ADDENDUM 2

VORTEX II Reference Manual

98A 9952 246

This addendum contains information relating to the 70-755x Rotating Memory Device. It also describes a new program RELINK, and changes toVZFMA and BLDSKD.

PAGE	ACTION
1-1	Add the following to section 1.1, item d: or a direct memory interface.
3-4	In section 3.2, add the following to the paragraph beginning: "The bad track table..." The 70-755x RMD driver uses a bad sector table (BST) instead of a bad track table. This RMD may also contain up to 63 partitions instead of 20, and so the PST occupies the first 2 logical 120 word records of the RMD. The third logical record contains the second half of the boot program, so that the BST begins on the fourth logical 120 word record instead of the second as for other RMDs. The BST format is logically the same as the BTT (bad track table).
3-7	Add the following to the macro table in section 3.5: The following I/O macros apply to the 70-755x RMD only: . DEMAND Demand the controller for this unit . RELEAS Release the controller after a DEMAND . BCB Generate Buffer Control Block . RESERV Reserve the controller for this unit . RELRSV Release the controller after a RESERV

ISSUED: NOVEMBER 1977

PAGE	ACTION

3-8 In section 3.5, add the following to the paragraph headed Bits 8-11 Op-Code:

1000-1111	used only by 70-755x RMD
1001	Demand
1010	Releas
1011	Reserv
1100	Relrsv

3-13 Add the following to the Function code table in section 3.5.8:

70-755x	0	Identify unit (see 3.6.4)
	1	Status of unit

3-16 Add new section 3.6.

3.6 70-755x RMD I/O

This section applies only to the 70-755x RMD, and to no other model RMD.

3.6.1 Alternate Sector Partition

Each 77-755x spindle contains an Alternate Sector Partition for that spindle. The first 2n sectors (where n = number of partitions per spindle) contain the Alternate Sector Directory for that spindle. The remainder of the Alternate Sector Partition is used for subsequent sector assignments and addition directories if needed. The Alternate Sector Partition is the first partition on the spindle and can either be specified at SYSGEN or defaulted at SYSGEN. The default size is 1 percent of the spindle capacity. The partition is an unkeyed partition with a partition designator of * (asterisk).

PAGE	ACTION
3-16	(continued)

Bad sectors are located and identified during the disc pack formatting operation. The Formatter creates a Table of bad sectors for the entire disc pack. During SGEN 1 execution, partitions and their associated PST entries are created, alternates to bad sectors are assigned, and the Alternate Sector Table is built. Alternates to bad sectors are always allocated from the alternate sector partition.

3.6.2 Alternate Sector Processing

When a bad sector is detected during a data transfer operation, the Alternate Sector Table for the partition being accessed is read into memory. The table is searched for an entry containing the address of the sector flagged bad. If it is not found, the I/O request is terminated with the appropriate error status; refer to appendix A.1 for a list of error conditions. When a match is obtained, the following actions are taken by the driver:

1. A Channel Control Block (CCB) specifying Command Buffer Out is created.

2. A Command Buffer containing the Position to Transfer Data command and an alternate sector address is created.

3. A channel Control Block specifying Data Buffer In/Out is created for the data buffer corresponding to the alternate sector.

4. The Command Buffer Out CCB is Command Chained to the Data Buffer In/Out CCB, then an Initiate Channel Function is performed.

5. Upon receipt of the Operation Complete interrupt, the read/write heads are repositioned to the sector following the bad sector and the data transfer operation is continued to normal completion.

PAGE	ACTION
3-16	(continued)

3.6.3 Data Record Blocking and Deblocking

The VORTEX II System utilizes records of 120 words and multiples of 120 words for I/O operations on RMD. The use of 120-word physical sectors for the 70-755x driver discs yields a utilization efficiency of approximately 73%. Increasing the physical sector size improves the disc utilization efficiency by increasing the data storage area with respect to fixed areas of sectors. The 70-755x controller provides the capability to select one of several different hardware sector sizes through the use of plug-in "PROM's". When sector sizes larger than 120 words are selected, the VORTEX II 70-755x driver will perform automatic blocking and deblocking of user data records on all I/O requests, unless this feature is overridden.

