1. `.OPEN TEMP`

`.DATA THIS IS DATA`

`.CLOSE`

2. `.OPEN COPY.TMP`

`.DATA RUN PIP`

`.DATA DY:FILE.TST/Z`

`.DATA DY:FILE.TST=DY1:FILE.TST/W/Y`

`DATA ^C`

`.CLOSE`

`^C@COPY.TMP`

В первом примере создается файл `TEMP.DAT` (номер файла (0), который будет состоять из строки символов `THIS IS DATA`).

Во втором примере создается файл `COPY.TMP`, который работает с программой `PIP`. В следующих командных строках снимается защита с файла `DY:FILE.TST` и этот файл копируется на `DY1:`.

6.9.10. Директива `.DEC`. По директиве `.DEC` значение числового имени уменьшается на единицу.

Формат директивы:

`.DEC числим`,

где числим — числовое имя, значение которого будет уменьшаться на единицу.

Программа `IND` будет выдавать сообщение об ошибке, если вместо числового имени будет использовано логическое или строковое имена.

6.9.11. Директива `.DELAY`. По директиве `.DELAY` обработка косвенного управляющего файла задерживается на указанный в директиве период времени. Директива действу-

ет, если в конфигурацию системы включен таймер и монитор осуществляет поддержку ожидания готовности устройства.

Формат директивы:

`.DELAY NNU`

где NN — интервал времени, на который задерживается выполнение управляющего файла;

U — единица времени (тики, секунды, часы, минуты).

Когда файл приостанавливает свою работу, на терминале печатается сообщение:

`DELAYING...`

После возобновления работы программа IND выдает сообщение:

`...CONTINUING`

Пример.

`.DELAY 10S`

Выполнение файла задерживается на восемь секунд.

6.9.12. Директива `.DISABLE` запрещает указанный операционный режим.

Формат директивы:

`.DISABLE опр[опр,опр ...]`

где опр — один или более операционных режимов (см. табл. 43).

Операционные режимы независимы друг от друга. В одной командной строке можно разрешить или запретить несколько операционных режимов за исключением режима DATA. Режим DATA нельзя указывать с другими операционными режимами. Если при запрещении нескольких операционных режимов возникает ошибка, то ни один из них не запрещается.

Пример.

`.DISABLE TIMEOUT, GLOBAL`

В этом примере запрещаются два операционных режима TIMEOUT и GLOBAL.

Операционные режимы бывают локальными и глобальными. Локальным операционным режимом считают тот, который определен внутри вложенного управляющего файла. После выполнения вложенного управляющего файла, автоматически устанавливаются ранее определенные операционные режимы. Глобальные операционные режимы действительны на всех уровнях управляющего файла. В табл. 43 приведена область действия каждого операционного режима. Если во время обработки управляющего файла, подать `CU/D`,

то вывод на терминал не будет производиться до тех пор, пока не встретится директива `.DISABLE QUIET`.

6.9.13. Директива `.DUMP`. По директиве `.DUMP` выводятся на печать таблицы глобальных, локальных или специальных имен.

Формат директивы:

а `.DUMP [имтабл]` ,

где имтабл — имя таблицы, которую надо воспроизвести: локальную, глобальную или специальную. Если имя таблицы не указано, то программа `IND` выводит содержимое всех трех таблиц.

Каждая строка таблицы имен формируется следующим образом:

им (тип): знач

где им — имя;

тип — тип имени. В таблице имен используются следующие обозначения типов имен:

L — логическое имя;

O — числовое имя в восьмеричной системе;

D — числовое имя в десятичной системе;

S — строковое имя;

знач — значение, присвоенное имени, T или F — для логического имени, что соответствует истинному или ложному значению, число — для числового имени, строка символов КОИ-7 в кавычках — для строкового имени.

Локальные имена воспроизводятся в таблице в обратном порядке, т. е. последнее определенное локальное имя указывается первым. По командной строке

`.DUMP LOCAL`

на терминал выводится таблица локальных имен, если она пуста, на терминал выдается сообщение:

***** THERE ARE NO LOCAL SYMBOLS ***

глобальные имена перечисляются в порядке их определения. По командной строке

`.DUMP GLOBAL`

на терминал выводится таблица глобальных имен, если она пуста, то выдается сообщение:

*****THERE ARE NO GLOBAL SYMBOL *****

Таблица специальных имен никогда не бывает пустой, так как специальные имена являются постоянными.

6.9.14. Директива `.ENABLE`. С помощью директивы `.ENABLE` задается один из операционных режимов, указанных в табл. 43.

Формат директивы:

.ENABLE опр [опр,опр...]

где опр — один или более операционных режимов (см. табл. 43).

Операционные режимы независимы друг от друга. Все режимы могут быть активны одновременно. По одной директиве .ENABLE можно задать несколько операционных режимов, за исключением режима DATA. Если задано несколько операционных режимов на одной строке и возникает ошибка, то ни один из операционных режимов не разрешается.

Пример.

.ENABLE LOWERCASE, TIMEOUT

Операционные режимы могут быть локальными или глобальными. При выходе из вложенного управляющего файла автоматически локальный режим запрещается и устанавливается ранее заданный режим. Локальный режим активен только в файле, внутри которого задан. Глобальный режим определен или запрещен на всех уровнях управляющего файла.

6.9.15. Режим DATA[#N]. В режиме данных программа IND посылает в выходной файл строки, следующие за директивой .ENABLE DATA. Пустые строки игнорируются. #N — произвольный номер файла в диапазоне от 0 до 3 (по умолчанию 0), если разрешена замена, можно вместо числа использовать любой другой символ, заключив его в кавычки.

Пример.

.OPEN SECFIL.DAT

.ENABLE DATA #«NUM»

REN B.MAC M.MAC

.DISABLE DATA #«NUM»

Создается файл SECFIL.DAT и в него записывается строка, заключенная между директивами .ENABLE DATA и .DISABLE DATA.

Примечание. Если разрешен режим данных для одного файла, а данные надо послать в другой, то нужно запретить режим для первого файла и разрешить для второго. Если не удалось запретить режим данных для первого файла, то данные, посылаемые во второй файл будут направляться в первый.

6.9.16. Командный режим DCL. В режиме DCL программа IND передает команды в KMON для выполнения.

Пример.

.ASKS DEV WHICH DEVICE YOU USE FOR THE LOG FILE?

ASSIGN DY0:LOG

Если режим DCL разрешен, то команда ASSIGN будет выполняться. Если режим запрещен или использован переключатель /N, команда игнорируется.

6.9.17. Режим DELETE. В режиме DELETE управляющий файл удаляется после его обработки. Этот режим действует, если программа IND встречает директиву .EXIT или достигает конца файла. Режим .ENABLE DELETE эквивалентен переключателю /D.

6.9.18. Режим ESCAPE. Если режим ESCAPE разрешен, то в качестве ответа на подсказку директив .ASK, .ASKN, .ASKS можно подавать AP2 возврат каретки. Тогда специальное имя <ESCAPE> устанавливается на «истинно». Если перед или после ответа AP2 возврат каретки указать любой символ, то программа IND выдает сообщение об ошибке и повторяет запрос. Пользователь должен подать правильный ответ.

Пример.

.ENABLE ESCAPE

.ASKS A ENTER OPTION

При обработке этого управляющего файла выдается подсказка:

*ENTER OPTION [S]: <AP2> <BK>

строковому имени A не присваивается значение.

6.9.19. Режим глобальных имен. В режиме GLOBAL все имена, начинающиеся с символа X, определяются как глобальные на всех уровнях управляющего файла. Все остальные имена считаются локальными.

Пример.

.ENABLE GLOBAL

.SETS X X «TEST»

@ MEN.COM

программа IND выводит на терминал:

; TEST

в этом примере X является глобальным именем.

6.9.20. Режим LOWERCASE. В этом режиме ответ на подсказку директивы .ASKS или строка символов в директиве .SETS выводятся в том виде, в котором они были введены (символы нижнего регистра на нижнем регистре, символы верхнего регистра на верхнем регистре). Если режим ниж-

него регистра запрещен, то все символы воспроизводятся на верхнем регистре. Директивы .IF и .TEST отличают символы верхнего и нижнего регистров независимо от того, разрешен или запрещен режим LOWERCASE. Если режим LOWERCASE запрещен, а ответ на запрос находится на нижнем регистре, то специальное имя <ALPHAN> устанавливается на «ложно».

6.9.21. Командный режим MCR. В режиме MCR программа IND передает в KMON команды для выполнения.

Пример.

```
.ASKS DEV DEVICE TO USE FOR LOG FILE?  
COPY SYS.MAC TOP.BAK
```

Если режим MCR задан, то команда COPY выполняется. Если данный режим запрещен и не используется переключатель /N, то команда игнорируется.

6.9.22. Режим OCTAL. С помощью режима OCTAL определяется основание системы счисления. Основание системы счисления определяется также посредством символов точка (.) и номер (#). Более подробно определение основания системы счисления описано в п. 6.5.3.1.

6.9.23. Режим PREFIX. В этом режиме программа IND печатает символ звездочка (*) перед подсказкой следующей после директив .ASK, .ASKN, .ASKS и точку с запятой (;) перед всеми внешними комментариями.

Пример.

```
.ASK COUNT DO YOU WANT TO CONTINUE
```

Если режим PREFIX не задан, то после обработки директивы .ASK на терминал выводится подсказка в следующем виде:

```
* DO YOU WANT TO CONTINUE? [Y/N D:N]:
```

Если режим PREFIX задан, то подсказка выводится в следующем виде:

```
DO YOU WANT TO CONTINUE? [Y/N D:N]
```

6.9.24. В режиме QUIET программа IND не воспроизводит команды.

Пример.

```
.OPENA COPY.TMP  
REN COPY.TMP SOS.MAC
```

Команда REN выполняется, но на терминале не воспроизводится.

6.9.25. В режиме SUBSTITUTION разрешено производить замену строкового имени на присвоенное ему значение. При этом строковое имя нужно каждый раз заключать в кавычки.

Например, если строковому имени A присвоена строка символов THIS IS A TEST, то каждое появление 'A' будет заменяться его значением THIS IS A TEST.

Пример.

```
.ENABLE SUBSTITUTION
.ASKS FIL SPECIFY SOURCE FILE
MACRO 'FIL'
```

после обработки этого управляющего файла на терминал выводится:

```
* SPECIFY SOURCE FILE [S]: SOURCE
.MACRO SOURCE
```

В качестве ответа указана строка SOURCE. Т. к. режим замены разрешен, то строковое имя FIL везде, где оно встречается в апострофах, заменяется на строку символов SOURCE.

6.9.26. Режим SUFFIX. При запрещенном режиме SUFFIX в подсказке, выдаваемой на терминал после обработки директив .ASK, .ASKN, .ASKS параметры не печатаются.

Пример.

```
.ENABLE SUFFIX
.ASKN [1:9:11] INIT NO, OF DISK1 TO INIT?
```

После обработки директивы на терминал выводится следующая строка:

```
*NO, OF DISKS TO INIT?
```

6.9.27. В режиме TIMEOUT программа IND разрешает использование счетчика ожидания готовности устройства с директивами .ASK, .ASKN, .ASKS, если в конфигурацию системы включен таймер и монитор осуществляет поддержку таймера. Если два последних условия не выполняются, то программа IND выдает предупреждающее сообщение при попытке использовать режим .ENABLE TIMEOUT. Если режим TIMEOUT запрещен, то счетчик ожидания готовности устройства игнорируется.

6.9.28. Режим TRACE. В режиме .ENABLE TRACE при обработке управляющего файла на терминал выводится каждая командная строка управляющего файла. При этом перед каждой командной строкой с директивной программа IND печатает восклицательный знак. Командные строки, содержащие только команды или только комментарии начинаются соответственно с точки или точки с запятой. Режим TRACE эквивалентен переключателю /T.

6.9.29. Директива .END отмечает конец блока управляющего файла. Директива .END имеет следующий формат:

.END

Информация следующая за директивой **.END** на той же строке игнорируется. Если программа **IND** встречает директив **.END** больше чем директив **.BEGIN**, на терминал выдается сообщение об ошибке.

6.9.30. Директива **.ERASE**. По директиве **.ERASE** из таблицы имен удаляются глобальные и локальные имена. После использования директивы **.ERASE** можно переопределять глобальные и локальные имена.

Формат директивы:

.ERASE LOCAL [им]

.ERASE GLOBAL [им]

где им — имя, которое надо удалить из таблицы имен.

Если в формате директивы не указано имя, то удаляются все глобальные или локальные имена из таблицы глобальных или локальных имен.

Примеры:

1. **.ERASE GLOBAL DEV**

2. **.ERASE GLOBAL**

3. **.ERASE**

В первом примере имя **DEV** удаляется из таблицы глобальных имен.

Во втором примере удаляются все глобальные имена из таблицы глобальных имен.

В третьем примере удаляются все глобальные и все локальные имена из таблицы локальных и глобальных имен.

Пример.

.ERASE LOCAL

Если эта командная строка указана в блоке управляющего файла, то удаляются локальные имена только из этого блока. Если эта командная строка задана вне блока управляющего файла, то удаляются все локальные имена, определенные вне блоков.

6.9.31. Директива **.EXIT**. По директиве **.EXIT** завершается обработка текущего управляющего файла или блока управляющего файла и управление передается предыдущему уровню управляющего файла. Если директива встречается в первом управляющем файле, программа **IND** передает управление **KMON**.

Формат директивы:

.EXIT [знач]

где знач — произвольное числовое выражение.

Пример.

В файле FILE1 появляется строка:

@ FILE2

файл FILE2 содержит:

.EXIT

Программа IND встречает директиву .EXIT в файле FILE2 и передает управление файлу FILE1.

Если в файле FILE2.COM директива .EXIT встречается в следующем виде:

.EXIT N+2, программа IND вычисляет выражение и присваивает значение специальному имени <EXSTAT>.

Когда программа IND достигает конца файла, то действия идентичны действиям директивы .EXIT.

6.9.32. Директива .GOSUB. По директиве .GOSUB осуществляется переход на метку, указанную в директиве, для вызова подпрограммы. Максимальная глубина вложения подпрограмм — восемь.

Формат директивы:

.GOSUB метка

где метка — точка входа в подпрограмму.

Для выхода из подпрограммы используется директива .RETURN.

Пример.

.GOSUB VOL

Управление передается на метку VOL.

6.9.33. Директива .GOTO. По директиве .GOTO осуществляется переход на другую строку управляющего файла. Все строки между директивой .GOTO и меткой, указанной в директиве игнорируются. С помощью этой директивы можно переходить как к начальным строкам файла так и к последним. По директиве .GOTO нельзя переходить к вложенному управляющему файлу, но можно переходить к любой строке управляющего файла после вложенного файла. Директива .GOTO, встречающаяся в блоке управляющего файла, просматривает блок от директивы .BEGIN до .END. В этом случае директива .GOTO действительна только в этом блоке и переход осуществляется только к ячейкам этого блока.

Формат директивы:

.GOTO метка ,

где метка — точка перехода на другую командную строку. Метка, используемая в директиве .GOTO не должна включать начальную точку и завершающее двоеточие.

Пример.

.GOTO 100

·
·
·

.100:

Метка 100 является точкой перехода на другую командную строку.

6.9.34. Директивы ветвления. Программа IND имеет ряд директив, которые выполняют логические проверки. Если в результате проверки устанавливается истинное значение, программа IND обрабатывает оставшуюся часть командной строки. Далее дается подробное описание директив ветвления.

6.9.35. Директива .IF. С помощью директивы .IF осуществляется сравнение числового имени или строкового с другим именем того же типа для определения истинности одного из нескольких возможных условий. Если после сравнения устанавливается истинное значение, программа IND выполняет оставшуюся часть командной строки. При сравнении строковых имен могут устанавливаться следующие отношения: меньше, больше, меньше или равно, больше или равно, не равно, а также различия между символами верхнего и нижнего регистров в строке символов.

Формат директивы:

.IF им оп врж дств

где им — числовое или строковое имя;
оп — один из следующих операторов:

EQ или = — равно;

NE или < > — не равно;

GE или > = — больше или равно;

LE или < = — меньше или равно;

GT или > — больше;

LT или < — меньше;

врж — выражение для имени того же типа;

дств — директива, по которой обработка продолжается.

Пример.

.SETS X «A»

.SETS Y «A»

.SETS Z «B»

.IF X EQ Y .GOTO 200

.IF Z NE Y .GOTO 100

Если значение, присвоенное строковому имени X, равно значению, присвоенному строковому имени Y, то управление переходит на метку 200. Если значение, присвоенное строковому имени Z, не равно значению, присвоенному строковому имени Y, то управление передается на метку 100.

6.9.36. Директивы .IFDF/.IFNDF. По директивам .IFDF и .IFNDF проверяют определено ли логическое, числовое или строковое имя или нет. Если в результате проверки получено истинное значение, то программа IND обрабатывает оставшуюся часть командной строки.

Формат директив:

.IFDF им дств

.IFNDF им дств

где им — имя, которое требует проверки;

дств — директива, по которой обработка продолжается.

Пример.

.IFDF A .GOTO 100

.IFNDF A .ASK A DO YOU WANT TO SET TIME .

Если имя A определено, то управление передается на метку 100, в противном случае программа IND обрабатывает директиву .ASK.

6.9.37. Директивы .IFENABLED и IFDISABLED. С помощью директив .IFENABLED и IFDISABLED проверяют разрешен ли определенный операционный режим (IFENABLED) или запрещен (IFDISABLED). Если при проверке получен истинный результат программа IND обрабатывает командную строку до конца.

Формат директив:

.IFENABLED опр дств

.IFDISABLED опр дств

где опр — операционный режим, который надо проверить;

дств — директива, по которой обработка продолжается.

Пример.

.IFENABLED DCL .GOTO COM

Управление передается на метку COM, если режим разрешен.

6.9.38. Директива .IFLOA и .IFNLOA. По этим директивам проверяют загружен драйвер или нет. Если при проверке устанавливается истинное значение, то IND продолжает обработку строки.

Формат директив:

.IFLOA уст дств

.IFNLOA уст дств

где уст — имя драйвера устройства, предназначенного для проверки;

дств — директива для продолжения обработки.

Пример.

.IFLOA DY1 .GOTO LOG

Если драйвер DY1 загружен, то осуществляется переход на метку LOG и обработка продолжается.

6.9.39. Директивы .IFT и .IFF проверяют является ли логическое имя истинным или ложным и соответствуют ли разряды, установленные в числовом имени, разрядам, установленным в маске.

Формат директив:

.IFT лимим дств

.IFT [маска] числим дств

.IFF лимим дств

.IFT [маска] числим дств

где лимим — логическое имя;

маска — аргумент, представляющий собой число или выражение в диапазоне от 0 до 177777 (восьмеричное) или от 0 до 65535 (десятичное), для которого надо проверить разряды на нуль и единицу. Аргумент должен быть заключен в квадратные скобки;

числим — числовое имя;

дств — директива для продолжения обработки.

Если при использовании директивы .IFT с логическим именем, значение логического имени устанавливается на истинно, то программа IND обрабатывает оставшуюся часть командной строки. В противном случае обрабатывается следующая командная строка.

Пример.

.SETT LOGA

.IFT LOGA .GOTO 100

При проверке значение логического имени устанавливается на истинно, управление передается на метку 100.

При использовании числового имени с директивой .IFT устанавливается соответствие между разрядами, установленными истинно в маске и в значении числового имени. Если в значении числового имени на истинно установлены те же разряды, что и в маске, то программа IND обрабатывает оставшуюся часть командной строки. В противном случае IND переходит на следующую командную строку.

Пример.

.SETT [5] NUMB

`.IFT [7] NUMB .GOTO 10`

Если числа 5 и 7 представить в двоичной системе, то получим соответственно 101 и 111. Разряды, установленные на «истинно» в маске директивы `.IFT` соответствуют разрядам, установленным на «истинно» в значении числового имени, следовательно управление передается на метку 10.

При использовании директивы `.IFF` с логическим именем программа `IND` осуществляет проверку на истинность значения логического имени. Если значение ложно, то программа `.IND` обрабатывает оставшуюся часть командной строки. В противном случае переходит на следующую командную строку.

Если в директиве `.IFF` используется числовое имя, проверяется соответствие на истинность разрядов, установленных в маске и значении числового имени. Если это соответствие не соблюдается, то программа `IND` обрабатывает оставшуюся командную строку, в противном случае переходит на другую строку.

Примеры:

1. `.SETF LOGA`
 `.IFF LOGA .GOTO 100`
2. `.SETF [5] NUMB`
 `.IFF [7] NUMB .GOTO 100`

В первом примере значение логического имени ложно, управление передается на метку 100.

Во втором примере по директиве `.SETF` первый и третий разряды в маске устанавливаются на нуль, т. е. значение числового имени в двоичной системе имеет вид 000. После проверки соответствия разрядов по директиве `.IFF` установлено, что разряды, установленные на истинно в маске директивы `.IFF` не соответствуют тем же разрядам в значении числового имени, следовательно управление передается на метку 100.

6.9.40. Директивы `.OR` и `.AND`. С помощью директив `.OR` и `.AND` можно объединять директивы ветвления на одной строке. Директива `.AND` обязательна при перечислении, если перед ней использована директива `.OR`.

Примеры:

1. `.IFT A .OR .IFT B .AND .IFT C .GOTO D`
2. `.IFT A .AND .IFF B .GOTO HELP`

При использовании нескольких директив ветвления на одной строке директиву `.AND` можно между ними опускать, как в следующем примере.

Пример.

.IFT A .IFF B .GOTO HELP

6.9.41. Директива .INC. По директиве .INC значение числового имени наращивается на единицу.

Пример.

.INC UNITS

значение числового имени UNITS будет увеличено на единицу.

6.9.42. Директива .ONERR. По директиве .ONERR обработка файла переходит на метку, указанную в директиве, если происходит одна из ошибок, перечисленных ниже:

?IND—F—BAD GANGE OR DEFAULT SPECIFICATION

?IND—F—DATA FILE ERROR

?IND—F—DATA FILE OPEN

?IND—F—DELETING SPESIAL SYMBOL

?IND—F—ERROR READING FROM TERMINAL

?IND—F—FILE ALREADY OPEN

?IND—F—FILE NOT OPEN

?IND—F—FILE READ ERROR

?IND—F—INVALID FILE NUMBER

?IND—F—INVALID KEYWORD

?IND—F—INVALID NESTING

?IND—F—LABEL NOT AT BEGINING OF LINE

?IND—F—NULL CONTROL STRING (TO .PARSE DIRECTIVE)

?IND—F—REDEFINING SYMBOL TO DIFFERENT TYPE
<SYMBOL>

?IND—F—.RETURN WITHOUT .GOSUB

?IND—F—STRING SUBSTITUTION ERROR

?IND—F—SUBROUTINE NESTING TOO DEEP

?IND—F—SWAP ERROR

?IND—F—SYMBOL TYPE ERROR <SYMBOL>

?IND—F—UNDEFINED LABEL <.LABEL>

?IND—F—UNDEFINED SYMBOL <SYMBOL>

формат директивы:

.ONERR метка ,

где метка — метка, указывающая переход на другую строку.

При обнаружении ошибки программа IND выдает сообщение об ошибке и останавливает выполнение косвенного управляющего файла. Если используется директива .ONERR, программа IND передает управление метке, указанной в директиве и продолжает выполнение косвенного управляющего

файла. Директива .ONERR должна быть помещена перед строкой, в которой программа IND обнаруживает ошибку. В управляющем файле можно использовать несколько директив .ONERR, но каждая директива обрабатывает только одну командную строку, в которой произошла ошибка. Причем, каждый раз когда IND обнаруживает ошибку, управление переходит к метке в директиве, указанной ранее командной строки, в которой произошла ошибка. Директива .ONERR более эффективна в пределах блока управляющего файла.

6.9.43. Директивы .OPEN, .OPENA, .OPENR. С помощью этих директив программа IND открывает файлы для работы с ними. Одновременно может быть открыто до четырех файлов.

Формат директив:

.OPEN [#N] спф

.OPENA [#N] спф.

.OPENR [#N] спф ,

где N — номер файла (по умолчанию 0). Если разрешен режим замены, то вместо числа можно использовать любой другой символ, заключив его в кавычки;

спф — спецификация файла (по умолчанию .DAT).

С помощью директивы .OPEN создается новый файл. Если в формате директивы указан уже существующий файл, то программа IND удаляет данные из этого файла.

Пример.

.OPEN#0 SEC

создается файл SEC.DAT для ввода данных.

По директиве .OPENA открывается файл для редактирования. Если директива используется с несуществующим файлом, она имеет тот же эффект, что и директива .OPEN. Перед открытием файла надо убедиться, что для него не установлена защита.

Пример.

.OPENA#0 FIL.BAK

файл FIL.BAK открывается для редактирования.

По директиве .OPENR файл открывается для чтения.

6.9.44. Директива .PARSE. По директиве .PARSE строка символов делится на подстроки и подстрокам присваиваются указанные строковые имена.

Формат директивы:

.PARSE стр «упстр» сим1 сим2 ...

где стр — строка, которую надо разбить. В качестве строки

можно использовать строку символов или строковое имя. Строка символов должна быть заключена в кавычки;

«упстр» — управляющая строка, состоящая из ограничителей (разделителей) подстроки. Управляющая строка всегда должна быть заключена в кавычки;

сим1 сим2 ... — имена, присваивающиеся подстрокам, отделяются друг от друга пробелами.

Пример.

```
.PARSE «DY1:LIBR.OBJ» «.:» DEV FILE TYPE
```

В этом примере строка символов «DY1:LIBR.OBJ» разделяется на подстроки. Управляющая строка содержит двоеточие и точку, служащие ограничителями строки. Двоеточие служит ограничителем для первой подстроки DY1, которой присваивается имя DEV. Точка служит ограничителем второй подстроки — LIBR и ей присваивается имя FILE. Оставшаяся часть — OBJ является третьей подстрокой, которой присваивается имя TYPE. Если имен больше чем подстрок, то программа IND устанавливает дополнительные подстроки на нуль. Специальное имя <STRLEN> содержит число обработанных строк, включая нулевые.

Пример.

```
.PARSE MACFIL «,» COM A1 A2 A3 A4 A5
```

MACFIL содержит строку символов COPY FILE1.MAC, FILE2.MAC,, FILE3.MAC. В этом примере строка символов разбивается следующим образом:

COM — COPY

A1 — FILE1.MAC

A2 — FILE2.MAC

A3 — нуль

A4 — FILE3.MAC

A5 — нуль

<STRLEN> — 6

6.9.45. Директива .PURGE. По директиве .PURGE закрывается указанный выходной файл и освобождается номер файла. Если .PURGE используется с файлом, открытым по директиве .OPENA, программа IND не вносит изменения в этот файл. Если .PURGE используется после директивы .OPENR, то .PURGE имеет тот же эффект, что и директива .CLOSE.

Формат директивы:

```
.PURGE [#N]
```

где N — номер файла от 0 до 3 (по умолчанию 0).

6.9.46. Директива `.READ`. По директиве `.READ` происходит чтение строки символов КОИ-7 из файла ранее открытого по `.OPENR`. Строке символов присваивается строковое имя, указанное в директиве `.READ`.

Формат директивы:

`.READ [##N] стрим`

где N — номер файла от 0 до 3 (по умолчанию 0);

стрим — строковое имя, назначенное строке символов.

Директива `.READ` игнорирует нулевые символы.

6.9.47. Директива `.RETURN` указывается в конце подпрограммы для передачи управления в косвенный управляющий файл.

Формат директивы:

`.RETURN`

6.9.48. Директивы `.SETD/.SETO`. Посредством этих директив изменяется основание системы счисления, значения числового имени на десятичное (`.SETD`) или восьмеричное (`.SETO`).

Формат директив:

`.SETD числим`

`.SETO числим`

где числим — числовое имя.

Пример.

`.SETD UNITS 10.`

`.SETO UNITS`

Числовое имя имеет значение 10 (десятичное). После использования директивы `.SETO` числовому имени присваивается значение 12 (восьмеричное).

6.9.49. Директива `.SETL`. По директиве `.SETL` значение логического имени устанавливается на «истинно» или «ложно» в зависимости от значения логического выражения. Если логическое имя не было определено, программа `IND` делает запись в таблицу имен и устанавливает значение логического имени в соответствии с логическим выражением. Если логическое имя было ранее определено, программа `IND` восстанавливает его значение. Если логическое имя было ранее определено как имя другого типа, то программа `IND` выдает сообщение об ошибке.

Формат директивы:

`.SETL лим лвр`

где лим — логическое имя;

лвр — логическое выражение, включающее буквенно-цифровые символы, соединенные логическими операторами &

(логическое И), ! (логическое ИЛИ), $\wedge$ (логическое НЕТ). Нельзя в логическом выражении использовать пробел или табуляцию. Программа IND обрабатывает логические выражения слева направо, если не использованы скобки, содержащее, которых обрабатывается в первую очередь. Любое значение в выражении может быть как восьмеричным так и десятичным, но все значения должны быть или восьмеричные или десятичные.

Пример.

.SETL MONT SJIFBIXM

Если любой из трех символов логического выражения установлен на «истинно», то логическое имя MONT также устанавливается на «истинно». Если ни один из этих символов не установлен на истинно, то логическое имя устанавливается на «ложно».

6.9.50. Директива .SETN. По директиве .SETN определяется значение числового имени. Если числовое имя ранее не определено, программа IND делает запись в таблицу имен. Если имя ранее определено программа IND восстанавливает его значение.

Формат директивы:

.SETN числим чвр

где числим — числовое имя;

чвр — числовое выражение. Числовые выражения получаются посредством объединения числовых имен и констант. В числовых выражениях нельзя использовать пробелы и табуляцию. Числовые имена обрабатываются слева направо, если в числовом выражении встречаются квадратные скобки, то сначала обрабатывается содержимое скобок. Все значения в директиве .SETN могут быть как десятичными так и восьмеричными.

Примеры:

1. **.SETN NUMBER 27.**

2. **.SETN A1 3*(A2—5)**

В первом примере числовому имени присваивается значение 27 (десятичное).

Во втором примере числовому имени A1 присваивается значение числового выражения.

6.9.51. Директива .SETS. По директиве .SETS определяется значение строкового имени. Если строковое имя не было определено, программа IND делает запись в таблицу имен и устанавливает значение строкового имени. Если строковое

имя было ранее определено, как имя другого типа, то программа IND выдает сообщение об ошибке.

Формат директивы:

.SETS стрим стрвр ,

где стрим — строковое имя;

стрвр — строковое выражение, создающееся путем объединения строковых имен и констант с помощью символа плюс (+). Константы должны быть заключены в кавычки.

Примеры:

1. .SETS A «ABCDEF»

2. .SETS X STR2+«ABC»

В первом примере строковому имени A назначается строка символов «ABCDEF».

Во втором примере строковому имени X назначается строковое выражение, состоящее из строкового имени STR2 (которое содержит ZZZ) и символов ABC. В результате строковому имени X присваивается значение ZZZABC.

6.9.52. Директивы .SETT и .SETF. По этим директивам значение логического имени устанавливается на истинно (.SETT) или ложно (.SETF). Также эти директивы используются для определения или переопределения числового имени посредством установки (.SETT) или очистки (.SETF) разрядов. Если логическое или числовое имя, указанные в этих директивах ранее не были определены, то программа IND делает запись в таблицу имен. Если логическое или числовое имя были ранее определены как имена другого типа, то программа IND выдает сообщение об ошибке.

Формат директив

.SETT лим

.SETT [маска] числим

.SETF лим

.SETF [маска] числим ,

где лим — логическое имя;

[маска] — значение в диапазоне от 0 до 177777 (восьмеричное) или от 0 до 65535 (десятичное), разряды которого устанавливаются или очищаются. Этот аргумент необходимо заключать в скобки;

числим — числовое имя, значение которого переопределяется посредством установки или очистки разрядов.

Пример.

.SETT X

.IFF X .GOTO 300

логическое имя X устанавливается на истинно, управление передается на метку 300.

При использовании директивы .SETT с параметром [маска] необходимо, чтобы каждому разряду, установленному на единицу в маске, соответствовал разряд в значении числового имени также, установленный на единицу. Каждому разряду в маске, установленному на нуль, соответствующий разряд в значении числового имени остается неизменным.

Пример.

```
.SETT [7] NUM  
.IFT [7] NUM .GOTO SUB1
```

По директиве .SETT разряды 1-ый, 2-ой и 3-ий в значении числового имени устанавливаются на единицу. При обработке директивы .IFT устанавливается соответствие между разрядами, установленными на единицу в маске и разрядами, установленными на единицу в значении числового имени, поэтому управление переходит на метку SUB1.

При использовании директивы .SETF с маской происходит очистка тех разрядов, установленных на единицу в значении числового имени, для которых соответствующие разряды в маске установлены на единицу.

Пример.

```
.SETF [12] NUM  
.IFF [22] NUM .GOTO SUB1
```

По директиве .SETF разряды 1-ый и 3-ий в значении числового имени устанавливаются на нуль. Остальные остаются без изменения. При обработке директивы .IFF разряды, установленные на нуль в значении числового имени, соответствуют разрядам, установленным на нуль в маске, поэтому управление передается на метку SUB1.

6.9.53. Директива STOP. Если во время обработки управляющего файла, встречается директива .STOP, обработка файла останавливается и на терминал выводится сообщение:

@ <EOF>

формат директивы:

```
.STOP
```

Директива .STOP имеет то же действие, что и директива логический конец файла (/).

6.9.54. Директива .TEST используется для определения типа имени, для проверки символов строкового имени на коды КОИ-7, для нахождения начальной позиции строки символов КОИ-7 или для установления основания системы счисления.

Формат директивы:

.TEST им

.TEST стрим стрсмв

где им — имя, тип которого надо установить;

стрим — строковое имя строки символов, которая проверяется на коды КОИ-7;

стрсмв — строка символов, для которой надо найти начальную позицию.

Для определения типа имени по директиве .TEST программа IND записывает соответствующий код в специальное имя <SYMTYP>. Если имя является строковым, программа IND соответственно устанавливает специальные имена <ALPHAN>, <RAD50> и <STRLEN>. Если имя — числовое, программа IND устанавливает специальное имя <OCTAL> на «истинно» или «ложно» в зависимости от того, в какой системе счисления задано значение числового имени.

Для определения начальной позиции строки символов в директиве .TEST указывается эта строка и в специальном имени <STRLEN> содержится абсолютное значение (восьмеричное) начальной позиции строки, если она найдена.

Пример.

.SETS ADDR «STREET»

.TEST ADDR «STREET»

.IF <STRLEN>=0 .GOTO NOT

.IF <STRLEN> <> 0 .GOTO POS

По директиве .SETS строковому имени ADDR присваивается строка символов «STREET» и затем осуществляется поиск начальной позиции этой строки. Если специальное имя <STRLEN> имеет значение 1, то строка STREET начинается с первой позиции в целевой строке. Строка «STREET» будет найдена в целевой строке, если они обе представлены или символами нижнего регистра, или верхнего.

Для определения основания системы счисления проверяют содержимое специального имени <OCTAL>. Если это специальное имя установлено на ложно, то значение числового имени дано в десятичной системе, если это специальное имя установлено на истинно, то значение числового имени дано в восьмеричной системе.

6.9.55. Директива .TESTDEVICE. По директиве .TESTDEVICE пользователь получает информацию об указанном устройстве.

Формат директивы:

.TESTDEVICE уст

где уст — имя устройства, о котором надо получить информацию. Двоеточие в имени устройства не обязательно.

Результат проверки хранится в специальном имени <EXSTRI>. Можно использовать директивы .PARSE и .TEST для разделения и проверки символов в строке. Если указанное устройство недействительно, специальное имя <EXSTRI> содержит NSD (нет такого устройства). Если устройство действительно, то в соответствующих полях устанавливается информация, приведенная в табл. 45.

Таблица 45

Поле	Содержимое
1	2
1	Имя устройства, если номер привода не указан, то устанавливается нуль
2	Размер устройства (в десятичной системе)
3	Нуль
4	Нуль
5	Нуль
6	LOD (если драйвер загружен) UNL (если драйвер не загружен)
7	ONL — если том установлен и его можно прочесть; OFL — если происходит ошибка чтения, если логический диск по какой-то причине не доступен; UNK — если указано устройство последовательного доступа или предназначенное только для записи
8	MTD — если на устройстве выделен логический диск; NMT — если на устройстве нет логического диска

Пример.

```
1. MOUNT LD2: DL1:MASTER.DSK SEK
   .TESTDEVICE SEC
   ; '<EXSTRI>'
   ; LD2,5000. ,0,0,0,UNL,ONL,MTD,
```

В данном примере файл DL1:MASTER.DSK ставится в соответствие логическому диску LD2. Логическому диску

LD2: назначается логическое имя SEC. Содержимое <EXSTRI> содержит следующую информацию:

LD2 — имя устройства;

5000 — размер устройства;

три нулевых поля;

ONL — устройство готово работать;

MTD — выделен логический диск LD2:

6.9.56. Директива .TESTFILE. По директиве .TESTFILE проверяется существование указанного файла. Затем имя указанного файла помещается в специальное имя <FILESPEC> и результат заносится в специальное имя <FILERR>.

Формат директивы:

.TESTFILE спф

где спф — спецификация файла. (По умолчанию использует-ся устройство DK: и тип файла DAT).

6.9.57. Директива .VOL. По директиве .VOL идентификатору присваивается указанное строковое имя и заносится в таблицу имен.

Формат директивы:

.VOL стрим уст

где стрим — строковое имя, которое присваивается идентификатору тома;

уст — логическое имя устройства, или присвоенное ему ранее строковое имя в кавычках (если разрешен режим замены) или строка символов. Двоеточие следующие за именем устройства игнорируется.

Директива .VOL используется только с устройствами файловой структуры. Программа IND только читает идентификатор указанного тома и присваивает ему указанное строковое имя. Имя владельца не записывается. Длина идентификатора тома заранее определена (12 символов). Если длина идентификатора тома меньше 12 символов, то остальные места заполняются пробелами.

ПЕРЕЧЕНЬ ССЫЛОЧНЫХ ДОКУМЕНТОВ

1. Операционная система ФОДОС-2
Описание применения
2. Операционная система ФОДОС-2
АССЕМБЛЕР
Руководство программиста
3. Операционная система ФОДОС-2
АССЕМБЛЕР
Описание языка

4. Операционная система ФОДОС-2
Программы работы с файлами
Руководство оператора
5. Операционная система ФОДОС-2
Редактор текста
Руководство оператора
6. Операционная система ФОДОС-2
Отладчик и виртуальный отладчик
Руководство оператора
7. Операционная система ФОДОС-2
Монитор расширенной памяти
Руководство программиста
8. ФОРТРАН/ФОДОС-2
Транслятор с ФОРТРАНА
Руководство программиста
9. Операционная система ФОДОС-2
Системная макробιβлиотека
Руководство программиста
10. БЕЙСИК
Руководство программиста
11. Операционная система ФОДОС-2
Экранный редактор текста
Руководство оператора
12. Операционная система ФОДОС-2
Вспомогательные программы
Руководство оператора
13. Программное обеспечение диалоговых вычислительных комплексов «ЭЛЕКТРОНИКА МС 0502» и «ЭЛЕКТРОНИКА МС 0501». Начальная загрузка и копирование систем
Руководство оператора

ІЯ ЗАМЕТОК

ДЛЯ ЗАМЕТОК

ВСПОМОГАТЕЛЬНЫЕ ПРОГРАММЫ

РУКОВОДСТВО ОПЕРАТОРА

1. ПЕЧАТЬ

1.1. Назначение программы и условия выполнения программы

Программа печати (DUMP) предназначена для вывода на терминал или построчно-печатающее устройство, а также для записи на внешнее ЗУ, содержимого всего или части файла (тома) в восьмеричном коде, а также символов в кодах КОИ-7 и RADIX-50.

Программа DUMP может использоваться для проверки справочников и файлов, содержащих двоичные данные.

1.2. Выполнение программы

Для вызова DUMP с системного устройства следует подать с терминала команду

R DUMP <BK> или DUMP <BK>

после того, как монитор напечатает на терминале точку.

После вызова DUMP печатает звездочку (*) и ожидает ввода командной строки. Если в это время нажать клавишу <BK>, то DUMP напечатает номер своей версии.

Примечание. Операции, выполняемые программой DUMP, могут быть выполнены монитором по команде DUMP (см. [2]).

1.3. Команды оператора

Режим работы программы DUMP задается ведением с терминала командной строки. Формат командной строки:

[выходспф=] входспф [/прк...]

где выходспф — спецификация выходного файла (устройство, имя и тип файла);

входспф — спецификация входного файла;

/прк — переключатель (см. табл. 1).

Если в выходной спецификации файла (выходспф) пропущено одно или более полей, то система по умолчанию предполагает следующее:

- 1) устройство — построчно-печатающее устройство:
- 2) тип файла — .DMP.

Таблица 1

ПЕРЕКЛЮЧАТЕЛИ ПРОГРАММЫ DUMP

Переключатель	Функция
1	2
/B	Вывод по байтам
/E:N	Завершение вывода на блоке N ¹
/G	Игнорирование ошибок чтения
/N	Запрещение вывода символов в кодах КОИ-7
/O:N	Вывод блока N
/S:N	Вывод с блока N
/T	Структура файлов на магнитной ленте отличается от используемой в ФОДОС
/W	Вывод по словам (по умолчанию)
/X	Вывод символов в кодах RADIX-50

1) N — восьмеричное число, указывающее номер блока в файле.

При выводе тома магнитной ленты, DUMP считывает данные до логического конца ленты. Логическим концом ленты являются три последовательные метки — метка конца файла и две метки ленты. Если структура записанных на ленте файлов отличается от используемой в ФОДОС, то логическим концом ленты считаются две последовательные метки ленты. DUMP помещает в распечатку сообщения о встреченных метках ленты

***** TAPE MARK *****

Если при выводе тома магнитной ленты используется переключатель /S:N и N превосходит количество блоков, записанных на ленте, то DUMP выдает сообщение

?DUMP—F—READ ERROR

и прекращает выполнение операции. При использовании переключателя /E:N DUMP считывает данные с ленты до блока N или логического конца ленты, что встретится первым.