On write requests which specify one of the access modes 0 through 3, this disc driver will pack as many whole and partial data records as a hardware sector can contain.

a. User data records which reside on the same hardware sector as the record to be written, are read into memory.

b. The new data record is data-chained to the other records which were read in during step a.

c. All of the data records are written onto the sector.

A write request requiring a blocking operation will take a minimum of 1, a maximum of 2 and an average of 1.5 revolutions to complete.

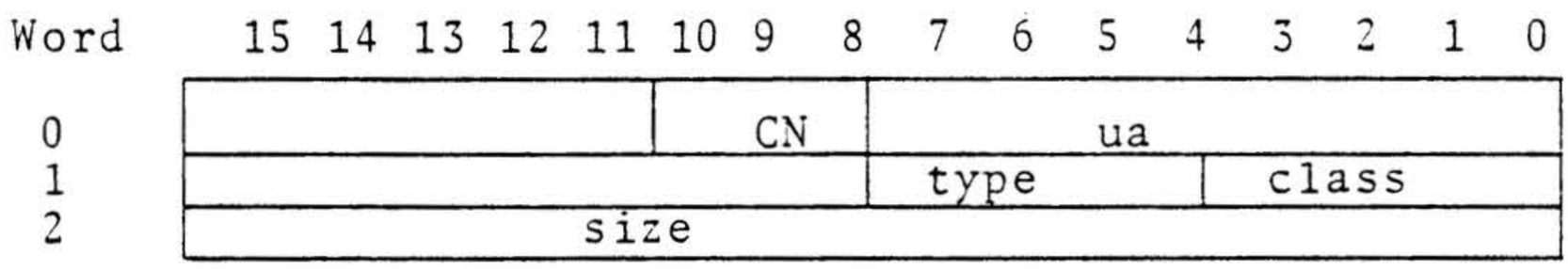
On read requests which specify one of the access modes 0 through 3, this disc driver will perform automatic deblocking of user data records. On deblocking operations, the driver utilizes the "transfer-in" channel capability of the 70-755x controller to eliminate data on the same sector preceding the desired record. Consequently, an additional revolution of the disc is not required to perform a deblocking operation.

PAGE	ACTION
3-16	(continued)

On read and write requests which specify access modes 4 and 5, user data records are read or written starting on hardware sector boundaries. Blocking and deblocking have no significance when these access modes are selected.

3.6.4 Identification Buffer Description

A three-word record containing unit information is placed in a user-supplied buffer in response to an Identify function (mode=0). The format of the three word buffer is:



Buffer Field Description

Word	Bit	Field	Description
0	10-8	CN	Controller number
0	7-0	ua	unit address
1	7-4	type	type of unit
			0 40 megabyte capacity
			1 80 megabyte capacity
			2 150 megabyte capacity
			3 300 megabyte capacity
1	3-0	class	class of device
			5 = 70-755x disc moving head RMD
2	15-0	size	is the unit sector size (in words)

3.6.5 Status Buffer Description

Status information up to a maximum of 4 words is returned to a user-supplied buffer in response to a status function (mode=1). The status information consists of a primary status word followed by three other words including the secondary status.

PAGE	ACTION
3-16	(continued)

Primary Status

The format of the primary status word is as follows:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UB	SS	EOU	EOF	SC				UNIT ADDR							

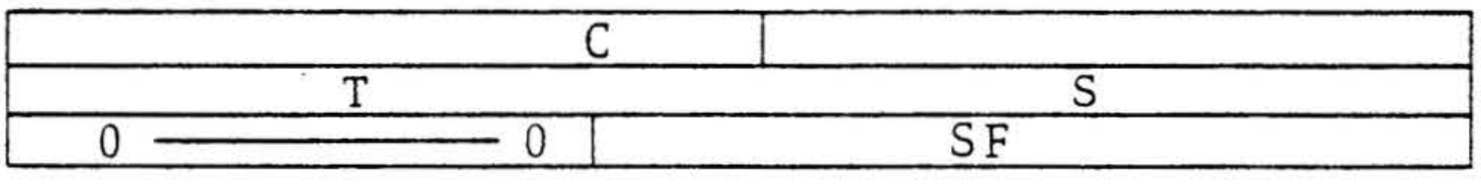
Bit	Field	Explanation
15	UB	Unit Busy
14	SS	Secondary status
13	EOU	End of Unit
12	EOF	End of File
11-8	Status Code	0 no error 1 Memory error 2 Map error 3 Unit data error 4 Attention 5 Off line 6 Chaining interrupt 7 Memory timeout 10 Command reject 11 Rate error 12-17 Not used
7-0	Unit Addr	Address of unit supplying status

For a detailed description of the status conditions, refer to the descriptions on input transfers in the V70-755x Disc Controller Manual (document number 98A 9906 51x).

PAGE	ACTION
3-16	(continued)

Secondary Status Buffer

The contents of the secondary status buffer are as follows:



Word	Bit	Field	Definition
0	15-0	C	Cylinder address of the last operation
1	15-8	T	Track address of the last operation
1	7-0	S	Sector address of the last operation
2	15-10	ZERO	All zeros
2	9-0	SF	Secondary status field

Bit 9 = Not used
Bit 8 = Write protect error
Bit 7 = Sector flagged bad
Bit 6 = Data synchronization fail
Bit 5 = Header synchronization fail
Bit 4 = Sector search error
Bit 3 = Track select error
Bit 2 = Cylinder seek error
Bit 1 = Data check error
Bit 0 = Header CRC error

For a detailed description of the secondary status buffer, refer to the description of the status input buffer in the V70-755x Disc Controller Manual (document number 98A 9906 51x).

PAGE	ACTION
3-16	(continued)

3.6.6 DEMAND

Function

The DEMAND request causes the controller of the disc unit associated with the designated logical unit to acquire immediate control of the interconnected channel of a dual channeled disc drive.

General Form

label DEMAND ,lun , wait

Parameter Description

Refer to section 3.5 for descriptions of parameters.

Error Conditions

Refer to appendix A.3 for error messages.

Comments

The demand request overrides alternate channel operations and retains control of the interconnected channel until a subsequent RELEAS request. It is the responsibility of tasks in systems with dual channel disc configurations to maintain the necessary coordination to avoid disruptions caused by the arbitrary and unwarranted use of the DEMAND request.

PAGE	ACTION
3-16	(continued)

Example

Acquire immediate control of the disc unit associated with logical unit 180.

```
LUN       EQU       180  (logical unit number)
            .
            .
            .
          DEMAND    ,LUN
```

3.6.7 RELEAS

Function

The RELEAS request relinquishes control of the interconnected channel for the disc unit associated with the designated logical unit.

General Form

```
label     RELEAS    ,lun , wait
```

Parameter Description

Refer to section 3.5 for descriptions of parameters.

Error Conditions

Refer to appendix A.3 for error messages.

Comments

The RELEAS request should be issued after a DEMAND request on the same logical unit. A RELEAS request without a preceeding DEMAND request has no effect.

PAGE	ACTION
3-16	(continued)

Example

Release logical unit 180 from a previous DEMAND condition.

```
LUN       EQU       180  (logical unit number)
            .
            .
            .
RELEAS    ,     LUN
```

3.6.8 FCB Macro

Function

The FCB (File Control Block) macro generates an additional list of parameters required by I/O requests involving disc file operations.

General Form

label FCB r1, $\frac{\text{buffer}}{\text{BCB addr}}$, acc, key, 'N_1','N_2','N_3', c

Parameter Description

r1	is the length in words of the record to be transmitted. If a READ request is specified with the Transfer In Channel option selected, r1 indicates the size of the file record; physical or logical record structure is designated by the access method parameter.
buffer	is the address of the user data area (C=0, no buffer chaining).
BCB adr	is the address of the first Buffer Control Block (C=1, buffer chaining specified).

PAGE	ACTION
3-16	(continued)

acc — is the access method to be used when referencing the file.

Values are:
0 direct access by logical record
1 sequential access by logical record
2 direct access by relative record number
3 sequential access by relative record number
4 direct access by physical record
5 sequential access by physical record.

The six access methods are described below.

key — is the protect code necessary to address those logical units which are protected.