При работе с программой DUMP полезно использовать специальные команды монитора (см. [2]).

Для выхода из DUMP и передачи управления монитору следует подать команду CY/C, если DUMP ожидает ввода с терминала, или дважды CY/C, если DUMP выполняет операцию. Для повторного пуска DUMP следует подать команду REENTER <BK> или R DUMP <BK>.

После выполнения операции DUMP печатает звездочку и ожидает ввода командной строки.

1.4. Примеры

Ниже приведен фрагмент распечатки первого блока файла SY:PIP.SAV. В первой строке распечатки записаны спецификация входного файла и переключатели, заданные в командной строке, во второй — номер выводимого блока. Числа в крайней левой колонке (начиная с 3-ей строки) указывают адрес первого слова в строке относительно начала блока. Информация, содержащаяся в блоке, записана в следующих восьми колонках. Она представлена в восьмеричной системе. Переключатель /N запрещает вывод символов в кодах КОИ-7.

PIP.SAV/N/O:1

BLOCK NUMBER 000001

```
000/ 060502 010046 010146 010246 000422 062701 001100 012102
020/ 022512 001406 012100 005046 011146 010246 104217 103405
040/ 012602 012601 012600 011505 000205 104376 175400 012767
060/ 011501 177724 016701 000012 005021 020167 000006 103774
100/ 000743 005562 015260 005562 000006 002112 005562 000013
```

```
760/ 000000 000000 000000 000000 000000 000000 000000 000000
```

Фрагмент распечатки четвертого блока файла

SY:SYSMAC.MAC, приведенный ниже, иллюстрирует вывод по байтам. Информация, содержащаяся в блоке, записана в шестнадцати колонках справа. Она представлена в восьмеричной системе и символах в кодах КОИ-7.

SYSMAC.MAC/B/O:4

BLOCK NUMBER 000004

```
000/015012056105116104115015012056115101103122117040
```

```
... E N D M ... M A C R O W
020/0560560561031150610111011 22105101054111103054103
```

```

. . . C M 1 . A R E A , I C , C
040/110101116054106114101107054101122107054111116123
    H A N , F L A G , A R G , I N S
060/054103123105124054102102015012056056056103115065
    , C S E T , B B . . . C M 5
100/011074101122105101076015012056056056126062075060
    . < A R E A > . . . V 2 = 0

```

```

760/060673015012056056056103115062011074101122107076
0 ; . . . C M 2 . < A R G >

```

Ниже приведен фрагмент распечатки шестого блока устройства DX0. Информация, содержащаяся в блоке, записана в восьми колонках справа. Она представлена в восьмеричной системе и символах в кодах RADIX-50.

DX0:/N/X/O:6

BLOCK NUMBER 000006

```

000/ 000004 000000 000001 000000 000016 002000 016300 000000
      D              A              N   YX   DX
020/ 075273 000004 000000 015454 002000 062170 000000 075273
      SYS        D              DM6 YX   PC              SYS
040/ 000002 000000 015454 002000 062074 012000 073376 000011
      B              DM6   YX   PAT   CH   SAV   I
060/ 000000 007711 002000 016130 000000 073376 000055 000000
      BUA   YX   DUP              SAV   AE
100/ 015454 002000 074070 062000 073376 000025 000000 015454
      DM6   YX   SIP   P   SAV   U              DM6

```

```

760/ 000000 000000 000000 000000 000000 000000 000000 000000

```

1.5 Сообщения оператору

Ниже описаны сообщения, выдаваемые программой DUMP.

?DUMP—F—DEVICE FULL DEV:FILNAM.TYP

Причина. На устройстве недостаточно места для выходного файла.

Действие. Увеличить размер свободной области на томе: удалить ненужные файлы или переписать их на другой том, сжать том по команде SQUEEZE.

?DUMP—F—FILE OR INPUT DEVICE NOT FOUND

Причина. Возможны следующие причины:

— не найден входной файл;

— в системе нет драйвера заданного устройства ввода;

— в мониторе не установлено обслуживание заданного устройства ввода.

Действие. Возможны следующие действия:

— ввести правильную командную строку;

— установить правильное обслуживание требуемого устройства.

?DUMP—F— INPUT ERROR DEV:FILNAM.TYP

Причина. Ошибка чтения.

Действие. Проверить готовность и исправность оборудования.

Повторить выполнение операции.

?DUMP—F— INSUFFICIENT MEMORY

Причина. Недостаточно оперативной памяти для выполнения заданной операции.

Действие. Освободить часть оперативной памяти (удалить ненужные драйверы, удалить основное задание, использовать монитор одного задания). Повторить выполнение операции.

?DUMP—F—INVALID COMMAND

Причина. В командной строке обнаружена ошибка.

Действие. Ввести правильную командную строку.

?DUMP—F—INVALID OPTION:/X

Причина. В командной строке обнаружен недопустимый переключатель.

Действие. Ввести правильную командную строку.

?DUMP—F—NO LP

Причина. В системе нет драйвера постстрочно-печатающего устройства.

Действие. Повторить выполнение операции, задав доступное устройство вывода.

?DUMP—F—OUTPUT ERROR DEV:FILNAM.TYP

Причина. Возможны следующие причины:

— ошибка записи;

— на том, используемом для вывода, нет свободной области для записи выходного файла.

Действие. Возможны следующие действия:

— проверить готовность и исправность оборудования;

— проверить том на плохие блоки. Повторить выполнение операции;

указать в командной строке размер выходного файла;

— использовать для вывода другой том.
?DUMP—F—PROTECTED FILE ALREADY EXISTS
DEV:FILNAM.TYP

Причина. Попытка создать файл с тем же именем, что и у имеющегося защищенного файла.

Действие. Снять защиту имеющегося файла по команде монитора UNPROTECT или по переключателю /Z программы PIP, или создать файл с другим именем.

?DUMP—F—UNEXPECTED EOF DEV:FILNAM.TYP

Причина. В заданной командной строке в переключателе /E:N (/O:N, /S:N) значение N превосходит длину входного файла.

Действие. Определить действительный размер входного файла. Повторить команду.

2. СРАВНЕНИЕ ТЕКСТОВЫХ ФАЙЛОВ

2.1. Назначение программы и условия выполнения программы.

Программа сравнения текстовых файлов (SRCCOM) предназначена для сравнения двух файлов, содержащих символическую информацию, и выявления различий между ними. Перечень различий программа может выдавать на терминал или формировать в выходной файл. Программа SRCCOM используется также для создания входного командного файла для выполнения его по программе модификации текстовых файлов (SLP).

Программа SRCCOM сравнивает два входных файла строка за строкой, отыскивая совпадающие строки. Когда программа обнаруживает несовпадение, она выводит отличающиеся строки каждого файла до тех пор, пока указанное число строк из первого файла не будет совпадать со строками из второго файла.

Программа SRCCOM применяется также для проверки правильности внесения изменений в исходные программы, сделанных во время редактирования.

2.2. Выполнение программы

Для вызова SRCCOM с системного устройства следует подать с терминала команду

R SRCCOM <БК> или SRCCOM <БК>

после того, как монитор напечатает на терминале точку.

После вызова SRCCOM печатает звездочку и ожидает ввода командной строки. Если в это время нажать клавишу <BK>, то SRCCOM напечатает номер своей версии.

Примечание. Операции, выполняемые программой SRCCOM, могут быть выполнены по команде монитора DIFFERENCES (см. [2]).

2.3. Команды оператора

Режим работы программы SRCCOM задается введением с терминала командной строки. Формат командной строки:

[[выходспф] [SLP—спф]=] входспф 1, входспф 2 [/прк...]

где выходспф — спецификация выходного файла (устройство, имя и тип файла);

SLP—спф — спецификация командного файла для программы SLP;

входспф1 — спецификация первого сравниваемого файла (старого);

входспф2 — спецификация второго сравниваемого файла (нового);

прк — переключатель (см. табл. 2).

Входные файлы указываются в любом порядке, если они сравниваются для выявления различий. Входные файлы указываются в порядке (старый), (новый), если создается командный файл для использования программой SLP.

Если в командной строке указаны «Выходспф» и «SLP—спф», программа SRCCOM создает файл различий и входной командный файл для программы SLP. Если «Выходспф» не указана, перед «SLP—спф» необходимо поместить (,) для обозначения отсутствия «Выходспф».

Таблица 2

ПЕРЕКЛЮЧАТЕЛИ ПРОГРАММЫ

Переключатель	Назначение
1	2
/A	Указывает строку символов, характеризующую каждую обновленную строку измененного текстового файла. Используется со спецификацией SLP, создаваемый файл может быть использован как входной командный файл для SLP

1	2
/B	Включает в сравнение пустые строки; обычно SRCCOM их игнорирует
/C	Игнорирует при сравнении комментария (весь текст которому предшествует точка с запятой), пробелы и табуляцию. Однако, строки, полностью состоящие из комментария, включаются в счетчик строк
/D	Печатает текст второго (нового) сравниваемого файла, указанного в командной строке, с отличиями от первого (старого) сравниваемого файла с указанием признаков [\] — для вставки и (O) — для вычеркивания строк
/F	Включает в выходной файл символы перевода формата; SRCCOM обычно сравнивает символы перевода формата, но не включает их в выходной файл
/L:N	Указывает число совпадающих строк; N — целое восьмеричное число от 1 до 310, по умолчанию N=3
/S	Игнорирует при сравнении пробелы и табуляцию
/T	Включает в сравнение пробелы и табуляцию, находящиеся в конце строки; обычно SRCCOM их игнорирует
/V:I:D	Используется с переключателем /D для указания символов, заменяющих признаки (\) и (O). I, D — символы в кодах КОИ-7 (от 40 до 176); I — код признака вставки, D — код признака вычеркивания

Устройство для вывода по умолчанию — системный терминал, тип файла по умолчанию для входных файлов — .MAC, тип файла по умолчанию для выходного файла — .DIF, тип файла по умолчанию для командного файла программы SLP — .SLP.

После выполнения операции SRCCOM печатает звездочку и ожидает ввода новой командной строки. Для выхода из программы SRCCOM и передачи управления монитору следует подать команду CY/C, если SRCCOM ожидает ввода с терминала, или дважды CY/C, если SRCCOM выполняет операцию.

Для повторного пуска программы SRCCOM следует подать с терминала команду монитора REENTER <BK> или R SRCCOM <BK>.

2.4. Использование конструкции (*) в командной строке

Конструкция (*) используется для многократного сравнения текстовых файлов при введении с терминала только одной командной строки.

При создании входного командного файла для программы SLP конструкция (*) не используется.

Конструкция (*) может использоваться в спецификации как первого, так и второго сравниваемого текстового файла.

Если (*) используется в одной из спецификаций входных файлов, то программа SRCCOM сравнивает текстовый файл, в спецификации которого (*) не указана, со всеми вариантами файла, в спецификации которого указана (*). Конструкция (*) указывает ту часть спецификации файла, которая будет изменяться.

Например, когда выполняется командная строка

TEST=DY0.TEST1.MAC,DY1:TEST2.

SRCCOM сравнивает текстовый файл TEST1.MAC на устройстве DY0: со всеми файлами с именем TEST2 на устройстве DY1:. Результаты всех сравнений программа может выдавать на терминал или формировать в выходной файл. В данном примере перечень различий программа SRCCOM формирует в выходной файл TEST.DIF.

Если (*) используется в обеих спецификациях входных файлов, то она указывает ту часть спецификации файлов, которая должна быть одинаковой в сравниваемых файлах. Например, когда выполняется командная строка

DY0:PROG1.,DY1:PROG2.*

SRCCOM сравнивает несколько пар текстовых файлов. Первый входной файл в каждой паре имеет имя PROG1, а второй — PROG2. Тип обоих файлов в каждой паре должен быть одинаковым. SRCCOM отыскивает первый файл с именем PROG1 на DY0:, затем отыскивает на DY1: файл с именем PROG2 и с тем же типом, что и PROG1. Если пара файлов найдена, SRCCOM сравнивает эти файлы и перечень

различий выдает на терминал или формирует в выходной файл, если он указан. Аналогично происходит сравнение следующих пар текстовых файлов.

2.5. Создание командного файла для SLP

Программа SRCCOM может использоваться для создания входного командного файла для SLP. В приведенном ниже примере командной строки SRCCOM выдает перечень различий на терминал и создает выходной файл MOD.SLP, который может быть использован как входной командный файл для SLP.

```
*.MOD=DEMO.BAK,DEMO.TXT  
.TYPE MOD.SLP
```

—2,2

—7,7

MOVB R0, (R12) +

—9,9

CLRB (R1) +

—12,12

/

Если используется переключатель /A, SRCCOM запрашивает признак модификации: AUDIT TRAIL?

Отвечать строкой не более 11 символов. Не использовать символ (/) в признаке модификации. Примером признака модификации могут являться авторские инициалы и дата модификации.

```
*.MOD=DEMO.BAK,DEMO.TXT/A  
AUDIT TRAIL? S.Z.27*MAY  
.TYPE MOD.SLP
```

—2,2,/,S.Z.27*MAY/

—7,7,/,S.Z.27*MAY/

MOVB R0, (R12) +

—9,9,/,S.Z.27*MAY/

CLRB (R1) +

—12,12,/,S.Z.27*MAY/

/

2.6. Примеры

В этом подразделе приведены результаты работы программы SRCCOM при сравнении текстовых файлов DEMO.BAK и DEMO.TXT.

```
. . . . .  
.TITLE TAIP  
.SBTTL KONTROL PRIMER  
.GLOBL SUBR1, SUBR2  
.CSECT PROG
```

```

START:  .MCALL  .TTYIN
        MOV     #BUFFER, R2
        MOVB    R0, (R2) +
        CMPB    R0, #LF
        CLRB    (R2) +
        JSR     PC, SUBR1
        BCS     START
        .TTYIN
BUFFER:  .BLKB   72
        .END    START
        . . . . .
        .TITLE  TAIP
        .GLOBL  SUBR1, SUBR2
        .CSECT  PROG
        .MCALL  .TTYIN
START:  MOV     #BUFFER, R2
        MOVB    R0, (R2) +
        CMPB    R0, #LF
        CLRB    (R2) +
        JSR     PC, SUBR1
        BCS     START
BUFFER: .BLKB   72
        .END    START

```

В первом примере число совпадающих строк устанавливается равным 1.

```

.R SRCCOM
*DEMO.BAK,DEMO.TXT/L:1
1) DK:DEMO.BAK
2) DK:DEMO.TXT
*****
1) 1 .SBTTL KONTROL PRIMER
1) .GLOBL SUBR1, SUBR2
****
2) 1 .GLOBL SUBR1, SUBR2
*****
1) 1 MOVB R0, (R2) +
1) CMPB R0, #LF
****
2) 1 MOVB R0, (R2) +
2) CMPB R0, #LF
*****
1) 1 CLRB (R2) +
1) JSR PC, SUBR1

```

```

*****
2) 1          CLR B      (R1) +
2)          JSR          PC, SUBR1
*****
1) 1          .TTYIN
1)          BUFFER: .BLKB      72
*****
2) 1          BUFFER: .BLKB      72
*****
?SRCCOM—W—FILES ARE DIFFERENT
*
```

Запись в левой части страницы имеет вид:

M)N

где M — номер исходного файла;

N — номер страницы.

Вначале SRCCOM печатает в качестве пояснения спецификации входных файлов (DK:DEMO.BAK и DK:DEMO.TXT), затем строку из десяти звездочек и различия между двумя файлами. Печатается строка различий из первого файла (.SBTTL KONTROL PRIMER). Следующая за ней, совпадающая в обоих файлах, строка (.GLOBL SUBR1, SUBR2) печатается для удобства отыскания строк различий в тексте. Четыре звездочки ограничивают печать строк различий из первого файла. После этого печатается строка различий и совпадающая строка из второго файла. Десять звездочек ограничивают печать строк второго файла.

Аналогично SRCCOM печатает остальные строки различий в файлах. По окончании печати различий на терминале печатается сообщение:

```
?SRCCOM—W—FILES ARE DIFFERENT
```

Во втором примере различия записываются на DK в файл DIFF.TXT, число совпадающих строк равно 3.

```
.R SRCCOM
```

```
*DIFF.TXT=DEMO.BAK, DEMO.TXT
```

```
?SRCCOM—W—FILES ARE DIFFERENT
```

```
*^C
```

```
.TYPE DIFF.TXT
```

```
1) DK:DEMO.BAK
```

```
2) DK:DEMO.TXT
```

```
*****
```

```
1) 1          .SBTTL      KONTROL PRIMER
```

```
1)          .GLOBL      SUBR1, SUBR2
```

```
*****
```



```
2) 1 .GLOBL SUBR1, SUBR2
```

```
*****
```

```
1) 1 MOVB R0, ((R2) +
1) CMPB R0, #LF
1) CLRB (R2) +
1) JSR PC, SUBR1
1) BCS START
1) .TTYIN
1) BUFFER: .BLKB 72
```

```
***
```

```
2) 1 MOVB R0, (R12) +
2) CMPB R0, #LF
2) CLRB (R1) +
2) JSR PC, SUBR1
2) BCS START
2) BUFFER: .BLKB 72
```

```
*****
```

Третий пример показывает результат использования переключателя программы (/D).

```
.R SRCCOM
```

```
*DEMO.BAK, DEMO.TXT/L:1/D
```

```

O .TITLE TAIP
  .GLOBL SUBR1, SUBR2
  .CSECT PROG
  .MCALL .TTYIN
  START: MOV #BUFFER, R2
  \      MOVB R0, (R12) +
  \      CMPB R0, #LF
  \      CLRB (R1) +
      JSR PC, SUBR1
      BCS START
O      BUFFER: .BLKB 72
      .END START
```

```
?SRCCOM—W—FILES ARE DIFFERENT
```

```
*
```

2.7. Сообщения оператору

Ниже описаны сообщения, выдаваемые программой SRCCOM.

```
?SRCCOM—F—CHANNEL IN USE DEV:FILNAM.TYP
```

Причина. Сбой в работе операционной системы.

Действие. Перезагрузить программу SRCCOM или систему и повторить выполнение операции. Если ошибка повторится снова, создать новую копию SRCCOM.SAV и повторить выполнение операции.

?SRCCOM—F—CHANNEL NOT OPEN DEV: FILNAM.TYP

Причина. Сбой в работе операционной системы.

Действие. Перезагрузить программу SRCCOM или систему и повторить выполнение операции. Если ошибка повторится снова, создать новую копию SRCCOM.SAV и повторить выполнение операции.

?SRCCOM—F—DEVICE FULL DEV: FILNAM.TYP

Причина. В справочнике выходного тома нет места для записи указанного файла.

Действие. Сжать том по команде SQUEEZE. Удалить или записать на другой том ненужные файлы. Использовать конструкцию [N] для выходного файла. Использовать другой том.

?SRCCOM—F—DEVICE NOT ACTIVE DEV:

Причина. Операция ввода—вывода невозможна на указанном устройстве или томе; устройство отключено; нет тома на устройстве; запрещена запись; устройство не описано в таблицах монитора.

Действие. Проверить правильность включения устройства. Установить обслуживание устройства по команде INSTALL.

?SRCCOM—F—DIRECTORY FULL DEV:

Причина. В справочнике выходного тома нет места для записи файла.

Действие. Сжать том по команде SQUEEZE. Удалить или записать на другой том ненужные файлы. Использовать другой том.

?SRCCOM—F—DIRECTORY I/O ERROR DEV: FILNAM.TYP

Причина. Ошибка чтения справочника указанного устройства.

Действие. Проверить готовность и исправность оборудования. Проверить диск на плохие блоки.

?SRCCOM—F—ERROR READING DIRECTORY DEV:

Причина. Аппаратная ошибка чтения справочника указанного устройства.

Действие. Проверить готовность и исправность оборудования. Проверить диск на плохие блоки.

?SRCCOM—F—FETCH ERROR DEV:

Причина. Сбой в работе операционной системы.

Действие. Перезагрузить программу SRCCOM или систему и повторить выполнение операции. Если ошибка повторится снова, создать новую копию SRCCOM.SAV и повторить выполнение операции.

?SRCCOM—F—FILE NOT FOUND DEV:FILNAM.TYP

Причина. Файл не найден на указанном устройстве.

Действие. Проверить командную строку.

?SRCCOM—F—ILLEGAL DEVICE DEV:FILNAM.TYP

Причина. В командной строке указано несуществующее или недопустимое устройство.

Действие. Проверить командную строку. Использовать другое устройство.

?SRCCOM—F—INPUT ERROR DEV:FILNAM.TYP

Причина. Аппаратная ошибка доступа к входному устройству.

Действие. Проверить готовность и исправность оборудования. Проверить диск на плохие блоки.

?SRCCOM—F—INSUFFICIENT MEMORY

Причина. Недостаточно оперативной памяти для выполнения операции.

Действие. Освободить часть оперативной памяти (удалить ненужные драйверы, удалить основное задание, использовать монитор одного задания). Повторить выполнение операции.

?SRCCOM—F—INVALID COMMAND

Причина. В командной строке обнаружена ошибка.

Действие. Ввести правильную командную строку.

?SRCCOM—F—INVALID DIRECTORY DEV:

Причина. Структура справочника тома устройства не является структурой ФОДОС.

Действие. Использовать справочник тома устройства со структурой ФОДОС.

?SRCCOM—F—INVALID OPTION:/X

Причина. В командной строке обнаружен недопустимый переключатель.

Действие. Ввести правильную командную строку.

?SRCCOM—F—INVALID VALUE SPECIFIED WITH OPTION:/X

Причина. К командной строке обнаружено недопустимое значение переключателя.

Действие. Ввести правильную командную строку, проверив границы допустимых значений переключателей.

?SRCCOM—F—OUTPUT ERROR DEV:FILNAM.TYP

Причина. Аппаратная ошибка записи. На томе недостаточно места для записи файла.

Действие. Проверить командную строку. Проверить готовность и исправность оборудования. Проверить диск на плохие блоки. Сжать том по команде SQUEEZE. Удалить или записать на другой том ненужные файлы.

?SRCCOM—F—OUTPUT FILE FULL DEV:FILNAM.TYP

Причина. На томе нет места для записи файла. Обнаружен маркер физического конца тома (для магнитной ленты).

Действие. Сжать том по команде SQUEEZE. Удалить или записать на другой том ненужные файлы. Использовать конструкцию [N] для выходного файла. Использовать другой том.

?SRCCOM—F—PROTECTED FILE ALREADY EXISTS DEV:FILNAM.TYP

Причина. Попытка создать файл с именем уже существующего защищенного файла.

Действие. Возможны следующие действия:
— использовать команду UNPROTECT или переключатель PIP/Z для отмены защиты существующего файла;
— использовать другое имя для создаваемого файла.

?SRCCOM—F—READ ERROR

Причина. Аппаратная ошибка чтения.

Действие. Проверить готовность и исправность оборудования. Повторить выполнение операции.

?SRCCOM—F—SYSTEM ERROR

Причина. Ошибка в системе.

Действие. Перезагрузить систему.

?SRCCOM—F—RECORD TOO LONG DEV:FILNAM.TYP

Причина. На томе недостаточно места для записи файла.

Действие. Сжать том по команде SQUEEZE. Удалить или записать на другой том ненужные файлы. Использовать конструкцию [N] для выходного файла. Использовать другой том.

?SRCCOM—F—TOO MANY DIFFERENCES

Причина. Программа встретила более 310 (восьмеричное) строк различий.

Действие. Возможны следующие действия:

- убедиться, что указана правильная командная строка;

- разбить файлы на части для сравнения.

?SRCCOM—F—UNEXPECTED EOF DEV:FILNAM.TYP

Причина. Ошибка при обращении к программе SRCCOM.

Действие. Перезагрузить систему и повторить выполнение операции. Создать новую копию SRCCOM.SAV и повторить выполнение операции.

?SRCCOM—F—WRITE ERROR

Причина. Возможны следующие причины:

- запрещена запись на устройство вывода;

- переполнен том устройства вывода;

- аппаратная ошибка записи.

Действие. Возможны следующие действия:

- разрешить запись на устройство вывода;

- использовать для вывода другой том или другое устройство;

- проверить диск на плохие блоки. Повторить выполнение операции.

?SRCCOM—F—WRONG VERSION

Причина. Попытка обращения системы ФОДОС к программе SRCCOM, принадлежащей системе ФОДОС-2.

Действие. Вызвать программу SRCCOM системы ФОДОС.

?SRCCOM—I—NO DIFFERENCES FOUND

Причина. Различий в файлах не найдено.

Действие. Не требуется.

?SRCCOM—W—AUDIT TRAIL TRUNCATED TO ELEVEN CHARACTERS

Причина. Признак модификации превысил одиннадцать символов и был усечен.

Действие. Ввести признак модификации верно, если усеченный признак модификации не подходит.

?SRCCOM—W—FILES ARE DIFFERENT

Причина. В файлах найдены различия.

Действие. Не требуется.

3. МОДИФИКАЦИЯ ФАЙЛОВ

3.1. Назначение программы и условия выполнения программы

Программа модификации файлов (PATCH) предназначена для изменения кодов любых типов файлов, допустимых в

ФОДОС (см. [1]). Программа позволяет открывать слова или байты файла, выводить информацию на печать в кодах КОИ-7 и RADIX-50 и вносить изменения в этих кодах в файл.

При работе с программой PATCH рекомендуется предварительно создавать копию модифицируемого файла, поскольку изменения вносятся непосредственно в файл.

Программа PATCH обслуживает только устройства с произвольным доступом к данным.

3.2. Выполнение программы

Для вызова PATCH с системного устройства следует подать с терминала команду

R PATCH <BK> или PATCH <BK>

после того, как монитор напечатает на терминале точку.

После вызова PATCH печатает

FILE NAME —

*

и ожидает ввода командной строки. Если в это время нажать клавишу <BK>, то PATCH напечатает номер своей версии.

3.3. Команды оператора

Режим работы программы PATCH задается введением с терминала командной строки. Формат командной строки:

спф [/прк ...]

где спф — спецификация модифицируемого файла (устройство, имя и тип файла);

прк — переключатель (см. табл. 3).

Тип файла по умолчанию — .SAV.

Таблица 3

ПЕРЕКЛЮЧАТЕЛИ ПРОГРАММЫ PATCH

Переключатель	Назначение
1	2
/A	Используется с именем файла или без него. Без имени файла применяется для восстановления разрушенных справочников на устройствах файловой структуры или для модификации начального загрузчика на нулевой блоке диска. С именем файла исполь-

1	2
/O	зается для модификации исходных файлов или файлов, имеющих тип, отличный от SAV. Используется с файлом, который является файлом монитора ФОДОС Указывает, что файл имеет оверлей- 1)
/C	ную структуру Запрос контрольной суммы. RATCH игнорирует переключатель, если изменения в файл не внесены
/D	Печать контрольной суммы конкрет- 2) ной «заплаты». RATCH игнорирует переключатель, если изменения в файл не внесены

1) оверлейным файлом будем называть файл, представляющий собой программу оверлейной структуры;

2) «заплатой» будем называть совокупность изменений, внесенных в файл в процессе модификации.

3.3.1. Команды RATCH

Для работы с программой RATCH используются команды, приведенные в табл. 4.

Таблица 4

КОМАНДЫ ПРОГРАММЫ RATCH

Команда	Функция
1	2
V;NR	Записать значение V в N-й регистр перемещения ($N=0, 1, \dots, 7$)
X;B	Установить младший адрес оверлейного файла равным X
R,O/	Открыть слово, адрес которого равен сумме содержимого регистра перемещения R и смещения O

1	2
R,O\	Открыть байт, адрес которого равен сумме содержимого регистра перемещения R и смещения O
S:R,O/	Открыть в оверлейном сегменте S слово, адрес которого равен сумме содержимого регистра перемещения R и смещения O. Номер оверлейного сегмента S определяется по карте загрузки
S:R,O\	Открыть в оверлейном сегменте S байт, адрес которого равен сумме содержимого регистра перемещения R и смещения O
<BK>	Заккрыть текущее открытое слово или байт
<PC>	Заккрыть текущее открытое слово или байт и открыть следующее слово или байт
Λ	Заккрыть текущее открытое слово или байт и открыть предыдущее слово или байт
@	Заккрыть текущее открытое слово и открыть слово, адресом которого является закрываемое слово (для оверлейных файлов адрес должен быть в том же самом сегменте)
F	Заккрыть текущий открытый файл и запросить новую командную строку. Если был указан переключатель /C, RATCH запрашивает, а если /D — выдает контрольную сумму
E	Заккрыть текущий открытый файл и выйти из программы RATCH (управление передается монитору). Если был указан переключатель /C, RATCH запрашивает, а если /D — выдает контрольную сумму
X;O	Установить значение X в оверлейном

1	2
&	драйвере или его таблицах ($X=0$ вызывает сообщение об ошибке) Прибавить к контрольной сумме содержимое всех последовательно открываемых адресов, пока не встретится второй &
A	Представление открытого слова или байта в виде символов в кодах КОИ-7 (если открыт байт — одного символа, слово — двух)
X	Представление открытого слова в виде трех символов в кодах RADIX-50
C(X[X])	Замена открытого слова или байта кодами КОИ-7 символов, введенных оператором (если открыт байт, подается один символ, если слово — два)
P(XXX)	Замена открытого слова кодами RADIX-50 трех символов, введенных оператором

В командах «С» и «Р» круглые скобки печатаются программой PATCH.

В процессе модификации строку с ошибочно напечатанными данными можно отменить по команде СУ/У, при этом ранее установленные регистры перемещения сохраняются. Для оверлейных файлов сохраняется номер сегмента, если среди использованных команд есть «/» или «ПС».

Примечания:

1. Если номер оверлейного сегмента в команде не указан, то предполагается корневой сегмент.

2. Если в команде «V;NR» вместо символа «;» указан символ «,», то указанное значение в регистр не запишется, и сообщение об ошибке выдано не будет.

3.3.2. Открытие слов и байтов в файле

Формат команды для открытия слова в неоверлейном файле:

[регистр перемещения] смещение/

PATCH печатает слово и ожидает подачи нового слова или другой команды.

Для оверлейных файлов формат команды:
 [номер сегмента:] [регистр перемещения,] смещение/
 аналогично можно открыть байт в файле. Для неоверлей-
 ных файлов формат команды:

[регистр перемещения,] смещение \

для оверлейных файлов:

[номер сегмента:] [регистр перемещения,] смещение \

как только слово или байт открыты, по адресу этого сло-
 ва или байта можно записать требуемую величину в формате:

[регистр перемещения,] восьмеричная величина
 используя одну из команд <BK>, <PC>, «@».

Если регистр перемещения установлен, последнее выра-
 жение имеет значение:

величина перемещения + восьмеричная величина

Примечание. Действие команд открытия и закрытия
 слов и байтов в файле аналогично действию соответствую-
 щих команд программы отладчик ([6]).

Ниже приводится пример модификации файла с исполь-
 зованием команд открытия слов и байтов в файле.

.R PATCH

FILE NAME —

*PROG/D <BK>

*1000; 1R

*1,102/	502	A=B?	C(M2)	<PC>
1,104/	50553	A=KQ	X=MAC	P(TXT) <PC>

1,106/	1120	@		
--------	------	---	--	--

1,120/	134750	X=??2	P(LDA) \
--------	--------	-------	----------

1,116/	4530	4570	<BK>
--------	------	------	------

*1,122 \	63	65	<BK>
----------	----	----	------

E

?PATCH—I—CHECKSUM=74245

3.3.3. Модификация оверлейных файлов

Под оверлейные сегменты редактор связей отводит память
 блоками по 256 слов. Каждый сегмент начинается на границе
 нового блока. Если какой-либо сегмент занимает не целое
 число блоков, то для «заплаты» можно использовать остав-
 шееся место до начала следующего блока. Для этого следует
 сначала изменить количество слов в сегменте, указанное в
 таблице оверлейного драйвера, чтобы выполняемая «запла-
 та» была включена в размер оверлейного сегмента. Не сле-
 дует вносить изменения в следующий блок, поскольку в нем
 начинается другой оверлейный сегмент.

По команде «X;O» вносятся изменения в оверлейный драйвер или его таблицы. Например, по команде

*616/1043 1100;O

оверлейная область увеличивается на 35 (восьмеричное) слов, что дает место для «заплаты». Происходит переустановка оверлейной структуры, оператор может подавать команды для модификации оверлейного файла. Если не указать аргумент, или указать X=0, то PATCH выдает сообщение об ошибке.

По команде «X;B» оператор сообщает программе PATCH младший адрес модифицируемого файла. Это делается в тех случаях, когда он отличается от начального значения указателя стека.

Ниже приводится пример модификации оверлейного файла. В оверлейном файле ROOT.SAV требуется изменить первое слово оверлейного сегмента 1. Команду безусловного перехода (код 403) необходимо заменить на команду (код 240). В карте загрузки, полученной при редактировании, указано, что оверлейный сегмент начинается с адреса 1236.

Модификация файла ROOT.SAV осуществляется с помощью следующих команд:

```
.R PATCH
FILE NAME —
*ROOT/O <BK>
*1236;1R
*1:1,0/                403            240 <BK>
*1:1,0/                240 <BK>
*E
```

3.3.4. Контрольная сумма

Программа PATCH производит подсчет значений каждой команды, аргумента и символа, введенных оператором. Эту сумму назовем контрольной суммой «Заплаты».

По команде «&» открытое слово или байт автоматически включается в контрольную сумму. При подаче этой команды PATCH закрывает слово или байт, если они были открыты, выполняет возврат каретки и перевод строки, печатает звездочку и ожидает очередной команды. Действие команды «&» прекращается после подачи второго «&».

По команде «E» или «F» (т. е. когда модификация текущего файла закончена) PATCH запрашивает ввод контрольной суммы, печатая «CHECKSUM», если в командной строке указан переключатель /C, или выдает контрольную сумму на

печать, если указан переключатель /D. В случае несовпадения контрольной суммы, введенной оператором, с той, которую вычислила программа PATCH, на терминале печатается сообщение, и PATCH снова запрашивает контрольную сумму. Если и в третий раз контрольные суммы не совпадают, то PATCH передает управление монитору, если подана команда «Е», или запрашивает ввод командной строки, если подана команда «F».

Примечание. Команды «А» или «Х» не изменяют слов и байтов, однако PATCH изменяет контрольную сумму, учитывая, что была введена новая команда.

3.4. Сообщения оператору

Ниже описаны сообщения, выдаваемые программой PATCH.

?PATCH—F—FILE PROTECTED

Причина. Попытка модифицировать защищенный файл.

Действие. Использовать команду RENAME/NOPROTECT или переключатель PIP/Z для отмены защиты файла.

?PATCH—F—INSUFFICIENT MEMORY

Причина. Недостаточно оперативной памяти для выполнения операции.

Действие. Освободить часть оперативной памяти (удалить ненужные драйверы, удалить основное задание, использовать монитор одного задания). Повторить выполнение операции.

?PATCH—F—READ ERROR

Причина. Ошибка чтения.

Действие. Проверить готовность и исправность оборудования. Повторить выполнение операции.

?PATCH—F—WRITE ERROR

Причина. Ошибка записи.

Действие. Проверить готовность и исправность оборудования. Повторить выполнение операции.

?PATCH—I—[+2K CORE]

Причина. Своппинг программы обслуживания пользователя (USR).

Действие. Не требуется.

?PATCH—I—CHECKSUM=NNNNNN

Причина. Вычисленное значение контрольной суммы равно NNNNNN.

Действие. Не требуется.

?PATCH—I—SEARCH FAILURE FOR NNNNNN

Причина. Программа PATCH не может обнаружить адрес NNNNNN.

Действие. Подать команду правильно.

?PATCH—W—ADDRESS NOT IN SEGMENT

Причина. Указанный в команде адрес не принадлежит сегменту.

Действие. Определить границы сегмента по карте загрузки. Подать команду правильно.

?PATCH—W—BOTTOM ADDRESS WRONG

Причина. Содержимое указанного адреса не соответствует первому слову оверлейного драйвера.

Действие. Установить соответствующий адрес по команде «X;B».

?PATCH—W—CHECKSUM ERROR

Причина. Вычисленное значение контрольной суммы не совпадает со значением, введенным на запрос CHECKSUM?

Действие. Удалить неправильно модифицированный файл. Повторно создать копию файла. Повторить модификацию.

?PATCH—W—ILLEGAL COMMAND

Причина. В командной строке обнаружена ошибка.

Действие. Ввести правильную командную строку.

?PATCH—W—ILLEGAL OPTION

Причина. В командной строке обнаружен недопустимый переключатель.

Действие. Ввести правильную командную строку.

?PATCH—W—INVALID OVERLAY HANDLER MODIFICATION

Причина. Попытка ввести нулевое значение в таблицы оверлейного драйвера.

Действие. Подать команду «X;O» с ненулевым значением X.

?PATCH—W—INVALID RELOCATION REGISTER

Причина. В команде указан недопустимый номер регистра перемещения.

Действие. Указать в команде допустимый номер регистра перемещения (от 0 до 7).

?PATCH—W—INVALID SEGMENT NUMBER

Причина. В модифицированном файле нет сегмента с указанным в команде номером.

Действие. Определить номер сегмента по карте загрузки. Подать команду правильно.

?PATCH—W—MUST OPEN WORD

Причина. Не открыто слово перед подачей команды «@», «Р» или «Х».

Действие. Открыть слово перед подачей указанных команд.

?PATCH—W—MUST SPECIFY SEGMENT NUMBER

Причина. Указанный в команде адрес не принадлежит корневому сегменту.

Действие. Определить по карте загрузки, какому сегменту принадлежит адрес. Подать команду правильно.

?PATCH—W—NO ADDRESS OPEN

Причина. Не открыто слово или байт перед подачей команды «ПС», «/», «@», «Х», «Р», «С» или «А».

Действие. Открыть слово или байт перед подачей указанных команд.

?PATCH—W—NOT IN PROGRAM BOUNDS

Причина. Адрес, указанный в команде, или начальное значение указателя стека превосходит старший адрес программы, записанный по адресу 50 в нулевом блоке файла.

Действие. Определить по карте загрузки старший адрес программы и подать команду правильно.

?PATCH—W—ODD ADDRESS

Причина. Попытка открыть байт с нечетным адресом по команде «/».

Действие. Открыть байт с нечетным адресом по команде «\».

?PATCH—W—ODD BOTTOM ADDRESS

Причина. Адрес, указанный в команде «Х;В» или содержащийся по адресу 42 в нулевом блоке оверлейного файла, нечетный.

Действие. Требуемый адрес указать четным.

?PATCH—W—PROGRAM HAS NO SEGMENTS

Причина. Обращение к сегменту в файле, который не определен командной строке как оверлейный, или обращение к оверлейному сегменту в файле, который их не имеет.

Действие. Определить по карте загрузки структуру файла. Ввести правильную командную строку.

4. МОДИФИКАЦИЯ ОБЪЕКТНЫХ МОДУЛЕЙ

4.1. Назначение программы и условия выполнения программы

Программа модификации объектных модулей (PAT) предназначена для внесения изменений в объектные модули.

РАТ позволяет заменять и/или включать в объектный модуль определения глобальных имен, команды, данные.

Входной информацией для РАТ являются два объектных файла — основной файл и файл изменений. Основной файл — это файл, который требуется модифицировать. В этот файл могут входить один или несколько объектных модулей. За один вызов РАТ можно модифицировать только один из этих модулей — основной модуль. В файл изменений входит один объектный модуль — модуль изменений. В этом модуле содержатся изменения и дополнения к основному модулю.

Выходной файл программы РАТ — это модифицированный основной файл (рис. 1).

СХЕМА ВНЕСЕНИЯ ИЗМЕНЕНИЙ

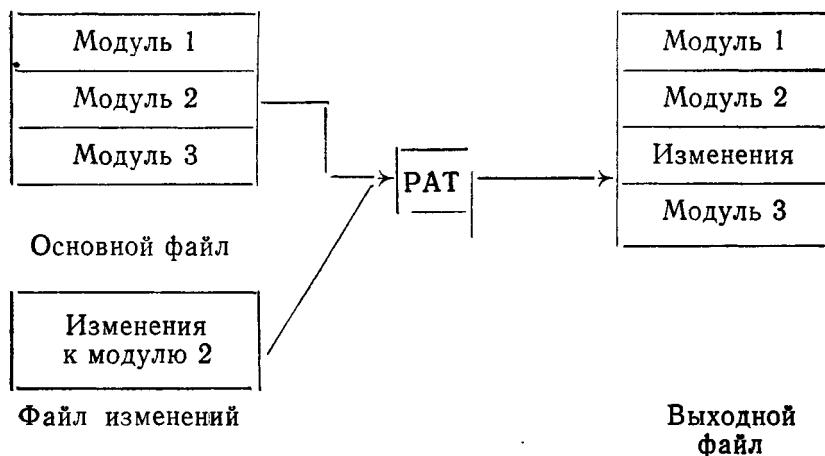


Рис. 1.

Программа РАТ не накладывает ограничений на использование устройств ввода-вывода.

4.2. Выполнение программы

Для вызова РАТ с системного устройства следует подать с терминала команду

R РАТ <ВК> или РАТ <ВК>

после того, как монитор напечатает на терминале точку.

После вызова РАТ печатает звездочку и ожидает ввода командной строки. Если в это время нажать клавишу <ВК>, то РАТ напечатает номер своей версии.

4.3. Команды оператора

Режим работы программы РАТ задается введением с терминала командной строки. Формат командной строки:

[выходспф=] оснспф/[C[:N]], измспф [/C[:N]]

где выходспф — спецификация выходного файла (устройство, имя и тип файла);

оснспф — спецификация основного файла;

измспф — спецификация файла изменений;

/C[:N] — переключатель, задаваемый для вычисления или проверки контрольной суммы.

Тип файла по умолчанию для входных и выходных файлов — .OBJ.

Если в командной строке не указана спецификация выходного файла, то результатом работы РАТ являются выдаваемые на терминал сообщения о встреченных ошибках.

Переключатель /C[:N] задают в командной строке для вычисления или проверки контрольной суммы основного модуля или модуля изменений. Под контрольной суммой модуля понимается сумма всех байтов данных, входящих в модуль.