N_1, N_2, N_3 — is the 6 ASCII character file name being referenced.

c — is the buffer chaining indicator
0 no buffer chaining (default value)
1 buffer chaining specified. When this option is selected, the buffer address parameter is the pointer to the first Buffer Control Block.

These parameters must appear in the order shown, separated by commas. If a parameter is omitted and it is not the last one in the sequence, it must be represented by its trailing comma.

Error Conditions

None

Comments

Words 3, 4, 5, and 6 are used by IOC as shown below. Relative Record Number begins with 1.

PAGE	ACTION
3-16	(continued)

Sequential Access Method

Word	OPEN	READ	WRITE	SREC	CLOSE	REW
3	Position of current record set per mode chosen (see open)	Increments Record # by one	Increments Record # by one	Adds or subtracts one	Position of file on directory is set per mode chosen (see close)	Current record is set to one or begin address of logical unit
4	Current position of file as noted on directory put in this word	Uses position to check end-of-file	no action	Uses position to check end-of-file	no action	Set to end of logical unit address
5	Beginning file address put in this word	no action	no action	no action	no action	Beginning logical unit address put in this word
6	End file address put in this word	no action	no action	no action	no action	End logical unit address put in this word

Direct Access Method

Word	OPEN	READ	WRITE	SREC	CLOSE	REW
3	Position of current record set per mode chosen (see OPEN)	no action	no action	Adds or subtracts one	Position of file on directory is set per mode chosen (see Close)	Current record set to one or begin address of logical unit
4	Current position of file as noted on directory put in this word	no action	no action	no action	no action	Set to end address of logical unit
5	Beginning file address put in this word	no action	no action	no action	no action	Beginning logical unit address put in this word
6	End file address put in this word	no action	no action	no action	no action	End logical unit address put in this word

PAGE	ACTION
3-16	(continued)

Example

Example: Create an FCB for a file named "FILE XB", using the sequential access method, logical record oriented. The record length is 128 and the user data area address if BUFF Protect key code is Z.

```
SEQR    EQU     1               (access method)
RECL    EQU     128
          .
          .
          .
DISC    FCB     RECL,BUFF,SEQR,'Z','F1','LE','XB'
          .
          .
          .
BUFF    BSS     128
```

Direct Access by Logical Record

In this method of 70-755x disc driver calculates the logical record address specified in word 3 of the FCB by using the record size specified in word 1 of the FCB. Words 3, 4, 5, and 6 are not updated by the disc driver and must be updated by the user accordingly.

Sequential Access by Logical Record

The 70-755x driver calculates the address of the logical record specified in Word 3 of the FCB by using the record size specified in word 1. Words 3 and 4 are updated as specified in the chart under comments in section 3.6.8.

PAGE	ACTION
3-16	(continued)

Direct Access by Relative Record Number

The relative sector number is specified in Word 3 of the FCB. The 70-755x driver calculates the absolute address by using the VORTEX System sector size (120 words), rather than the FCB record size as in the direct access by logical record method. Words 3, 4, 5 and 6 of the FCB are not updated by this disc driver and must be updated by the user accordingly.

Sequential Access by Relative Record Number

The 70-755x driver determines the absolute sector address for the relative sector specified in Word 3 of the FCB by using the VORTEX system sector size (120 words) and not the FCB record size. Words 3 and 4 are updated as specified in the chart under comments in section 3.6.8.

Direct Access by Physical Record

The 70-755x driver uses the contents of Word 3 of the user-supplied FCB as a recored number within the specified file, and converts it to a physical sector address. All READ and WRITE operations begin on physical sector boundaries and may extend across several hardware records. Word 3 of the FCB is not altered following the completion of I/O operations.

The size of physical sectors is variable: it depends upon the current selection in effect for the controller. Regardless of the ratio of user record length to physical sector size, blocking and deblocking of user records will not be performed by the driver when this access method is designated.

Note: Both direct and sequential access by physical record may result in data files that are not compatible with logical and relative record access methods due to sector boundary considerations. Therefore, care should be taken to use only direct or sequential access by physical record on files which have been created using these access methods.

PAGE	ACTION
3-16	(continued)

Sequential Access by Physical Record

The 70-755x driver determines the physical sector address corresponding to the record number specified in Word 3 of the user-supplied FCB. Read and Write operations always begin on physical sector boundaries, and may extend across several hardware sectors. Words 3 and 4 of the FCB are updated by the driver after the requested operation has been completed. Refer to the comments in section 3.6.8.

The size of the physical sector is variable: it is dependent upon the current selection in effect for the controller. Even if the user record length is different from the physical sector size, automatic blocking and deblocking of user records will not be performed by the driver.

Note: See note under Direct Access by Physical Record.

3.6.9 BCB Macro

Function

The BCB macro generates a four word Buffer Control Block which describes a user data buffer in main memory. Non-contiguous data buffers to be used for a data transfer operation are chained together through their associated Buffer Control Blocks.

General Form

```
label   BCB   buf adr, buf len ,  offset,  nxt bcb
```

Parameter Description

buf adr	is the address of the user data buffer
buf len	is the length of the user data buffer (in words)

PAGE	ACTION
3-16	(continued)

offset — is the displacement value of the first word within a file record which is to be read. The offset parameter has relevance only for READ operations utilizing the "Transfer-In Channel" capability. The default value for offset is zero. The range of offset is $0 \leq \text{offset} < \text{file record length}$.

nxt bcb — is address of the next Buffer Control Block in the chain. If this parameter is zero, it indicates the end of the chain. The default value of nxt bcb is zero.

Error Conditions

None

Comments

1. The BCB macro does not contain any executable instructions. Therefore, Buffer Control Blocks must be generated out of line.

2. The BCB macros must be used to link and describe non-contiguous buffers when the Chain option has been selected; refer to section 3.6.8 for a description of the FCB macro. If the "Transfer-In Channel" capability for a READ request is desired, the BCB macro must be used to generate the Buffer Control Block regardless of the number of buffers utilized.

3. The "c" parameter in the FCB macro must be set to indicate Buffer Chaining. In addition, the buffer address parameter in the FCB macro must point to the first Buffer Control Block.

4. All Buffer Control Blocks which are to be used for data transfer requests must be created and linked before the I/O request is issued.

PAGE	ACTION
3-16	(continued)

Examples

Example 1. Write a 128 word record on file "FILEXC" from three non-contiguous buffers labeled BUFF1 (30 words), BUFF2 (40 words), and BUFF3 (58 words). The logical sequential access method and buffer chaining options are selected and the protection code for the file is Z. The logical unit number is 28 and the wait option is specified.

```
LUN       EQU        28  (logical unit number)
CHAIN     EQU         1
RL        EQU       128
SEQR      EQU         1  (logical sequential access method)
BUFL1     EQU        30  (length of buffer 1)
BUFL2     EQU        40  (length of buffer 2)
BUFL3     EQU        58  (length of buffer 3)
                .
                .
                .
          WRITE    FCBXC,LUN
                .
                .
                .
FCBXC     FCB      RL,BCB1,SEQR,'Z','F1','LE','XC',CHAIN
                .
                .
                .
BCB1      BCB      BUF1,BUFL1,,BCB2
BCB2      BCB      BUF2,BUFL2,,BCB3
BCB3      BCB      BUF3,BUFL3
                .
                .
                .
```

PAGE	ACTION
3-16	(continued)

Example 1, continued

```
BUF1        BSS         30

BUF2        BSS         40

BUF3        BSS         58
```

Example 2. Read 3 non-contiguous entries from a record in file "FILEXD" on logical unit 180. The entries start on words 20, 40, and 60 of the record and 10, 15, and 20 word entries respectively are read using the "Transfer In Channel" option. The logical sequential access method, wait, and buffer chaining options are selected. The three buffers are labeled BUFA, BUFB, and BUFC.

```
LUN         EQU         180   (logical unit number)

RL          EQU         256   (file record length)

SEQR        EQU           1   (logical sequential access method)

CHAIN       EQU           1   (buffer chaining option)

BUFASZ      EQU          10   (size of buffer 1)

BUFBSZ      EQU          15   (size of buffer 2)

BUFCSZ      EQU          20   (size of buffer 3)

DISP1       EQU          20   (displacement of first entry)

DISP2       EQU          40   (displacement of second entry)

DISP3       EQU          60   (displacement of third entry)

            .
            .
            .

            READ        FCBXD,LUN

            .
            .
            .
```