Пример.

*A/C, AI

Выполняя указанную операцию, РАТ вычисляет контрольную сумму основного модуля и выдает сообщение

?PAT—W—INPUT MODULE CHECKSUM IS NNNNNN

где NNNNNN — восьмеричное число, являющееся контрольной суммой.

Если требуется проверить значение контрольной суммы, в командной строке задают переключатель /C:N, где N — восьмеричное число. РАТ вычисляет контрольную сумму соответствующего модуля и сравнивает ее со значением N. При несопадении РАТ выдает сообщение об ошибке.

После выполнения операции РАТ печатает звездочку (*) и ожидает ввода новой командной строки.

Для выхода из программы РАТ и передачи управления монитору следует подать команду СУ/C, если РАТ ожидает ввода с терминала, или дважды СУ/C, если РАТ выполняет операцию.

4.4. Способ использования

Использование программы РАТ предполагает создание модуля изменений. Модуль изменений создается таким образом: данные, необходимые для модификации объектного

модуля, записывают в виде последовательности операторов языка АССЕМБЛЕРА. Модуль изменений получается как результат трансляции этих операторов. Единственным ограничением является то, что имя модуля изменений (т. е. аргумент директивы .TITLE в исходной записи) должно совпадать с именем основного модуля.

Формируя выходной файл, РАТ выполняет следующие действия:

все программные секции, входящие в модуль изменений, добавляются к основному модулю. Если имя программной секции, входящей в модуль изменений, дублирует имя программной секции, входящей в основной модуль, то должны выполняться условия:

— обе программные секции имеют одинаковые значения признака перемещения (ABS или REL);

— обе программные секции определены посредством одной и той же директивы (.PSECT или .CSECT).

Если эти условия не выполнены, РАТ прекращает выполнение операции, выдает сообщение об ошибке и передает управление монитору.

Все глобальные имена, определенные в модуле изменений, добавляются к каталогу глобальных имен основного модуля. Если глобальное имя, определенное в модуле изменений, дублирует глобальное имя, определенное в основном модуле, и оба имени имеют абсолютное или относительное значение, то РАТ заменяет определение в основном модуле на его аналог из модуля изменений. Если же одно глобальное имя имеет абсолютное значение, а другое относительное, то РАТ прекращает выполнение операции, выдает сообщение об ошибке и передает управление монитору.

Если в модуле изменений определена точка входа (пусковой адрес), то РАТ заменяет определение точки входа в основном модуле на его аналог из модуля изменений.

4.4.1. Примеры

Ниже приведены два примера, иллюстрирующие использование программы РАТ. Для наглядности вместо объектных и загрузочных модулей приведены соответствующие им исходные модули на языке АССЕМБЛЕР. Предположим, что в объектный модуль

.TITLE	ABC
.IDENT	/01/
.ENABL	GBL

```

ABC: :   MOV      A,C
        JSR      PC,XYZ
        RTS      PC
        .END

```

требуется вставить команду MOV A,B между командами JSR PC,XYZ и RTS PC. Для этого можно использовать такой модуль изменений

```

.TITLE   ABC
.IDENT   /01.01/
.ENABL   GBL
.=.+12
MOV      A,B
RTS      PC
.END

```

результатом работы PAT в этом случае будет объектный модуль

```

ABC: :   .TITLE   ABC
        .IDENT   /01.01/
        .ENABL   GBL
        MOV      A,C
        JSR      PC,XYZ
        RTS      PC
        .=ABC
        .=.+12
        MOV      A,B
        RTS      PC
        .END

```

результатом редактирования этого объектного модуля будет загрузочный модуль

```

ABC: :   .TITLE   ABC
        .IDENT   /01.01/
        .ENABL   GBL
        MOV      A,C
        JSR      PC,XYZ
        MOV      A,B
        RTS      PC
        .END

```

В данном примере происходит наложение кодов команд MOV A,B и RTS PC на код команды RTS PC. При этом размер изменяемой секции увеличивается на шесть байтов.

В тех случаях, когда объем изменений значителен, выгодно пользоваться следующим приемом. Вносимые изменения оформляют в виде подпрограммы (например создают новую

программную секцию) и с помощью PAT включают ее в основной модуль.

Предположим, что в объектный модуль

```

        .TITLE      ABC
        .IDENT      /01/
        .ENABL      GBL
ABC:    :   MOV      A,B
        JSR      PC,XYZ
        MOV      C,R0
        RTS      PC
        .END

```

нужно вставить команды

```

        MOV      X,R0
        ASL      R0

```

между командами MOV A,B и JSR PC,XYZ. Для этого можно использовать такой модуль изменений

```

        .TITLE      ABC
        .IDENT      /01.01/
        .ENABL      GBL
        JSR      PC,PATCH
        NOP
        .PSECT      PATCH
PATCH:  MOV      A,B
        MOV      X,R0
        ASL      R0
        RTS      PC
        .END

```

результатом работы PAT в этом случае будет объектный модуль

```

        .TITLE      ABC
        .IDENT      /01.01/
        .ENABL      GBL
ABC:    :   MOV      A,B
        JSR      PC,XYZ
        MOV      C,R0
        .RTS      PC
        .=ABC
        JSR      PC,PATCH
        NOP
        .PSECT      PATCH
PATCH:  MOV      A,B
        MOV      X,R0
        ASL      R0

```

	RTS	PC
	.END	

результатом редактирования этого объектного модуля будет
загрузочный модуль

	.TITLE	ABC
	.IDENT	/01.01/
	.ENABL	GBL
ABC: :	JSR	PC,PATCH
	NOP	
	JSR	PC,XYZ
	MOV	C,R0
	RTS	PC
	.PSECT	PATCH
PATCH:	MOV	A,B
	MOV	X,R0
	ASL	R0
	RTS	PC
	.END	

В этом примере коды команд JSR PC,PATCH и NOP (три слова) накладываются на код команды MOV A, B (три слова). Команда NOP используется здесь для соответствия количества слов. Команда MOV A,B, замененная командой JSR PC,PATCH должна быть первой выполняемой командой в подпрограмме PATCH.

4.5. Сообщения оператору

Ниже описаны сообщения, выдаваемые программой PAT.

?PAT—F—CORRECTION FILE HAS INVALID RECORD

Причина. Формат файла изменений отличен от объектного формата, используемого в ФОДОС.

Действие. Проверить правильность введенной командной строки. Заново получить файл изменений и повторить выполнение операции.

?PAT—F—CORRECTION FILE MISSING

Причина. В командной строке не указана спецификация файла изменений.

Действие. Ввести правильную командную строку.

?PAT—F—CORRECTION FILE MISSING RLD RECORD

Причина. В файле изменений отсутствует каталог перемещений перед первой записью TXT.

Действие. Вновь протранслировать файл изменений и повторить выполнение операции.

?PAT—F—CORRECTION FILE READ ERROR

Причина. Аппаратная ошибка чтения файла изменений.

Действие. Проверить готовность и исправность оборудования.

Повторить выполнение операции.

?PAT—F—DEVICE FULL DEV:FILNAM.TYP

Причина. На томе, используемом для вывода, нет свободной области для размещения выходного файла.

Действие. Возможны следующие действия:

— указать в командной строке размер выходного файла;

— освободить место на томе;

— использовать для вывода другой том.

?PAT—F—INCOMPATIBLE REFERENCE TO GLOBAL AAAAAA

Причина. В файле изменений неправильно определено глобальное имя AAAAAA.

Действие. Исправить файл изменений, определив глобальное имя AAAAAA в соответствии с определением этого имени в основном модуле. Повторить выполнение операции.

?PAT—F—INCOMPATIBLE REFERENCE TO SECTION AAAAAA

Причина. В файле изменений неправильно определена программная секция AAAAAA.

Действие. Исправить файл изменений, определив программную секцию AAAAAA в соответствии с определением этой секции в основном модуле. Повторить выполнение операции.

?PAT—F—INPUT FILE HAS INVALID RECORD

Причина. Формат основного файла отличен от объектного формата, используемого в ФОДОС.

Действие. Проверить правильность введенной командной строки. Заново получить основной файл и повторить выполнение операции.

?PAT—F—INPUT FILE MISSING

Причина. В командной строке не указана спецификация основного файла.

Действие. Ввести правильную командную строку.

?PAT—F—INSUFFICIENT MEMORY

Причина. Недостаточно оперативной памяти для выполнения операции.

Действие. Освободить часть оперативной памяти (удалить ненужные драйверы, удалить основное задание, ис-

пользовать монитор одного задания). Повторить выполнение операции.

?PAT—F—INTERNAL ERROR

Причина. Сбой в работе операционной системы. Возможно, что ошибка в PAT.

Действие. Повторить выполнение операции. При появлении этой же ошибки получить новую копию программы PAT.

?PAT—F—INVALID COMMAND

Причина. В командной строке обнаружена ошибка.

Действие. Ввести правильную командную строку.

?PAT—F—INVALID GSD SYMBOL IN DEV:FILNAM.TYP

Причина. Обнаружена ошибка в каталоге глобальных имен основного модуля.

Действие. Проверить правильность введенной командной строки. Заново получить основной модуль и повторить выполнение операции.

?PAT—F—INVALID RLD SYMBOL IN DEV:FILNAM. TYP

Причина. Обнаружена ошибка в каталоге перемещений основного модуля.

Действие. Проверить правильность введенной командной строки. Заново получить основной модуль и повторить выполнение операции.

?PAT—F—ONLY /C ALLOWED

Причина. В командной строке задан переключатель, отличный от /C[:N].

Действие. Ввести правильную командную строку.

?PAT—F—OUTPUT ERROR DEV:FILNAM.TYP

Причина. Возможны следующие причины:

- заблокирована запись на устройстве вывода;
- аппаратная ошибка записи выходного файла.

Действие. Проверить готовность и исправность оборудования. Повторить выполнение операции.

?PAT—F—UNABLE TO LOCATE MODULE AAAAAA

Причина. В основном файле нет модуля с именем AAAAAA. AAAAAA — имя модуля изменений.

Действие. Определить в файле изменений правильное имя модуля. Повторить выполнение операции.

?PAT—F—WRONG VERSION

Причина. Попытка обращения системы ФОДОС к программе PAT, принадлежащей системе ФОДОС-2.

Действие. Вызвать программу PAT системы ФОДОС.

?PAT—W—ADDITIONAL INPUT FILE IGNORED

Причина. В командной строке указано более двух входных файлов.

Действие. Не требуется. PAT игнорирует все входные файлы, кроме двух первых.

?PAT—W—ADDITIONAL OUTPUT FILE IGNORED

Причина. В командной строке указано более одного выходного файла.

Действие. Не требуется. PAT игнорирует все выходные файлы, кроме первого.

?PAT—W—CORRECTION FILE CHECKSUM ERROR

Причина. Вычисленное значение контрольной суммы модуля изменений не совпадает со значением, указанным в командной строке.

Действие. Не требуется.

?PAT—W—CORRECTION FILE CHECKSUM IS NNNNNN

Причина. Вычисленное значение контрольной суммы модуля изменений равно NNNNNN.

Действие. Не требуется.

?PAT—W—INPUT FILE CHECKSUM ERROR

Причина. Вычисленное значение контрольной суммы основного модуля не совпадает со значением, указанным в командной строке.

Действие. Не требуется.

?PAT—W—INPUT MODULE CHECKSUM IS NNNNNN

Причина. Вычисленное значение контрольной суммы основного модуля равно NNNNNN.

Действие. Не требуется.

5. МОДИФИКАЦИЯ ЗАГРУЗОЧНЫХ МОДУЛЕЙ

5.1. Назначение программы и условия выполнения программы

Программа модификации загрузочных модулей SIPP предназначена для изменения кодов любых типов файлов, допустимых в ФОДОС-2, на устройствах с произвольным доступом. Главным образом программа SIPP используется для модификации загрузочных модулей типа .REL, .SAV и .LDA, созданных как с помощью редактора связей ФОДОС-2, так и с помощью редактора связей ФОДОС, последние не должны иметь оверлейную структуру.

Программа SIPP позволяет проверять ячейки внутри файла, не изменяя их. Программа SIPP может быть запущена из косвенного командного файла, из пакета заданий BATCH или с терминала.

Во время выполнения программы SIPP можно с помощью соответствующих переключателей ввести указанные изменения кодов в файл или создать косвенный командный файл из указанных изменений и команд программы SIPP для ввода изменений в файл. При необходимости следует лишь запустить косвенный командный файл для изменения кодов файла. При модификации загрузочных модулей по программе SIPP дата создания файла меняется на текущую системную дату.

Примечание. Не рекомендуется модифицировать содержимое ячеек 50, 64, 66, а также слово состояния задания, оверлейный драйвер и блоки определения окон. Программа SIPP при необходимости автоматически модифицирует их. Если используется переключатель /A (см. п. 5.3), программа не модифицирует эти ячейки.

5.2. Выполнение программы

Для вызова SIPP с системного устройства в ответ на точку (.) монитора следует подать команду:

R SIPP <BK> или SIPP <BK>

после вызова SIPP интерпретатор командной строки печатает звездочку (*) и ожидает ввода командной строки. Если нажать только <BK>, SIPP напечатает номер своей версии; если CY/C — управление возвращается монитору. CY/C может быть подана в то время, когда SIPP ожидает действия от пользователя. В ответ на это SIPP печатает:

?SIPP—ARE YOU SURE?

Если напечатать Y (или любую строку, начинающуюся с Y) и <BK>, SIPP прерывает текущую процедуру модификации и передает управление монитору, не выполняя изменений в файле. Любой другой ответ возвращает управление к прерванной процедуре модификации. Два последовательных нажатия CY/C во время выполнения какого-либо действия программой SIPP (например, поиск кода), включая работу SIPP из косвенного командного файла, также вызывает печать сообщения:

?SIPP—ARE YOU SURE?

5.3. Команды оператора

Режим работы программы SIPP задается введением с терминала командной строки следующего формата:

[спфком=]входспф[/прк..]

где спфком — спецификация косвенного командного файла, создаваемого программой SIPP. Тип файла по умолчанию — .COM. Если спфком не указана, косвенный командный файл не создается;

входспф — спецификация модифицируемого загрузочного модуля;

/прк — один или несколько переключателей (см. табл. 5).

Если после печати звездочки (*) пользователь указал только имя устройства, SIPP открывает первый блок указанного тома и предполагает переключатель /A.

В табл. 5 перечисляются все переключатели программы SIPP.

Таблица 5

Переключатель	Функция
1	2
/A	Запрещает автоматическую модификацию ячейки 50, блоков определения окон, оверлейной таблицы или оверлейного драйвера. Используется для модификации файлов любых типов кроме .SAV, .REL и .LDA
/C	Запрашивает контрольную сумму по окончании модификации. Если модификация не производилась, SIPP игнорирует переключатель /C. Косвенный командный файл автоматически будет содержать /C. Нельзя использовать вместе переключатели /C и /D
/D	Вычисляет и печатает контрольную сумму «заплаты» по окончании модификации. Если модификация не производилась, SIPP игнорирует переключатель /D
/L	Создает косвенный командный файл, сохраняя входной файл неизменным

Примечание. «Заплата» — совокупность изменений, которые должны быть внесены в загрузочный модуль в процессе модификации.

5.4. Диалог программы SIPP

После ввода первой командной строки, SIPP печатает серию подсказок на терминале.

Примечание. Подсказка — сообщение программы SIPP, подсказывающее, какую информацию необходимо ввести.

Если входной файл имеет оверлейную структуру, на терминале печатается: SEGMENT?

Следует ввести номер оверлейного сегмента (номер сегмента следует искать в карте загрузки программы), коды которого будут модифицированы, и ВК. Для модификации корневого сегмента программы необходимо ввести «ВК» или «0» и «ВК».

Программа SIPP не печатает эту подсказку, если входной файл не оверлейной структуры, если введен переключатель /A или, если модифицируется том.

После ответа на первую подсказку на терминале печатается: BASE?

Если файл оверлейной структуры, следует ввести адрес базы, которой указывается в карте загрузки модифицируемого сегмента. Если файл не оверлейной структуры, следует ввести адрес загрузки программной секции, которую необходимо модифицировать или проверить.

После ввода адреса базы SIPP печатает: OFFSET?

Следует ввести смещение от адреса текущей базы до ячейки, код которой будет модифицирован или проверен.

Если указанное смещение — четное, SIPP открывает соответствующее слово, если — нечетное, SIPP открывает соответствующий байт.

После ввода смещения SIPP печатает заголовок:

SEGMENT BASE OFFSET OLD NEW?

а ниже значения, указанные в диалоге.

Если файл не оверлейной структуры, программа SIPP не печатает колонку SEGMENT.

Пример.

```
.R SIPP <BK>
*PROG=PROG<BK>
SEGMENT? 2<BK>
BASE? 20000<BK>
OFFSET? 100<BK>
```

SEGMENT	BASE	OFFSET	OLD	NEW?
000002	20000	20100	103425	

В колонке OLD печатается содержимое ячейки, открытой в настоящий момент. В колонку NEW? можно ввести новое значение для данной ячейки и/или команду (см. п. 5.5).

Примечание. Программа SIPP не вносит в файл изменения, которые были напечатаны в колонке NEW?; она лишь хранит их в буфере. По окончании модификации следует нажать СУ/С или несколько раз СУ/З для внесения указанных выше изменений в файл.

5.5. Команды программы SIPP

Для работы с программой SIPP используются команды, перечисленные в табл. 6. Более подробное описание команд будет приведено ниже. Команды программы SIPP следует заканчивать <ВК> или <ПС>.

Таблица 6

Команда	Функция
1	2
<ВК> или <ПС>	*Закрывает текущее слово или байт и открывает следующее слово или байт
N<ВК>	Заносит значение N в текущее слово или байт, закрывает его и открывает следующее слово или байт
Λ<ВК>	Закрывает текущее открытое слово или байт и открывает предыдущее слово или байт
NΛ<ВК>	Заносит значение N в текущее слово или байт, закрывает его и открывает предыдущее слово или байт
\<ВК>	Повторно открывает младший байт текущего слова. После этой команды SIPP будет открывать ячейки по байтам (режим байта).
/<ВК>	В режимах RAD-50 или КОИ-7 эта команда не используется. Повторно открывает текущее слово. После этой команды SIPP будет открывать ячейки по словам (режим слова). В режимах RAD-50 или

1	2
; O<BK>	КОИ-7 эта команда не используется Повторно открывает текущее слово и печатает код в восьмеричном виде (восьмеричный режим печати). Это режим по умолчанию. Используется для возврата к восьмеричному режиму печати после режимов RAD-50 или КОИ-7
; A<BK>	Открывает текущий байт и печатает его код в символах КОИ-7 (режим КОИ-7). Последующие байты будут также распечатываться в символах КОИ-7
; AX<BK>	Заносит код КОИ-7 указанного символа (X) в текущий байт, закрывает его и открывает следующий байт. Эта команда используется также для поиска кода КОИ-7
; R<BK>	Открывает текущее слово и печатает его код в символах RAD-50 (режим RAD-50).
; RYYY<BK>	Последующие слова будут также распечатываться в символах RAD-50 Заносит код RAD-50 указанных символов (YYY) в текущее слово, закрывает его и открывает следующее слово. Эта команда используется также для поиска кода RAD-50 (см. п. 5.5.7)
; S<BK>	Используется для поиска указанного кода в файле. После ввода данной команды SIPP опрашивает, какой код надо найти и в каком месте файла будет выполняться поиск
; V<BK>	Печатает внесенные изменения. Не следует подавать эту команду в ответ на подсказку CHECKSUM?
CY/Z<BK>	Печатает подсказки: OFFSET?, BASE? или SEGMENT? в указанном порядке.

1	2
CU/Y<BK>	Эта команда позволяет модифицировать разные области программы Завершает модификацию, вносит «заплату» в файл, создает косвенный командный файл (если требуется) и печатает звездочку

5.5.1. Команды <BK> (<ПС>) и N<BK>

Как было описано выше, по окончании диалога SIPP печатает заголовок и ниже — адрес указанной ячейки и ее содержимое. В последней колонке с заголовком NEW? следует подать: — <BK> или <ПС>, чтобы закрыть текущее слово или байт и открыть следующее слово или байт; — N<BK>, чтобы занести значение N в текущее слово или байт. Указанные команды могут выполняться как по словам, так и по байтам в зависимости от установленного режима.

Пример:

BASE	OFFSET	OLD	NEW?
001000	001200	004767	240<BK>
001000	001202	003106	

В данном примере по адресу 2200 заносится значение 240 и открывается следующее слово.

5.5.2. Команды ^<BK> и N^<BK>

Когда печатается ^<BK> в колонке с заголовком NEW?, SIPP закрывает открытое слово или байт и открывает предыдущее слово или байт. Если указать значение N^<BK>, SIPP записывает N в текущее слово или байт, закрывает его и открывает предыдущее слово или байт.

Пример:

BASE	OFFSET	OLD	NEW?
002000	002134	020027	112000 ^<BK>
002000	002132	001732	

В данном примере по адресу 4134 заносится значение 112000 и открывается предыдущее слово по адресу 4132.

Чтобы открыть предыдущий байт, если в данный момент открыто слово, следует использовать команду \^<BK>. Чтобы открыть предыдущее слово, если в настоящий момент открыт байт, следует использовать команду /^<BK>.

5.5.3. Команды \<BK> и /<BK>

По умолчанию команды программы SIPP выполняются в режиме слова. По команде \<BK> SIPP закрывает текущее слово и повторно открывает младший четный байт этого слова. После этой команды значения воспроизводятся на терминале и заносятся в ячейки по байтам.

Чтобы перейти к режиму слова, следует подать команду /<BK>, по которой SIPP повторно открывает текущую ячейку как 16-разрядное слово.

Программа SIPP печатает заголовок:
BASE OFFSET OLD NEW?

Каждый раз при переходе от режима байта к режиму слова или наоборот.

Пример:

BASE	OFFSET	OLD	NEW?
002000	002112	003002	\<BK>
BASE	OFFSET	OLD	NEW?
002000	002112	002	<BK>
002000	002113	006	<BK>
002000	002114	132	/<BK>
BASE	OFFSET	OLD	NEW?
002000	002114	003132	

В данном примере после команды \<BK> программа SIPP перешла к режиму байта, после /<BK> — к режиму слова.

Если в режиме байта подана команда CY/Z, SIPP печатает подсказку OFFSET? и автоматически переходит к режиму слова.

5.5.4. Команда ;O<BK>. По команде ;O<BK> SIPP повторно открывает текущее слово и печатает его код в восьмеричном виде (режим по умолчанию). Необходимость в этой команде возникает в том случае, если требуется перейти из режима КОИ-7 или RAD-50 к печати в восьмеричном виде. Команда ;O используется также для перехода от режима байта к режиму слова.

Пример:

В следующем примере по команде ;O SIPP переходит от режима КОИ-7 к восьмеричному режиму печати.

BASE	OFFSET	OLD	NEW?
002000	002100	051101	;A<BK>
002000	002100	<A>	;O<BK>
BASE	OFFSET	OLD	NEW?
002000	002100	051101	

Если в режиме КОИ-7 или RAD-50 подать CY/Z, SIPP печатает подсказку OFFSET? и автоматически переходит к восьмеричному режиму печати.

5.5.5. Команды ;A<BK> и ;AX<BK>

По команде ;A<BK> SIPP открывает текущий байт и печатает его код в символах КОИ-7. Коды последующих байтов SIPP продолжает распечатывать в символах КОИ-7 до тех пор, пока не будет подана команда ;O или ;R.

Пример:

В следующем примере по команде ;A открывается текущий младший, четный байт и SIPP воспроизводит код байта символом КОИ-7.

BASE	OFFSET	OLD	NEW?
003000	003100	050524	;A<BK>
003000	003100	<T>	<BK>
003000	003100	<Q>	

По команде ;AX<BK> SIPP заносит код КОИ-7 указанного символа (X) в текущий байт, закрывает его и открывает следующий байт в восьмеричном режиме, режиме КОИ-7 или RAD-50 (в зависимости от того, какой режим использовался программой SIPP до команды ;AX).

Примечание. По одной команде ;AX можно занести код только одного символа, и в качестве X нельзя указывать управляющий символ.

Пример:

В следующем примере по команде ;AX код символа 'W' записывается в текущий байт.

BASE	OFFSET	OLD	NEW?
003000	003100	050524	;AW<BK>
003000	003101	121	

Команда ;AX используется также для поиска символа КОИ-7 (см. п. 5.5.7).

5.5.6. Команды ;R<BK> и ;YYYY<BK>

По команде ;R<BK> SIPP открывает текущее слово и печатает его код в символах RAD-50 (режим RAD-50). После команды ;R SIPP продолжает работать в режиме RAD-50 до тех пор, пока не будет использована команда ;A или ;O.

Пример:

В следующем примере по команде ;R повторно открывается ячейка 6220 и печатается ее код символами RAD-50.

BASE	OFFSET	OLD	NEW?
001000	005220	071070	;R<BK>

001000	005220	<RK>	<BK>
001000	005222	<TES>	

если код ячейки — недопустимый код для RAD-50, SIPP печатает <??>.

По команде ;RYYY<BK> SIPP заносит код RAD-50 указанных символов (YYY) в текущее слово, закрывает его, открывает следующее слово и печатает его код в восьмеричном виде, символами КОИ-7 или RAD-50 (в зависимости от того, какой режим использовался программой SIPP до команды ;RYYY).

Если в команде ;RYYY указано только два символа, SIPP предполагает пробел вместо третьего символа. Если указан только один символ, SIPP предполагает пробелы вместо второго и третьего символов. Если пробел указан вторым символом (например, X Z), SIPP заносит код всех указанных символов. По команде ;RYYY можно занести код RAD-50 не более трех символов, начиная с младшего байта текущей ячейки. В противном случае (открыт старший байт текущей ячейки) SIPP печатает сообщение об ошибке.

Код следующих символов RAD-50 может быть занесен в ячейку по команде ;RYYY:

- буквы от A до Z;
- цифры от 0 до 9;
- символы: 'X', '*', '.', '%'.

Примечание. Пробел также допустим, и SIPP преобразует символ процент (%) в символ точка (.).

Пример:

В данном примере по команде ;RYYY код RAD-50 символов ABC заносится в ячейку 6332 и открывается следующая ячейка.

BASE	OFFSET	OLD	NEW?
001000	005332	000240	;RABC<BK>
001000	005334	002110	

Команда ;RYYY используется для поиска кода символа RAD-50 (см. п. 5.5.7).

5.5.7. Команда ;S<BK> предназначена для поиска кода в указанных границах файла. Используя эту команду, можно найти слово, код которого необходимо изменить.

Команду ;S можно ввести в ответ на вопрос диалога программы SIPP или в колонке NEW?. Эту команду не следует вводить в ответ на подсказку CHECKSUM?.

После ввода ;S SIPP печатает подсказку: SEARCH FOR? В ответ следует ввести код, который необходимо найти; можно также ввести команду ;AX или ;RYYY для поиска кода КОИ-7 символа X или кода RAD-50 символов YYY. Если указать символ '\ ' после ввода разыскиваемого кода, SIPP выполняет поиск по байтам, иначе — по словам. Заметим, что по команде ;AX поиск также выполняется в режиме байта.

После ввода кода SIPP печатает START? В ответ следует ввести адрес начала поиска и <BK> или только <BK>. Если указан только <BK>, SIPP начинает поиск с начала файла; если указан адрес и <BK>, поиск начинается с указанного адреса файла. Для поиска в оверлейном сегменте начальный адрес следует указать в следующем виде: N:M
где N — номер просматриваемого сегмента;

M — смещение от начального адреса сегмента.

После ввода начального адреса поиска SIPP печатает: END? В ответ следует ввести конечный адрес поиска (в виде N:M для поиска в оверлейном сегменте) и/или <BK>. Если указан адрес и <BK> поиск будет выполняться до указанного адреса, не включая его. Если введен только <BK>, SIPP выполняет поиск до конца файла или тома включительно. Если используется переключатель /A, SIPP выполняет поиск до конца последнего блока файла или тома.

После ввода начального и конечного адресов поиска SIPP начинает поиск указанного кода. Каждый раз, когда SIPP находит этот код, на терминале печатается его адрес. Если используется переключатель /A или происходит поиск в корневом сегменте программы, SIPP печатает адрес найденного кода в виде: FOUND AT NNNNNN

при нахождении нужного кода в оверлейном сегменте SIPP печатает следующее: FOUND AT SEG:MMM, NNN
где SEG — номер сегмента;

MMM — адрес сегмента из карты загрузки;

NNN — смещение от начального адреса сегмента.

Примечание. Если выполняется поиск в файле оверлейной структуры, и в строке команды был указан переключатель /A, SIPP не использует обозначение SEG: MMM, NNN.

5.5.8. По команде ;V программа SIPP на терминале печатает внесенные изменения, т. е. адреса измененных слов и их новый код (в том числе адреса и коды тех слов, коды

которых были изменены, а позже восстановлены их первоначальные значения). После этого SIPP возвращается к операции, которая была прервана командой ;V.

Команду ;V можно подать в ответ на подсказку SEARCH FOR?,

но не следует подавать в ответ на подсказки CHECKSUM?, START? и END?.

По команде ;V печать всех изменений происходит по словам в восьмеричном виде.

Пример:

В следующем примере по команде ;V на терминале печатаются все внесенные изменения.

BASE	OFFSET	OLD	NEW?
003000	003200	003112	240<BK>
003000	003202	002300	<BK>
003000	003204	002300	<BK>
003000	003206	002300	240<BK>
003000	003210	000101	;V<BK>
BASE	OFFSET	OLD	NEW?
003000	003200	003112	000240
003000	003206	000230	000240
BASE	OFFSET	OLD	NEW?
003000	003210	000101	

5.5.9. По команде CY/Z SIPP печатает подсказки OFFSET?, BASE? или SEGMENT? в указанном порядке.

Если CY/Z указать в колонке NEW?, SIPP печатает OFFSET?. Если CY/Z указать в качестве ответа на подсказку OFFSET?, SIPP печатает подсказку BASE?. Если опять указать CY/Z, SIPP печатает:

— подсказку SEGMENT?, если файл оверлейной структуры;

— подсказку для ввода контрольной суммы, если в командной строке был указан переключатель /C, затем вносит все изменения в файл (если контрольная сумма «Заплаты» и указанная пользователем совпала). Если использовался переключатель /L, SIPP не вносит изменения в файл, а создает косвенный командный файл (если требуется).

Если файл оверлейной структуры и введена команда CY/Z в ответ на подсказку SEGMENT?, SIPP печатает подсказку для ввода контрольной суммы (если в командной строке указан переключатель /C), сравнивает введенную контрольную сумму с контрольной суммой заплаты, если они совпадают — вносит изменения в файл.

Команда `SU/Z` предназначена, главным образом, для перехода к подсказкам `OFFSET?`, `BASE?` или `SEGMENT?`. Более эффективный путь внесения изменений в файл — ввод команды `SU/Y`.

5.5.10. По команде `SU/Y<BK>` SIPP завершает текущую модификацию, т. е. вносит «Заплату» в файл. Если в строке команды указан переключатель `/C`, SIPP печатает подсказку для контрольной суммы перед внесением изменений в файл. Если указана правильная контрольная сумма, SIPP вносит изменения в файл. Если в строке команды указан переключатель `/L` и в ответ на подсказку введена правильная контрольная сумма, SIPP не вносит изменения, а создает косвенный командный файл.

После того, как SIPP внесет «Заплату» в файл, на терминале печатается звездочка (*), т. е. можно вводить новую командную строку программе SIPP.

5.5.11. Расширение файлов и оверлейных сегментов

Границы расширения программ и оверлейных сегментов в процессе модификации зависят от того, является ли программа:

- неоверлейной;
- оверлейной с оверлеями только в нижней памяти (см. [3]);
- оверлейной с оверлеями только в расширенной памяти (см. [3]);
- оверлейной с оверлеями как в нижней, так и в расширенной памяти.

Ниже следует подробное описание ограничений на расширение программ, корневых и оверлейных сегментов.

Перечисленные в табл. 7 данные SIPP может автоматически модифицировать по мере расширения программы.

Таблица 7

Данные	Назначение
1	2
Ячейка 50	Содержит старший адрес программы, если программа неоверлейной структуры. Если программа с оверлеями в нижней памяти, эта ячейка содержит старший адрес оверлейной(ых) области(ей) в нижней памяти. Если про-

1	2
Размер области	грамма с оверлеями в расширенной памяти, ячейка 50 содержит старший адрес корневого сегмента Указывает размер оверлейной области в расширенной памяти. Эти данные находятся в оверлейном драйвере расширенной памяти
Старший адрес корневого сегмента +2	Указывает адрес первой свободной ячейки за корневым сегментом. Эти данные находятся в оверлейном драйвере нижней памяти или оверлейном драйвере расширенной памяти
Старший адрес последней оверлейной области +2	Указывает адрес первой свободной ячейки за последней оверлейной областью в нижней памяти. Эти данные находятся в оверлейном драйвере нижней памяти или оверлейном драйвере расширенной памяти
Размер текущего оверлейного сегмента в словах	Указывает размер текущего оверлейного сегмента в словах. Эти данные находятся в таблице сегментов оверлейного драйвера
Длина и размер в блоке определения окна (600к)	Указывает длину и размер окна в блоке определения окна (600к) для отображения окна в расширенную память для использования оверлеями в расширенной памяти, которые необходимо расширить (см. [3])
Смещение в 600к	Указывает в области расширенной памяти смещение окон, следующих за расширяемым сегментом (см. [3])

5.5.11.1. Расширение неоверлейных программ

В случае необходимости SIPP автоматически расширяет программу до конца последнего блока памяти, занятого программой (SIPP автоматически модифицирует ячейку 50). Подробное описание расширения неоверлейной программы см. [4].

5.5.11.2. Расширение программы с оверлеями в нижней памяти

Если программа оверлейная, но с оверлеями в нижней памяти, SIPP не позволяет расширять корневой сегмент. Если оверлейный сегмент загружается не в последнюю оверлейную область, можно расширить этот сегмент до размера наибольшего сегмента соответствующего этой области. Если оверлейный сегмент загружается в последнюю оверлейную область, можно расширить этот сегмент до последнего блока этого оверлейного сегмента.

В табл. 8 показано, какие ячейки SIPP модифицирует (если необходимо) при расширении оверлейных сегментов в нижней памяти. В табл. 8 две колонки: для оверлейных сегментов, которые не являются наибольшими в этой оверлейной области (не наибольший сегмент в области), и для оверлейных сегментов, которые расширяются за пределы наибольшего оверлейного сегмента в этой области (свыше наибольшего сегмента в последней области). Содержимое колонок следует понимать так: да — SIPP модифицирует данные, нет — SIPP не модифицирует данные, н/д — данные не пригодны для модификации.

Таблица 8

Данные	Не наибольший сегмент в области	Свыше наибольшего сегмента в последней области
1	2	3
Ячейка 50	нет	да
Размер области	н/д	н/д
Старший адрес корневого сегмента +2	нет	нет
Старший адрес последней оверлейной области +2	нет	да
Размер текущего оверлейного сегмента в словах	да	да
Длина и размер окна в 600к	н/д	н/д
Смещение в 600к	н/д	н/д

5.5.11.3. Расширение программы с оверлеями в расширенной памяти

Если программа с оверлеями в расширенной памяти, то можно расширить:

— корневой сегмент до конца последнего блока, занимаемого корневым сегментом;

— оверлейный сегмент до конца последнего блока данного сегмента.

В табл. 9 показано, какие ячейки SIPP модифицирует (если необходимо) при расширении корневого или оверлейного сегмента программы с оверлеями в расширенной памяти. В табл. 9 колонка 'корневой сегмент' — для корневого сегмента; колонка 'не наибольший сегмент' — для оверлейного сегмента, который не является наибольшим в данной оверлейной области; колонка 'свыше наибольшего' — для оверлейного сегмента, который расширяется свыше размера наибольшего оверлейного сегмента соответствующего данной области. Содержимое колонок следует понимать так: да — SIPP модифицирует данные (если необходимо); нет — SIPP не модифицирует данные.

Таблица 9

Данные	Корневой сегмент	Не наи- больший сегмент	Сегмент свыше наи- большего
1	2	3	4
Ячейка 50	да	нет	нет
Размер области	нет	да	да
Старший адрес корнево- го сегмента +2	да	нет	нет
Старший адрес послед- ней оверлейной области +2	да	нет	нет
Размер текущего овер- лейного сегмента в сло- вах	нет	да	да
Длина и размер окна в 600к	нет	да	да
Смещение в 600к	нет	да	да

5.5.11.4. Расширение программы с оверлеями в нижней и расширенной памяти

Если оверлейная программа имеет сегменты как в нижней, так и в расширенной памяти, SIPP не позволяет расширять корневой сегмент. Оверлейные сегменты в нижней памяти можно расширить до размера наибольшего сегмента соответствующего данной области. Если такой оверлейный сегмент загружается в последнюю оверлейную область, то его можно расширить до конца последнего блока данного оверлейного сегмента.

Оверлейный сегмент в расширенной памяти можно расширить до конца последнего блока данного сегмента.

В табл. 10 показано, какие ячейки SIPP модифицирует (если необходимо) при расширении оверлейных сегментов как в расширенной, так и в нижней памяти. В табл. 10 колонка '/O не наибольший' — для оверлейного сегмента в нижней памяти, если сегмент не является наибольшим в данной оверлейной области; колонка '/O свыше наибольшего' — для оверлейного сегмента в нижней памяти, размер которого превышает размер наибольшего сегмента соответствующего данной оверлейной области; колонка '/V не наибольший' — для оверлейного сегмента в расширенной памяти, если он не является наибольшим в данной области; колонка '/V свыше наибольшего' — для оверлейного сегмента в расширенной памяти, размер которого превышает размер наибольшего сегмента соответствующего данной области. Содержимое колонок следует понимать так: да — SIPP модифицирует данные (если необходимо); нет — SIPP не модифицирует данные.

Таблица 10

	/O не наи- боль- ший	/O свыше наиболь- шего	/V не наи- боль- ший	/V свыше наиболь- шего
1	2	3	4	5
Ячейка 50	нет	да	нет	нет
Размер области	нет	нет	да	да
Старший адрес корне- вого сегмента +2	нет	нет	нет	нет
Старший адрес послед- ней оверлейной обла- сти +2	нет	да	нет	нет

1	2	3	4	5
Размер текущего оверлейного сегмента в словах	да	да	да	да
Длина и размер окна в 600к	нет	нет	да	да
Смещение в 600 к	нет	нет	да	да

Примечание. Если при расширении оверлейных сегментов в расширенной памяти будет превышен физический адрес области 96К слов, SIPP печатает предупреждающее сообщение. В этом случае следует подать команду СУ/С для прерывания модификации, поскольку многие ячейки области связи с системой уже будут содержать неверные данные.

5.6. Контрольная сумма программы SIPP

Программа SIPP вычисляет контрольную сумму после внесения «Заплаты». Контрольная сумма помогает проверить сделанную работу, т. е. напоминает, что ошибка существует, не указывая место ошибки. Алгоритм вычисления контрольной суммы использует адрес измененной ячейки и ее новое содержимое (в контрольную сумму включаются только ячейки, которые были изменены в процессе модификации). Если во время модификации код слова менялся много раз, SIPP вводит в контрольную сумму последний указанный код данного слова.

Если в строке команды был указан переключатель /D, то по завершении модификации SIPP печатает: CHECKSUM=NNNNNN.

Если в строке команды был указан переключатель /C, SIPP по завершении модификации печатает: CHECKSUM? В ответ следует ввести контрольную сумму для «Заплаты». Если указана неправильная контрольная сумма, SIPP печатает:

?SIPP—F—CHECKSUM ERROR

затем SIPP возвращается к началу диалога к подсказке SEGMENT? или BASE?. Это позволяет, не прекращая модификации, найти и исправить ошибки. Программа SIPP не вводит «Заплату» в файл до тех пор, пока не будет введена правильная контрольная сумма. Для прерывания процедуры модификации следует напечатать СУ/С.

5.7. Выполнение программы SIPP из косвенного командного файла

Косвенный командный файл, созданный программой SIPP, содержит команды, необходимые для ввода «Заплаты» в файл или на том. Порядок появления команд в косвенном командном файле может не совпадать с последовательностью их введения. Однако, это не влияет на правильность вносимых изменений. Косвенный командный файл распечатывается в восьмеричном виде. Для создания косвенного командного файла следует указать его спецификацию (спфком) в командной строке. Тип создаваемого косвенного командного файла по умолчанию — .COM.

Косвенный командный файл всегда содержит контрольную сумму, вычисленную после ввода изменений с терминала. Если в командной строке был указан переключатель /C, SIPP запрашивает контрольную сумму по окончании модификации. Если указана правильная контрольная сумма, SIPP завершает создание косвенного командного файла. Если в командной строке был указан переключатель /A, SIPP вводит /A в косвенный командный файл.

Пример.

В следующем примере создается косвенный командный файл TEST.COM. Заметим, что файл TEST.COM не модифицируется, т. к. указан переключатель /L.

. R SIPP

*TEST=TEST/L

BASE? 5000<BK>

OFFSET? 20<BK>

BASE	OFFSET	OLD	NEW
005000	000020	032764	240<BK>
005000	000022	177400	240<BK>
005000	000024	000002	1016<BK>
005000	000026	001016	CU/Y<BK>

*CU/C

копия файла TEST.COM следующая:

.TYPE TEST.COM

RUN SIPP

DK:TEST.SAV/C

5000

20

240

240

1016

^Y

165617

^C

Число 165617 — контрольная сумма для даниой «Заплаты». Для выполнения косвенного командного файла TEST.COM, следует в ответ на точку (.) монитора подать команду: @TEST<BK>

Если косвенный файл TEST.COM выполняется во время действия установки SET TT:QUIET, то выполнение косвенного командного файла не сопровождается печатью строк на терминале.

5.8. Выполнение программы SIPP из пакета заданий

Можно внести изменения в файл из потока заданий. Для этого требуется создать задание — получить косвенный командный файл и отредактировать его с помощью редактора текста:

— ввести команды программы

BATCH (⌘ JOB/FODS, TTYIO, ⌘ EOJ);

— поставить точку (.) перед командой RUN и звездочку перед каждой последующей командной строкой;

— удалить CY/C (ΛC).