PAGE	ACTION
3-16	(continued)

Example 2, continued

```
FCBXD     FCB       RL,BCBA,SEQR,'J','F1','LE','XD',CHAIN
                .
                .
                .
BCBA      BCB       BUFA,BUFASZ,DISP1,BCBB
BCBB      BCB       BUFB,BUFBSZ,DISP2,BCBC
BCBC      BCB       BUFC,BUFCSZ,DISP3
                .
                .
                .
BUFA      BSS       10    (buffer 1)
BUFB      BSS       15    (buffer 2)
BUFC      BSS       20    (buffer 3)
```

3.6.10 RESERV

Function

The RESERV request causes the controller of the disc unit associated with the designated logical unit to acquire immediate control of the interconnected channel of a dual channeled disc drive.

General Form

```
label     RESERV      ,lun , wait
```

Parameter Description

Refer to section 3.5 for descriptions of similarly named parameters.

PAGE	ACTION
3-16	(continued)

Error Conditions

Refer to appendix A.3 for error messages.

Comments

The "reserve" condition will remain in effect on the controller and disc unit designated by the "RESERV" request until a subsequent "RELRSV" request is issued. The "reserve" condition on the designated disc unit and controller may be overridden by a "DEMAND" request issued to the alternate controller.

Example

Place the disc unit associated with logical unit 180 in the "reserve" condition

```
LUN         EQU         180   (logical unit number)
                .
                .
                .
            RELEAS      ,LUN
```

3.6.11 RELRSV

Function

The RELRSV request relinquishes control of the interconnected channel for the disc unit associated with the designated logical unit. This request resets the "reserve" condition set by the "RESERV" request.

General Form

```
label       RELRSV      ,lun , wait
```

PAGE	ACTION
3-16	(continued)

Parameter Description

Refer to section 3.5 for descriptions of similarly named parameters.

Error Conditions

Refer to appendix A.3 for error messages.

Comments

The RELRSV request should be issued after a "RESERV" request on the same disc unit. A RELRSV request without a preceeding RESERV request has no effect.

Example

Reset the "reserve" condition on the disc unit associated with logical unit 180.

```
LUN       EQU       180  (logical unit number)

                .
                .
                .

          RELRSV    ,LUN
```

6-7 Insert new section 6.4:

6.4 RELINK

RELINK is a background program which is used to "relink" (to reestablish VORTEX nucleus pointers) an existing load module with a modified nucleus (one recreated with a SYSGEN and for which changes have been introduced that moved certain nucleus pointers or addresses). The load module may be restored on a library partition by any copy process such as FMUTIL, IOUTIL, or equivalent program. RELINK makes the load module executable on the new system without the need to go through another entire LMGEN process. RELINK needs to be executed only once for each load module after the nucleus has been modified and the load module

PAGE	ACTION
6-7	(continued)

restored. RELINK may also be used for relinking a load module that has been transferred from another system, e.g. through CPU to CPU data links, on disc pack, or on magnetic tape. This is particularly useful in a master-slave configuration where the slave system is not capable of supporting development activities: programs are compiled and LMGENed on the host system and transferred to the slave system.

RELINK uses a Core Resident Symbol Table (CRST) located at the end of the load module file -- this is not part of the user program space -- which has been established by LMGEN. RELINK extracts entries from this table and searches for the nucleus pointer name in the system CL directory. When this name is found, RELINK "patches" the appropriate locations in the load module RMD file and continues this process until the CRST table is complete. The VORTEX system loader (V$SAL) does not use the CRST, since the load module has been patched. There is no limit to the number of times an existing load module may be relinked. However, relinking is possible only if all load module referenced nucleus pointers exist on the new system on which the load module is to be relinked.