Пример.

В данном примере приведено задание, созданное по редактору текста из косвенного командного файла TEST.COM.

```
⌘ JOB/FODS
      TTYIO
.RUN SIPP
*DK:TEST.SAV/C
*5000
*20
*240
*240
*1016
*ΛY
*165617
⌘EOJ
```

5.9. Сообщения оператору

Ниже описаны сообщения, выдаваемые программой SIPP.

?SIPP—E—BELOW SEGMENT BOUNDARY

Причина. Попытка открыть ячейку, адрес которой меньше младшего адреса сегмента; SIPP печатает подсказку OFFSET?

Действие. Проверить, чтобы указанный адрес был не меньше младшего адреса сегмента.

?SIPP—E—CHECKSUM ERROR

Причина. Указана неправильная контрольная сумма; SIPP печатает подсказку SEGMENT? или BASE? Для нахождения и исправления ошибки.

Действие. Если указана неправильная контрольная сумма, следует подать CY/C и <BK> для возврата к подсказке CHECKSUM?, и ввести правильную контрольную сумму. Если была сделана ошибка во время модификации, следует подать команду ;V для воспроизведения на терминале внесенных изменений. Исправить ошибку, напечатать CY/Y и <BK>, а затем ввести правильную контрольную сумму.

?SIPP—E—END OF FILE

Причина. Попытка модификации ячейки, адрес которой больше старшего адреса файла.

Действие. SIPP печатает подсказку OFFSET?, ранее сделанные изменения не отменяются. Проверить, чтобы указанный адрес ячейки был не больше старшего адреса файла.

?SIPP—E—EXCEEDS PROGRAM LIMIT

Причина. Адрес ячейки, которая модифицируется или проверяется, больше старшего адреса программы.

Действие. SIPP печатает подсказку OFFSET?, ранее сделанные изменения не отменяются. Ячейка, открытая в данный момент, не модифицируется.

?SIPP—E—EXCEEDS SEGMENT BOUNDARY

Причина. Попытка проверить или модифицировать ячейку, адрес которой вне границ сегмента.

Действие. SIPP печатает подсказку OFFSET?, ранее сделанные изменения не отменяются.

?SIPP—E—INVALID ADDRESS

Причина. Указан недопустимый адрес в качестве адреса сегмента, базы или смещения.

Действие. SIPP печатает подсказку OFFSET?. Ввести правильный адрес.

?SIPP—E—INVALID BOUNDARY SIZE

Причина. Область поиска превышает размер файла.

Действие. Повторно ввести команду поиска и ответить на подсказку, указав правильную область поиска.

?SIPP—E—INVALID COMMAND

Причина. Введена недопустимая командная строка.

Действие. Ввести правильную командную строку.

?SIPP—E—INVALID EXTENSION OF ROOT SEGMENT

Причина. Попытка модификации корневого сегмента вне границ этого сегмента. SIPP печатает подсказку OFFSET?.

Действие. Если файл содержит оверлеи в расширенной памяти, размер корневого сегмента может быть увеличен до старшего адреса последнего блока области. Нельзя расширять корневой сегмент, если файл содержит оверлеи в нижней памяти.

?SIPP—E—INVALID INPUT

Причина. При вводе данных имела место ошибка. SIPP печатает подсказки.

Действие. Проверить введенные данные и повторить операцию.

?SIPP—E—INVALID OPTION: /X

Причина. Указан недопустимый переключатель.

Действие. Ввести правильную командную строку.

?SIPP—E—INVALID OPTION COMBINATION

Причина. Переключатели /C и /D использованы одновременно.

Действие. Повторно ввести командную строку, указав лишь один из этих переключателей.

?SIPP—E—INVALID SEARCH COMMAND

Причина. Команда или данные, введенные на подсказку SIPP для поиска, недопустимы.

Действие. Проверить введенную команду или данные. Повторить операцию.

?SIPP—E—INVALID SEGMENT NUMBER

Причина. Указан несуществующий номер сегмента. SIPP повторно печатает подсказку SEGMENT? или START?.

Если находится в режиме поиска.

Действие. Проверить карту загрузки программы для определения и исправления номера оверлейного сегмента.

?SIPP—E—NOT B3 OVERLaid FILE

Причина. Модифицируемый файл — файл оверлейной структуры, созданный редактором связей ФОДОС.

Действие. Повторно отредактировать программу редактором связей ФОДОС-2 или модифицировать программу по PATCH.

?SIPP—E—OLD BASE ADDRESS

Причина. Указан нечетный адрес базы. SIPP вновь печатает подсказку BASE?.

Действие. Указать четный адрес базы. Нечетный адрес может быть указан в ответ на подсказку OFFSET?

?SIPP—E—PATCH BUFFER FULL

Причина. Вся память, предназначенная для хранения изменений, использована.

Действие. Подать СУ/У и <ВК> для внесения изменений, напечатанных ранее. Если это сообщение имело место во время расширения файла, сегмента или программы, следует повторно открыть файл и ввести ноль (0) в ячейку, следующую за той, которая вызвала печать этого сообщения об ошибке.

?SIPP—E—REGION SIZE EXCEEDS

Причина. Модифицируемая программа расширена сверх максимального адреса оверлейной области.

Действие. Остановить текущую операцию по команде СУ/С и повторно выполнить модификацию, или продолжить текущую операцию, исправив размер этого сегмента до размера указанного в оверлейной таблице и два слова размера для этого сегмента в блоке определения окна драйвера.

?SIPP—F—DEVICE FULL DEV:FILNAM.TYP

Причина. Устройство вывода не имеет достаточно свободной области для создания выходного файла.

Действие. Использовать команду SQUEEZE для сжатия диска или использовать другое устройство для этой операции.

?SIPP—F—FILE NOT FOUND DEV:FILNAM.TYP

Причина. Входного файла нет на указанном устройстве.

Действие. Проверить наличие файла на указанном устройстве и ввести повторно командную строку.

?SIPP—F—INPUT ERROR DEV:FILNAM.TYP

Причина. При чтении модифицируемого файла произошла аппаратная ошибка.

Действие. Проверить исправность и готовность оборудования. Повторить операцию.

?SIPP—F—INSUFFICIENT MEMORY

Причина. Недостаточно памяти для работы программы SIPP.

Действие. Разгрузить драйверы временно неиспользуемых устройств, удалить основное и системные задания.

?SIPP—F—INVALID DEVICE DEV:

Причина. Указанное устройство не является устройством прямого доступа.

Действие. Использовать допустимое устройство.

?SIPP—F—OUTPUT ERROR DEV: FILNAM.TYP

Причина. При модификации файла по программе SIPP произошла аппаратная ошибка.

Действие. Проверить исправность и готовность оборудования. Повторить операцию.

?SIPP—F—PATCH ABORTED, NO MODIFICATION MADE

Причина. Произошла аппаратная ошибка. Изменения в файл не введены.

Действие. Проверить исправность и готовность оборудования. Повторить операцию.

?SIPP—F—PROTECTED FILE ALREADY EXISTS
DEV: FILNAM.TYP

Причина. Попытка создать косвенный командный файл с именем уже существующего защищенного файла.

Действие. Использовать команду монитора UNPROTECT или переключатель /Z программы PIP для отмены защиты файла.

?SIPP—F—SIZE FUNCTION FAILED

Причина. Аппаратная ошибка произошла в то время, когда программа SIPP определяла размер тома.

Действие. Проверить исправность и готовность оборудования. Повторить операцию.

?SIPP—I—APPROACHING SEGMENT BOUNDARY

Причина. Программа SIPP превысила верхнюю границу (или старший адрес блока) оверлейного сегмента не более чем на 5 слов.

Действие. Не требуется. Операция продолжается.

?SIPP—I—END OF FILE

Причина. В ответ на подсказку OFFSET? введен адрес больше старшего адреса файла.

Действие. SIPP вновь печатает подсказку OFFSET?. Ввести правильный адрес.

?SIPP—I—EXTENDING HIGH LIMIT

Причина. Попытка изменить код ячейки, адрес которой больше старшего адреса программы.

Действие. Не требуется.

?SIPP—I—EXTENDING OVERLAY SEGMENT

Причина. Попытка изменить код ячейки, адрес которой больше старшего адреса сегмента, но меньше старшего адреса последнего блока оверлейной области.

Действие. Не требуется.

?SIPP—I—NO MODIFICATION MADE

Причина. После создания «Заплаты» или после команды ;V изменения не были внесены в файл.

Действие. Не требуется.

?SIPP—W—PATCH BUFFER APPROACHING LIMIT

Причина. Переполнение памяти для хранения изменений.

Действие. Можно внести не более пяти изменений. Для расширения файла следует подать команду SU/Y и ввести оставшиеся изменения.

6. МОДИФИКАЦИЯ ТЕКСТОВЫХ ФАЙЛОВ

6.1. Назначение программы и условия выполнения программы

Программа модификации текстовых файлов (SLP) предназначена для модификации файлов, содержащих символьную информацию. Программа SLP использует входной текстовый файл и командный файл, который создается при сравнении двух исходных файлов по программе сравнения текстовых файлов (SRCCOM).

6.2. Выполнение программы.

Для вызова SLP с системного устройства следует подать с терминала команду

R SLP <BK> или SLP <BK>

после того, как монитор напечатает на терминале точку.

После вызова SLP печатает звездочку (*) и ожидает ввода командной строки. Если в это время нажать клавишу <BK>, то SLP напечатает номер своей версии.

6.3. Команды оператора

Режим работы программы SLP задается введением с терминала командной строки. Формат командной строки:

[выходспф][листспф]=входспф,комспф[/прк...]

где выходспф — спецификация модифицированного текстового файла (устройство, имя и тип файла);

листспф — спецификация файла листинга; листинг файла состоит из последовательно пронумерованных строк модифицированного текстового файла с указанием признака модификации для каждой удаляемой строки;

- признак модификации — произвольный набор символов, идентифицирующий удаляемую строку;
- входспф — спецификация входного текстового файла, который модифицируется с помощью программы SLP;
- комспф — спецификация командного файла, содержащего строки текста и команды для модификации текстового файла; командный файл может быть создан по программе SRCCOM и будет иметь тип .SLP;
- /прк... — переключатель (см. табл. 11).

Устройство для вывода по умолчанию — системный терминал, тип файла по умолчанию для модифицированного текстового файла — .MAC, тип файла по умолчанию для файла листинга — .LST, тип файла по умолчанию для входного текстового файла — .MAC, тип файла по умолчанию для командного файла — .MAC.

Таблица 11

ПЕРЕКЛЮЧАТЕЛИ ПРОГРАММЫ SLP

Переключатель	Функция
1	2
/A	Запрещает печать признака модификации в выходных файлах
/B	Заменяет табуляции на пробелы между последним печатаемым символом строки и признаком модификации
/C[:N]	Определяет или оценивает содержимое входного текстового файла программы SLP, командного файла программы SLP или обоих, используя контрольные суммы. Переключатель /C используется для определения контрольной суммы файла. Переключатель /C:N используется для проверки содержимого файла. SLP вычисляет контрольную сумму для файла и сравнивает результат со зна-

1	2
	чением, которое указывается числом N
/D	Печатает «пустую» строку после каждой строки файла
/L:N	Устанавливает размер буфера строки, где N — максимальное число символов в строке (включая пробелы, табуляции и символы признака модификации); по умолчанию N=200.
/N	Запрещает создание копии файла с типом .BAK, когда SLP изменяет входной файл
/P:N	Устанавливает начало поля признака модификации, N — номер колонки, с которой печатается признак модификации. По умолчанию N=73. Заметим, что признак модификации может «забить» текстовую строку, если указать начало поля признака модификации с колонки, занимаемой строкой. SLP округляет указанное число N до ближайшей позиции табулирования. Например, если вы указываете N, равное 46., программа SLP округляет это число до 49 .
/S:N	Устанавливает размер признака модификации; N — количество символов в признаке модификации. По умолчанию N=12.. Максимальное значение N равно 16.
/T	Сохраняет завершающие пробелы и символы табуляции при корректировке входного текстового файла. По умолчанию, SLP перемещает завершающие пробелы и символы табуляции на конец строки при введении входного текстового файла

Примечание. Для всех переключателей, перечисленных в табл. 11, N — целое восьмеричное (N) или десятичное (N.) число.

После выполнения операции SLP печатает звездочку (*) и ожидает ввода новой командной строки.

Для выхода из SLP и передачи управления монитору следует подать команду CY/C, если SLP ожидает ввода терминала, или дважды CY/C, если SLP выполняет операцию. Для повторного вызова SLP следует подать команду REENTER <BK> или R SLP <BK>.

6.4. Пример

В данном примере SLP используется для модификации исходного файла KONTFG.MAC, который сравнивается с файлом CONTFG.MAC по программе SRCCOM. CONTFG.MAC содержит следующие строки:

```

        .TITLE      KONTFG
        .MCALL      .SDATC, .PRINT, .MRKT, .QSET, .SPND
START:  .PRINT      #MSG
        .QSET       #QUEUE, #100.
        .MRKT       #AREA, #TIME, #MKTC, #1
        .SPND
MKTC:   CMP         MSGCNT, #90.
        BGT         MKTC1
        .SDATC      #AREA, #BUFFER, #30., #SDATC
        INC         MSGCNT
MKTC1:  .MRKT       #AREA, #TIME, #MKTC, #1
        RETURN

```

Модифицируемый файл KONTFG.MAC имеет вид:

```

        .TITLE      KONTFG
        .MCALL      .SDATC, .PRINT, .MRKT, .QSET, .SPND
START:  .PRINT      #MSG
        .QSET       #QUEUE, #111.
        .MRKT       #EREA, #TIME, #MKTC, #1
        .SPND
MKTC:   CMP         MSGCNT, #99.
        BGT         MKTC1
        .SDATC      #EREA, #BUFER, #30., #SDATC
        INC         MSGCNT
MKTC1:  .MRKT       #EREA, #TIME, #MKTC, #1
        RETURN

```

Полученный в результате сравнения (с помощью программы SRCCOM) командный файл CONTFG.SLP содержит

необходимую информацию для модификации KONTRG.MAC таким образом, чтобы KONTRG.MAC был идентичен CONTRG.MAC.

Командная строка для SLP имеет следующий вид:

*KONTRG,KONTRG=KONTRG,CONTRG.SLP

после выполнения командной строки SLP присваивает тип .BAK входному текстовому файлу KONTRG.MAC, а тип .MAC — модифицированному текстовому файлу. SLP также создает файл листинга KONTRG.LST, в котором все строки последовательно нумеруются и к каждой новой строке добавляется признак модификации. По умолчанию признак модификации — «**NEW**». Модифицированный файл KONTRG.MAC теперь идентичен CONTRG.MAC.

Файл KONTRG.LST имеет вид:

```
SLP B03.00 KONTRG,KONTRG=KONTRG.MAC,CONTRG.SLP
1. .TITLE CONTF ;**NEW**
2. ;**NEW**
3. .MCALL .SDATC,.PRINT,.MRKT,.QSET,.SPND ;**NEW**
4. ;**NEW**
5. START: .PRINT #MSG ;**4
6. .QSET #QUEUE,#100. ;**NEW**
7. .MRKT #AREA,#TIME,#MKTC,#1 ;**NEW**
8. .SPND ;**2
9.
10.
11. MKTC: CMP MSGCNT,#90. ;**NEW**
12. BGT MKTC1 ;**1
13. .SDATC #AREA,#BUFFER,#30.,#SDATC ;**NEW**
14. INC MSGCNT ;**1
15. MKTC1: .MRKT #AREA,#TIME,#MKTC,#1 ;**NEW**
16. RETURN ;**1
```

где ;**NEW** — признак модификации по умолчанию;
;**1 и ;**2 — дополнительные признаки модификации.

Примечание. Дополнительный признак модификации печатается под признаком модификации и представляет число последовательно замененных строк.

Файл CONTRG.SLP, полученный с помощью программы SRCCOM, имеет вид:

—1,4

```
.TITLE CONTF
.MCALL .SDATC,.PRINT,.MRKT,.QSET,.SPND
```

—6,7 .QSET #QUEUE, #100.
 .MRKT #AREA, #TIME, #MKTC, #1
 —11, 11
 MKTC: CMP MSCNT, #90.
 —13, 13 .SDATC #AREA, #BUFFER, #30., #SDATC
 —15, 15
 MKTC1: .MRKT #AREA, #TIME, #MKTC, #1
 /

6.5. Создание командного файла

Командный файл можно получить с помощью программы SRCCOM или создать его, используя программу редактор текста.

Табл. 12 содержит операторы, используемые для создания командного файла.

Таблица 12

Оператор	Функция
1	2
—	Признак начала замены строк. Любые символы, предшествующие признаку начала замены строк в командном файле для программы SLP, игнорируются. В этом случае SLP печатает предупреждающее сообщение и символы, которые игнорируются. Наличие признака начала замены строк обязательно
↖	Запрещает печать признака модификации.
%	Этот оператор должен занимать первую позицию строки Разрешает печать признака модификации

1	2
/ //	Признак конца командного файла Указывает конец одной из серии текстов изменений в единственном командном файле.
<	Каждый текст изменяет один входной текстовый файл. Этот оператор используется, когда необходимо включить изменения для более чем одного файла в единственном командном файле. Этот оператор печатается после каждой серии изменений. На следующей строке печатается командная строка, указывающая следующий изменяемый входной файл, на последующих строках печатается текст изменений для этого файла Операторы (командного файла), следующие за оператором «<» рассматриваются программой SLP как текст (и не идентифицируются)

6.5.1: Формат строки командного файла

Общий формат строки командного файла программы SLP:

— идентификатор1, [идентификатор2], [/признак модификации/]
[;] входная строка

где:

—

идентификатор1

идентификатор2

- признак заменяемой строки;
- часть текстовой строки, используемая в качестве идентификатора строки;
- часть текстовой строки, используемая в качестве идентификатора строки. При наличии идентификатора1 определяет диапазон строк, которые могут быть заменены или удалены. Строка, ука-

- званная идентификатором2, должна встречаться в исходном файле после строки, указанной идентификатором1;
- /признак модификации/ — представляет набор символов, который используется как признак модификации. SLP печатает признак модификации справа от каждой заменяемой строки. Наклонная черта (/) до и после признака модификации является обязательным ограничителем;
- входная строка — представляет строку текста, которую SLP вводит в файл сразу за указанной строкой;
- ! — произвольный ограничитель командной строки (может быть опущен).

Все поля в строке изменений имеют точное местоположение, если, например, указывается только идентификатор1 и признак модификации, то необходимо поставить две запятые, ограничивая поле идентификатора2. Если указывается только признак модификации, то перед признаком модификации необходимо поставить две запятые.

Идентификаторы строк могут использовать один из следующих видов:

/подстрока/[+N]
 /подстрока... подстрока/[+N]
 номер [+N]
 .+N

где:

/подстрока/[+N]

— представляет часть строки, ограниченную слева и справа символом «наклонная черта» (/). SLP устанавливает указатель строки на строку, которая содержит указанную подстроку. +N означает смещение от строки, содержащей подстроку;

/подстрока... подстрока/[+N]

— представляет часть строки, ограниченную слева и справа символом «наклонная че-

рта» (/). SLP находит строку, в которой две подстроки ограничивают большую подстроку. Многоточие используется для того, чтобы отделить две подстроки. +N представляет смещение от строки, определенной указателем
 /подстрока... подстрока/;
 номер [+N] — представляет последовательный номер строки, на которую SLP перемещает указатель строки. +N представляет смещение от строки, определенной номером;
 .+N — представляет смещение от указателя строки. SLP интерпретирует точку (.) как текущее местоположение указателя строки, а +N — как смещение от него.

6.5.2. Создание файла листинга

Программу SLP можно использовать для создания файла листинга с указанием номеров строк. При создании файла листинга входным файлом для SLP является текстовый файл.

Командная строка для создания файла листинга имеет следующий формат:

* ,листспф=входспф

где: листспф — спецификация файла листинга;

входспф — спецификация текстового файла.

Например, имеем текстовый файл PROG.MAC:

```

      .TITLE    PROG.MAC
      .MCALL    .TTYOUT, .EXIT, .PRINT
EXP:   .PRINT   #MESSAG
      MOV      #N,R5
FIRST: MOV      #N+1,R0
      MOV      #A,R1
```

В результате выполнения командной строки

* ,PROG=PROG

Получаем файл листинга PROG.LST:

```
SLP      B03.00      ,PROG=PROG.MAC
1.              .TITLE  PROG.MAC
2.              .MCALL  .TTYOUT, .EXIT, .PRINT
3.
4.      EXP:  .PRINT  #MESSAG
5.              MOV    #N,R5
6.      FIRST: MOV    #N+1,R0
7.              MOV    #A,R1
```

6.5.3. Введение строк в файл

Для введения в файл новых строк можно использовать в командном файле один из следующих видов идентификаторов:

- номер
- {+N}, , [/признак модификации/]
- /строка/

Заметим, что второй вид содержит две запятые между идентификатором и признаком модификации. Если не указывается признак модификации, то эти запятые не обязательны.

Файл NUMBER.PAS, в который SLP вводит новые строки:

```
PROGRAM NUMBER;
TYPE      TEXT      =FILE OF CHAR;
          PTR        =^WORDNODE;
          WORDNODE=REKORD
                        WORD:ARRAY[1..30] OF CHAR;
                        NEXT:PTR;
          END;
VAR       P,TOP      :PTR;
          INTEXT     :TEXT;
          I           :INTEGER;
```

командный файл OMEGA.MAC для выполнения этой процедуры:

```
—/PTR/
(*POINTER TO NODE*)
/
```

После выполнения командной строки создается файл листинга, в котором строка с номером 5 — новая:

```
SLP B03.00
NUMBER.PAS, NUMBER=NUMBER.PAS, OMEGA.MAC
1. PROGRAM NUMBER;
2.
```



```

3. TYPE   TEXT      =FILE OF CHAR;
4.        PTR       =^WORDNODE;
5. (*POINTER TO NODE*)                      ;**NEW**
6.        WORDNODE=RECORD
7.        WORD:ARRAY[1 ... 30] OF CHAR;
8.        NEXT:PTR;
9.        END;
10.
11. VAR    P,TOP     :PTR;
12.        INTEXT    :TEXT;
13.        I         :INTEGER;

```

SLP нумерует строки, вставляет новый текст и присоединяет признак модификации (по умолчанию — ;**NEW**) к новой строке.

Следующий пример использует тот же файл NUMBER.PAS и командный файл SIGMA.MAC. SIGMA.MAC содержит следующие строки:

```

—/WORDNODE=/+2
      ID :INTEGER;

```

/

В результате выполнения командной строки создается следующий файл листинга:

```

SLP B03.00 NUMBER.PAS, NUMBER=NUMBER.PAS, SIGMA.MAC
1. PROGRAM NUMBER;
2.
3. TYPE   TEXT      =FILE OF CHAR;
4.        PTR       =^WORDNODE;
5.        WORDNODE=RECORD
6.        WORD:ARRAY[1 .. 30] OF CHAR;
7.        NEXT:PTR;
8.        ID        :INTEGER;      ;**NEW**
9.        END;
10.
11. VAR    P,TOP     :PTR;
12.        INTEXT    :TEXT;
13.        I         :INTEGER;

```

SLP вставляет новую строку через две строки после строки, содержащей подстроку WORDNODE. Можно включить в один командный файл текст изменений для нескольких входных файлов. Необходимо печатать символ «двойная косая черта» (//) в строке в конце текста изменений для каж-

дого файла. На следующей строке печатается командная строка, указывающая следующий изменяемый входной файл, на последующих строках печатается текст изменений для этого файла. В конце командного файла необходимо напечатать символ косой черты (/).

Например, следующий командный файл MTMTST.MAC содержит тексты изменений для модификации входных файлов NUMBER.PAS и GTMSG.MAC в следующем порядке:
—/WORDNODE=/+2

```

ID :INTEGER;
//
DY0:NEWMSG=GTMSG.MAC, TMST.MAC
—2,2
.IDENT /01.01/
—7
ADD A,B
—14
B: .WORD ()
/
```

6.5.4. Удаление строк из текстового файла

Формат командной строки:

—идентификатор1, идентификатор2, [/признак модификации/][:]

Строки, первая из которых определяется идентификатором1 и последняя — идентификатором2, удаляются.

Например, для удаления строк с пятой по восьмую из файла NUMBER.PAS, листинг которого имеет вид:

SLP B03.00 ,NUMBER=NUMBER.PAS

```

1. PROGRAM NUMBER;
2.
3. TYPE TEXT = FILE OF CHAR;
4. PTR = ^WORDNODE;
5. WORDNODE=RECORD
6. WORD:ARRAY[1..30] OF CHAR;
7. NEXT:PTR;
8. END;
9.
10. VAR P,TOP :PTR;
11. INTXT :TEXT;
12. I :INTEGER;
```

используя командный файл GAMMA.MAC

—/WORDNODE=/,/END/

/

Получаем листинг модифицированного файла
NUMBER.PAS:

SLP B03.00 NUMBER.PAS, NUMBER=NUMBER.PAS, GAMMA.MAC

```
1. PROGRAM NUMBER;  
2.  
3. TYPE    TEXT    =FILE OF CHAR;  
4.        PTR      =^WORDNODE;  
5.  
6. VAR     P,TOP    :PTR;  
7.        INTTEXT   :TEXT;  
8.        I         :INTEGER;
```

;**—4

6.5.5. Замена строк в файле

При замене строк происходит удаление строк, а затем введение нового текста. Удаление строк происходит со строки, указанной идентификатором1. Идентификатор2 показывает не только количество удаляемых строк, но и местоположение нового текста, который вводит программа SLP.

Например, по команде (—4, +4) в командном файле SLP удаляет строки, начиная со строки, указанной идентификатором1, т. е. со строки 4, и происходит удаление следующих четырех строк (обозначено +4), включая строку 4. Окончательно SLP вводит новые строки, которые были в командном файле и помещает их, начиная со строки, указанной идентификатором1.

Следующий пример иллюстрирует замену строк в файле. Исходный файл BETA.MAC состоит из следующих строк:

```
.TITLE    BETA.MAC  
.MCALL    .TTYOUT, .PRINT, .EXIT  
START:    .PRINT  #MESSAG  
          MOV     #5, R0
```

Командный файл DELTA.MAC состоит из следующих строк:

```
—6,2,/,NEW LINE/  
BNE          START  
MOVB (R2), —(R3)  
/
```

В результате выполнения командной строки для исходного файла BETA.MAC и командного файла DELTA.MAC создается следующий файл листинга:

```

SLP B03.00      BETA, BETA=BETA, DELTA.MAC
1.              .TITLE BETA.MAC
2.
3.              .MCALL .TTYOUT, .PRINT, .EXIT
4.
5. START: .PRINT  #MESSAG
6.              BNE      START                      ;NEW LINE
7.              MOVB     (R2), — (R3)                ;NEW LINE

```

6.6. Определение и оценка содержимого файла

Для определения или оценки содержимого файла используется переключатель контрольной суммы (/C[:N]). Он вычисляет сумму всех символов в кодах КОИ-7 файла. Переключатель контрольной суммы в форме /C:N используется для вычисления контрольной суммы и сравнения ее со значением, которое указывается как N.

Если указанное в переключателе /C:N значение N не совпадает со значением контрольной суммы, вычисленной программой SLP, то программа SLP выдает сообщение:

?SLP—F—SOURCE FILE CHECKSUM ERROR

или

?SLP—F—CORRECTION FILE CHECKSUM ERROR

Для определения контрольной суммы файла введите командную строку программы SLP с переключателем /C для файла, чью контрольную сумму необходимо вычислить.

Например, SLP отвечает на следующую команду

INFILE, INFILE=INFILE.MAC/C, INFILE.SLP

сообщением

?SLP—I—DEV:FILE.TYP CHECKSUM IS N,

где N — контрольная сумма файла INFILE.MAC.

6.7. Сообщения оператору

Ниже описаны сообщения, выдаваемые программой SLP.

?SLP—F—AUDIT TRAIL EXCEEDS LINE LIMIT

Причина. Указана недопустимая комбинация переключателей /P, /S или /L в командной строке.

Действие. Проверить командную строку.

?SLP—F—AUDIT TRAIL OVERWRITES LINE

Причина. Недопустимое расположение признака модификации: он «забывает» текстовую строку

Действие. Использовать переключатель /P для изменения позиции признака модификации.

?SLP—F—BACK UP FILE SUPPRESSION
UNSUCCESSFUL

Причина. Попытка использовать переключатель /N для защищенного файла.

Действие. Снять защиту файла по команде монитора UNPROTECT или использовать переключатель /Z программы PIP.

?SLP—F—CHARACTERS BEFORE THE «—» IN DEV:FILNAM.TYP

Причина. В командном файле изменений перед символом дефис (—) находятся символы, которые игнорируются.

Действие. Установить дефис в начало командного файла изменений.

?SLP—F—COMMAND SYNTAX ERROR

Причина. Ошибка в командной строке.

Действие. Проверить командную строку. Подать команду правильно.

?SLP—F—CORRECTION FILE CHECKSUM ERROR

Причина. Вычисленная контрольная сумма исходного файла, который обновляется, не совпадает со значением, указанным в переключателе /C:N командной строки программы SLP.

Действие. Проверить командную строку. Проверить значение контрольной суммы в переключателе /C:N и имя указанного файла. Повторить командную строку.

?SLP—F—DEVICE FULL DEV:FILNAM.TYP

Причина. На устройстве недостаточно места для выходного файла.

Действие. Увеличить размер свободной области на томе: удалить ненужные файлы или переписать их на другой том, сжать том по команде SQUEEZE.

?SLP—F—EXTRA FILE(S) SPECIFIED

Причина. В командной строке указано более двух выходных или более двух входных файлов.

Действие. Проверить командную строку.

?SLP—F—FILE NOT FOUND DEV:FILNAM.TYP

Причина. Входной файл не найден.

Действие. Проверить, существует ли файл с указанным именем. Подать команду правильно.

?SLP—F—FILE PROTECTED

Причина. Попытка модификации защищенного файла.

Действие. Снять защиту файла по команде монитора RENAME/NOPROTECT или использовать переключатель /Z программы PIP.

?SLP—F—HARD ERROR ON DEVICE

Причина. Ошибка во время выполнения операции ввода-вывода.

Действие. Проверить готовность и исправность оборудования. Повторить операцию.

?SLP—F—INSUFFICIENT MEMORY

Причина. Недостаточно оперативной памяти для выполнения операции.

Действие. Освободить часть оперативной памяти (удалить ненужные драйверы, удалить основное задание, использовать монитор одного задания). Повторить выполнение операции.

?SLP—F—/C INVALID OPTION

Причина. Переключатель /C был указан с выходным файлом в командной строке программы SLP.

Действие. Убедитесь в том, что переключатель /C указан в командной строке программы SLP с входным текстовым файлом программы SLP или с командным файлом.

?SLP—F—INVALID OPTION:/X

Причина. В команде указан недопустимый переключатель.

Действие. Проверить командную строку.

?SLP—F—INVALID RENAME

Причина. Попытка переименовать файл, находящийся на магнитной ленте.

Действие. Операция переименования недопустима для магнитной ленты.

?SLP—F—INVALID VALUE SPECIFIED WITH OPTION:/X

Причина. Недопустимое значение переключателя.

Действие. Подать команду правильно.

?SLP—F—LINE NUMBER ERROR

Причина. Неверный номер строки.

Действие. Проверить командную строку. Подать команду правильно.

?SLP—F—LINE TOO LONG

Причина. Командная строка имеет длину более 80 символов.

Действие. Используйте переключатель /L, чтобы увеличить буфер строки.

?SLP—F—PROTECTED FILE ALREADY EXISTS
DEV:FILNAM.TYP

Причина. Попытка создать файл с тем же именем, что и у имеющегося защищенного файла.

Действие. Отменить защиту имеющегося файла по команде монитора RENAME/NOPROTECT или использовать переключатель /Z программы PIP, либо создать файл с другим именем.

?SLP—F—SEARCH FAILURE IN SOURCE FILE

Причина. Не найдена указанная строка во входном текстовом файле.

Действие. Проверить указанную строку и, если необходимо, указать другую строку.

?SLP—F—SOURCE FILE CHECKSUM ERROR

Причина. Вычисленная контрольная сумма исходного файла, который обновляется, не совпадает со значением, указанным в переключателе /C:N командной строки программы SLP.

Действие. Проверить командную строку. Проверить значение контрольной суммы в переключателе /C:N и имя указанного файла. Повторить командную строку.

?SLP—F—SOURCE FILE NOT SPECIFIED

Причина. В командной строке не указан входной текстовый файл.

Действие. Подать команду правильно.

?SLP—F—SYSTEM ERROR

Причина. Системный файл монитора имеет плохие блоки.

Действие. Использовать новую копию файла монитора.

?SLP—F—UNEXPECTED END OF CORRECTION FILE

Причина. Командный файл не содержит строки с признаком конца файла («/»).

Действие. Изменить командный файл, включив строку с признаком конца файла («/»).

?SLP—F—VALUE REQUIRED:/X

Причина. Не указано значение аргумента для переключателя.

Действие. Указать значение аргумента и подать команду правильно.

?SLP—I—DEV:FILNAM.TYP CHECKSUM IS N

Причина. Указан переключатель /C в командной строке программы SLP для определения контрольной суммы входного файла.

Действие. Не требуется.

?SLP—W—AUDIT TRAIL OVERWRITE EXISTING AUDIT TRAIL

Причина. Попытка записи признака модификации в конце строки, уже имеющей признак модификации. Новый признак модификации «забывает» старый.

Действие. Не требуется.

PSLP—W—NO UPDATE TEXT FOUND

Причина. Командный файл изменений не содержит символ дефис (—). Модификация файла не выполняется и управление передается монитору.

Действие. Установить дефис в командном файле изменений.

7. СРАВНЕНИЕ ДВОИЧНЫХ ФАЙЛОВ

7.1. Назначение программы и условия выполнения программы

Программа сравнения двоичных файлов (BINCOM) предназначена для сравнения двух файлов, содержащих двоичную информацию и выявления различий между ними. Перечень различий программа может выдавать на терминал, построочно-печатающее устройство или формировать выходной файл различий.

Программа BINCOM проверяет два входных файла по словам (или по байтам) и ищет различия. Когда BINCOM находит различие, она печатает номер блока и смещение внутри блока, содержащего различия, восьмеричное значение каждого сравниваемого файла и результат логической операции «исключающее ИЛИ» двух значений. Последнее число помогает определить разряды, содержащее которых различие в двух значениях.

Программа BINCOM используется также для создания командного файла программы SIPP.

7.2. Выполнение программы

Для вызова BINCOM с системного устройства следует подать с терминала команду

R BINCOM <BK> или BINCOM <BK>

после того, как монитор напечатает на терминале точку.

После вызова BINCOM печатает звездочку (*) и ожидает ввода командной строки. Если в это время нажать клавишу <BK>, то BINCOM напечатает номер своей версии.

7.3. Команды оператора

Режим работы программы BINCOM задается введением с терминала командной строки. Формат командной строки: [выходспф[/прк]][,комспф[/прк]]=входспф1, входспф2[/прк...]

где выходспф — спецификация выходного файла (устройство, имя и тип файла);
 комспф — спецификация командного файла. Он содержит команды, необходимые для модификации входных файлов;
 входспф1 — спецификация первого сравниваемого файла (старого);
 входспф2 — спецификация второго сравниваемого файла (нового);
 /прк — переключатель (см. табл. 13).

Таблица 13

ПЕРЕКЛЮЧАТЕЛИ ПРОГРАММЫ BINCOM

Переключатель	Функция
1	2
/B	Сравнивает два входных файла по байтам (при отсутствии переключателя сравнивает по словам)
/D	Сравнивает два тома, начиная с нулевого блока. Если один том имеет больший размер, то BINCOM печатает сообщение и сравнение будет закончено, как только все блоки меньшего из томов будут прочитаны. Командный файл для программы SIPP не создается
/E:N	Заканчивает сравнение на блоке с номером N, где N — восьмеричное число (при отсутствии переключателя сравнение завершается при обнаружении конца одного из входных файлов или конца одного из входных томов)
/H	Печатает на терминале список имеющихся переключателей
/O	Создает выходной файл даже при отсутствии различий между двумя сравниваемыми файлами. Если указать этот переключатель после выходного файла, BINCOM создаст выходной

1	2
/Q	файл, несмотря на отсутствие различий между сравниваемыми файлами. Если указать этот переключатель после командного файла, BINCOM создаст командный файл независимо от существования различий. Для получения выходного и командного файлов достаточно указать этот переключатель в конце командной строки. Запрещает печать файла различий и печатает одно из следующих сообщений: ?BINCOM—W—FILES ARE DIFFERENT или ?BINCOM—W—DEVICES ARE DIFFERENT IF APPLICABLE (или ?BINCOM—I—NO DIFFERENCES FOUND)
/S:N	Начинает сравнение с блока под номером N, где N — восьмеричное число

Устройство для вывода по умолчанию — системный терминал, тип файла по умолчанию для выходного файла — .DIF, тип файла по умолчанию для командного файла — .COM.

После выполнения операции BINCOM печатает звездочку (*) и ожидает ввода новой командной строки.

Для выхода из BINCOM и передачи управления монитору следует подать команду SU/C, если BINCOM ожидает ввода с терминала, или дважды SU/C, если BINCOM выполняет операцию. Для повторного вызова BINCOM следует подать команду REENTER <BK> или R BINCOM <BK>.

7.4. Использование конструкции (*) в командной строке

Конструкция (*) используется для многократного сравнения двоичных файлов при введении с терминала только одной командной строки. При создании командного файла программы SIPP конструкция (*) не используется.

Конструкцию (*) можно использовать в любой из спецификаций сравниваемых файлов (старого и нового). Раз-

личные типы сравнений представляют зависимость от того, будет ли использоваться конструкция (*) только в одном или обоих сравниваемых файлах.

Если использовать конструкцию (*) только в одной спецификации сравниваемых файлов, BINCOM сравнивает файл, который вы указываете без конструкции (*), со всеми вариантами файла, в спецификации которого указана конструкция (*). Конструкция (*) указывает ту часть спецификации файла, которая будет изменяться. Можно использовать этот метод для сравнения одного файла с несколькими другими файлами. Например, когда следующая командная строка выполняется, BINCOM сравнивает файл TEST1.SAV на устройстве DY0: со всеми файлами устройства DY1: с именем TEST2:

TEST=DY0:TEST1.SAV, DY1:TEST2.

Результаты всех сравнений программа может выдавать на терминал или формировать в выходной файл. В данном примере все различия при сравнении формируются в файл TEST.DIF.

Если использовать конструкцию (*) в обеих спецификациях сравниваемых файлов, то она будет представлять часть спецификации файла, которую необходимо оставить такой же в обоих сравниваемых файлах. Можно использовать этот метод для сравнения нескольких частей файлов. Например, когда следующая командная строка выполняется, BINCOM сравнивает части файлов; первый сравниваемый файл каждой части имеет имя PROG1, второй имеет имя PROG2. Типы обоих файлов из каждой части должны совпадать.

DY0:PROG1.*,DY1:PROG2.

BINCOM отыскивает первый файл на DY0: с именем PROG1, замечает тип этого файла. Затем отыскивает имя файла PROG2 на устройстве DY1: с тем же типом, что и PROG1. Если совпадение найдено, BINCOM сравнивает два файла и выдает различия на терминал или формирует в выходной файл, если он указан. Аналогично происходит сравнение последующих пар файлов.

7.5. Формат выходного файла

При включении спецификации выходного файла в командную строку BINCOM создает файл, содержащий различия между двумя сравниваемыми файлами. Если не указать выходной файл, BINCOM выдает различия на терминал. Если указать переключатель /Q, BINCOM не выдает различий и не создает выходной файл.

BINCOM сравнивает файлы по байтам или по словам. Первая строка в листинге файла различий идентифицирует сравниваемые файлы. Далее BINCOM напечатает строку пробелов и список различий между двумя сравниваемыми файлами. Каждая строка файла различий имеет следующий вид:

BBBBBBB OOO/FFFFFF SSSSSS XXXXXX

где BBBBBB — номер блока, содержащего различия;

OOO — адрес слова или байта относительно начала блока, содержащего различия;

FFFFFF — значение первого сравниваемого файла (тома);

SSSSSS — значение второго сравниваемого файла (тома);

XXXXXX — результат логической операции «исключающее ИЛИ» для двух значений.

При наличии нескольких различий в одном блоке BINCOM напечатает номер этого блока только один раз.

7.6. Примеры.

В первом примере сравниваются файлы TEST1.TST и TEST3.TST на устройстве DK. Выходной файл и переключатель не указаны.

.R BINCOM

*TEST1.TST, TEST3.TST

BINCOM COMPARING/DK:TEST1.TST — — DK:TEST3.TST

000000 002/ 051511 051502 000013

?BINCOM—W—FILES ARE DIFFERENT

Число 051511 — содержимое второй ячейки блока 0 в файле TEST1.TST.

Число 051502 — содержимое второй ячейки блока 0 в файле TEST3.TST.

Число 000013 — результат логической операции «исключающее ИЛИ» для двух значений.

Число 002 представляет номер слова в блоке.

В следующем примере указан выходной файл FOO1 как файл различий для файлов TEST1.TST и TEST3.TST.

.R BINCOM

*FOO1=TEST1.TST, TEST3.TST

?BINCOM—W—FILES ARE DIFFERENT

Тип выходного файла по умолчанию — .DIF. Он имеет следующий вид:

BINCOM COMPARING/DK:TEST1.TST — — DK:TEST3.TST

000000 002/ 051511 051502 000013

7.7. Создание командного файла

Программу BINCOM можно использовать для создания командного файла, который вызывает программу модификации файлов SIPP.

В следующем примере создается командный файл FOO2. Перед спецификацией командного файла FOO2 необходимо поставить запятую.

```
.R BINCOM
*,FOO2=TEST1.TST, TEST3.TST
BINCOM COMPARING/DK:TEST1.TST — DK:TEST3.TST
000000 002/ 051511 051502 000013
?BINCOM—W—FILES ARE DIFFERENT
```

Программа BINCOM присваивает файлу FOO2 тип .COM. Он содержит следующие строки:

```
R SIPP
DK:TEST1.TST/A
000000
000000002
051502
^Y
^C
```

Файл FOO2 из приведенного примера может быть использован как командный файл для модификации файла TEST1.TST. Для выполнения командного файла подать команду @FOO2.COM.

7.8. Сообщения оператору

Ниже приведены сообщения, выдаваемые программой BINCOM.

```
?BINCOM—E—DEVICE FULL DEV:FILNAM.TYP
```

Причина. На устройстве недостаточно места для выходного файла.

Действие. Увеличить размер свободной области на томе: удалить ненужные файлы или переписать их на другой том, сжать том по команде SQUEEZE.

```
?BINCOM—E—DEVICE NOT ACTIVE DEV:
```

Причина. Операция ввода-вывода невозможна на указанном устройстве или томе; устройство отключено; нет тома на устройстве; запрещена запись; устройство не описано в таблицах монитора.

Действие. Проверить правильность включения устройства. Установить обслуживание устройства по команде INSTALL.

- ?BINCOM—E—ERROR READING DIRECTORY DEV:**
 Причина. Аппаратная ошибка чтения справочника указанного устройства.
 Действие. Проверить готовность и исправность оборудования.
 Проверить диск на плохие блоки.
- ?BINCOM—E—INPUT ERROR DEV: FILNAM.TYP**
 Причина. Ошибка чтения входного файла FILNAM.TYP.
 Действие. Проверить готовность и исправность оборудования.
 Повторить операцию.
- ?BINCOM—E—INPUT FILE NOT FOUND DEV: FILNAM.TYP**
 Причина. Входной файл, указанный в команде, не найден.
 Действие. Проверить, существует ли файл с указанным именем на указанном устройстве. Подать команду правильно.
- ?BINCOM—E—INSUFFICIENT MEMORY**
 Причина. Недостаточно оперативной памяти для выполнения заданной операции.
 Действие. Освободить часть оперативной памяти (удалить ненужные драйверы, удалить основное задание, использовать монитор одного задания). Повторить выполнение операции.
- ?BINCOM—E—INVALID COMMAND**
 Причина. В командной строке обнаружена ошибка.
 Действие. Ввести правильную командную строку.
- ?BINCOM—E—INVALID DIRECTORY DEV:**
 Причина. Структура справочника тома устройства не является структурой ФОДОС.
 Действие. Использовать справочник тома устройства со структурой ФОДОС.
- ?BINCOM—E—INVALID OPTION/X**
 Причина. В командной строке обнаружен недопустимый переключатель.
 Действие. Ввести правильную командную строку.
- ?BINCOM—E—INVALID OPTION/X: VAL**
 Причина. Не указан аргумент переключателя, либо указано неверное значение аргумента.
 Действие. Ввести правильную командную строку.
- ?BINCOM—E—OUTPUT ERROR DEV: FILNAM.TYP**
 Причина. Возможны следующие причины:
 — ошибка записи;
 — на томе, используемом для вывода, нет свободной области для записи выходного файла.

Действие. Возможны следующие действия:

- проверить готовность и исправность оборудования;
- проверить том на плохие блоки. Повторить выполнение операции;
- указать в командной строке размер выходного файла;
- использовать для вывода другой том.

?BINCOM—E—OUTPUT FILE FULL DEV:FILNAM .TYP

Причина. На томе нет места для записи файла. Обнаружен маркер физического конца тома (для магнитной ленты).

Действие. Сжать том по команде SQUEEZE. Удалить или записать на другой том ненужные файлы. Использовать конструкцию [N] для выходного файла. Использовать другой том.

?BINCOM—I—COMPARING/DEV:FILNAM.TYP—DEV:FILNAM.TYP

Причина. Сравнимые входные файлы имеют различия.

Действие. Не требуется.

?BINCOM—I—NO DIFFERENCES FOUND

Причина. Сравнимые файлы идентичны. BINCOM не создает выходной файл, если не указывается переключатель /O в командной строке или используется команда монитора DIFFERENCES/ALWAYS.

Действие. Не требуется.

?BINCOM—W—DEVICE IS LONGER DEV:

Причина. Сравнимые устройства имеют различный размер буфера.

Действие. Не требуется.

?BINCOM—W—DEVICES ARE DIFFERENT

Причина. Сравнимые устройства различны.

Действие. Не требуется.

?BINCOM—W—FILE IS LONGER DEV:FILNAM.TYP

Причина. Сравнимые файлы имеют различный размер.

Действие. Не требуется.

?BINCOM—W—FILES ARE DIFFERENT

Причина. Сравнимые файлы различны.

Действие. Не требуется.

8. ФОРМАТИРОВАНИЕ

8.1. Назначение программы и условия выполнения программы

Программа форматирования (FORMAT) предназначена для форматирования кассетных дисков на устройстве СМ5400 и гибких дисков на устройстве «ЭЛЕКТРОНИКА ГМД7012».

Для того, чтобы том можно было использовать операционной системой ФОДОС, его следует отформатировать. Под форматированием тома понимается запись заголовков в каждый блок формируемого тома. Возможно форматирование гибкого диска как с одинарной, так и с двойной плотностью.

Повторное форматирование по программе FORMAT может ликвидировать плохие блоки на дисках и гибких дисках, которые образуются в результате длительной эксплуатации томов. Форматирование не гарантирует восстановление каждого плохого блока, но может уменьшить число плохих блоков.

Примечание. Программа FORMAT разрушает все данные, которые находятся на томе.

8.2. Выполнение программы

Для вызова программы FORMAT с системного устройства в ответ на точку (.) монитора следует подать команду:

R FORMAT <BK> или FORMAT <BK>

После вызова программы FORMAT интерпретатор командной строки напечатает звездочку (*) и ожидает ввода командной строки. Если нажать только <BK>, программа FORMAT напечатает номер своей версии; если — СУ/С, управление возвращается системе. Не рекомендуется прерывать операцию форматирования, подавая дважды СУ/С.

8.3. Команды оператора

Формат командной строки: уст:[/прк]

где уст — физическое или логическое имя устройства, которое используется для форматирования тома;

/прк — один или несколько переключателей (табл. 14).

После ввода командной строки перед началом операции форматирования программа FORMAT печатает сообщение:

уст: /ARE YOU SURE?

где уст — физическое имя устройства, указанное в командной строке. Если в командной строке было указано логическое имя, то в опрашивающем сообщении будет напечатано физическое имя указанного устройства.

Для выполнения форматирования в ответ на это сообще-

ние следует ввести Y или любую строку, начинающуюся с Y, и <BK>. Ввод N или CY/C запрещает форматирование. Интерпретатор командной строки печатает (*) и ожидает ввода командной строки. Любой другой ответ вызывает повторную печать сообщения.

Допустимо использовать программу FORMAT из косвенного командного файла. В этом случае ответ Y и <BK> должны следовать на следующей строке после командной строки программы FORMAT. По переключателю /Y можно запретить печать опрашивающего сообщения (тогда и в косвенном командном файле после командной строки, если она содержит переключатель /Y, не следует вводить ответ Y и <BK>).

При попытке форматирования тома, когда загружено основное задание, система выдает сообщение:

BACKGROUND LOADED.

<UCT:>/FORMAT—ARE YOU SURE?

Пользователю следует ввести:

— Y (или любую строку, начинающуюся с Y) и <BK> для выполнения форматирования;

— N (или любую строку, начинающуюся с N) и <BK> или CY/C для запрещения операции;

— любой другой ответ вызывает повторную печать сообщения.

Примечание. Хотя форматирование или проверка тома возможны, когда загружено основное задание, однако, делать этого не рекомендуется.

При попытке форматирования тома, который содержит защищенные файлы, система выдает сообщение:

VOLUME CONTAINS PROTECTED FILES;ARE YOU SURE?

Пользователю следует ввести:

— Y (или любую строку, начинающуюся с Y) и <BK> для выполнения форматирования;

— N (или любую строку, начинающуюся с N) и <BK> или CY/C для запрещения операции;

— любой другой ответ вызывает повторную печать сообщения.

После завершения форматирования следует использовать команду INITIALIZE, чтобы подготовить том для использования его операционной системой ФОДОС.

Таблица 14

ПЕРЕКЛЮЧАТЕЛИ ПРОГРАММЫ FORMAT

Переключатель	Функция
1	2
Отсутствует	Форматирует при отсутствии переключателя указанный том (для гибких дисков форматирование по умолчанию выполняется с двойной плотностью). В командной строке можно указывать также переключатели /Y и /W
/P:N	Задаёт эталон (эталон) проверки (проверок) (N — целое восьмеричное число от 0 до 177777, по умолчанию N=200)
/S	Форматирует дискеты с одинарной плотностью
/V:[ONL]	Выполняет на указанном томе форматирование, а затем проверку (если в командной строке указан /V) или только проверку на плохие блоки указанного тома (если в командной строке указан /V:ONL)
/W	Используется в том случае, если необходимо произвести замену томов для выполнения операции форматирования
/Y	Запрещает печать опрашивающего сообщения: ARE YOU SURE? — и обеспечивает немедленное выполнение операции

8.3.1. Форматирование по умолчанию

Для форматирования гибкого диска с двойной плотностью или кассетных дисков в командной строке следует указать только имя устройства. Можно использовать также переключатели /Y для запрещения печати опрашивающего сообщения и /W для замены тома на формируемый.

Примеры.

1. *DY1:

DY1:/FORMAT—ARE YOU SURE?Y
 ?FORMAT—I—FORMATTING COMPLETE

*

В данном примере выполняется форматирование гибкого диска с двойной плотностью на первом приводе устройства DY:.

2. *RK1:

RK1:/FORMAT—ARE YOU SURE?Y
 ?FORMAT—I—FORMATTING COMPLETE

*

В данном примере выполняется форматирование кассетного диска на первом приводе устройства RK:.

8.3.2. Переключатель/P:N позволяет задавать эталон для проверки тома и используется с переключателем /V:[ONL]. N — целое восьмеричное число от 0 до 177777, которое определяет один или несколько последовательных эталонов проверки для программы FORMAT. В табл. 15 приведены эталоны проверок, используемые программой FORMAT.

Таблица 15

Порядковый номер эталона	Номера разрядов, установленных в числе N (в скобках — число N)	Значение эталона
1	2	3
1	0 (1)	000000
2	1 (2)	177777
3	2 (4)	163126
4	3 (10)	125252
5	4 (20)	052525
6	5 (40)	007417
7	6 (100)	021042
8	7 (200)	104210
9	8 (400)	155555
10	9 (1000)	145454
11	10 (2000)	146314

Определение эталонов проверки выполняется следующим образом:

— аргумент N, указанный пользователем, преобразуется в двоичное число;

— номер каждого установленного разряда полученного двоичного числа определяет эталон для проверки.

Примеры.

1. N=25

Двоичное число, соответствующее данному N, равно 010101, т. е. установлены биты 0, 2, 4. Из табл. 15 видно, что:
0 биту соответствует 1 эталон проверки;
2 биту соответствует 3 эталон проверки;
4 биту соответствует 5 эталон проверки.

2. N=255

Двоичное число, соответствующее данному N, равно 010101101, т. е. установлены биты 0, 2, 3, 5, 7. Программа FORMAT выполняет проверки по эталонам 1, 3, 4, 6, 8.

3. N=377

Программа FORMAT выполняет восемь проверок по восьми эталонам.

Если аргумент N не указан, FORMAT выполняет одну проверку с помощью эталона 8.

Если пользователь по переключателю /P:N указал больше одного эталона, проверки выполняются последовательно. После завершения каждой проверки FORMAT печатает номера найденных плохих блоков. Сообщение о результате каждой проверки имеет следующий формат:

PATTERN #X
NNNNNN

где X — номер эталона проверки;
NNNNNN — номер найденного плохого блока.

Примеры.

1. *RK1:/V/P:70

RK1:/FORMAT—ARE YOU SURE?Y
?FORMAT—I—FORMATTING COMPLETE
PATTERN #6
PATTERN #5
PATTERN #4
?FORMAT—I—VERIFICATION COMPLETE

*

В данном примере выполняется форматирование кассетного диска и последующая его проверка по эталонам 4, 5 и 6.

2. *DY1:/V:ONL/P:2

DY1:/VERIFY—ARE YOU SURE?Y
PATTERN #2
?FORMAT—I—VERIFICATION COMPLETE

*

В данном примере выполняется только проверка гибкого диска по эталону 2 без форматирования.

8.3.3. Переключатель /S предназначен для форматирования гибкого диска с одинарной плотностью. Можно указать с /S также /Y для запрещения печати опрашивающего сообщения и /W для замены тома на формируемый.

Пример.

```
*DY1:/S
DY1:/FORMAT—ARE YOU SURE?Y
?FORMAT—I—FORMATTING COMPLETE
*
```

В данном примере формируется гибкий диск с одинарной плотностью на первом приводе устройства DY:.

8.3.4. Переключатель /V:ONL] предназначен для форматирования тома с последующей проверкой всех блоков. Если указан переключатель /V:ONL, программа FORMAT выполняет только проверку тома без форматирования. Программа FORMAT позволяет выполнять проверку как кассетных дисков, так и гибких дисков.

Проверка тома программой FORMAT выполняется следующим образом: сначала FORMAT записывает значение эталона в каждый блок указанного тома и затем его считывает. Если происходит ошибка чтения или записи, FORMAT печатает на терминале номер блока, в котором произошла ошибка.

Примеры:

```
1. *DY1:/V:ONL
DY1:/VERIFY—ARE YOU SURE?Y
PATTERN #8
?FORMAT—I—VERIFICATION COMPLETE
```

*

В данном примере выполняется только проверка гибкого диска на первом приводе устройства DY: по эталону 8 без форматирования.

```
2. *RK1:/V
RK1:/FORMAT—ARE YOU SURE?Y
?FORMAT—I—FORMATTING COMPLETE
PATTERN #8
?FORMAT—I—VERIFICATION COMPLETE
```

*

В данном примере выполняется форматирование кассетного диска и последующая проверка по эталону 8.

8.3.5. Переключатель /W используется при форматировании томов в том случае, если необходимо произвести замену томов для выполнения операции форматирования или отформатировать том на приводе устройства, на котором находится системный том. По переключателю /W операция форматирования инициируется, затем приостанавливается и печатается сообщение:

MOUNT INPUT VOLUME IN <УСТ:>; CONTINUE?

где уст — имя устройства, на котором будет форматироваться том.

Пользователю следует заменить том формируемым и ввести Y (или любую строку, начинающуюся с Y) и <BK> для выполнения форматирования. Для запрещения операции следует ввести N (или любую строку, начинающуюся с N) и <BK> или CY/C. Любой другой ответ вызывает повторную печать сообщения. После завершения форматирования на терминале печатается:

MOUNT SYSTEM VOLUME IN <УСТ:>; CONTINUE?

следует заменить отформатированный том системным и ввести Y (или любую строку, начинающуюся с Y) и <BK>

Переключатель /W можно использовать с любым другим переключателем.

Пример.

*DY1:/W/S

DY1:/FORMAT—ARE YOU SURE?Y

MOUNT INPUT VOLUME IN <УСТ:>; CONTINUE?Y

?FORMAT—I—FORMATTING COMPLETE

MOUNT SYSTEM VOLUME IN <УСТ:>; CONTINUE?Y

*

В данном примере выполняется замена тома на формируемый и форматирование тома с одинарной плотностью.

8.3.6. Переключатель /Y запрещает печать опрашивающего сообщения: ARE YOU SURE?

перед началом операции форматирования и обеспечивает немедленное выполнение операции.

Пример.

*DY1:/Y

?FORMAT—I—FORMATTING COMPLETE

*

В данном примере операция форматирования выполняется немедленно без печати предварительного опрашивающего сообщения.

8.4. Сообщения оператору

Ниже описаны сообщения, выдаваемые программой **FORMAT**.

?FORMAT—F—DEVICE ERROR

Причина. Произошла аппаратная ошибка при попытке форматирования или привод с указанным номером не существует.

Действие. Проверить наличие привода с указанным номером. Если имела место аппаратная ошибка, проверить исправность и готовность оборудования.

?FORMAT—F—DEVICE INVALID OR NOT SUPPORTED BY FORMAT

Причина. В командной строке указано недопустимое для программы **FORMAT** имя устройства.

Действие. Ввести командную строку, указав допустимое имя устройства.

?FORMAT—F—DEVICE NOT READY

Причина. Указанное в командной строке устройство не готово для форматирования или проверки, т. к. устройство выключено или запись на него запрещена; для кассетного диска — скорость вращения меньше допустимой.

Действие. Проверить готовность устройства к операции: устройство должно быть включено и запись на него разрешена; кассетный диск должен набрать нужную скорость.

?FORMAT—F—INSUFFICIENT MEMORY

Причина. Для выполнения указанной операции недостаточно памяти.

Действие. Следует увеличить область свободной памяти, удалив драйверы неиспользуемых устройств, а также основное и системные задания.

?FORMAT—F—INVALID COMMAND LINE

Причина. Недопустимый формат командной строки.

Действие. Следует ввести командную строку следующего формата: ***уст:[/прк]**.

?FORMAT—F—INVALID OPTION:/X

Причина. Переключатель, указанный в командной строке, недопустим для указанного устройства.

Действие. Вновь ввести командную строку и использовать только допустимые переключатели.

?FORMAT—F—INVALID OPTION /X:VAL

Причина. Для указанного переключателя в командной строке аргумент недопустим.

Действие. Ввести командную строку, указав переключатель без аргумента (см. табл. 14).

?FORMAT—F—INVALID VALUE SPECIFIED WITH
OPTION:/X

Причина. С переключателем /X указано недопустимое значение

Действие. Ввести командную строку, указав допустимое значение для переключателя.

?FORMAT—E—OPERATION NOT COMPLETED

Причина. Выполнение операции форматирования прекращено в ответ на печать пользователем N (или любой строки, начинающейся с N) или СУ/С после сообщения системы, вызванного переключателем /W.

Действие. Ответ пользователя на сообщение системы по переключателю /W должен начинаться с Y для выполнения операции.

?FORMAT—F—UNIT NUMBER MUST BE IN RANGE 0—7

Причина. В командной строке указан недопустимый номер привода для устройства.

Действие. Ввести командную строку, указав номер привода в диапазоне от 0 до 7.

?FORMAT—F—UNIT NUMBER TOO LARGE

Причина. В командной строке указан номер привода, превышающий допустимый.

Действие. Проверить номер привода, указанный в командной строке, и вновь ввести командную строку.

?FORMAT—I—FORMATTING ABORTED

Причина. Форматирование прервано из-за ошибки устройства.

Действие. Не требуется.

?FORMAT—I—FORMATTING COMPLETE

Причина. Форматирование завершено.

Действие. Не требуется.

?FORMAT—I—VERIFICATION ABORTED

Причина. Проверка прервана из-за ошибки устройства или по желанию пользователя.

Действие. Если необходимо, повторить операцию проверки. Если имела место ошибка устройства, проверить исправность и готовность оборудования.

?FORMAT—I—VERIFICATION COMPLETE

Причина. Программа FORMAT закончила проверку.

Действие. Не требуется.

?FORMAT—U—DEVICE HANDLER FETCH ERROR

Причина. Драйвер устройства не может быть считан с периферийного устройства в основную память.

Действие. Проверить наличие драйвера устройства на системном томе; проверить системный том на наличие плохих блоков. Вновь повторить операцию проверки устройства.

?FORMAT—U—FORMATTING/VERIFYING THE SYSTEM VOLUME IS NOT ALLOWED

Причина. Попытка форматирования или проверки системного тома.

Действие. Указать в командной строке другой привод устройства (отличный от привода системного тома) или использовать переключатель /W (или команду FORMAT/WAIT см. [2]) для замены системного тома томом, который требуется отформатировать.

?FORMAT—U—TOO MANY BAD BLOCKS DEV:

Причина. Число плохих блоков превысило допустимое значение.

Действие. Использовать другой том.

?FORMAT—W—DUPLICATE OPTION:/X

Причина. Переключатель /X дважды указан в командной строке.

Действие. Проверить переключатели в командной строке и вновь ввести командную строку.

9. ПОЛУЧЕНИЕ ИНФОРМАЦИИ О СОСТОЯНИИ СИСТЕМЫ

9.1. Назначение программы и условия выполнения программы

Программа RESORC предназначена для выдачи информации о ресурсах системы. При использовании программы RESORC пользователь может получить следующие сведения о системе:

- номер версии монитора;
- варианты команды SET;
- аппаратную конфигурацию системы;
- размер памяти системы;
- организацию физической памяти;
- задания, загруженные в данный момент;
- сгенерированные особенности системы;
- назначение устройств;

— состояние активных терминалов (в мультитерминальных системах);

— состояние драйвера устройства;

— набор логических дисков.

9.2. Выполнение программы

Вызов программы RESORC с системного устройства осуществляется по команде:

R RESORC <BK> или RESORC <BK>

после того, как монитор напечатает на терминале точку.

После вызова программа RESORC печатает звездочку и ожидает ввода командной строки. Если в это время подать возврат каретки, то программа напечатает номер своей версии.

9.2.1. Команды оператора

Режим работы программы RESORC задается введением с терминала командной строки следующего формата:

[уст:]/прк,

где уст — имя устройства;

прк — переключатель.

Если в командной строке используется два или более переключателя, то программа RESORC выводит информацию по мере выполнения переключателей в следующей последовательности: /M, /C, /H, /O, /L, /J, /T, /X, /S независимо от того, в каком порядке они были записаны в командной строке.

Таблица 16

ПЕРЕКЛЮЧАТЕЛИ ПРОГРАММЫ RESORC

Переключатель	Функция
1	2
/A	Результат комбинации всех переключателей программы RESORC
/C	Выдает информацию об устройстве, с которого загружена система, и вариантах команды SET
[уст:]/D	Выдает список драйверов устройств, их состояние, адреса векторов и регистров состояний или информацию только об указанном устройстве

1	2
/H	Выдает информацию об аппаратной конфигурации системы, включая весь объем памяти (в байтах)
/J	Выдает информацию о работающих в данное время и загруженных заданиях
/L	Выдает информацию о назначениях устройств
/M	Сообщ. тип монитора и номер версии.
/O	Выдает информацию об особенностях системной генерации
/S	Выдает информацию о наборе логических дисков
/T	Выдает информацию о состоянии активных терминалов
/X	Выдает информацию об организации физической памяти
/Z	Результат комбинации переключателей /M, /C, /H, /J, /O

Для выхода из программы RESORC и передачи управления монитору следует подать команду CY/C, если RESORC ожидает ввода командной строки или дважды CY/C, если RESORC выполняет операцию.

9.2.2. Переключатель /A является комбинацией следующих переключателей программы RESORC: /M, /C, /H, /O, /D, /J, /T, /X, /S, которые описаны в последующих разделах. По переключателю /A на терминал выводится следующая информация в ниже приведенном порядке:

— конфигурация монитора (т. е. тип монитора и номер версии, устройство, с которого загружен монитор, варианты команды SET, загружено задание или нет, глубина вложения косвенного файла);

- аппаратная конфигурация;
- особенности системной генерации;
- назначение устройств;
- состояние задания;
- состояние активных терминалов;
- организация физической памяти;

— набор логических дисков, выделенных из физических дисков.

9.2.3. Переключатель /C. По переключателю /C выдается следующая информация:

- устройство, с которого загружена система;
- варианты команды SET;
- загружено основное задание или нет;
- глубина вложения косвенного файла.

Пример.

***/C**

загрузка с RK1:FMONXM

основное задание загружено

команда: SET USR SWAP

команда: SET EXIT SWAP

команда: SET KMON IND

команда: SET TT QUIET

команда: SET ERROR ERROR

глубина вложения 3

9.2.4. Переключатель /D. По переключателю /D выводится список драйверов системных устройств со следующей информацией о них:

- состояние;
- адреса регистров состояний;
- адреса векторов.

По переключателю /D распечатывается информация только о тех драйверах, чьи особенности соответствуют загруженному в настоящий момент монитору. Если в командной строке указано имя устройства, то программа RESORC выдает информацию только об этом устройстве.

Пример.

***/D**

Устройство	Состояние	PC	Вектор
LD	установлен	000000	000
LP	не установлен	177514	200
PC	не установлен	177550	070 074
RK	резидентный	177400	220
MT	установлен	172520	224
NL	установлен	000000	000
DX	не установлен	177170	264
BA	установлен	000000	000
DY	122620	177170	264
VM	не установлен	177572	000
SL	установлен	000000	000

В этом примере в колонке «Состояние» может быть напечатано одно из сообщений, приведенных в табл. 17.

Таблица 17

Состояние	Значение
1	2
Установлен	Драйвер устройства, имеющийся в системных таблицах, не загружен в основную память (его можно загрузить по команде LOAD)
Не установлен	В системных таблицах монитора данного драйвера нет, его можно установить по команде INSTALL (если в таблицах монитора есть свободное место)
Не установлен	Драйвер устройства не соответствует существующему монитору. Этот драйвер нельзя установить, о чем свидетельствует символ минус перед сообщением
Резидентный	Драйвер устройства постоянно находится в памяти. Его нельзя удалить или загрузить
NNNNNN	Начальный адрес загруженного драйвера

В последней колонке указываются адреса векторов, если драйвер многовекторный, то адреса дополнительных векторов печатаются в этой же колонке.

9.2.5. Переключатель /Н. При использовании переключателя /Н программа RESORC выдает тип процессора, общий объем памяти системы (в байтах) и особенности аппаратной конфигурации системы. В системе используется один из следующих процессоров:

- ЭЛЕКТРОНИКА 60;
- ЭЛЕКТРОНИКА 100—25;
- ЭЛЕКТРОНИКА 79;
- МС 1601;
- МС 1601.01.

Аппаратные особенности системы выбираются из следующего списка и воспроизводятся в порядке, указанном в этом списке:

- процессор плавающей запятой;
- расширенная арифметика (РА);
- плавающая запятая (ПЗ);
- диспетчер памяти
- проверка четности памяти;
- кэш памяти

Пример:

*/Н

процессор ЭЛЕКТРОНИКА 100—25

128КВ памяти

расширенная арифметика (РА)

плавающая запятая (ПЗ)

диспетчер памяти

проверка четности памяти

9.2.6. Переключатель /J. По переключателю /J на терминал выводится таблица со следующей информацией о загруженных заданиях:

- номер задания;
- имя задания (если запрещена поддержка системного задания в мониторе, то основное задание появляется с именем FORE и его приоритет равен единице при работе с монитором основного — фоновго задания (FB) или монитором расширенной памяти (XM);
- терминал, на котором работает задание (для немультитерминальных систем — 0);
- приоритет задания;
- состояние задания (завершено, приостановлено, работает);
- нижний и верхний пределы памяти для этого задания;
- начальный адрес смешанной области задания.

9.2.7. Переключатель /L. По переключателю /L программа RESORC выдает информацию о назначениях системных устройств, имеющих в системных таблицах. Листинг включает также дополнительную информацию об особенностях устройств. В табл. 18 приведены сообщения, которые могут быть выданы по переключателю /L.

Таблица 18

Сообщение	Значение
1	2
(RESORC) или=RESORC	Устройство или привод, назначенный фоновому заданию RESORC (для мониторов FB или XM)
(F) или=F	Устройство или привод, назначенный основному заданию (для мониторов FB или XM, которые не поддерживают системные задания)
(имя задания) или= имя задания	Устройство или привод, назначенный системному или основному заданию, (только для мониторов FB или XM, которые поддерживают системные задания), где имя задания является именем основного или системного задания
загружен	Драйвер устройства загружен в память по команде LOAD
резидентный	Драйвер устройства включен в резидентный монитор
1) =лог.имя (1) =лог.имя (2) =лог.имя (3) XX свободных мест	Устройству или приводу назначено логическое имя по команде ASSIGN В последней строке указывается число свободных мест в таблицах монитора

1) лог. имя — логическое имя устройства.

Пример.

*/L

TT резидентный

RK резидентный
RKI=DK, SY

MQ резидентный

LD

SL
MT
BA
NL

8 свободных мест

Этот пример показывает состояние всех устройств, имеющих в системе, при загруженном мониторе FB.

9.2.9. Переключатель /0. По переключателю /0 на терминал выводится тип и номер версии работающего монитора.

Пример.

*/M

ФОДОС Ф В03.00

9.2.9. Переключатель /O. По переключателю /O на терминал выводятся особенности системы, выбранные во время генерации системы. Особенности, действующие в данное время, выводятся на терминал в том порядке, в котором они приведены в табл. 19.

Таблица 19

Особенность	Назначение
1	2
Поддержка ожидания готовности устройства	Разрешает драйверам опрашивать готовность устройства через равные промежутки времени
Поддержка регистрации ошибок	Хранит статистическую запись о всех операциях ввода/вывода на устройствах, которые поддерживают эту особенность, обнаруживает и хранит данные о всех ошибках, которые имеют место в течении операций ввода-вывода
Мультитерминальная поддержка	Разрешает использовать до восьми терминалов
Поддержка четности памяти	Система выдает сообщение об ошибке, когда встречается ошибка четности памяти
Поддержка таймера монитором SJ	Поддерживает программные запросы, использующие таймер
Поддержка системного задания	Разрешает работу с шестью системными заданиями в дополнение к основному и фоновому заданиям

Пример.

***/O**

поддержка ожидания готовности устройства

поддержка регистрации ошибок

мультитерминальная поддержка

поддержка четности памяти

поддержка системного задания

9.2.10. Переключатель /S. По переключателю /S выдается сообщение о наборе логических дисков, выделенных из физических дисков. При использовании переключателя /S программа RESORC выдает имя файла, которое назначено логическому диску и размер логического диска в десятичных блоках.

В следующем примере показано, что логические диски выделены из физических дисков RK0: и RK1:.

Пример.

***/S**

LD0: — RK0:L551.DSK[600.]

LD1: — RK1:DISK.SEC[1200.]*

Символ звездочка (*) означает, что логический диск назначен, но файл не существует на томе, который установлен в текущий момент. Символ «#» указывает, что драйвер устройства не загружен. В основном символы «#» или «*» выдаются после использования команд SET LD CLEAN или LD/C для определения состояния значений логического диска.

9.2.11. Переключатель /T. По переключателю /T выдается информация об активных в данный момент терминалах. Если в системе нет мультитерминальной поддержки, RESORC выдает:

НЕТ МУЛЬТИТЕРМИНАЛЬНОЙ ПОДДЕРЖКИ

Если в систему включена мультитерминальная поддержка, то по переключателю /T на терминал выводится таблица. В первой графе таблицы печатается номер терминала. Можно использовать в системе до шестнадцати терминалов. В следующей графе выдаются имена заданий (основного, фонового или системного), для которых назначен терминал. Если системное задание не поддерживается, то в графе с именем «Владелец» появляется имя системного задания FORE или RESORC в зависимости от применения.

В графе с именем «Тип» указывается тип терминала: локальный, удаленный, консоль, S-консоль (консоль, совместно используемая основным, системным и фоновым заданиями).

В графе с именем «Длина» указывается количество символов (до 255) в строке.

В следующих четырех графах указаны варианты команды SET, находящиеся в действии в данное время. Если установлен TAB, то автоматически выполняется табуляция. Если установлен SRLF, то происходит автоматический перевод строки, всякий раз когда пользователь пытается печатать за пределами правого поля. При установлении варианта FORM возможен автоматический перевод формата. На графических терминалах возможно отражение символов «ЗБ» посредством операции шаг назад — пробел — шаг назад.

В последней графе выдается скорость печати в бодах, N/A — если постоянна.

9.2.12. Переключатель /X. По переключателю /X выводится информация об организации физической памяти. Пользователь получает информацию о том, где находятся загруженные задания KMON, USR, а также количество слов, занимаемых этими программами. Адреса памяти даны в восьмеричной системе.

Пример.

1. */X

Расширенная память		
Адрес	Модуль	Слова
00166100		35296.
00160000	SL	1568.

Ядро памяти		
Адрес	Модуль	Слова
160000	IOPAGE	4096.
157172	RK	195.
117044	RMON	8235.
106044	USR	2304.
001000	. .BG. .	17682.

9.2.13. Переключатель /Z имеет то же действие, что и комбинация переключателей /M, /C, /H, /J, /O.

Пример.

*/Z

ФОДОС Ф В03.00

Загрузка с RK1:FMONXM

основное задание загружено

команда: SET USR SWAP
команда: SET EXIT SWAP
команда: SET KMON NOIND
команда: SET TT NOQUIET
команда: SET ERROR ERROR
команда: SET SL OFF
команда: SET EDIT EDIT

глубина вложения 3

процессор ЭЛЕКТРОНИКА 100—25

128 КВ памяти

расширенная арифметика (РА)

плавающая запятая (ПЗ)

диспетчер памяти

проверка четности памяти

поддержка ожидания готовности устройства

поддержка четности памяти

поддержка таймера монитором SJ

9.3. Сообщения оператору

Ниже приведены сообщения, выдаваемые программой RESORC.

?RESORC—F—ERROR READING DIRECTORY

Причина. Ошибка чтения справочника системного устройства.

Действие. Проверить готовность и исправность оборудования. Проверить системный том на плохие блоки, используя команды DIRECTORY/BAD BLOCKS или DUP/K.

?RESORC—F—INPUT ERROR FILNAM.TYP

Причина. Ошибка чтения указанного файла.

Действие. Проверить системный том на плохие блоки.

?RESORC—F—INSUFFICIENT MEMORY

Причина. Недостаточно памяти для работы программы RESORC.

Действие. Разгрузить основное или системное задания и неиспользуемые в данный момент драйверы.

?RESORC—F—INSUFFICIENT MEMORY TO FETCH HANDLER

Причина. Недостаточно места для размещения драйвера в основной памяти.

Действие. Разгрузить основное или системное задания и неиспользуемые в данный момент драйверы.

?RESORC—F—INVALID OPTION

Причина. Переключатель, указанный в командной строке, не является переключателем RESORC.

Действие. Исправить и вновь ввести командную строку.

?RESORC—F—NO AUDIT INFORMATION AVAILAB

Причина. Сбой в работе операционной системы. Возможно заперчен текст RESORC.

Действие. Повторить операцию. При появлении той же ошибки получить новую копию программы RESORC.

?RESORC—F—SYSTEM ERROR

Причина. Сбой при работе запроса .DASTATUS или .LOOKUP. Либо устройство не готово к работе, либо монитор, загруженный в память, заперчен.

Действие. Проверить исправность и готовность оборудования, перезагрузить монитор.

?RESORC—F—WRONG VERSION

Причина. Попытка обращения системы ФОДОС к программе RESORC, принадлежащей системе ФОДОС-2.

Действие. Использовать программу RESORC с системой ФОДОС-2.

Перечень ссылочных документов

1. Операционная система ФОДОС-2
Описание применения
2. Операционная система ФОДОС-2
Командный язык системы
3. Операционная система ФОДОС-2
Редактор связей
Руководство оператора
4. Операционная система ФОДОС-2
Программы работы с файлами
Руководство оператора
5. Операционная система ФОДОС-2
Редактор текста
Руководство оператора
6. Операционная система ФОДОС-2
Отладчик и виртуальный отладчик
Руководство оператора

ПРОГРАММЫ РАБОТЫ С ФАЙЛАМИ РУКОВОДСТВО ОПЕРАТОРА

1. ПРОГРАММА ОБМЕНА РІР

1.1. Назначение программы и условия выполнения программы.

Программа обмена (RIP) предназначена для копирования файлов между любыми периферийными устройствами операционной системы ФОДОС-2, а также для объединения, удаления, переименования и изменения состояния защиты файлов.

1.2. Выполнение программы

Для вызова RIP с системного устройства следует подать с терминала команду:

R RIP <ВК>

после того, как монитор напечатает на терминале точку. После вызова RIP печатает звездочку и ожидает ввода командной строки. Если в это время нажать клавишу <ВК>, то RIP печатает номер своей версии.

1.3. Команды оператора

Режим работы программы RIP задается введением с терминала командной строки. В командной строке может быть указано до шести имен входных файлов и одно имя выходного файла. Переключатели в командной строке могут следовать за спецификацией любого файла, но обычно их указывают в конце командной строки. Операции с магнитной лентой являются исключением из этого правила, т. к. переключатель /M имеет различные значения, в зависимости от того, будет ли он указан в выходной или во входной спецификациях командной строки.

Некоторые переключатели программы RIP допускают в качестве аргумента дату.

Синтаксис, определяющий дату, следующий:

[[:дд.][:ммм]][:гг.]

где дд. — день (десятичное число от 1 до 31);

ммм — месяц (первые три буквы названия месяца на английском языке);

гг. — год (десятичное число от 73 до 99).

Датой по умолчанию является текущая дата. Если пренебречь каким-либо элементом даты (дд, ммм или гг), система использует элемент из текущей даты. Например, если указать

только год, : 82., и текущая дата 4 мая 1983 года, то система будет использовать 4.:MAY:82.. Если текущая дата не установлена, то ее значение считается нулевым (то же самое для файлов с отсутствующей датой создания в листинге справочника).

В устройствах с произвольным доступом к данным (например, диск) и при копировании с магнитной ленты операции РІР сохраняют дату создания файла. Если дата создания файла отсутствует — РІР присваивает ему текущую дату. Однако при копировании на магнитную ленту РІР всегда присваивает файлам текущую дату.

Если сгенерированная система поддерживает таймер, но не поддерживает автоматического изменения даты при смене месяца, то установить дату начала каждого месяца можно с помощью команды монитора DATE. Если начало месяца в дате не установлено, система печатает «—BAD—» в колонке даты создания файла, созданного позже конца месяца. (Установить текущую дату создания файла можно по команде монитора RENAME/SETDATE).

В командной строке для спецификаций файлов могут быть использованы конструкции «*» и «%». Выходные спецификации файлов не могут содержать встроенную конструкцию «*», если входные спецификации ее не содержат. Если в командной строке в спецификации входного файла используется конструкция «*», соответствующее имя или тип выходного файла также должны быть определены через «*». Операция по переключателю /U является исключением из этого правила, т. к. не допускает использование конструкции «*» в выходной спецификации файла. Конструкцию «*» нельзя использовать для обращения к файлам с типом .BAD (*.BAD). В этом случае в командной строке необходимо точно определить имя и тип файла.

Для выхода из программы РІР и передачи управления монитору следует подать команду СУ/С, если РІР ожидает ввода с терминала, или дважды СУ/С, если РІР выполняет операцию.

Для повторного пуска РІР следует подать команду REENTER.

1.4. Переключатели программы РІР (табл. 1) позволяют выполнять самые различные операции на периферийных устройствах операционной системы. Если в командной строке переключатель не указан, РІР копирует файл в режиме ото-

бражения памяти. Необходимо помнить, что при таком копировании нельзя обеспечить правильную передачу файла на перфоленту, с перфоленты, на ТТ: или на LP:.

Таблица 1

Переключатель	Функция
1	2
/A	Копирует файлы в кодах КОИ-7, игнорируя нули и забори. Воспринимает СУ/Z (код 32) как логический конец входного файла
/B	Копирует файлы объектного и абсолютного форматов
/C[:дата]	Выполняет операцию на файлах с указанной датой создания. Если используется /C и не указана дата, PIP выполняет операции на файлах с текущей датой создания
/D	Удаляет файлы из справочника указанного тома. Недопустим для магнитных лент
/E	Используется для запуска и остановки операции копирования. Это полезно в небольших, однодисковых, системах, т. к. позволяет заменить системный том другим томом перед выполнением операции
/F	Защищает файлы от удаления. Недопустим для магнитных лент и кассет
/G	Игнорирует входные ошибки во время операции копирования
/H	Сравнивает выходные файлы с соответствующими входными файлами после операции копирования. Недопустимо использование с переключателями /A или /B
/I[:дата]	Выполняет операции с файлами, созданными во время или после указанной даты

1	2
/J[:дата]	Выполняет операции с файлами, созданными раньше указанной даты
/K:N	Делает N копий выходного файла на LP:, TT: или PC:
/M:N	Используется в операциях с магнитной лентой
/N	Не копирует или не переименовывает файл, если на устройстве вывода уже существует файл с тем же именем
/O	Удаляет файл на устройстве вывода, если его имя совпадает с именем выходного файла командной строки. Операция удаления происходит перед операцией копирования
/P	Копирует или удаляет все файлы, кроме указанных в командной строке
/Q	Опрашивает пользователя о включении в указанную операцию каждого файла. Используется только с другими переключателями
/R	Переименовывает указанный файл. Недопустим для магнитных лент
/S	Копирует файлы по одному блоку в один прием
/T[:дата]	Приписывает указанную дату всем файлам, включенным в операции копирования или переименования. Недопустим при копировании на магнитную ленту (операции, включающие устройства магнитной ленты, всегда используют текущую дату)
/U	Копирует и объединяет указанные файлы
/V	Копирует файлы с входного тома на 2 или более меньших по размеру выходных тома

1	2
/W	Печатает на терминале список файлов, включенных в операции копирования, удаления или переименования
/X	Печатает информационное сообщение взамен ошибочного сообщения, если указанный файл не найден
/Y	Включает в указанную операцию файлы с типом .SYS. Нельзя изменить или удалить эти файлы без /Y, если используется конструкция «*» вместо типа входного файла
/Z	Удаляет состояние защиты выходного файла. Недопустим для магнитных лент

Копирование в режиме отображения — копирование данных без изменения их структуры, «слово в слово».

1.4.1. Переключатель /A используется для копирования файлов в кодах КОИ-7. При этом PIP игнорирует нули и забори, а CY/Z (код 32) воспринимает как логический конец входного файла.

Может использоваться с переключателем /V.

Пример:

*DX1:M2.MAC=M1.MAC/A

в этом примере PIP копирует файл M1.MAC с DK: на DX1: и присваивает ему имя M2.MAC.

1.4.2. Переключатель /B используется для копирования файлов объектного и абсолютного форматов. Во время выполнения операции PIP производит проверку на совпадение контрольной суммы и печатает сообщение, если произошла ошибка. По переключателю /B нельзя копировать библиотечные файлы. Для этой цели необходимо воспользоваться копированием в режиме отображения памяти.

Может использоваться с переключателем /V.

Пример:

*DK:FILE.OBJ=PC:/B

в этом примере PIP копирует файл объектного формата с PC: на DK: и присваивает ему имя FILE.OBJ.

1.4.3. Переключатель /C[:дата] выполняет операцию на файлах только с указанной датой создания. Если дата не указана, используются файлы только с текущей датой создания. В командной строке он указывается один раз и будет относиться ко всем спецификациям файлов. Может быть использован с другим переключателем.

Пример:

*DY1:NN3.MAC=DX0:*MAC/C:12.:JAN:83., DY0:RDWR.MAC/A/U

В этом примере PIP копирует (в кодах КОИ-7) все файлы с типом .MAC с DX0:, созданные 12 января 1983 г., и копируется файл RDWR.MAC, если он имеет дату создания 12 января 1983 г., с DY0: на DY1:. Их комбинацией является файл с именем NN3.MAC на DY1:.

1.4.4. Переключатель /D используется для удаления файлов из справочника указанного тома. В командной строке должна быть определена спецификация файла; командная строка, включающая только имя устройства, уст:/D, — недопустима. Переключатель /D нельзя использовать в операциях с магнитной лентой.

Пример:

* *.MAC/D

В этом примере PIP удаляет из справочника все файлы с типом .MAC, находящиеся на DK:.

1.4.5. Переключатель /E рекомендуется использовать при копировании файлов в том случае, когда необходимо произвести замену томов во время операции копирования. Переключатель /E может использоваться с любым переключателем PIP. Запрещается использовать конструкцию «*» в спецификации входного файла.

Формат командной строки:

*выходспф=входспф/E

где выходспф — представляет собой спецификацию выходного файла (устройство, имя и тип файла);

входспф — представляет собой спецификацию входного файла (устройство, имя и тип файла).

Операция копирования с использованием переключателя /E выполняется в следующем порядке:

1) PIP печатает сообщение:

MOUNT INPUT VOLUME IN <УСТ>;CONTINUE?

и ожидает ответа пользователя.

В это время входной том, с которого будет выполняться копирование, должен быть установлен на устройство, указан-

ное во входной спецификации командной строки. Для продолжения операции необходимо напечатать «Y» и нажать клавишу <BK>. Если напечатать «N» или нажать CY/C, операция прервется. Будет напечатано сообщение:

MOUNT SYSTEM VOLUME IN <UCT>;CONTINUE?

Необходимо установить системный том на привод устройства, с которого была загружена система, напечатать на терминале «Y» и нажать клавишу <BK>.

Любой другой ответ приводит к повтору сообщения:

MOUNT INPUT VOLUME IN <UCT>;CONTINUE?

2) После печати программой PIP очередного сообщения:

MOUNT OUTPUT VOLUME IN <UCT>;CONTINUE?

Необходимо установить выходной том на устройство, указанное в выходной спецификации командной строки, напечатать «Y» и нажать клавишу <BK> для продолжения операции копирования. Если напечатать «N» или нажать CY/C, операция прервется. Будет напечатано сообщение:

MOUNT SYSTEM VOLUME IN <UCT>;CONTINUE?

Необходимо установить системный том на привод устройства, с которого была загружена система, напечатать на терминале «Y» и нажать клавишу <BK>.

3) По окончании копирования файлов PIP печатает:

MOUNT SYSTEM VOLUME IN <UCT>;CONTINUE?

Необходимо установить системный том на привод устройства, с которого была загружена система, напечатать на терминале «Y» и нажать клавишу <BK> для завершения операции копирования.

Примечание. Копирование файла с использованием одного привода устройства (ВЗУ) выполняется в несколько приемов, что влечет неоднократное повторение действий, описанных выше.

1.4.6. Переключатель /F используется для защиты файлов от удаления. Указанные в командной строке файлы становятся защищенными файлами в справочнике тома, на котором эти файлы находятся. Буква «P», следующая за размером файла в листинге справочника тома, служит признаком защищенного файла.

К защищенному файлу неприменимы операции, результатом которых является его удаление. Нельзя отменить состояние защиты, если не использовать переключатель /Z.

Переключатель /F может использоваться с переключателем /R.

Если переключатели /F или /Z не используются в командной строке, то после завершения операции копирования выходной файл сохраняет состояние защиты входного файла.

Примеры:

1. *** *.TXT/F**

В этом примере все файлы с типом .TXT становятся защищенными на DK:.

2. ***DY0:* .BAK=DY0:* .MAC/F/R**

В этом примере все файлы с типом .MAC с DY0: переименовываются в защищенные файлы с типом .BAK.

3. ***DY1:RAS.SAV/F/T:11.:AUG:83.**

В этом примере файл RAS.SAV, созданный на DY1: 11 августа 1983 года, будет защищен; если он был защищен еще раньше, то операция прервется, и будет напечатано сообщение об ошибке.

1.4.7. Переключатель /G игнорирует входные ошибки во время операции копирования. Его использование обеспечивает копирование файла по одному блоку в один прием (подобно /S), что повышает возможность безошибочной передачи данных. Если, тем не менее, ошибка произойдет, PIP напечатает сообщение:

?PIP—W—INPUT ERROR

и продолжит копирование.

Пример:

***ABC.SAV=DX1:TOP.SAV/G**

В этом примере PIP копирует файл TOP.SAV с DX1: на DK: и присваивает ему имя ABC.SAV. Ошибки при копировании игнорируются.

1.4.8. Переключатель /H используется для сравнения выходных файлов с соответствующими входными файлами после операции копирования. Если два файла отличны, на термине печатается соответствующее сообщение.

Этот переключатель не может быть использован с переключателями /A и /B.

Пример:

***DY1:A.BAK=DY0:A.MAC/H**

В этом примере сравнивается выходной файл A.BAK на DY1: с входным файлом A.MAC на DY0:.

1.4.9. Переключатель /I[:дата] включает в операции только файлы, созданные во время или после указанной даты. Если дата не указана, PIP использует текущую дату.

Пример:

***DY0:* .MAC=* .MAC/I:4.:JAN:83.**

В этом примере копируются с DK: на DY0: файлы с типом .MAC, созданные 4 января 1983 г. и позже.

1.4.10. Переключатель /J[:дата] включает в операции только файлы, созданные до указанной даты. Если дата не указана, PIP использует текущую дату.

Пример:

*DY0:*MAC=*MAC/J:14.:JAN:83.

В этом примере копируются с DK: на DY0: файлы с типом .MAC, созданные до 14 января 1983 г.

1.4.11. Переключатель /K:N используется для получения N копий указанного файла. Допустимыми устройствами вывода являются перфоратор, терминал и построчно-печатающее устройство. Копии разделяются переводом формата.

Пример:

*LP:=FORM.LST/K:3

В этом примере PIP печатает три копии файла FORM.LST на построчно-печатающем устройстве.

1.4.12. Переключатель /M:N. Используется в операциях с магнитной лентой. Выполняет различные функции в командной строке для входных и выходных спецификаций. Применяется на том, а не на файл, поэтому один /M:N может быть определен для входного файла, а другой — для выходного.

Для операций чтения магнитной ленты (копирования с ленты) использование переключателя /M:N вызывает следующие процедуры:

1) N=0

лента перематывается на начало и PIP осуществляет поиск указанного файла. Если в командной строке определено несколько файлов, PIP перематывает ленту перед поиском каждого файла. Если спецификация файла содержит конструкцию «*», лента перематывается один раз.

2) N>0

PIP производит поиск файла с последовательным номером N (N-го файла от начала ленты). Если найденный файл тот, который указан в командной строке, PIP копирует его. Иначе печатает сообщение:

?PIP—F—FILE NOT FOUND DEV:FILNAM.TYP

если спецификация файла содержит конструкцию «*», PIP производит поиск файла с последовательным номером N, а затем начинает поиск всех файлов, соответствующих операции «*».

3) N=-1

PIP начинает поиск с текущего положения магнитной ленты.

Для операций записи на магнитную ленту (копирование на ленту) использование переключателя /M:N вызывает следующие процедуры:

1) $N=0$

PIP перематывает ленту перед копированием каждого файла. Если на ленте уже существует файл с тем же именем, что и выходной файл, PIP печатает предупреждающее сообщение: ?PIP—W—OUTPUT FILE FOUND, NO OPERATION PERFORMED DEV:FILNAM.TYP и не будет выполнять операцию.

2) $N>0$

PIP производит поиск файла с последовательным номером N и записывает на его место указанный файл. Если перед тем как найти N-ый файл PIP обнаружит метку LEOT (логический конец ленты), она печатает сообщение:

?PIP—F—FILE SEQUENCE NUMBER NOT FOUND

и файл копировать не будет. Если в командной строке определено несколько файлов или используется конструкция «*», PIP не перематывает ленту перед записью каждого файла и не производит проверку на дублирование имен файлов.

3) $N=-1$

PIP производит поиск LEOT и записывает на ее место указанный файл.

4) $N=-2$

PIP перематывает ленту перед каждой операцией копирования и записывает файл на место первого встретившегося файла с тем же именем или на место LEOT.

Если перед завершением операции копирования PIP обнаружит маркер физического конца ленты, она сотрет неполный файл путем обратной перемотки по зонам и записи метки LEOT перед меткой заголовка файла.

Если в процессе любой операции вывода на магнитную ленту дважды подана команда SU/C, PIP не запишет метку LEOT в конце данных. Следовательно, на эту магнитную ленту уже невозможно будет записать информацию, если не использовать одну из двух следующих процедур восстановления:

1) записать все файлы прерванной ленты на другой том и инициализировать ее:

уст:.*=MT:*. *
^C

.R DUP
*MT:/Z/Y

где уст: — вспомогательный том устройства.

2) Определить последовательный номер прерванного файла и использовать переключатель /M:N для записи на его место фиктивного файла. PIP запишет фиктивный файл и после него метку LEOT. В следующем примере предполагается, что плохой файл — четвертый на ленте:

*MT:FILE.NEW/M:4=FILE.DUM

Перед выполнением процедуры восстановления драйвер магнитной ленты необходимо удалить из памяти по команде монитора UNLOAD, если до этого он был загружен по команде LOAD.

Примечания:

1. Если переключатель /M:N не указан в командной строке, PIP перематывает ленту между каждой операцией.

2. В связи с последовательным доступом к данным на магнитной ленте обеспечивается доступ одновременно только к одному файлу на каждом приводе устройства. Командная строка, в которой входные и выходные файлы определены на одном и том же томе магнитной ленты, недопустима.

1.4.13. Переключатель /N запрещает выполнение операции копирования или переименования, если файл с тем же именем, что и выходной файл, уже существует на устройстве вывода. Этот переключатель недопустим в операциях с магнитной лентой.

Пример:

*DX1:DIN.MAC=DK:DIN.MAC/N
?PIP—W—OUTPUT FILE FOUND, NO OPERATION PERFORMED DK:DIN.MAC

Файл DIN.MAC уже существует на DX1:, PIP не будет выполнять операцию копирования.

1.4.14. Переключатель /O. По переключателю /O PIP удаляет файл из справочника выходного устройства, если его имя совпадает с именем выходного файла. Операция удаления происходит перед операцией копирования. Обычно PIP удаляет файл с тем же именем после операции копирования.

Переключатель /O недопустим в операциях с магнитной лентой.

Пример:

*DX1:TEST.MAC=TEST.MAC/O

Если файл TEST.MAC уже существует на DX1:, PIP удалит его перед выполнением копирования файла TEST.MAC с DK: на DX1:.

1.4.15. Переключатель /P. По переключателю /P PIP копирует или удаляет все файлы, кроме указанных в командной строке. Заметим, что, если необходимо включить системные файлы (с типом .SYS), когда используется /P, нужно использовать также и переключатель /Y.

Пример:

DX1:.*=*.MAC/P

В этом примере PIP копирует с DK: на DX1: все файлы, кроме файлов с типом .MAC. Файлы с типом .SYS исключены из операции, потому что не указан переключатель /Y.

1.4.16. Переключатель /Q. По переключателю /Q PIP печатает вопросы пользователю и ждет ответа. Печать «Y» и нажатие клавиши <BK> вызывает включение файла в указанную операцию, печать другого знака или просто нажатие клавиши <BK> исключает файл из операции.

Пример:

DX1:.*/D/Q
FILES DELETED:
DX1:CAR.SAV?
DX1:MELV.SAV?Y
DX1:COPY.SAV?

*

PIP удаляет файл MELV.SAV с DX1:.

1.4.17. Переключатель /R используется для переименования файлов. После выполнения этой операции файл, определенный во входной спецификации, будет иметь имя, указанное в выходной спецификации. В командной строке должно быть одинаковое число входных и выходных файлов, причем они должны находиться на одном и том же томе устройства. /R недопустим в операциях с магнитной лентой.

Пример:

*DX1:SAM.MAC=DX1:PLK21.MAC/R

В этом примере PIP переименовывает файл PLK21.MAC на DX1: и присваивает ему новое имя SAM.MAC.

1.4.18. Переключатель /S. По переключателю /S PIP осуществляет копирование файлов по одному блоку в один прием. Это повышает возможность безошибочной передачи данных. Переключатель /S может быть использован с другими переключателями программы PIP.

Пример:

***DX1:POISK.MAC=POISK.MAC/S**

РІР выполняет операцию копирования файла POISK.MAC по одному блоку в один прием с ДК: на DX1:.

1.4.19. Переключатель /T[:дата]. По переключателю /T[:дата] РІР приписывает указанную дату всем файлам, включенным в операции копирования или переименования. Обычно в этих операциях РІР сохраняет дату создания файла. Если дата не указана, РІР использует текущую дату.

Переключатель /T недопустим в операциях с магнитной лентой, т. к. РІР всегда использует текущую дату для файлов на ленте.

Пример:

****.*=*.*/Y/T**

РІР приписывает текущую дату всем файлам на ДК:.

1.4.20. Переключатель /U используется для объединения нескольких файлов в один файл. При этом в выходной спецификации файла нельзя использовать конструкцию «*».

Пример:

***AA.OBJ=DX1:DD.OBJ,CC.OBJ,BB.OBJ/U**

В этом примере РІР передает файлы DD.OBJ, CC.OBJ, BB.OBJ с DX1: на ДК: как один файл и присваивает ему имя AA.OBJ.

1.4.21. Переключатель /V копирует файлы с входного тома на два или более меньших по размеру выходных тома. РІР производит поиск файлов в справочнике входного тома, копирует их на выходной том и, после его заполнения, печатает сообщение:

MOUNT NEXT OUTPUT VOLUME IN <УСТ>;CONTINUE?

и ожидает ответа.

Для продолжения операции копирования необходимо: установить вместо заполненного тома новый выходной том того же типа, напечатать «Y» и нажать клавишу <BK>. Если напечатать «N» или нажать СУ/С, то операция прервется. После печати «Y» РІР продолжает копировать на выходной том файлы с того же сегмента справочника входного тома. Если все файлы с этого сегмента были скопированы, РІР начинает копирование со следующего сегмента справочника. Копирование продолжается до тех пор, пока все указанные входные файлы не будут скопированы.

Пример:

***DX1:*.*=DY1:*.*/V/Y**

MOUNT NEXT OUTPUT VOLUME IN DX1:;CONTINUE?Y
*

В этом примере копируются все файлы с DY1: на две дискеты DX1:.

1.4.22. Переключатель /W. При использовании переключателя /W в командной строке, PIP печатает на терминале список файлов, включенных в операции копирования, переименования или удаления.

Пример:

```
*DX1:*.*/D/W
?PIP—W—NO .SYS ACTION
FILES DELETED:
DX1:TEXT.MAC
DX1:TEXT1.MAC
DX1:TEXT2.MAC
```

*

1.4.23. Переключатель /X используется для замены сообщения о неисправимой (фатальной) ошибке на информационное сообщение в том случае, если какой-либо из указанных в командной строке входных файлов отсутствует.

Использование переключателя /X обеспечивает окончание операции. Если же переключатель /X не указан, то в случае обнаружения ошибки выполнение операции прерывается, печатается сообщение о неисправимой ошибке и управление передается клавиатурному монитору.

Пример:

```
*DY1:*.*=DY0:FILE1.TXT,FILE2.TXT,FILE3.TXT/X
?PIP—I—FILE NOT FOUND DY0:FILE2.TXT
```

*

В этом примере входные файлы FILE1.TXT и FILE3.TXT с DY0: были скопированы на DY1:, несмотря на отсутствие входного файла FILE2.TXT.

1.4.24. Переключатель /Y. Переключатель /Y используется в том случае, если необходимо выполнить операцию на системных файлах, имеющих тип .SYS.

Заметим, что нужно всегда использовать переключатель /Y с переключателем /P для включения файлов с типом .SYS, даже если не используется конструкция «*».

Пример:

```
*DX1:*.*=*.*/G/Y
```

в этом примере PIP копирует с DK: на DX1: в режиме отображения памяти все файлы, включая файлы с типом .SYS. Ошибки при вводе игнорируются.

1.4.25. Переключатель /Z. Использование переключателя /Z снимает защиту файла и делает возможным удаление или переименование этого файла. Может использоваться с переключателем /R.

Примеры:

1. *BOO.BAK=BOOT.MAC/R/Z

В этом примере файл BOOT.MAC переименовывается в незащищенный файл BOO.BAK на DK:.

2. **.MAC/Z

В этом примере отменяется защита всех файлов с типом .MAC на DK:.

1.5. Сообщения оператору

Ниже описаны сообщения, выдаваемые программой PIP.

?PIP—E—OPERATION NOT COMPLETED

Причина. Прервана операция с переключателем /E, потому что ответ на запрос начинался с «N» или нажато СУ/С.

Действие. Для продолжения операции на тот же самый запрос необходимо напечатать «Y» и нажать клавишу <BK>.

?PIP—E—PROTECTED FILE ALREADY EXISTS
DEV:FILNAM.TYP

Причина. На выходном томе уже существует защищенный файл с именем, указанным в командной строке. Операция создания защищенного выходного файла невозможна. PIP выполнит любую другую операцию, указанную в той же самой командной строке, кроме создания защищенного выходного файла.

Действие. Использовать переключатель /Z для снятия защиты уже существующего файла или воспользоваться другим именем для создания нового файла.

?PIP—F—CHANNEL IN USE

Причина. Ошибка при обращении к программе PIP.

Действие. Перезагрузить систему и повторить операцию. Если ошибка повторится снова, создать новую копию PIP.SAV и повторить операцию.

?PIP—F—CHANNEL NOT OPEN DEV:FILNAM.TYP

Причина. См. ?PIP—F—CHANNEL IN USE.

Действие. См. ?PIP—F—CHANNEL IN USE.

?PIP—F—CHECKSUM ERROR DEV:FILNAM.TYP

Причина. Ошибка контрольной суммы во время копирования, при использовании переключателя /B.

Действие. Проверить введенную командную строку. Повторить операцию по переключателю /G.

?PIP—F—DEVICE FULL DEV:FILNAM.TYP

Причина. На томе недостаточно места для размещения копируемого файла.

Действие. Увеличить размер свободной области на томе; удалить или переписать на другой том ненужные файлы, сжать том по команде SQUEEZE. Использовать конструкцию [N] для выходного файла. Использовать другой том.

?PIP—F—DEVICE IN USE

Причина. Требуемое устройство (обычно MT:) используется другим заданием.

Действие. Повторить операцию после выполнения другого задания.

?PIP—F—DIRECTORY FULL DEV:

Причина. В справочнике выходного тома нет места для записи выходного файла.

Действие. Использовать команду SQUEEZE для объединения неиспользуемых областей выходного тома или скопировать выходной том на том с большим числом сегментов справочника. Повторить команду.

?PIP—F—DIRECTORY INPUT ERROR DEV:

Причина. Аппаратная ошибка во время чтения справочника.

Действие. Проверить исправность и правильность включения аппаратуры. Проверить том на плохие блоки. Повторить операцию. Если ошибка повторяется снова, скопировать максимально возможное число файлов на другой том. Отформатировать том и повторить операцию. Если ошибка повторяется снова, заменить том другим.

?PIP—F—FETCH ERROR DEV:

Причина. Ошибка при обращении к программе PIP.

Действие. См. ?PIP—F—CHANNEL IN USE.

?PIP—F—FILE NOT FOUND DEV:FILNAM.TYP

Причина. Указанный входной файл не найден или нет файлов с указанной датой создания, когда используются переключатели /I[:дата], J[:дата] или T[:дата].

Действие. Проверить командную строку и сравнить имеющиеся файлы с указанными в командной строке. Повторить операцию.

?PIP—F—FILE SEQUENCE NUMBER NOT FOUND

Причина. Входной файл с указанным последовательным номером не найден на томе.

Действие. Проверить введенную командную строку. Получить справочник магнитной ленты по команде монитора DIRECTORY/POSITION и повторить операцию.

?PIP—F—INPUT ERROR DEV:FILNAM.TYP

Причина. Аппаратная ошибка во время чтения файла.

Действие. См. ?PIP—F—DIRECTORY INPUT ERROR.

?PIP—F—INSUFFICIENT MEMORY

Причина. Недостаточно памяти для выполнения требуемой операции (обычно, если спецификация файла содержит конструкцию «*»).

Действие. Удалить ненужные драйверы устройств. Удалить основное задание. Использовать монитор SJ. Разрешить свопинг USR по команде монитора SET USR SWAP. Копировать файлы отдельно.

?PIP—F—INVALID COMMAND

Причина. В командной строке обнаружена ошибка.

Действие. Ввести правильную командную строку.

?PIP—F—INVALID DELETE DEV:FILNAM.TYP

Причина. Операция удаления недопустима для магнитных лент.

Действие. Не требуется.

?PIP—F—INVALID DEVICE DEV:

Причина. Указано недопустимое или несуществующее устройство.

Действие. Проверить введенную командную строку. Использовать команду монитора SHOW для получения списка устройств, обслуживаемых системой.

?PIP—F—INVALID DIRECTORY

Причина. Структура справочника тома устройства (метки тома для магнитной ленты) не является структурой ФОДОС-2.

Действие. Инициализировать том по команде монитора INITIALIZE.

?PIP—F—INVALID OPTION:/X

Причина. В командной строке обнаружен недопустимый переключатель.

Действие. Ввести правильную командную строку.

?PIP—F—INVALID OUTPUT FILE

Причина. Недопустимое имя выходного файла для указанной операции.

Действие. Проверить и повторить командную строку.
 ?PIP—F—INVALID PROTECT OR UNPROTECT DEV:
 FILNAM.TYP

Причина. Ошибка при обращении (вызове) к программе PIP.
 Действие. См. ?PIP—F—CHANNEL IN USE.
 ?PIP—F—INVALID RENAME DEV:FILNAM.TYP

Причина. Недопустимая операция переименования.
 Действие. Проверить введенную командную строку (входной
 и выходной тома должны совпадать).
 ?PIP—F—INVALID SET DATE DEV:FILNAM.TYP

Причина. Ошибка при вызове программы PIP.
 Действие. См. ?PIP—F—CHANNEL IN USE.
 ?PIP—F—INVALID USE OF WILDCARDS DEV:FIL-
 NAM.TYP

Причина. Спецификация выходного файла содержит встроен-
 ную конструкцию «*» и не соответствует входной
 спецификации.

Действие. Проверить введенную командную строку.
 ?PIP—F—INVALID VALUE SPECIFIED WITH OP-
 TION:/X

Причина. Во время копирования с магнитной ленты значение
 последовательного номера файла в аргументе пе-
 реключателя /M:N выходит за пределы от —2 до
 999.

Действие. Проверить командную строку.
 ?PIP—F—LIBRARY FILE NOT COPIED DEV:FIL-
 NAM.TYP

Причина. Копирование библиотечных файлов по переключателю /B.

Действие. Копировать в режиме отображения памяти.
 ?PIP—F—OUTPUT ERROR DEV:FILNAM.TYP

Причина. Аппаратная ошибка во время операции записи. На
 томе недостаточно места для создания файла.

Действие. Проверить исправность и правильность включения
 аппаратуры. Проверить защиту записи. Проверить
 диск на плохие блоки. Сжать том по команде мо-
 нитора SQUEEZE. Удалить или записать на другой
 том ненужные файлы. Использовать конструкцию
 [N] для выходного файла. Использовать другой
 том.
 ?PIP—F—OUTPUT FILE FULL DEV:FILNAM.TYP

Причина. Выходной файл заполнен; на томе нет места для файла. Обнаружен маркер физического конца тома (для магнитной ленты).

Действие. Сжать том по команде монитора SQUEEZE. Удалить или записать на другой том ненужные файлы. Использовать конструкцию [N] для выходного файла. Использовать другой том.

?PIP—F—PROTECTED FILE DEV:FILNAM.TYP

Причина. Попытка удаления защищенного файла.

Действие. Снять защиту файла, используя переключатель /Z, и повторить команду.

?PIP—F—SIZE FUNCTION FAILED

Причина. Ошибка при определении размера тома на приводе, который не поддерживается монитором.

Действие. Установить том на привод с номером, который поддерживается монитором. Выполнить генерацию системы, увеличив количество поддерживаемых монитором приводов требуемого устройства.

?PIP—F—SYSTEM ERROR

Причина. Ошибка в системе.

Действие. Перезагрузить систему.

?PIP—F—TOO MANY OUTPUT FILES

Причина. В командной строке определено несколько выходных спецификаций.

Действие. Проверить введенную командную строку.

?PIP—F—VERIFICATION ERROR DEV:FILNAM.TYP

Причина. При использовании переключателя /V после операции копирования обнаружены различия между входными и соответствующими выходными данными.

Действие. Проверить том на плохие блоки. Повторить операцию копирования, используя переключатель /V. Если различия повторяются, заменить выходной том.

?PIP—F—WRONG VERSION

Причина. Попытка обращения системы ФОДОС к программе PIP, принадлежащей системе ФОДОС-2.

Действие. Вызвать программу PIP системы ФОДОС.

?PIP—I—DEVICE FULL DEV:FILNAM.TYP

Причина. При использовании переключателя /V установленный выходной том заполнен и не имеет участка памяти для размещения входных файлов. PIP про-

должает копирование и запрашивает установку следующего выходного тома.

Действие. Снять заполненный выходной том и установить новый выходной том.

?PIP—I—FILE NOT FOUND DEV:FILNAM.TYP

Причина. Указанный файл не найден.

Действие. Не требуется. Информационное сообщение.

?PIP—I—PROTECTED FILE NOT FOUND DEV:FILNAM.TYP

Причина. Указанный файл найден, однако он уже защищен.

Действие. Не требуется.

?PIP—I—UNPROTECTED FILE NOT FOUND DEV:FILNAM.TYP

Причина. Указанный файл найден, однако он уже защищен.

Действие. Не требуется.

?PIP—W—FILE LARGER THAN OUTPUT DEVICE DEV:FILNAM.TYP

Причина. При использовании переключателя /V размер указанного файла больше выходного устройства. Файл не скопирован, операция продолжается.

Действие. Использовать для копирования файла тома большего размера.

?PIP—W—INPUT ERROR DEV:FILNAM.TYP

Причина. Аппаратная ошибка во время операции чтения по переключателю /G. Предупреждающее сообщение.

Действие. Не требуется.

?PIP—W—NO .SYS ACTION

Причина. В командной строке не указан переключатель /Y. Операция выполнится для всех файлов, кроме файлов с типом .SYS. Предупреждающее сообщение.

Действие. Использовать переключатель /Y, если необходимо включить в операцию файлы с типом .SYS.

?PIP—W—OUTPUT FILE FOUND, NO OPERATION PERFORMED DEV:FILNAM.TYP

Причина. На устройстве вывода уже существует файл с тем же именем, что и выходной файл. Предупреждающее сообщение.

Действие. Переименовать файл. Использовать другой том.

?PIP—W—REBOOT

Причина. Файлы с типом .SYS перемещены, переименованы или удалены на системном томе. Предупреждающее сообщение.

Действие. Если это файлы, с которыми в данный момент работает система, перезагрузить систему, иначе — сообщение игнорировать.

2. ПРОГРАММА ОБСЛУЖИВАНИЯ УСТРОЙСТВ DUP

2.1. Назначение программы и условия выполнения программы

Программа обслуживания периферийных устройств (DUP) предназначена для создания и расширения файлов на устройствах файловой структуры операционной системы ФОДОС-2 (диски), сжатия и перезагрузки этих устройств, а также инициализации и копирования в режиме отображения устройств ФОДОС-2 файловой структуры (диски и магнитная лента).

DUP не обслуживает устройства ФОДОС-2 нефайловой структуры (терминал, перфоленточные устройства ввода-вывода, построочно-печатающее устройство).

2.2. Выполнение программы. Для вызова DUP с системного устройства следует подать с терминала команду

R DUP <BK>

после того, как монитор на терминале напечатает точку. После вызова DUP печатает звездочку и ожидает ввода командной строки.

Если в это время нажать клавишу <BK>, то DUP печатает номер своей версии.

2.3. Команды оператора. Режим работы программы DUP задается введением с терминала командной строки. В командной строке может быть указана только одна спецификация для входного файла и одна для выходного.

Для выхода из программы DUP и передачи управления монитору необходимо подать команду СУ/С, если DUP ожидает ввода с терминала, или дважды СУ/С, если DUP выполняет операцию. Исключения составляют операции по переключателям /S, /T и /C.

Для повторного пуска DUP следует подать команду REENTER.

2.3.1. Переключатели, используемые в программе DUP (табл. 2), делятся на две группы:

1) переключатели действия;

2) переключатели режима.

Переключатели действия выполняют специфические операции. Переключатели режима модифицируют переключатели

действия. В командной строке пользователь может определить только один переключатель действия вместе с соответствующими переключателями режима или без них. В табл. 3 приведены переключатели режима, которые могут быть использованы с определенными переключателями действия. Заметим, что переключатель /V может быть как переключателем действия, так и переключателем режима.

Таблица 2

Переключатель	Функция
1	2
/B[:RET]	Используется с переключателем /Z для создания файлов с типом .BAD на месте плохих блоков диска, найденных программой DUP на диске во время его инициализации. Использование :RET позволяет сохранять при инициализации все содержащиеся на диске файлы с типом .BAD, созданные ранее по /B
/C	Используется с переключателем /G для создания файла на указанном пользователем устройстве
/D	Используется с переключателем /Z для восстановления справочника инициализированного тома (если на этом томе ничего не записывалось с тех пор, как он был инициализирован)
/E:N	Используется с переключателями /I и /K для указания номера блока, на котором должна завершиться выполняемая операция
/F	Используется с переключателем /K для вывода на печать имени файла, содержащего плохой блок, и номера плохого блока относительно начала файла или с переключателем /I для копирования файла на выходное устройство или для копирования тома устройства в выходной файл

1	2
/G:N	Используется с переключателями /C, /I и /K для указания номера начального блока в выполняемой операции. N — целое число, представляющее номер начального блока
/H	Используется с переключателем /I для сравнения входных и выходных данных
/I	Копирует том диска на другой диск или магнитную ленту или с магнитной ленты на диск. (Используется с переключателями /G и /E)
/K	Скайирует том устройства на плохие блоки и печатает восьмеричные адреса плохих блоков на устройстве вывода. Может использоваться с переключателями /G и /E
/N:N	Используется с переключателем /Z для задания требуемого числа сегментов справочника. N — целое, от 1 до 37 (восьмеричное)
/O	Загружает указанный том устройства или файл
/Q	Используется с переключателем /O для загрузки тома более ранней версии системы ФОДОС
/S	Сжимает том диска на другой диск или на тот же самый
/T:N	Расширяет указанный файл на N блоков
/U[:XX]	Записывает начальный загрузчик файла монитора в блоки 0 и 2—5 указанного тома устройства; «XX» — двухсимвольное имя устройства с произвольным доступом к данным, для которого записывается начальный загрузчик
/V[:ONL]	Печатает идентификацию тома. Используется с переключателем /Z для

1	2
	записи идентификации тома в первый блок инициализируемого диска или в нулевой блок инициализируемой магнитной ленты. Использование аргумента ONL изменяет идентификацию тома без инициализации устройства (последнее не применимо для магнитных лент) /W Используется с любым переключателем действия для запуска и остановки операции. Это полезно в однодисковых системах, т. к. позволяет заменить системный диск другим диском перед выполнением операции /X Используется с переключателем /S для запрещения автоматической загрузки системного тома после его сжатия /Y Используется вместе с переключателями /C, /I, /O, /S, /T и /Z для запрещения печати сообщений: DEV:/XXXX;ARE YOU SURE? и FOREGROUND JOB LOADED;CONTINUE? и немедленного выполнения операции /Z[:N] Инициализирует том указанного устройства. Аргумент N используется для размещения дополнительных слов в записи справочника о файле

Примечания:

1. Инициализировать том — значит подготовить его для использования в операционной системе ФОДОС-2. Для магнитной ленты — записать метку тома VOL1 и метку LEOT (логический конец ленты). Для диска — записать идентификацию тома и создать на нем справочник.

2. Операция сжатия есть операция перезаписи всех файлов тома в начало диска.

3. Идентификация тома включает идентификатор тома и имя владельца. Используется для опознавания тома и определения его принадлежности.

Таблица 3

Переключатель действия	Переключатель режима
1	2
C	W, Y, G, E
D	W, Y
I	W, Y, G, E, F, H
K	W, F, G, E
O	W, Y
S	W, Y, X
T	W, Y
U	W, Y
V	W, Y
Z	W, B, N, V, Y, D

2.3.2. Переключатель /C[/G:N] используется для создания файла с указанным именем, размером и в определенном месте указанного устройства (для устройств с произвольным доступом к данным). Переключатель /C создает в справочнике тома только запись о файле. Запись данных в файл не происходит.

Формат командной строки:

спф[L]=/C[/G:N]

где спф — спецификация создаваемого файла; [L] — размер создаваемого файла в блоках. Заметим, что квадратные скобки здесь являются частью команды. Если размер файла не указывается, DUP создает одноблочный файл (L — десятичное число).

N — номер начального блока создаваемого файла (N — восьмеричный или N. — десятичный); если /G:N не используется, DUP создает файл на месте ближайшей свободной области, достаточной для размещения данного файла.

Переключатель /C может быть использован для создания файлов с типом .BAD на месте плохих блоков диска.

Он также используется для восстановления случайно удаленных файлов. Для этого необходимо получить справочник

удаленных файлов тома и присвоить имя той области тома, которая содержит необходимые данные.

Перед использованием переключателя /C необходимо убедиться, чтобы область справочника, в которой будет создаваться файл, была неиспользуемой (<UNUSED>). Если для создания файла требуется меньшее число блоков, чем размер неиспользуемой области, оставшиеся блоки будут описаны в справочнике тома как неиспользуемая область (<UNUSED>). Если же места на томе не достаточно для создания файла, DUP напечатает сообщение об ошибке:

?DUP—F—NO ROOM FOR FILE DEV:FILNAM.TYP
и не будет создавать файл.

По переключателю /C DUP производит проверку на дублирование имени файла. Если на томе уже существует файл с тем же именем, что и вновь создаваемый, DUP напечатает сообщение об ошибке и не будет создавать файл.

Если произведена попытка создания файла на месте временного файла (открытого и не закрытого) при загруженном основном задании, то система запрашивает подтверждения выполнения операции. Если напечатать «Y» и нажать клавишу <BK>, то DUP создаст файл на месте временного файла. В этом случае, если временный файл использовался основным заданием, файл будет испорчен и результат будет непредсказуем.

Пример:

*DK1:FILE.MAC[3]=/C/G:140

В этом примере, используя /C, DUP создает файл с именем FILE.MAC в блоках 140, 141, 142 на устройстве DK1:.

2.3.3. Переключатель /I. По переключателю /I DUP копирует «блок в блок» (режим отображения) том с одного устройства на другое без изменения структуры файлов и их положения на томе. Если DUP встретит плохой блок на входном или выходном томе, она напечатает сообщение об ошибке и повторит операцию. Если в процессе копирования будет напечатано только одно сообщение об ошибке, можно предположить, что передача данных завершена правильно.

Переключатель /I можно использовать для копирования с тома диска, информация на котором записана не в формате ФОДОС-2.

Формат командной строки:

уст:имфайл[/G:N1[/F]=уст:[имфайл]/I[/G:N2/E:N3] [/F[/H]

*

где имфайл — имя входного или выходного файла; необходимо указывать имя файла вместе с переключателем /F; если переключатель /F не используется и указан входной файл, то в выходной спецификации используется только фиктивное имя файла (*); разрешается использовать одно из двух имен файла, входное или выходное, но не оба сразу;

* — фиктивное имя файла (обязательно звездочка; если в качестве выходного устройства используется магнитная лента, то обязательно нужно указывать имя выходного файла); на выходном устройстве может быть определено или имя файла, или фиктивное имя файла, но не оба сразу;

N1 — номер начального блока для операции записи;

N2 — номер начального блока для операции чтения;

N3 — номер последнего блока для операции чтения;

при использовании переключателя /I командная строка должна включать как входную, так и выходную спецификацию; если необходимо указать номер какого-либо блока, то должны быть указаны все три переключателя, определяющие номера граничных блоков для операций чтения и записи.

Если емкость одного тома меньше, чем другого, DUP осуществляет копирование в соответствии с емкостью меньшего тома.

Переключатель /I используется также для копирования информации между диском и магнитной лентой. DUP записывает данные на ленту, формируя их в блоки по 1K слов. При этом на магнитной ленте, независимо от ее размера, возможно хранение только одного тома диска.

Переключатель /H используется с переключателем /I для сравнения входных данных с соответствующими выходными данными после операции копирования

Примеры:

1. *DX1:* = DX0:/I

DX1:/COPY;ARE YOU SURE?Y

В этом примере DUP копирует содержимое всех блоков диска DX0: на DX1:. Имя файла * — несущественно; оно является фиктивным и требуется интерпретатором командной строки (CSI).

2. *DX1:*/G:101 = DX0:/I/G:0/E:200

DX1:/COPY;ARE YOU SURE?Y

В этом примере DUP копирует содержимое блоков с 0 по 200 диска DX0: на диск DX1:, записывая их со 101 блока.

3. *DY1:FLOPPY.BAK/F = DY0:/I

DY1:/COPY;ARE YOU SURE?Y

В этом примере DUP копирует устройство DY0: на DY1: с именем файла FLOPPY.BAK.

2.3.4. Переключатель /K. В результате использования дисков и их износа на них могут появиться плохие блоки. Операцию по проверке дисков на плохие блоки DUP выполняет по переключателю /K. DUP печатает сообщение о плохих блоках на системном терминале или записывает информацию о плохих блоках в выходной файл.

Необходимо помнить, что номера блоков представляют собой восьмеричные числа, и первый блок диска есть блок с номером 0. Если DUP не обнаружит плохие блоки, печатается соответствующее сообщение (см. п. 2.4). Полное сканирование диска занимает от одной до нескольких минут, в зависимости от размера тома. DUP не разрушает информацию, хранящуюся на диске.

Можно для сканирования выбрать часть устройства, указав номера начального и конечного блоков.

Формат командной строки:

[выходспф=]входуст:/K[/G:N1[/E:N2]][/F]

где выходспф — спецификация выходного файла для сообщения о плохих блоках (устройство, имя и тип файла);

N1 — начальный блок для операции сканирования;

N2 — конечный блок для операции сканирования.

Если в командной строке указан только переключатель /G:N1, DUP просканирует том от указанного блока N1 до конца.

Вместе с переключателем /K можно использовать переключатель /F для вывода на печать имени файла, содержащего плохой блок и номера плохого блока, относительно начала файла. Иногда блок, о котором сообщается как о плохом, может быть восстановлен. Для того, чтобы узнать какой тип ошибки имел плохой блок — «SOFT» или «HARD» (может он быть восстановлен или нет), нужно выполнить повторное сканирование тома на плохие блоки и сравнить оба сообщения. Если в обоих сообщениях о блоке сообщается как о плохом, то это означает ошибку типа «HARD», и восстановление блока невозможно. Если о блоке сообщается как о плохом лишь в первом сообщении, и не сообщается во втором, то это означает, что произошла ошибка типа «SOFT», и блок восстановлен.

Пример:

*DY1:/K/G:100./E:200.

В этом примере сканируется на плохие блоки устройство DY1: с 100-го блока по 200-ый.

2.3.5. Переключатель /F используется с переключателем /K для проверки устройства на наличие плохих блоков (см. п. 2.3.4).

Пример:

*DX0:/K/F

BLOCK	TYPE	FILE	BLOCK	
000717	463.	HARD NUMBER.PAS	000546	358.
000725	469.	HARD ANTONY.MAC	000554	364.
000732	474.	HARD CAESAR.MAC	000561	369.
000743	483.	HARD <UNUSED>	000572	378.

?DUP—I—BAD BLOCKS DETECTED 4.

В этом примере приведен перечень плохих блоков устройства DX0:. В колонке BLOCK DUP печатает номер плохого блока диска; в колонке TYPE — тип ошибки («HARD» или «SOFT»); в колонке FILE — имя и тип файла, содержащего плохой блок; в следующей колонке BLOCK — номер плохого блока относительно начала файла.

При использовании /F вместе с /I выполняется копирование входного файла на выходное устройство или копирование входного устройства в выходной файл.

Заметим, что при использовании магнитной ленты во входной и выходной спецификациях обязательно указывается имя файла.

2.3.6. Переключатель /O выполняет две функции: аппаратного загрузчика определенного тома устройства и загрузчика определенного файла монитора.

Формат командной строки для загрузки тома устройства:
уст:/O

формат командной строки для загрузки определенного файла монитора без изменения загрузчика системы на томе:
уст:монитор/O

где монитор — имя загружаемого монитора.

Если указанная операция загрузки тома устройства или монитора недействительна по ряду причин, DUP печатает сообщение об ошибке и выходит в режим ожидания командной строки.

После применения переключателя /O пользователю нет необходимости вновь определять дату и время по командам монитора DATE и TIME. Однако, во время перезагрузки, часы отстанут на несколько секунд.

Пример:

*DK1:FMONSJ.SYS/O
ФОДОС Ф В03.00

В этом примере произведена загрузка файла монитора SJ.

2.3.7. Использование переключателя /Q вместе с переключателем /O позволяет загрузить том, содержащий монитор системы ФОДОС (предыдущей версии системы).

Пример:

*DX0:/O/Q
ФОДОС Ф В02.00

В этом примере загружается ФОДОС второй версии.

DUP сохраняет дату и время после использования переключателя /Q.

2.3.8. Переключатель /S используется для сжатия диска на другой диск или на тот же самый. Для этого DUP переписывает все файлы в начало диска, создавая тем самым одну неиспользуемую область после всех файлов. Загрузчик системы в начале тома остается без изменения. Перед операцией сжатия том устройства вывода должен быть инициализирован.

Если устройство вывода не указано в командной строке, DUP напечатает сообщение:

<УСТ>/SQUEEZE; ARE YOU SURE?

и будет ждать ответа. Для продолжения операции необходимо напечатать «Y» и нажать клавишу <BK>.

Для того, чтобы быть уверенным в безошибочном выполнении операции сжатия, перед использованием /S необходимо проверить том устройства на плохие блоки по переключателю /K.

Переключатель /S не перемещает файлы с типом .BAD.

Если во время сжатия произойдет ошибка, DUP напечатает сообщение об ошибке и повторит операцию. Если, таким образом, будет напечатано только одно сообщение об ошибке, можно предположить, что операция завершена правильно.

Формат командной строки:

[выходуст:=]входуст:/S

где выходуст: — устройство вывода;

входуст: — устройство ввода.

Переключатель /S неприменим для системного устройства, когда загружено основное задание. В этом случае DUP печатает сообщение об ошибке:

**?DUP—F—CAN'T SQUEEZE SY: WHILE FOREGROUND
LOADED**

и не будет выполнять операцию сжатия. Перед использованием /S необходимо разгрузить основное задание.

Если операция сжатия выполняется на системном устройстве, то после ее завершения автоматически произойдет перезагрузка операционной системы. Это делается для того, чтобы предотвратить возможное нарушение в работе системы в результате перемещения на томе файла монитора.

Для запрещения автоматической перезагрузки системы вместе с переключателем /S используется переключатель /X. Тем не менее, использование /X возможно лишь в том случае, если известно, что файл монитора и драйверы устройств не будут перемещены. Однако, если в командной строке определен переключатель /X, а программа обслуживания пользователя (USR) по каким-либо причинам не может быть резидентной, DUP все равно перезагрузит операционную систему.

При сжатии системного тома необходимо, чтобы программа обслуживания устройств, находящаяся на нем, имела имя DUP.SAV. Если это условие не выполняется, происходит отказ системы.

Пример:

***DX1:* = DX0:/S**

В этом примере DUP сжимает DX0: на DX1:, при этом содержимое DX0: остается без изменения. Имя файла * — несущественно; оно является фиктивным и требуется интерпретатором командной строки (CSI).

Пример:

***SY:/S**

SY:SQUEEZE;ARE YOU SURE?Y

ФОДОС Ф B03.00

В этом примере сжимается системный том и система перезагружается.

2.3.9. Переключатель /T:N используется для расширения файла. Формат командной строки:

спф = /T:N

где спф — спецификация расширяемого файла;

N — число дополнительных блоков для файла.

Расширять файл таким образом можно только в том случае, если за ним на томе есть неиспользуемая область размером, по крайней мере, в N блоков.

Пример:

***DX1:СОРУ.МАС=Т:20**

В этом примере DUP отведет для файла СОРУ.МАС на DX1: дополнительно 20 блоков.

2.3.10. Переключатель /U[:XX]. Для того, чтобы использовать диск в качестве системного устройства, необходимо скопировать на диск начальный загрузчик системы. Чтобы сделать это, необходимо убедиться, что на диске находится соответствующий файл монитора и драйвер.

«XX» — двухсимвольное имя устройства произвольного доступа к данным, для которого записывается начальный загрузчик на устройстве произвольного доступа к данным другого типа. Например, для записи начального загрузчика на устройстве DY: для устройства DX:

Пример:

***DX1:* =DX0:FMONFB.SYS/U**

По этой команде начальный загрузчик монитора основного/фонового задания записывается в блоки начального загрузчика (блоки 0 и 2—5) на DX1:. Имя файла * — несущественно, оно является фиктивным и требуется интерпретатором командной строки (CSI).

2.3.11. Переключатель /V[:ONL] используется (как переключатель действия) для вывода на печать идентификации тома устройства или для изменения идентификации тома без инициализации тома.

Формат командной строки:

уст: [/Z]/V[:ONL]

где уст: — имя устройства, идентификацию которого требуется изменить или распечатать.

Если в командной строке указан переключатель /V, DUP печатает на терминале идентификатор тома и имя владельца тома указанного устройства.

Если в командной строке указаны переключатели /Z и /V, DUP инициализирует устройство и запрашивает новый идентификатор тома и имя владельца.

Если в командной строке указаны переключатели /Z и /V:ONL, DUP допускает лишь замену идентификатора тома и имени владельца и не инициализирует устройство.

Если в командной строке указаны переключатели /Z/V или /Z/V:ONL, DUP запрашивает новый идентификатор тома: VOLUME ID?

Количество символов в идентификаторе тома не должно превышать 12-ти после нажатия клавиши <BK> (для магнитных лент — шести).

DUP запрашивает новое имя владельца:

OWNER?

Количество символов в имени владельца не должно превышать 12-ти после нажатия клавиши <BK> (для магнитных лент — десяти).

DUP игнорирует символы, выходящие за допустимую длину. Невозможно изменить идентификацию тома магнитной ленты без инициализации ленты. Команда с переключателем /V:ONL изменяет лишь идентификатор тома и имя владельца; устройство не инициализируется.

Пример:

*RK1:/Z/V:ONL

RK1:/VOLUME ID CHANSE; ARE YOU SURE?Y

VLUME ID? FODOS

OWNER? Петров

В этом примере с помощью переключателя /V:ONL получены новый идентификатор тома RK1: — «FODOS», и новое имя владельца — «Петров».

2.3.12. Переключатель /W рекомендуется использовать в том случае, если при выполнении операции возникает необходимость смены томов. Важным случаем подобной ситуации является выполнение операции с использованием одного (единственного) привода устройства.

По /W DUP инициирует выполнение операции, затем приостанавливает ее и печатает сообщение:

MOUNT INPUT VOLUME IN <УСТ>;CONTINUE?

где <УСТ> — устройство, на которое необходимо установить входной том.

После установки входного тома на устройство <УСТ> для продолжения операции нужно напечатать «Y» и нажать клавишу <BK> (для запрещения операции необходимо напечатать «N» или CY/C, операция не выполнится; будет напечатано сообщение MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE? (см. ниже); в случае CY/C управление возвратится к клавиатурному монитору; любой другой ответ приводит к повторению первого сообщения).

После завершения операции (кроме операции загрузки системы по переключателю /O), DUP печатает сообщение:

MOUNT SYSTEM VOLUME IN <УСТ>;CONTINUE?

где <УСТ> — устройство, на которое необходимо установить системный том.

Необходимо вновь установить системный том, напечатать «Y» и нажать клавишу <BK>. DUP печатает звездочку (*) и выходит в режим ожидания командной строки.

Если используется переключатель /W, DUP должна обязательно находиться на системном томе.

Пример:

*DX1:/K/F/W

MOUNT INPUT VOLUME IN DX1;;CONTINUE?Y

?DUP—I—NO BAD BLOCKS DETECTED

MOUNT SYSTEM VOLUME IN DX1;;CONTINUE?Y

*

В этом примере DUP сканирует DX1: на плохие блоки. Во время первой паузы системный диск заменяется другим диском, печатается «Y» и нажимается клавиша <BK>. После этого выполняется операция сканирования. Во время второй паузы устанавливается системный диск (на котором находится программа DUP), печатается «Y» и нажимается клавиша <BK>. DUP печатает звездочку (*), что означает готовность к приему новой командной строки.

2.3.13. Переключатель /Y используется для запрещения печати сообщений, требующих подтверждения на выполняемую операцию. Например, если в памяти находится основная программа, по переключателям /C, /I, /O, /S, /T и /Z DUP печатает сообщение:

BACKGROUND JOB LOADED, CONTINUE?

Для продолжения операции необходимо напечатать «Y» и нажать клавишу <BK>.

При использовании переключателей /C, /I, /O, /S, /V и /Z DUP печатает сообщение:

ARE YOU SURE?

и также будет ждать ответа. Если в это время в памяти находится основная программа, DUP печатает оба эти сообщения. Для запрещения этих сообщений и обеспечения немедленного выполнения требуемой операции в командной строке следует указать переключатель /Y.

2.3.14. По переключателю /Z DUP инициализирует том указанного устройства. Эта операция всегда должна быть первой операцией на новом (отформатированном) томе. После использования переключателя /Z файлы удаляются из справочника тома.

Формат командной строки:

уст:/Z[:N]

где уст: — устройство, которое нужно инициализировать;

N — восьмеричное число ($N > 0$): указывает на изменение размера каждой записи о файле в справочнике тома. DUP прибавляет это число к числу слов по умолчанию, отведенных для каждой записи справочника (неприменим для устройств несправочной структуры).

Число файлов, которые могут храниться на томе устройства, определяется, в частности, размером справочника. Система допускает 31 сегмент в справочнике тома устройства и до 72 записей файлов в каждом сегменте. Сегмент представляет собой 2 блока диска. Если в командной строке аргумент N не определен, каждая запись справочника имеет длину 7 слов (для информации об имени файла, дате его создания и размере файла). Если определены дополнительные слова, число записей в сегменте справочника уменьшится. Формула определения числа записей в сегменте имеет вид:

$$(512 - 7) / (\text{дпс} + 7)$$

где дпс — число дополнительных слов в записи.

Например, при использовании /Z:1, в сегменте справочника можно сделать только 63 записи.

Число сегментов в справочнике, отводимое системой по умолчанию для устройств ФОДОС-2, приведено в табл. 4.

Таблица 4

Имя устройства	Размер справочника в сегментах (десятич.)
1	2
RK	16
DP	31
DX	1
DY	4
MX	1
MY	2

Если пользователя не удовлетворяет размер справочника по умолчанию, он может с помощью переключателя /N:N (N — целое десятичное число от 1 до 31) задать требуемое число сегментов.

Пример:

*DY1:/Z/N:6

DY1:INITIALIZE;ARE YOU SURE?Y

В этом примере DUP инициализирует DY1: и отводит 6 сегментов для справочника.

Во время инициализации диска или магнитной ленты DUP сохраняет в блоке номер 0 (метка тома) неизменными идентификатор тома и имя владельца.

Если во время инициализации тома необходимо заменить идентификатор тома и имя владельца, вместе с переключателем /Z используется переключатель /V.

Пример:

*DX1:/Z/V

DX1:INITIALIZE;ARE YOU SURE?Y

VOLUME ID? FODOS

OWNER? Иванов

В этом примере инициализируется устройство DX1: и устанавливается идентификатор тома — «FODOS», и имя владельца — «Иванов».

2.3.15. Переключатель /B[:RET]. Для сканирования тома на плохие блоки и записи на них файлов вместе с переключателем /Z используется переключатель /B. Для каждого встретившегося плохого блока на томе DUP создает файл FILE.BAD. После того, как том будет инициализирован и операция сканирования завершится, справочник тома будет содержать только файлы FILE.BAD (если на томе есть плохие блоки). Если DUP обнаружит плохой блок в начальном загрузчике системы или в справочнике тома, она печатает сообщение об ошибке, которое информирует пользователя о непригодности тома к использованию.

Если используется /B:RET, DUP сохраняет во время инициализации все файлы FILE.BAD, созданные ранее по переключателю /B.

Если данный том содержит плохие блоки, система печатает адреса плохих блоков (восьмеричные и десятичные).

Пример:

*DY0:/Z/B

DY0:/INITIALIZE;ARE YOU SURE?Y

	BLOCK	TYPE
--	-------	------

000120	80.	HARD
--------	-----	------

000471	313.	HARD
--------	------	------

000521	337.	HARD
--------	------	------

?DUP—W—BAD BLOCKS DETECTED 3.

В этом примере в левой колонке приведены восьмеричные, в средней — десятичные адреса плохих блоков. В правой колонке печатается тип плохого блока: «HARD» или «SOFT».

2.3.16. Переключатель /D используется для восстановления справочника тома, если после инициализации тома на него не копировались никакие файлы. DUP пытается восстановить содержимое справочника, если данный том был инициализирован. (Этот переключатель используется, если том инициализирован ошибочно).

Заметим, что по переключателю /D не будут восстанавливаться блоки начального загрузчика. Поэтому, если /D используется для восстановления системного тома, то необходимо воспользоваться переключателем /U:XX] для записи начального загрузчика.

Пример:

*DY1:/Z/D

В этом примере восстанавливается справочник DY1:.

2.4. Сообщения оператору

Ниже приведены сообщения, выдаваемые программой DUP.

?DUP—E—OPERATION NOT COMPLETED

Причина. При использовании переключателя /W в ответ на запрос на выполнение операции напечатано «N» или нажато **CV/C**.

Действие. Повторить команду. Для выполнения операции напечатать «Y» и нажать клавишу **<BK>** в ответ на запрос.

?DUP—F—BAD BLOCK IN SYSTEM AREA DEV:

Причина. Плохие блоки в системной области диска.

Действие. Отформатировать диск и повторить операцию. Если ошибка повторяется, диск необходимо заменить.

**?DUP—F—CAN'T SQUEEZE SY: WHILE
FOREGROUND LOADED**

Причина. Попытка сжать системный том, когда системное или основное задание находится в памяти.

Действие. Удалить основное или системное задание из памяти и повторить команду.

?DUP—F—CHANNEL IN USE DEV:FILNAM.TYP

Причина. Ошибка при обращении (вызове) к программе DUP.

Действие. Перезагрузить систему и повторить операцию. Если ошибка повторится снова, создать новую копию

DUP.SAV и повторить операцию.
?DUP—F—CHANNEL NOT OPEN
DEV:FILNAM.TYP

Причина. Ошибка при обращении к программе DUP.

Действие. Перезагрузить систему и повторить операцию. Если ошибка повторится снова, создать новую копию DUP.SAV и повторить операцию.

?DUP—F—CONFLICTING SYSGEN OPTIONS

Причина. Используются несовпадающие варианты генерации драйвера устройства и RMON.

Действие. Проверить соответствие вариантов генерации.

?DUP—F—DEVICE FULL DEV:FILNAM.TYP

Причина. Недостаточно места на выходном томе для размещения указанного файла или данных.

Действие. Использовать переключатель /C, указав меньший размер файла или использовать переключатель /S для объединения неиспользуемых областей на выходном томе. Если использовался переключатель /S для сжатия входного тома на выходной, инициализировать выходной том, указав меньшее число для сегментов справочника. Если использовались переключатели /I/F, скопировать на другой выходной том или использовать переключатель /S для создания достаточно большой смежной неиспользуемой области на имеющемся выходном томе.

?DUP—F—DIRECTORY FULL DEV:

Причина. Нет места в справочнике выходного тома для создания выходного файла.

Действие. С помощью переключателя /S объединить неиспользуемые области на выходном томе или использовать другой том.

?DUP—F—DIRECTORY INPUT ERROR DEV:

Причина. Ошибка чтения справочника указанного устройства.

Действие. Проверить исправность и правильность включения аппаратуры. Проверить том на плохие блоки. Повторить операцию. Если ошибка повторяется, скопировать максимально возможное число файлов на другой том. Отформатировать том и повторить операцию. Если ошибка повторяется снова, заменить том другим.

?DUP—F—DIRECTORY OUTPUT ERROR DEV:

Причина. См. ?DUP—F—DIRECTORY INPUT ERROR DEV:.

Действие. См. ?DUP—F—DIRECTORY INPUT ERROR DEV:.
 ?DUP—F—FETCH ERROR DEV:

Причина. Произошел сбой в работе операционной системы.

Действие. Перезагрузить систему и повторить операцию. Если ошибка повторится, использовать новую копию DUP.SAV и драйвера указанного устройства. Повторить операцию.
 ?DUP—F—FILE NOT FOUND DEV:FILNAM.TYP

Причина. Указанный файл не найден.

Действие. Проверить командную строку. Повторить операцию.
 ?DUP—F—INPUT ERROR DEV:FILNAM.TYP

Причина. Аппаратная ошибка во время операции чтения.

Действие. Проверить исправность и правильность включения аппаратуры. Проверить диск на плохие блоки.
 ?DUP—F—INSUFFICIENT MEMORY

Причина. Недостаточно памяти под буферы для выполнения требуемой операции.

Действие. Разгрузить ненужные драйверы устройств. Разгрузить основное задание. Использовать монитор SJ. Разрешить свопинг USR по команде монитора SET USR SWAP.
 ?DUP—F—INVALID COMMAND

Причина. В командной строке обнаружена ошибка.

Действие. Ввести правильную командную строку.
 ?DUP—F—INVALID DEVICE DEV:

Причина. Указанное устройство не занесено в таблицу устройств монитора.

Действие. Проверить командную строку. Используя команду монитора INSTALL, занести указанное устройство в таблицу устройств монитора.
 ?DUP—F—INVALID DIRECTORY DEV:

Причина. Том указанного устройства имеет недопустимую в системе ФОДОС-2 структуру справочника.

Действие. Инициализировать том, используя переключатель /Z.
 ?DUP—F—INVALID OPTION:/X

Причина. В командной строке указан недопустимый переключатель.

Действие. Проверить командную строку.
 ?DUP—F—INVALID RESTORE DATA DEV:

- Причина. DUP не в состоянии восстановить справочник тома с помощью переключателя /D, потому что в блоке номер 1 тома хранятся недопустимые данные.
- Действие. Процедура восстановления тома невозможна. (Справочник тома инициализирован безвозвратно).
?DUP—F—INVALID VALUE SPECIFIED WITH OPTION:/X
- Причина. В командной строке указан недопустимый аргумент переключателя.
- Действие. Проверить командную строку.
?DUP—F—NON BOOTABLE DRIVER
DEV:FILNAM.TYP
- Причина. Попытка загрузить том, который не содержит начальный загрузчик. Отсутствует двоеточие (:) в имени физического устройства.
- Действие. Использовать другое устройство для выполнения операции. Повторить команду, указав двоеточие в имени физического устройства.
?DUP—F—NO ROOM FOR FILE
DEV:FILNAM.TYP
- Причина. Размер неиспользуемой области памяти меньше указанного значения [N] по переключателю DUP /C или указанную область уже занимает файл.
- Действие. Удалить файл или файлы с указанной области тома. Повторить команду.
?DUP—F—NO SPACE FOR EXTENSION
DEV:FILNAM.TYP
- Причина. Размер неиспользуемой области, следующей за расширяемым (по переключателю /T:N) файлом, меньше значения аргумента переключателя.
- Действие. Указать меньшее значение аргумента переключателя /T:N или удалить файл или файлы с указанной области тома.
?DUP—F—NO B3 BOOT ON VOLUME
- Причина. Попытка загрузить том, не содержащий начального загрузчика системы ФОДОС-2.
- Действие. Использовать переключатель /U для записи на том начального загрузчика системы ФОДОС-2 или использовать переключатель /Q для загрузки системы ФОДОС.
?DUP—F—OUTPUT ERROR DEV:FILNAM.TYP
- Причина. Аппаратная ошибка во время операции записи.

- Действие. См. ?DUP—F—INPUT ERROR DEV:FILNAM.TYP.
?DUP—F—OUTPUT FILE EXISTS
DEV:FILNAM.TYP
- Причина. Файл с именем, указанным в командной строке с использованием переключателя /C, уже существует на выходном томе.
- Действие. Удалить существующий на томе файл или указать другое имя выходного файла.
?DUP—F—SIZE FUNCTION FAILED
- Причина. Ошибка при определении размера монтируемого тома. (Монитор, драйвер устройства или DUP могут быть разрушены). Это сообщение возникает, когда привод с указанным номером не поддерживается монитором, созданным при генерации системы.
- Действие. Перезагрузить систему и повторить операцию. В случае повторения ошибки создать новую копию DUP.SAV и повторить операцию.
?DUP—F—UNEXPECTED EOF DEV:FILNAM.TYP
- Причина. Ошибка при обращении к программе DUP.
- Действие. Перезагрузить систему и повторить операцию. Если ошибка повторится снова, создать новую копию DUP.SAV и повторить операцию.
?DUP—F—ININITIALIZED VOLUME DEV:
- Причина. Справочник указанного тома не инициализирован.
- Действие. Инициализировать справочник тома, используя переключатель /Z.
?DUP—F—VOLUME NOT FODOS FORMAT DEV:
- Причина. Указан том, имеющий недопустимый формат в системе ФОДОС-2.
- Действие. Отформатировать том, используя команду монитора FORMAT, или использовать том системы ФОДОС-2.
?DUP—I—NO BAD BLOCKS DETECTED DEV:
- Причина. Во время операции сканирования устройства (с использованием переключателей /K или /B) плохих блоков не обнаружено.
- Действие. Информационное сообщение. Действий не требуется.
?DUP—U—CHANNEL NOT OPEN
- Причина. Ошибка при обращении к программе DUP.
- Действие. Перезагрузить систему и повторить операцию. Если ошибка повторится снова, создать новую ко-

пию DUP.SAV и повторить операцию.

?DUP—U—SYSTEM ERROR

Причина. См. ?DUP—U—CHANNEL NOT OPEN.

Действие. См. ?DUP—U—CHANNEL NOT OPEN.

?DUP—U—WRONG VERSION

Причина. Произошла попытка обращения системы ФОДОС к программе DUP, входящей в состав системы ФОДОС-2.

Действие. Использовать DUP системы ФОДОС.

?DUP—W—BAD BLOCKS DETECTED NNNNN.

Причина. Во время операции сканирования устройства (с использованием переключателей /К или /В) обнаружено NNNNN плохих блоков.

Действие. Информационное сообщение. Действий не требуется.

?DUP—W—DEVICE FULL DEV:FILNAM.TYP

Причина. На выходном томе недостаточно места для размещения передаваемых файлов.

Действие. Использовать другой том, больший или равный входному.

?DUP—W—OUTPUT ERROR DEV:FILNAM.TYP

Причина. Ошибка во время операции сжатия.

Действие. DUP создает файл с типом .BAD на месте плохого блока и продолжает операцию.

?DUP—W—OWNER NAME TRUNCATED AT TEN CHARACTERS

Причина. Имя пользователя для магнитной ленты содержит более десяти символов, DUP «отсекает» лишние символы.

Действие. Если сокращение не приемлемо, использовать переключатели /Z/V. Указать не более 10-ти символов.

?DUP—W—OWNER NAME TRUNCATED AT TWELVE CHARACTERS

Причина. Имя пользователя для диска содержит более двенадцати символов, DUP «отсекает» лишние символы.

Действие. Если сокращение не приемлемо, использовать переключатели /Z/V. Указать не более 12-ти символов.

?DUP—W—TOO MANY BAD BLOCKS DEV:

Причина. Обнаружено более 128 плохих блоков во время сканирования.

Действие. Том непригоден для использования и подлежит замене.

?DUP—W—VERIFICATION ERROR AT
RELATIVE BLOCK NNNNNN

Причина. При использовании переключателей /I/H обнаружено несоответствие входных и выходных данных.

Действие. Проверить выходной том на плохие блоки. Повторить операцию.

?DUP—W—VOLUME ID TRUNCATED AT
SIX CHARACTERS

Причина. В идентификаторе тома магнитной ленты более шести символов. DUP «отсекает» лишние символы.

Действие. Если сокращение не приемлемо, использовать переключатели /Z/V. Указать не более 6-ти символов.

?DUP—W—VOLUME ID TRUNCATED AT
TWELVE CHARACTERS

Причина. Идентификатор тома для диска содержит более двенадцати символов, DUP «отсекает» лишние символы.

Действие. Если сокращение не приемлемо, использовать переключатель /V для изменения идентификации тома. Указать не более 12-ти символов.

3. ПРОГРАММА ПОЛУЧЕНИЯ СПРАВОЧНИКА DIR

3.1. Назначение программы и условия выполнения программы

Программа получения справочника (DIR) предназначена для получения справочной информации об определенном томе устройства и файлах, хранящихся на нем. Эта информация включает число сегментов в справочнике тома, число файлов на томе, имена, даты создания и размер файлов. DIR позволяет организовать листинги справочника в алфавитном или хронологическом порядке.

3.2. Выполнение программы. Для вызова DIR с системного устройства следует подать с терминала команду

R DIR <BK>

после того, как монитор напечатает на терминале точку. После вызова DIR печатает звездочку и ожидает ввода командной строки.

Если в это время нажать клавишу <BK>, то DIR печатает номер своей версии.

3.3. Команды оператора. Режим работы программы DIR задается введением с терминала командной строки. В командной строке может быть указано только одно устройство ввода и одно устройство вывода и до шести имен входных файлов. Устройством вывода по умолчанию является терминал. Тип выходного файла по умолчанию — .DIR. Конструкция «*» (звездочка) допустима только во входной спецификации.

Для выхода из программы DIR и передачи управления монитору следует подать команду CY/C, если DIR ожидает ввода с терминала, или дважды CY/C, если DIR выполняет операцию.

Для повторного пуска DIR следует подать команду REENTER.

3.4. Переключатели DIR. Переключатели программы DIR (см. табл. 5) позволяют выполнять различные операции со справочниками устройств. Если в командной строке переключатель не указан, DIR выполняет операцию по переключателю /L.

Некоторые переключатели допускают в качестве своего аргумента дату. Синтаксис, определяющий дату, следующий:

[дд.][:ммм][:гг.]

где дд. — день (десятичное число от 1 до 31);

ммм — месяц (первые три буквы названия месяца на английском языке);

гг. — год (десятичное число от 73 до 99).

Если сгенерированная система поддерживает таймер, но не поддерживает автоматического изменения даты при смене месяца, то установить дату начала каждого месяца можно с помощью команды монитора DATE. Если начало месяца в дате не установлено, DIR печатает «—BAD—» в колонке даты создания файла, созданного позже конца месяца. (Установить текущую дату создания файла можно по команде монитора RENAME/SETDATE).

Обычно листинги справочников DIR печатает на терминале в две колонки. Читать такие листинги необходимо слева направо, всю строку сразу. Однако листинги справочников, которые упорядочены определенным образом (по переключателям /A, /R или /S), исключаются из этого правила. Такие листинги следует читать следующим образом: сначала левую колонку сверху вниз, затем аналогично правую.

Таблица 5

Переключатель	Функция
1	2
/A	Печатает имена файлов, содержащихся в справочнике тома в алфавитном порядке (аналогично /S:NAM)
/B	Печатает справочник тома, включая десятичные номера начальных блоков файлов. Для магнитной ленты номер начального блока — последовательный номер файла
/C:N	Печатает справочник в N колонок (N от 1 до 9). По умолчанию N=2 для обычного листинга и N=5 для краткого листинга
/D[:дата]	Печатает справочник файлов с указанной датой создания. Если дата не указана, DIR использует текущую дату
/E	Печатает справочник, включая неиспользуемые области тома и их размеры
/F	Печатает в пять колонок краткий справочник тома (только имена и типы файлов)
/G	Включает в листинг справочника указанный файл и все файлы, которые следуют за ним в справочнике тома
/J[:дата]	Печатает справочник файлов, созданных в течение или после указанной даты
/K[:дата]	Печатает справочник файлов, созданных до указанной даты
/L	Печатает справочник тома, включая имена файлов, их число, даты создания и размеры файлов
/M	Печатает справочник неиспользуемых областей тома
/N	Печатает информационный справочник тома

1	2
/O	Аналогичен переключателю /L, но печатает размеры и номера начальных блоков файлов в восьмеричной системе
/P	Печатает справочник тома, исключая указанные файлы
/Q	Печатает справочник файлов, которые были удалены, но справочная информация о которых не была разрушена
/R	Печатает справочник в обратном порядке по отношению к упорядоченному, определенному по /A или /S
/S[.XXX]	Печатает справочник тома в определенном порядке; аргумент XXX определяет порядок файлов в листинге справочника (см. табл. 6)
/T	Печатает справочник только защищенных файлов тома
/U	Печатает справочник только незащищенных файлов тома
/V[.ONL]	Печатает идентификатор тома и имя владельца как часть заголовка в листинге справочника. При использовании аргумента :ONL печатается только идентификатор тома и имя владельца

3.4.1 Переключатель /A используется для получения справочника тома в алфавитном порядке по имени файла (аналогично /S:NAM). Цифры сортируются вслед за буквами.

ПРИМЕР:

```

*DXC: /A
14-DEC-82
BUILD .MAC 102 06-SEP-82 SWAP .SYS 25 05-DEC-82
DX .SYS 3 06-SEP-82 SYSMAC.MAC 41 19-NOV-82
FMONSJ.SYS * 67P 19-NOV-82 TT .SYS 2 19-NOV-82
4 FILES, 238 BLOCKS
248 FREE BLOCKS

```

3.4.2. Переключатель /В используется для получения справочника тома, включающего десятичные номера начальных блоков файлов. Для магнитных лент номер начального блока представляет собой последовательный номер файла на ленте.

```
Пример:
*DX0:/В
14-DEC-82
FSM .MAC 31P 19-NOV-82 20 BATCH .MAC 102P 19-NOV-82 51
FLCOPY.MAC 8P 19-NOV-82 153 ELINIT.MAC 15P 14-OCT-82 161
SYSTBL.BL 4F 19-NOV-82 176 SYCND .DIS 5P 19-NOV-82 180
PCTAL .SAV 36 11-SEP-82 185 LAMP .SA 29 16-MAR-82 221
HUMPUS.SAV 30 16-MAR-82 250
7 FILES, 260 BLOCKS
226 FREE BLOCKS
```

В этом примере приведен справочник устройства DX0; числа, стоящие после даты создания файла, представляют собой номера начальных блоков.

3.4.3. Переключатель /C[:N] используется для организации справочника тома в N колонок (N — целое, от 1 до 9). Если он не указан в командной строке, DIR печатает обычный справочник в две колонки, краткий — в пять колонок.

Пример:

```
*DX1:/C:1
29-NOV-82
SWAR .SYS 25P 19-NOV-82
FMONSJ .SYS 67P 19-NOV-82
FMONFB .SYS 80P 19-NOV-82
TT .SYS 2P 19-NOV-82
LP .SYS 2P 19-NOV-82
5 FILES, 176 BLOCKS
310 FREE BLOCKS
```

В этом примере DIR печатает справочник DX1: в одну колонку.

3.4.4. Переключатель /D[:дата] используется для получения справочников файлов с указанной датой создания. Датой по умолчанию является текущая дата.

```
Пример:
*DX0:/D:13.:AUG:82.
15-SEP-82
FMONSJ.SYS 67P 13-AUG-82 FMONFB.SYS 80P 13-AUG-82
SWAP .SYS 25P 13-AUG-82 DX .SYS 3P 13-AUG-82
LP .SYS 2P 13-AUG-82 TT .SYS 2P 13-AUG-82
6 FILES, 179 BLOCKS
203 FREE BLOCKS
```

В этом примере приведена команда, по которой DIR печатает файлы устройства DX0: с датой создания 13 августа 1982 года.

3.4.5. Переключатель /E используется для получения справочника тома, включающего неиспользуемые области тома и их размеры в блоках.

```

Пример:
*DX1:/E
14-DEC-82
SWAP .SYS      25P 23-OCT-82      FMONSJ.SYS      67P 23-OCT-82
FMONFB.SYS     80P 19-NOV-82      DX .SYS         3P 19-NOV-82
RK .SYS        3P 19-NOV-82      LP .SYS        2P 19-NOV-82
TT .SYS        2P 23-OCT-82      NUM3 .LST      3 13-DEC-82
< UNUSED >     4                  NUMBER.PAS     1 13-DEC-82
DISMT1.COM     9P 27-NOV-82      < UNUSED >     287
10 FILES, 195 BLOCKS
291 FREE BLOCKS

```

В этом примере приведен справочник DX1:, включающий неиспользуемые области.

3.4.6. Переключатель /F используется для получения краткого справочника тома (включает только имена и типы файлов).

```

Пример:
*DX0:/F
16-AUG-82
DX .SYS PIP .SAV DIR .SAV DUP .SAV SWAP .SYS
FMONSJ.SYS FMONFB.SYS LP .SYS TT .SYS RK .SYS
10 FILES, 260 BLOCKS
226 FREE BLOCKS

```

В этом примере приведен справочник, содержащий имена и типы файлов устройства DX0:.

3.4.7. Переключатель /G включает в листинг справочника указанный файл и все файлы, которые следуют за ним в справочнике тома.

```

Пример:
*DX0:TT.SYS/G
10-DEC-82
TT .SYS      2 10-AUG-82      DIR .SAV      17 03-AUG-82
RK .SYS      3 13-AUG-82      EDIT .SAV     19 03-AUG-82
STARTS.COM   1 27-AUG-82      BINCOM.SAV    11 05-OCT-82
6 FILES, 53 BLOCKS
73 FREE BLOCKS

```

В этом примере приведена команда, по которой DIR печатает системный файл (TT.SYS) и все файлы, следующие за ним в справочнике DX0:.

3.4.8. Переключатель /J[:дата] используется для получения справочника файлов, созданных в течение или после указанной даты. Датой по умолчанию является текущая дата.

Пример:

```
*DX0:/J:10.:DEC:82.  
13—DEC—82  
AREL .MAC 2P 10—DEC—82  
MUR .MAC 5 11—DEC—82  
2 FILES, 7 BLOCKS  
401 FREE BLOCKS
```

В этом примере DIR включает в листинг справочника файлы, созданные 10 декабря 1982 г. и позже.

3.4.9. Переключатель /K[:дата] используется для получения справочника файлов, созданных до указанной даты. Датой по умолчанию является текущая дата.

Пример:

```
*DX1:/K:15.:MAR:83.  
13—APR—83  
FORTRA .SAV 191 14—MAR—83  
BASIC .SAV 51 25—FEB—83  
2 FILES, 242 BLOCKS  
38 FREE BLOCKS
```

В этом примере DIR включает в листинг справочника файлы, созданные до 15 марта 1983 г.

3.4.10. Переключатель /L используется для получения справочника тома, который включает имена файлов, даты их создания и размеры, количество файлов на томе, общее число используемых и свободных блоков тома (для дисков). Числовая информация указывается в десятичном исчислении.

Пример:

```
*LP:=DX1:/L  
03—DEC—82  
DIR .SAV 17P 01—DEC—82 DUP .SAV 41P 01—DEC—82  
ABC .SAV 4 02—DEC—82 A\F .MAC 2 01—DEC—82  
PIP .SAV 16P 02—DEC—82 MERGE .FOL 6 01—DEC—82  
6 FILES, 86 BLOCKS  
400 FREE BLOCKS
```

В этом примере приведена командная строка, по которой DIR печатает справочник DX1: на LP:.

3.4.11. Переключатель /M печатает справочник неиспользуемых областей тома и их размеры.

Пример:

*DX0:/M

14—DEC—82

<UNUSED>	11	<UNUSED>	2
<UNUSED>	26	<UNUSED>	32
<UNUSED>	1	<UNUSED>	25
<UNUSED>	0	<UNUSED>	65

0 FILES, 0 BLOCKS

162 FREE BLOCKS

В этом примере DIR печатает информацию о неиспользуемых областях на DX0:.

3.4.12. Переключатель /N используется для получения информационного справочника тома. Такой справочник включает число сегментов в справочнике тома, число используемых сегментов, число файлов в каждом сегменте, общее число файлов на томе, и общее число используемых и свободных блоков тома.

Пример:

*/N

14—NOV—82

44 FILES IN SEGMENT 1

37 FILES IN SEGMENT 2

4 AVAILABLE SEGMENTS, 2 IN USE

81 FILES, 364 BLOCKS

122 FREE BLOCKS

В этом примере приведен информационный справочник устройства DK:.

3.4.13. Переключатель /O используется для печати размеров и номеров начальных блоков файлов на томе (если указан переключатель /B) в восьмеричном исчислении.

Пример:

- *DX1:/O

10-DEC-82 OCTAL

SWAP .SYS	31	05-SEP-82	ANTON .MAC	4	19-NOV-82
MONSJ .SYS	103P	19-NOV-82	TT .SYS	2P	19-NOV-82
DX .SYS	3P	29-AUG-82	BUILD .MAC	144	19-NOV-82

6 FILES, 311 BLOCKS

435 FREE BLOCKS

В этом примере приведен справочник с восьмеричными размерами файлов, находящихся на DX1:.

3.4.14. Переключатель /P используется для получения справочника файлов тома, кроме тех, которые указаны в командной строке.

Пример:

```
*DX1:*.SAV/P
03—DEC—82
ABCD .MAC 4 01—DEC—82 KUF .MAC 2 01—DEC—82
BIM .MAC 7 14—NOV—82
3 FILES, 13 BLOCKS
60 FREE BLOCKS
```

В этом примере DIR печатает справочник файлов устройства DX1:, кроме файлов, имеющих тип .SAV.

3.4.15. Переключатель /Q используется для получения справочника удаленных файлов, справочная информация о которых не была разрушена. Такой справочник может быть полезен для восстановления случайно удаленных файлов (см. раздел 2).

Пример:

```
*DX1:/Q
13—DEC—82
AB .MAC 2 01-DEC-82 15 KUF1 .MAC 4 01-DEC-82 104
BIMY .MAC 11 01-DEC-82 138
0 FILES, 0 BLOCKS
120 FREE BLOCKS
```

3.4.16. Переключатель /R используется для получения справочника тома, организованного в обратном порядке по отношению к упорядоченному, определенному по /A или /S.

Пример:

```
*DX:/S:DAT/R
13—DEC—82
KUBE .MAC 4 10-DEC-82 CHIS .MAC 7P 02-DEC-82
LOO .MAC 14P 07-DEC-82 SEEK .MAC 23 01-DEC-82
4 FILES, 48 BLOCKS
438 FREE BLOCKS
```

В этом примере DIR печатает справочник DX: в обратном хронологическом порядке.

3.4.17. Переключатель /S[:XXX] используется для получения упорядоченного справочника тома согласно определенному пользователем трехсимвольному коду XXX (см. табл. 6).

Таблица 6

Значение кода	Функция
1	2
DAT	Организует справочник по датам создания файлов (по хронологии). Файлы, имеющие одну и ту же дату создания, располагаются в алфавитном порядке по имени и типу файла
NAM	Организует справочник в алфавитном порядке по имени файлов. Файлы, имеющие одно и то же имя, располагаются в алфавитном порядке по типу файла (аналогично /A)
POS	Организует справочник файлов в порядке их расположения на томе (аналогично /S без кода)
SIZ	Организует справочник по длине файлов. Файлы, имеющие одну и ту же длину, располагаются в алфавитном порядке по имени и типу файлов
TYP	Организует справочник в алфавитном порядке по типам файлов. Файлы с одним и тем же типом располагаются в алфавитном порядке по именам

Примеры:

1. *DX0:/S:DAT

14—DEC—82

BUILD .MAC 100 06-SEP-82 SYSMAC .MAC 41 19-NOV-82

DX .SYS 3 06-SEP-82 TT .SYS 2 19-NOV-82

FMONSJ .SYS 67P 19-NOV-82 SWAP .SYS 25 05-DEC-82

6 FILES, 238 BLOCKS

248 FREE BLOCKS

2. *DX0:/S:NAM

14—DEC—82

BUILD .MAC 100 06-SEP-82 SWAP .SYS 25 05-DEC-82

DX .SYS 3 06-SEP-82 SYSMAC .MAC 41 19-NOV-82

FMONSJ .SYS 67P 19-NOV-82 TT .SYS 2 19-NOV-82

6 FILES, 238 BLOCKS

248 FREE BLOCKS

3. *DX0:/S:POS
14—DEC—82

FMONSJ .SYS 67P 19-NOV-82 BUILD .MAC 100 06-SEP-82
DX .SYS 3 06-SEP-82 SYSMAC .MAC 41 19-NOV-82
SWAP .SYS 25 05-DEC-82 TT .SYS 2 19-NOV-82
6 FILES, 238 BLOCKS
248 FREE BLOCKS

4. *DX0:S/:TYP
14—DEC—82

BUILD .MAC 100 06-SEP-82 FMONSJ .SYS 67P 19-NOV-82
SYSMAC .MAC 41 19-NOV-82 SWAP .SYS 25 05-DEC-82
DX .SYS 3 06-SEP-82 TT .SYS 2 19-NOV-82
6 FILES, 238 BLOCKS
248 FREE BLOCKS

5. *DX0:/S:SIZ
14—DEC—82

TT .SYS 2 19-NOV-82 SYSMAC .MAC 41 19-NOV-82
DX .SYS 3 06-SEP-82 FMONSJ .SYS 67P 19-NOV-82
SWAP .SYS 25 05-DEC-82 BUILD .MAC 100 06-SEP-82
6 FILES, 238 BLOCKS
248 FREE BLOCKS

В этих примерах используется переключатель /S.

3.4.18. Переключатель /T используется для получения справочника тома, включающего только защищенные файлы. Буква «Р», следующая за размером файла в справочнике тома, служит признаком защищенного файла.

Пример:

*DK:/S:SIZ/R/T
5—JAN—83

BUILD .MAC 100P 06-SEP-82 FMONSJ .SYS 67P 19-NOV-82
SWAP .SYS 25P 05-DEC-82 DX .SYS 3P 06-SEP-82
TT .SYS 2P 19-NOV-82
5 FILES, 197 BLOCKS
182 FREE BLOCKS

В этом примере распечатаны все защищенные файлы устройства DK:.

3.4.19. Переключатель /U используется для получения справочника тома, включающего только незащищенные файлы.

Пример:

* /S: SIZ/R/U

14—DEC—82

SHARM .MAC 80 11-DEC-82 CORN .MAC 9 13-DEC-82

CORN .OBJ 7 13-DEC-82

3 FILES, 96 BLOCKS

52 FREE BLOCKS

В этом примере распечатаны незащищенные файлы системного устройства.

3.4.20. Переключатель /V[:ONL] печатает идентификатор тома и имя владельца как часть заголовка в листинге справочника. При использовании аргумента :ONL печатается только идентификатор тома и имя владельца. Возможны комбинации переключателя /V с другими переключателями.

Примеры:

*DX:/V

14—DEC—82

VOLUME ID: FODOS

OWNER : PETROW

SWAP .SYS 25P 19-NOV-82 FMONSJ. SYS 67P 19-NOV-82

FMONFB. SYS 80P 19-NOV-82 DX .SYS 3P 19-NOV-82

TT .SYS 2P 19-NOV-82 DY .SYS 4P 19-NOV-82

RK .SYS 3P 19-NOV-82

7 FILES, 184 BLOCKS

302 FREE BLOCKS

*DX0:/V:ONL

VOLUME ID: FODOS

OWNER : DOEV

В первом случае используется /V; во втором — вместе с аргументом :ONL.

3.5. Сообщения оператору

Ниже описаны сообщения, выдаваемые программой DIR оператору.

?DIR—F—DEVICE NOT ACTIVE

Причина. Операция ввода—вывода невозможна на указанном устройстве или томе; устройство отключено; нет тома на устройстве; запрещена запись; устройство не описано в таблицах монитора.

Действие. Проверить правильность включения устройства. Установить обслуживание устройства по команде монитора INSTALL.

?DIR—F—ERROR READING DIRECTORY

Причина. Ошибка чтения справочника тома.

- Действие. Проверить исправность и правильность включения аппаратуры. Проверить диск на плохие блоки.
 ?DIR—F—INPUT ERROR DEV:
- Причина. Произошла аппаратная ошибка при чтении входного файла.
- Действие. Проверить исправность и правильность включения аппаратуры. Проверить том на плохие блоки.
 ?DIR—F—INSUFFICIENT MEMORY
- Причина. Недостаточно памяти для получения упорядоченного справочника.
- Действие. Использовать другой переключатель.
 ?DIR—F—INVALID COMMAND
- Причина. В командной строке обнаружена ошибка.
- Действие. Ввести правильную командную строку.
 ?DIR—F—INVALID DEVICE DEV:
- Причина. Операция недопустима на указанном устройстве: устройство несправочной структуры (как, например, TT: или LP:) или в системе не установлено обслуживание указанного устройства.
- Действие. Проверить введенную командную строку. Использовать команду монитора SHOW для получения списка устройств, обслуживаемых системой.
 ?DIR—F—INVALID DIRECTORY
- Причина. Структура справочника тома устройства (метки тома для магнитной ленты) не являются структурой ФОДОС-2.
- Действие. Не требуется.
 ?DIR—F—INVALID OPTION:/X
- Причина. В командной строке обнаружен недопустимый переключатель или комбинация переключателей.
- Действие. Ввести правильную командную строку.
 ?DIR—F—OUTPUT ERROR DEV:FILNAM.TYP
- Причина. Произошла аппаратная ошибка во время записи выходного файла.
- Действие. См. ?DIR—F—INPUT ERROR DEV:FILNAM.TYP.
 ?DIR—F—OUTPUT FILE FULL
- Причина. Выходной файл заполнен: на томе нет места для файла справочника.
- Действие. Вывести справочник на печатающее устройство. Использовать конструкцию [N] для выходного файла.
 ?DIR—F—PROTECED FILE ALREADY EXISTS
 DEV:FILNAM.TYP

Причина. Попытка создания защищенного файла, имеющего такое же имя, что и у уже существующего защищенного файла.

Действие. Воспользоваться командой монитора **RENAME/NOPROTECT** или переключателем **PIP /Z** для снятия защиты уже существующего файла или воспользоваться другим именем для создания нового файла.

?DIR—I—BACKUP FILE—USE/BACKUP OPTION

Причина. Файл, указанный в командной строке, был создан с помощью программы **BUP** (см. р. 4).

Действие. Использовать команду монитора **DIRECTORY/BACK**

4. ПРОГРАММА СОЗДАНИЯ КОПИЙ BUP

4.1. Назначение программы и условия выполнения программы

Программа создания копий (**BUP**) предназначена для специализированной передачи файлов с целью хранения больших файлов или томов. Программа **BUP** позволяет копировать файл или том на несколько томов, размер которых меньше, чем размер входного файла или тома. Программа **BUP** выполняет также и обратную операцию восстановления фрагментированного файла или тома в его первоначальную форму на отдельном большом томе. Восстановленный файл или том можно снова использовать, в отличие от фрагментированного. Поэтому программа **BUP** наиболее полезна для резервирования информации, которую необходимо сохранить.

4.2. Выполнение программы. Для вызова **BUP** с системного устройства следует подать с терминала команду:

R BUP<BK>

после того, как монитор напечатает на терминале точку. После вызова **BUP** печатает звездочку (*) и ожидает ввода командной строки. Если в это время нажать клавишу **<BK>**, то **BUP** печатает номер своей версии.

4.3. Команды оператора. Режим работы программы **BUP** задается введением с терминала командной строки.

Формат командной строки следующий:

выхспф=вхспф/прк[/прк]

где **выхспф** — спецификация выходного файла (устройство, на которое устанавливаются тома для получения копий входного тома, имя и тип файла); если имя выходного файла не указано, BUP использует имя входного файла; тип выходного файла по умолчанию — .BUP; конструкция «*» недопустима в спецификации выходного файла;

вхспф — спецификация входного файла (устройство, на котором установлен том-оригинал для создания томов-копий, имя и тип файла); конструкция «*» недопустима в спецификации входного файла;

прк — переключатель программы BUP (см. табл. 7).

Необходимо обязательно указывать имя и тип входного файла. Устройством по умолчанию является ДК:. Конфигурация «*» игнорируется при инициализации или при получении справочника тома-копии. Использование конструкции «*» во входной и выходной спецификации файла недопустимо для операций копирования и восстановления.

Магнитные ленты могут использоваться лишь как выходные тома при операции копирования или лишь как входные тома при операции восстановления.

Для выхода из программы BUP и передачи управления монитору следует подать команду **СУ/С**, если BUP ожидает ввода с терминала, или дважды **СУ/С**, если BUP выполняет операцию.

Для повторного пуска BUP следует подать команду **REENTER**.

4.4. Переключатели программы BUP (табл. 7) позволяют выполнять самые различные операции на периферийных устройствах операционной системы. Если в командной строке переключатель не указан, BUP копирует файл или том на несколько меньших томов.

Таблица 7

Переключатель	Функция
1	2
/I	Копирует том или файл на несколько меньших по размеру томов в режиме отображения. С переключателем /X используется для восстановления тома
/L	Печатает листинг справочника тома-копии.
/X	Использование его с другими переключателями недопустимо. Восстанавливает файл, который был скопирован по переключателю /I на несколько томов-копий или используется с переключателем /I для восстановления тома
/Y	Используется с переключателем /Z для запрещения печати опрашивающих сообщений и немедленного выполнения операции
/Z	Инициализирует указанный том для использования его в качестве выходного тома в операции копирования по BUP. Используется только с переключателем /Y

4.4.1. Переключатель /I используется для создания копий тома в режиме отображения. Сначала необходимо инициализировать выходной том (см. п. 1.4.5), используя переключатель /Z (недопустимо для магнитных лент), после чего BUP будет воспринимать выходной том как том-копию, не содержащий плохих блоков.

Формат командной строки следующий:

выходспф=входуст:/I

где выходспф — спецификация выходного файла (устройство, на котором установлен том-копия, имя и тип файла); конструкция «*» недопустима в выходной спецификации;

входуст: — устройство и номер привода, на котором установлен том-оригинал.

Имя выходного файла обязательно как для файлов, так и для томов, если имя файла в выходной спецификации отсутствует, BUP использует двухсимвольную мнемонику входного устройства в качестве имени файла (например, «DX» для гибких дисков с одинарной плотностью записи). Тип выходного файла по умолчанию — .BUP.

Пример:

*MTO:=DX1:/I

В этом примере на магнитной ленте создается копия тома DX1: с именем выходного файла DX1.BUP.

BUP копирует входной том на первый выходной том до его заполнения. Перед началом создания первого тома-копии BUP сообщает:

?BUP—I—CREATING VOLUME 1

Если весь входной том умещается на одном выходном томе, BUP печатает сообщение об ошибке, если только выходной том не является магнитной лентой:

?BUP—F—ENOUGH SPACE ON ONE VOLUME — —
USE PIP

Когда первый выходной том будет заполнен, BUP печатает:

MOUNT OUTPUT VOLUME IN <УСТ>;CONTINUE?

Для продолжения операции необходимо установить новый выходной том на указанное устройство <УСТ>, напечатать «Y» и нажать клавишу <BK>, после чего BUP сообщает номер создаваемой копии:

?BUP—I—CREATING VOLUME N

BUP повторяет этот процесс до тех пор, пока весь входной том не будет скопирован. Датой создания выходного файла, записанной в справочнике тома-копии, является текущая дата. Если дата не была установлена, она отсутствует в справочнике.

Если установлен выходной том, который не был предварительно инициализирован с помощью переключателя /Z, BUP печатает следующее сообщение:

?BUP—W—NOT A BACKUP VOLUME DEV:
<УСТ:>/BUP INITIALIZE; ARE YOU SURE?

Для инициализации тома необходимо напечатать «Y» и нажать клавишу <BK>. Если инициализация нежелательна, необходимо заменить выходной том другим, и, если, он

уже инициализирован, то достаточно нажать клавишу <BK>.

Если в качестве выходного тома используется том-копия не системы ФОДОС-2 или том, который уже содержит файлы, BUP печатает:

VOLUME NOT FODOS FORMAT. ARE YOU SURE?

или

VOLUME CONTAINS FILES. ARE YOU SURE?

Для инициализации тома необходимо напечатать «Y» и нажать клавишу <BK>. BUP автоматически выполняет операцию копирования, как только инициализация завершится.

Если же нажать клавишу <BK> (или любой другой ответ), то BUP печатает:

MOUNT OUTPUT VOLUME IN <УСТ>; CONTINUE?

Для продолжения операции необходимо на устройстве <УСТ> установить выходной том, предварительно инициализированный с помощью переключателя /Z, напечатать «Y» и нажать клавишу <BK>.

Пример:

*DX:=DY1:/I

В этом примере том с устройства DY1: копируется на несколько томов меньшего размера на DX:. Каждый том-копия будет содержать выходной файл DY1.BUP.

4.4.2. Переключатель /L печатает справочник указанного тома-копии. Формат командной строки следующий:

уст:/L

где уст: — том, справочник которого необходимо получить.

Справочник устройств произвольного доступа к данным начинается с текущей даты и номер указанного тома-копии. Номер тома показывает место тома в наборе томов, содержащих копию отдельного тома или файла.

Далее в четырех колонках содержится информация о каждом томе-копии набора. Первая колонка перечисляет номера томов-копий. Вторая — имя и тип файла, часть которого содержится на этом томе. Третья колонка сообщает число блоков на каждом томе. И последняя колонка содержит дату копирования файла.

Далее в справочнике BUP печатает число свободных блоков на указанном томе.

Так как информация справочника определяется в самом начале операции копирования, то может печататься инфор-

мация даже тогда, когда операция копирования не закончена.

Пример:

```
*DY0:/L
23—JAN—83
VOLUME 3 OF 4
VOLUME  FILENAME      BLOCKS      DATE
V1       COPFIL.BUP    980         21—JAN—83
V2       COPFIL.BUP    980         21—JAN—83
V3       COPFIL.BUP    980         21—JAN—83
V4       COPFIL.BUP    400         21—JAN—83
1 FILE, 980 BLOCKS
0 FREE BLOCKS
```

В этом примере приведен справочник третьего тома-копии из набора четырех томов, содержащих файл COPFIL.TXT.

Для магнитных лент вид справочника такой же. Однако, воспроизводится только текущая дата и информация указанной магнитной ленты. Третья колонка перечисляет общее число блоков, использованных в наборе магнитных лент, содержащих копии файла или тома.

Пример:

```
*MT1:/L
23—JAN—83
VOLUME      FILENAME      BLOCKS      DATE
V1          DY1 .BUP      950 23—JAN—83
```

В этом примере приведен справочник магнитной ленты, содержащей копию DY1:.

4.4.3. Переключатель /X восстанавливает файл или том из нескольких томов-копий отдельного файла или тома в формате системы ФОДОС-2.

Формат командной строки для восстановления файла следующий:

выходспф=входспф/X

где выходспф — спецификация выходного файла (устройство, имя и тип файла), который будет восстановлен из томов-копий; если имя и тип выходного файла не указаны, BUB использует имя и тип входного файла; конструкция «*» недопустима в выходной спецификации;

входспф — спецификация входного файла (устройство, имя и тип файла), который будет восста-

навливаться; конструкция «*» недопустима.

Формат командной строки для восстановления тома следующий:

выходуст: = входспф/I/X

где выходуст: — выходной том, на который копируются тома-копии с целью восстановления; конструкция «*» недопустима в выходной спецификации;

входспф — входной том (и произвольное имя файла, под которым он записан) для восстановления; конструкция «*» недопустима во входной спецификации.

BUP начинает восстановление и печатает сообщение:

MOUNT INPUT VOLUME N IN <УСТ>;CONTINUE?

Для продолжения операции необходимо установить том-копию номер 1 из набора томов, содержащих том или файл, напечатать «Y» и нажать клавишу <BK>. После чего BUP печатает:

?BUP—I—RESTORE OPERATION STARTED FROM VOLUME N

Когда том-копия 1 полностью скопирован на выходной том, BUP печатает вновь первое сообщение и повторяет этот процесс до тех пор, пока весь набор томов-копий, содержащих файл-оригинал (или том-оригинал), не будет скопирован на отдельный том.

Если установлен том-копия с неправильным номером, BUP печатает:

?BUP—E—WRONG VOLUME NUMBER

MOUNT INPUT VOLUME N IN <УСТ>;CONTINUE?

Для продолжения операции необходимо установить том-копию номер N из набора томов, напечатать «Y» и нажать <BK>.

Если установлен том-копия с правильным номером, но с ошибочным именем файла, BUP печатает:

?BUP—W—FILE NOT FOUND DEV:FILNAM.TYP

MOUNT INPUT VOLUME NUMBER N IN <УСТ>CONTINUE?

Для продолжения операции необходимо установить том-копию номер N из набора томов, содержащих том-оригинал (или файл-оригинал), напечатать «Y» и нажать клавишу <BK>.

Если напечатать «N» и нажать клавишу <BK>, операция будет прервана. При любом другом ответе BUP повтор-но печатает:

MOUNT INPUT VOLUME NUMBER N IN <УСТ>CONTINUE?

ожидает ответ на запрос.

Примеры:

1. *DY1:=DX1:REFIL.DAT/X

MOUNT INPUT VOLUME 1 IN DX1;CONTINUE?Y
BUP—I—RESTORE OPERATION STARTED FROM
VOLUME 1

MOUNT INPUT VOLUME 2 IN DX1;CONTINUE?Y
BUP—I—RESTORE OPERATION STARTED FROM
VOLUME 2

MOUNT INPUT VOLUME 3 IN DX1;CONTINUE?Y
BUP—I—RESTORE OPERATION STARTED FROM
VOLUME 3

BUP—I—COPY OPERATION IS COMPLETE

В этом примере файл REFIL.DAT с трех томов-копий DX1: восстанавливается на томе DY1:.

2. *DY1:=DX0:/X/I

По этой команде восстанавливается том DY1: с нескольких томов-копий DX0:.

4.4.4. Переключатель /Z используется для инициализации любого тома перед использованием этого тома в качестве выходного в операции копирования. (Переключатель /Z недопустим для магнитных лент, BUP инициализирует магнитную ленту во время операции копирования). По переключателю /Z очищается справочник тома, записывается информация в блок номер 1 (блок номер 0 для магнитных лент). Кроме того, том сканируется на плохие блоки (том-копия не должен содержать плохих блоков).

Формат командной строки следующий:

уст:/Z

где уст: — том, который требуется инициализировать.

Далее печатается запрос на подтверждение выполнения операции инициализации. Для выполнения операции необходимо напечатать «Y» и нажать клавишу <BK>. Для прекращения операции необходимо нажать клавишу <BK> или СУ/С. В последнем случае управление передается клавиатурному монитору.

Для запрещения печати запросов на подтверждение выполнения операции с переключателем /Z используется переключатель /Y.

Если на томе обнаружены плохие блоки, BUP печатает:
?BUP—I—BAD BLOCKS DETECTED; USE ANOTHER
VOLUME

необходимо установить другой том и инициализировать его.

Если в справочнике тома, который необходимо инициализировать, обнаружены файлы, BUP печатает:

VOLUME CONTAINS FILES. ARE YOU SURE?

Для инициализации этого тома необходимо напечатать «Y» и нажать клавишу <BK>. Любой другой ответ прерывает операцию.

Чтобы том, инициализированный с помощью программы BUP (используя переключатель /Z), можно было использовать в системе ФОДОС-2, его необходимо инициализировать, используя команду монитора INITIALIZE или переключатель /Z программы DUP (см. п. 2.3.14).

4.5. Сообщения оператору

Ниже приведены сообщения, выдаваемые программой BUP.

?BUP—E—WRONG VOLUME NUMBER

Причина. Во время операции восстановления установлен том-копия, номер которого ошибочен.

Действие. Установить соответствующий номер тома-копии. Повторить операцию.

?BUP—F—CHANNEL IN USE DEV:FILNAM.TYP

Причина. Ошибка при обращении к программе BUP.

Действие. Перезагрузить систему и повторить операцию. Если ошибка повторится, создать новую копию BUP.SAV и повторить операцию.

?BUP—F—DEVICE IN USE

Причина. Указанное устройство используется другим заданием.

Действие. Прервать выполнение задания или дождаться его завершения. Повторить операцию.

?BUP—F—DIRECTORY INPUT ERROR DEV:

Причина. Аппаратная ошибка во время чтения справочника.

Действие.. Проверить исправность и правильность включения аппаратуры. Проверить том на плохие блоки. Повторить операцию. Если ошибка повторяется, скопировать максимально возможное число файлов на другой том. Отформатировать том и повторить опе-

рацию. Если ошибка повторяется снова, заменить том другим.

?BUP—F—DIRECTORY OUTPUT ERROR DEV:

Причина. Аппаратная ошибка во время записи справочника.

Действие. См. ?BUP—F—DIRECTORY INPUT ERROR DEV:.

?BUP—F—DUPLICATE OPTION

Причина. Один и тот же переключатель указан в командной строке более одного раза.

Действие. Ввести правильную командную строку.

**?BUP—F—ENOUGH SPACE ON ONE VOLUME—
USE PIP**

Причина. Входные данные могут разместиться на одном выходном томе.

Действие. Необходимо использовать программу PIP (см. гл. 1).

?BUP—F—ERROR READING DIRECTORY

Причина. См. ?BUP—F—DIRECTORY INPUT ERROR DEV:.

Действие. См. ?BUP—F—DIRECTORY INPUT ERROR DEV:.

?BUP—F—FETCH ERROR DEV:

Причина. Ошибка при обращении к программе BUP.

Действие. Перезагрузить систему и повторить операцию. Если ошибка повторится, создать новую копию BUP.SAV и повторить операцию.

?BUP—F—FILE NOT FOUND DEV:FILNAM. TYP

Причина. На томе-копии не существует входного тома, указанного в командной строке. Управление передано монитору.

Действие. Заменить том-копию другим, на котором находится указанный входной файл. Повторить операцию.

?BUP—F—INPUT ERROR DEV:FILNAM.TYP

Причина. Аппаратная ошибка во время операции чтения.

Действие. См. ?BUP—F—DIRECTORY INPUT ERROR DEV:.

?BUP—F—INSUFFICIENT MEMORY

Причина. Недостаточно памяти, необходимой для выполнения требуемой операции.

Действие. Удалить ненужные драйверы устройств. Удалить основное задание. Использовать монитор SJ. Разрешить свопинг USR по команде монитора SET USR SWAP.

?BUP—F—INVALID COMMAND

Причина. В командной строке обнаружена ошибка.

Действие. Ввести правильную командную строку.

?BUP—F—INVALID DEVICE DEV:

Причина. В командной строке указано недопустимое или не-
существующее устройство.

Действие. Проверить введенную командную строку. Использо-
вать команду монитора SHOW для получения
списка устройств, обслуживаемых системой.

?BUP—F—INVALID DEVICE COMBINATION

Причина. Одно и то же устройство и один и тот же привод
указаны во входной и в выходной спецификациях.

Действие. Проверить командную строку.

?BUP—F—INVALID DIRECTORY

Причина. Справочник указанного устройства не структуры
ФОДОС-2.

Действие. Инициализировать том перед использованием по
команде монитора INITIALIZE.

?BUP—F—INVALID OPTION

Причина. В командной строке указан недопустимый пере-
ключатель.

Действие. Ввести правильную командную строку.

?BUP—F—INVALID OPTION COMBINATION

Причина. В командной строке обнаружена недопустимая
комбинация переключателей.

Действие. Ввести правильную командную строку.

?BUP—F—NO ROOM ON OUTPUT DEVICE DEV:

Причина. Недостаточно места на выходном томе для завер-
шения операции восстановления.

Действие. Заменить выходной том другим, с большим числом
свободных блоков. Повторить операцию.

?BUP—F—NOT A BACKUP VOLUME DEV:

Причина. Попытка получения с помощью переключателя /L
листинга справочника тома, который не является
томом-копией.

Действие. Заменить том на нужную том-копию. Повторить
операцию.

?BUP—F—OPERATION NOT COMPLETED

Причина. Прервано выполнение операции программы BUP,
т. к. ответ на запрос начинался с «N» или нажато
СУ/С.

Действие. Для продолжения операции на тот же запрос не-
обходимо напечатать «Y» и нажать клавишу
<BK>.

?BUP—F—OUTPUT ERROR DEV: FILNAM.TYP

Причина. Аппаратная ошибка во время записи файла.

Действие. См. ?BUP—F—DIRECTORY INPUT ERROR DEV.:
?BUP—F—TOO MANY BAD BLOCKS DEV:

Причина. Во время операции восстановления или операции копирования с диска на диск было обнаружено более 25 плохих блоков.

Действие. Проверить, правильно ли установлен диск. Если правильно, то заменить его другим.

?BUP—F—VOLUME DOES NOT BELONG TO THIS SET

Причина. Во время операции восстановления установлен том-копия с именем файла и номером тома-копии, соответствующими указанному входному файлу. Однако число блоков, находящихся на установленном томе-копии, не соответствует числу блоков тома, установленного для восстановления (из другого набора томов-копий).

Действие. Использовать том-копию, являющийся частью данного набора томов-копий для восстановления.

?BUP—F—WILDCARDS NOT PERMITTED ON COMMAND LINE

Причина. В командной строке присутствует недопустимая конструкция «*».

Действие. Указать в командной строке имя файла вместо конструкции «*».

?BUP—F—WRONG VERSION

Причина. Попытка вызова программы BUP системой ФОДОС.

Действие. Исключить такую попытку. Работать с программой BUP можно только в системе ФОДОС-2.

?BUP—I—BAD BLOCK SCAN STARTED...

Причина. Программа BUP начала сканирование тома на плохие блоки.

Действие. Информационное сообщение. Действий не требуется.

?BUP—I—BAD BLOCKS DETECTED; USE ANOTHER VOLUME

Причина. Программа BUP обнаружила плохие блоки на томе.

Действие. Заменить том другим и повторить операцию.

?BUP—I—COPY OPERATION IS COMPLETE

Причина. Программа BUP закончила копирование или восстановление файла (или тома).

Действие. Информационное сообщение. Действий не требуется.

?BUP—I—CREATING VOLUME N

Причина. Программа BUP копирует информацию на указанный том-копию номер N.

Действие. Информационное сообщение. Действий не требуется.

?BUP—I—NO BAD BLOCKS DETECTED

Причина. На указанном томе плохих блоков не обнаружено.

Действие. Информационное сообщение. Действий не требуется.

?BUP—I—RESTORE OPERATION STARTED FROM VOLUME N

Причина. Начинается операция восстановления с тома-копии номер N.

Действие. Информационное сообщение. Действий не требуется.

?BUP—W—BAD BLOCK READ AT NNNNNN

Причина. Во время операции восстановления или операции копирования с диска на диск во входном томе обнаружен плохой блок номер NNNNNN.

Действие. Информационное сообщение. Действий не требуется.

?BUP—W—FILE NOT FOUND DEV:FILNAM.TYP

Причина. Во время операции восстановления на установленном входном томе-копии не обнаружен указанный в командной строке входной файл. Был использован том-копия из другого набора томов-копий или повторно установлен один из томов-копий.

Действие. Установить соответствующий том-копию (содержащий указанный вх.файл и указанный номер тома-копии).

?BUP—W—INCOPATIBLE OUTPUT VOLUME*SIZE

Причина. На выходном устройстве установлен том, размер которого отличен от размера первого используемого в операции копирования тома-копии. Например, это сообщение появляется, если на выходном устройстве во время операции копирования были последовательно установлены гибкие диски как с одинарной, так и с двойной плотностью.

Действие. Использовать тома того же размера, что и первый том-копия. Повторить операцию.

?BUP—W—NOT A BACKUP VOLUME DEV:

Причина. Установленный том не инициализирован для использования его в операциях копирования или восстановления с помощью программы BUP.

Действие. В ответ на запрос напечатать «Y» и нажать клавишу <BK> для инициализации тома.

ПЕРЕЧЕНЬ ССЫЛОЧНЫХ ДОКУМЕНТОВ

1. Операционная система ФОДОС-2
Описание применения.
2. Операционная система ФОДОС-2
Командный язык системы.

Для заметок

Для заметок

СОДЕРЖАНИЕ

КОМАНДНЫЙ ЯЗЫК СИСТЕМЫ

	Стр.
1. Назначение командного языка системы (КЯС)	3
2. Средства командного языка системы	3
2.1. Формат команды	4
2.2. Использование конструкций * и % в командах КЯС	5
2.3. Ввод с терминала и редактирование командной строки	7
2.3.1. Действие клавиши <СЛУЖЕБНАЯ 1>	9
2.3.2. Действие клавиши <СПРАВОЧНАЯ>	10
2.3.3. Перемещение курсора	10
2.3.4. Действие клавиши <СТРЕЛКА ВВЕРХ>	11
2.3.5. Действие клавиши <СЛУЖЕБНАЯ 2>	11
2.3.6. Действие клавиш <СЛУЖЕБНАЯ 1> и <СЛУЖЕБНАЯ 2> или <СЛУЖЕБНАЯ 1> и СУ/У	11
2.3.7. Действие клавиши <ЗБ>	12
2.3.8. Действие комбинации клавиши <СУ> и <Н>	12
2.3.9. Действие комбинации клавиш <СУ> и <У>	12
2.3.10. Действие клавиши <СЛУЖЕБНАЯ 1> и <ВК>	12
2.3.11. Действие клавиши <ВК>	13
2.3.12. Действие комбинаций клавиш <СУ> и <R>, <СУ> и <W>	13
2.4. Косвенные файлы	13
2.4.1. Создание косвенных файлов	13
2.4.2. Обработка косвенных файлов	14
3. Пуск системы	17
4. Описание команд КЯС	17
4.1. Команда ABORT	18
4.2. Команда ASSIGN	18
4.3. Команда B	19
4.4. Команда BACKUP	20
4.5. Команда BASIC	23
4.6. Команда BOOT	23
4.7. Команда CLOSE	25
4.8. Команда COMPILE	25
4.9. Команда COPY	34
4.10. Команда CREATE	48
4.11. Команда D	50
4.12. Команда DATE	51
4.13. Команда DEASSIGN	51
4.14. Команда DELETE	52
4.15. Команда DIBOL	56
4.16. Команда DIFFERENCES	58
4.17. Команда DIRECTORY	63
4.18. Команда DISMOUNT	72

4.19.	Команда DUMP	72
4.20.	Команда E	74
4.21.	Команда EDIT	75
4.22.	Команда EXECUTE	77
4.23.	Команда FORMAT	86
4.24.	Команда FORTRAN	89
4.25.	Команда FRUN	92
4.26.	Команда GET	95
4.27.	Команда INITIALIZE	96
4.28.	Команда INSTALL	100
4.29.	Команда LIBRARY	101
4.30.	Команда LINK	107
4.31.	Команда LOAD	114
4.32.	Команда MACRO	115
4.33.	Команда MOUNT	118
4.34.	Команда PRINT	119
4.35.	Команда PROTECT	125
4.36.	Команда R	130
4.37.	Команда REENTER	130
4.38.	Команда REMOVE	131
4.39.	Команда RENAME	131
4.40.	Команда RESET	135
4.41.	Команда RESUME	136
4.42.	Команда RUN	136
4.43.	Команда SAVE	137
4.44.	Команда SET	138
4.45.	Команда SHOW	152
4.46.	Команда SQUEEZE	160
4.47.	Команда SRUN	162
4.48.	Команда START	163
4.49.	Команда SUSPEND	164
4.50.	Команда TIME	164
4.51.	Команда TYPE	165
4.52.	Команда UNLOAD	168
4.53.	Команда UNPROTECT	169
5.	Описание ККАС	173
6.	Процессор косвенных управляющих файлов (IND)	174
6.1.	Создание косвенного управляющего файла	174
6.1.1.	Метки	175
6.1.2.	Директивы программы IND	175
6.1.3.	Команды	178
6.1.4.	Комментарии	179
6.2.	Пуск программы	179
6.2.1.	Команды оператора	180
6.2.2.	Переключатель /D	181
6.2.3.	Переключатель /N	181
6.2.4.	Переключатель /Q	181
6.2.5.	Переключатель /T	181
6.2.6.	Параметры	181
6.2.7.	Вложенные косвенные управляющие файлы	182
6.3.	Вызов косвенных командных файлов из управляющих файлов	182
6.4.	Специальные символы	183
6.5.	Имена	183
6.5.1.	Локальные и глобальные имена	183

6.5.2.	Логические имена	184
6.5.3.	Числовые имена	184
6.5.3.1.	Определение системы счисления	184
6.5.3.2.	Цифровые выражения	184
6.5.4.	Строковые имена	185
6.6.	Операционные режимы	186
6.7.	Специальные имена	188
6.8.	Замена	192
6.9.	Директивы программы IND	193
6.9.1.	Директива метка	193
6.9.2.	Логический конец файла (/)	193
6.9.3.	Директива .ASK	193
6.9.4.	Директива .ASKN	195
6.9.5.	Директива .ASKS	196
6.9.6.	Директива .BEGIN	197
6.9.7.	Директива .CHAIN	198
6.9.8.	Директива .CLOSE	198
6.9.9.	Директива .DATA	199
6.9.10.	Директива .DEC	199
6.9.11.	Директива .DELAY	199
6.9.12.	Директива .DISABLE	200
6.9.13.	Директива .DUMP	201
6.9.14.	Директива .ENABLE	201
6.9.15.	Режим DATA[#N]	202
6.9.16.	Командный режим DCL	202
6.9.17.	Режим DELETE	203
6.9.18.	Режим ESCAPE	203
6.9.19.	Режим глобальных имен	203
6.9.20.	Режим LOWERCASE	203
6.9.21.	Командный режим MCR	204
6.9.22.	Режим OCTAL	204
6.9.23.	Режим PREFIX	204
6.9.24.	Режим QUIET	204
6.9.25.	Режим SUBSTITUTION	204
6.9.26.	Режим SUFFIX	205
6.9.27.	Режим TIMEOUT	205
6.9.28.	Режим TRACE	205
6.9.29.	Директива .END	205
6.9.30.	Директива .ERASE	206
6.9.31.	Директива .EXIT	206
6.9.32.	Директива .GOSUB	207
6.9.33.	Директива .GOTO	207
6.9.34.	Директивы ветвления	208
6.9.35.	Директива .IF	208
6.9.36.	Директивы .IFDF/.IFNDF	209
6.9.37.	Директивы .IFENABLED и IFDISABLED	209
6.9.38.	Директива .IFLOA и .IFNLOA	209
6.9.39.	Директива .IFT и .IFF	210
6.9.40.	Директивы .OR и .AND	211
6.9.41.	Директива .INC	212
6.9.42.	Директива .ONERR	212
6.9.43.	Директивы .OPEN, .OPENA, .OPENR	213
6.9.44.	Директива .PARSE	213
6.9.45.	Директива .PURGE	214

6.9.46.	Директива .READ	215
6.9.47.	Директива .RETURN	215
6.9.48.	Директивы .SETD/.SETO	215
6.9.49.	Директива .SETL	215
6.9.50.	Директива .SETN	216
6.9.51.	Директива .SETS	216
6.9.52.	Директивы .SETT и .SETF	217
6.9.53.	Директива .STOP	218
6.9.54.	Директива .TEST	218
6.9.55.	Директива .TESTDEVICE	219
6.9.56.	Директива .TESTFILE	221
6.9.57.	Директива .VOL	221
	Перечень ссылочных документов	221

ВСПОМОГАТЕЛЬНЫЕ ПРОГРАММЫ РУКОВОДСТВО ОПЕРАТОРА

1.	Печать	225
1.1.	Назначение программы и условия выполнения программы	225
1.2.	Выполнение программы	225
1.3.	Команды оператора	227
1.4.	Примеры	228
1.5.	Сообщения оператору	230
2.	Сравнение текстовых файлов	230
2.1.	Назначение программы и условия выполнения программы	230
2.2.	Выполнение программы	231
2.3.	Команды оператора	233
2.4.	Использование конструкции (*) в командной строке	233
2.5.	Создание командного файла для SLP	234
2.6.	Примеры	234
2.7.	Сообщения оператору	237
3.	Модификация файлов	241
3.1.	Назначение программы и условия выполнения программы	241
3.2.	Выполнение программы	242
3.3.	Команды оператора	242
3.3.1.	Команды PATCH	243
3.3.2.	Открытие слов и байтов в файле	245
3.3.3.	Модификация оверлейных файлов	246
3.3.4.	Контрольная сумма	247
3.4.	Сообщения оператору	248
4.	Модификация объектных модулей	250
4.1.	Назначение программы и условия выполнения программы	250
4.2.	Выполнение программы	251
4.3.	Команды оператора	252
4.4.	Способ использования	252
4.4.1.	Примеры	253
4.5.	Сообщения оператору	256
5.	Модификация загрузочных модулей	259
5.1.	Назначение программы и условия выполнения программы	259
5.2.	Выполнение программы	260
5.3.	Команды оператора	260
5.4.	Диалог программы SIPP	262
5.5.	Команды программы SIPP	263
5.5.1.	Команды <BK> (<PC>) и N<BK>	265

5.5.2.	Команды $\wedge<BK>$ и $N\wedge<BK>$	265
5.5.3.	Команды $\backslash<BK>$ и $/<BK>$	265
5.5.4.	Команда $;O<BK>$	266
5.5.5.	Команды $;A<BK>$ и $;AX<BK>$	267
5.5.6.	Команды $;R<BK>$ и $;YYYY<BK>$	267
5.5.7.	Команда $;S<BK>$	268
5.5.8.	Команда $;V<BK>$	269
5.5.9.	Команда $CU/Z<BK>$	270
5.5.10.	Команда $CU/Y<BK>$	271
5.5.11.	Расширение файлов и оверлейных сегментов	271
5.5.11.1.	Расширение неоверлейных программ	272
5.5.11.2.	Расширение программы с оверлеями в нижней памяти	272
5.5.11.3.	Расширение программы с оверлеями в расширенной памяти	273
5.5.11.4.	Расширение программы с оверлеями в нижней и расширенной памяти	274
5.6.	Контрольная сумма программы SIPP	276
5.7.	Выполнение программы SIPP из косвенного командного файла	277
5.8.	Выполнение программы SIPP из пакета заданий	278
5.9.	Сообщения оператору	278
6.	Модификация текстовых файлов	283
6.1.	Назначение программы и условия выполнения программы	283
6.2.	Выполнение программы	283
6.3.	Команды оператора	283
6.4.	Пример	286
6.5.	Создание командного файла	288
6.5.1.	Формат строки командного файла	289
6.5.2.	Создание файла листинга	291
6.5.3.	Введение строк в файл	292
6.5.4.	Удаление строк из текстового файла	294
6.5.5.	Замена строк в файле	295
6.6.	Определение и оценка содержимого файла	296
6.7.	Сообщения оператору	296
7.	Сравнение двоичных файлов	300
7.1.	Назначение программы и условия выполнения программы	300
7.2.	Выполнение программы	300
7.3.	Команды оператора	300
7.4.	Использование конструкции (*) в командной строке	302
7.5.	Формат выходного файла	303
7.6.	Примеры	304
7.7.	Создание командного файла	305
7.8.	Сообщения оператору	305
8.	Форматирование	308
8.1.	Назначение программы и условия выполнения программы	308
8.2.	Выполнение программы	308
8.3.	Команды оператора	308
8.3.1.	Форматирование по умолчанию	310
8.3.2.	Переключатель /P:N	311
8.3.3.	Переключатель /S	313
8.3.4.	Переключатель /V[:ONL]	313
8.3.5.	Переключатель /W	314
8.3.6.	Переключатель /Y	314
8.4.	Сообщения оператору	314
9.	Получение информации о состоянии системы	317
9.1.	Назначение программы и условия выполнения программы	317

9.2. Выполнение программы	318
9.2.1. Команды оператора	318
9.2.2. Переключатель /A	319
9.2.3. Переключатель /C	320
9.2.4. Переключатель /D	320
9.2.5. Переключатель /H	321
9.2.6. Переключатель /J	322
9.2.7. Переключатель /L	322
9.2.8. Переключатель /M	324
9.2.9. Переключатель /O	324
9.2.10. Переключатель /S	325
9.2.11. Переключатель /T	325
9.2.12. Переключатель /X	326
9.2.13. Переключатель /Z	326
9.3. Сообщения оператору	327
Перечень ссылочных документов	328

ПРОГРАММЫ РАБОТЫ С ФАЙЛАМИ РУКОВОДСТВО ОПЕРАТОРА

1. Программа обмена RIP	329
1.1. Назначение программы и условия выполнения программы	329
1.2. Выполнение программы	329
1.3. Команды оператора	329
1.4. Переключатели RIP	330
1.4.1. Переключатель /A	333
1.4.2. Переключатель /B	333
1.4.3. Переключатель /C[:DATA]	334
1.4.4. Переключатель /D	334
1.4.5. Переключатель /E	334
1.4.6. Переключатель /F	335
1.4.7. Переключатель /G	336
1.4.8. Переключатель /H	336
1.4.9. Переключатель /I[:DATA]	336
1.4.10. Переключатель /J[:DATA]	337
1.4.11. Переключатель /K:N	337
1.4.12. Переключатель /M:N	337
1.4.13. Переключатель /N	339
1.4.14. Переключатель /O	339
1.4.15. Переключатель /P	340
1.4.16. Переключатель /Q	340
1.4.17. Переключатель /R	340
1.4.18. Переключатель /S	340
1.4.19. Переключатель /T[:DATA]	341
1.4.20. Переключатель /U	341
1.4.21. Переключатель /V	341
1.4.22. Переключатель /W	342
1.4.23. Переключатель /X	342
1.4.24. Переключатель /Y	342
1.4.25. Переключатель /Z	343
1.5. Сообщения оператору	343
2. Программа обслуживания устройств DUP	349
2.1. Назначение программы и условия выполнения программы	349
2.2. Выполнение программы	349
2.3. Команды оператора	349

2.3.1.	Переключатели	DUP	349
2.3.2.	Переключатель	/C/G:N]	353
2.3.3.	Переключатель	/I	354
2.3.4.	Переключатель	/K	356
2.3.5.	Переключатель	/F	357
2.3.6.	Переключатель	/O	357
2.3.7.	Переключатель	/Q	358
2.3.8.	Переключатель	/S	358
2.3.9.	Переключатель	/T:N	359
2.3.10.	Переключатель	/U:XX]	360
2.3.11.	Переключатель	/V[:ONL]	360
2.3.12.	Переключатель	/W	361
2.3.13.	Переключатель	/Y	362
2.3.14.	Переключатель	/Z[:N]	362
2.3.15.	Переключатель	/B[:RET]	364
2.3.16.	Переключатель	/D	365
2.4.	Сообщения оператору		365
3.	Программа получения справочника	DIR	371
3.1.	Назначение программы и условия выполнения программы		371
3.2.	Выполнение программы		371
3.3.	Команды оператора		372
3.4.	Переключатели	DIR	372
3.4.1.	Переключатель	/A	374
3.4.2.	Переключатель	/B	375
3.4.3.	Переключатель	/C[:N]	375
3.4.4.	Переключатель	/D[:DATA]	375
3.4.5.	Переключатель	/E	376
3.4.6.	Переключатель	/F	376
3.4.7.	Переключатель	/G	377
3.4.8.	Переключатель	/J[:DATA]	377
3.4.9.	Переключатель	/K[:DATA]	377
3.4.10.	Переключатель	/L	378
3.4.11.	Переключатель	/M	378
3.4.12.	Переключатель	/N	378
3.4.13.	Переключатель	/O	379
3.4.14.	Переключатель	/P	379
3.4.15.	Переключатель	/Q	379
3.4.16.	Переключатель	/R	380
3.4.17.	Переключатель	/S[:XXX]	380
3.4.18.	Переключатель	/T	382
3.4.19.	Переключатель	/U	383
3.4.20.	Переключатель	/V[:ONL]	383
3.5.	Сообщения оператору		383
4.	Программа создания копий	BUP	385
4.1.	Назначение программы и условия выполнения программы		385
4.2.	Выполнение программы		385
4.3.	Команды оператора		386
4.4.	Переключатели	BUP	386
4.4.1.	Переключатель	/I	387
4.4.2.	Переключатель	/L	389
4.4.3.	Переключатель	/X	390
4.4.4.	Переключатель	/Z	392
4.5.	Сообщения оператору		393
	Перечень ссылочных документов		398

Изд. № 86. Сдано в набор 28.08.89. Подп. в печать 20.12.89.
Формат 60×84¹/₁₆. Бум. тип. № 1. Гарнитура литературная.
Печать высокая. Объем 23,6 усл. печ. л. Тираж 20 000 экз. Заказ 102.

Ленинградское отделение РППО «Союзбланкоиздат».

Великолукская городская типография управления издательств,
полиграфии и книжной торговли Псковского облисполкома
182100, г. Великие Луки, ул. Полиграфистов, 78/12.