DIRECTIVE FORMAT:

/LOAD,RELINK

RELINK will respond with the prompt:

LUN,KEY,LD MODULES

The user then inputs the name of the load modules that are to be relinked in the format:

LUN,KEY,mod name, mod name,...

where

LUN	is the logical unit number on which the load module resides (may be lun name)
KEY	is the protection key, if any, required for that LUN
mod name	is the name of the load module on that LUN

PAGE	ACTION
6-7	(continued)

Up to ten mod names may be specified on a single input line; however, the directive may not contain any embedded blanks, since a blank character terminates the directive line processing; and all the "mod names" must reside on the specified LUN.

A continuation directive may be used for "mod names" on the previously specified LUN by starting the directive with a comma and omitting the LUN and KEY parameters; however, a continuation line may not be used as the first directive line. Each directive line is processed before the next directive -- either new or continuation -- is accepted. If a load module resides on a different LUN, it must be specified by a new directive giving the different LUN and KEY.

As each load module is relinked, the following message appears on the SO device:

xxxxxx RELINKED

where

xxxxxx is the name of the module which has been relinked.

The LUN,KEY,LD MODULES prompt occurs only once.

To exit from RELINK and return to JCP the user may enter the command

END

If a directive beginning with a slash (/) is encountered, an I013 diagnostic message is output and RELINK exits.

Example

Relink the load modules PROG1, PROG2, and PROG3 from the background library, and load modules PROG4 and PROG5 from the foreground library.

PAGE	ACTION

6-7 (continued)

```
/LOAD,RELINK                    user input
LUN,KEY,LD MODULES              program prompt
BL,E,PROG1,PROG2,PROG3          user input
**PROG1     RELINKED**          program output
**PROG2     RELINKED**          program output
**PROG3     RELINKED**          program output
FL,F,PROG4,PROG5                user input
**PROG4     RELINKED**          program output
**PROG5     RELINKED**          program output
END                             user input
```

9-5 Replace the title and first paragraph of section 9.3 with the following:

9.3 VORTEX FILE MAINTENANCE DRIVER (VZFMA)

The VORTEX File Maintenance driver provides a user-programmable subset of the VORTEX FMAIN services. VZFMA operates as a system driver assigned to logical unit 115. All requests to VZFMA must be made through an OM library resident interface routine, V$FILE. Direct calls to VZFMA should not be used, to prevent conflicts in calling sequences if VZFMA services should be augmented.

9-6 In section 9.3, the paragraph beginning "Upon exit from a file request,..." delete from "The interface program allows..." through "... process the new request."

10-2 Add the following at the end of section 10.2.1:

When specifing random length on input, the input unit should not be an RMD device. Also, when specifying blocking/deblocking, the input and output record lengths should be multiples of each other.

PAGE	ACTION
15-7	Add the following to the table in section 15.4:

DcuH Rotating memory 70-755x

In section 15.4.4, add the following after "device":

the even device address should be used for the 70-755x.

In section 15.4.4, add the following after "bic":

this parameter should be 1 for the 70-755x.

15-11 Add the following to table 15-2:

DnH 70-755x Rotating memory

15-12 In section 15.5.3, add the following to the Dcup(n) information:

or (1) through (20). (For 70-755x, "p" may be (1) through (63).) Note: when listing the partitioning to the LIS device, SYSGEN always displays partitions with the numeric format regardless of how the partition was specified on the PRT directive.

Add the following at the end of section 15.5.3:

On 70-755x RMDs an alternate sector partition (Dcu*) is automatically created for each spindle (as the first partition) that is equal to 1% of that spindle's capacity. This size may be overridden by specifically defining the alternate sector partition with a PRT directive. For example, to allocate 10 tracks of alternate sectors on controller zero unit zero, the directive would be PRT,D00*,10,*.

15-19 Add the following to section 15.5.18:

Systems using the multitask spooler, FMUTIL, TSS, or any other task which makes a VORTEX File Maintenance driver request (e.g. a V$FILE call) must include the following directives:

```
EQP,FMOA,0,1,0,0
ASN,115=FM00
```

PAGE	ACTION
17-6	Replace section 17.3 with sections 17.3 and 17.4 below:

17.3 BLDSKD

The standard VORTEX Teletype drivers support the foreground scheduling task BLDSKD. BLDSKD allows the operator to define selected control characters (A-Z excluding G,J,M) to cause scheduling of specified tasks without the lengthly OPCOM SCHED format. Use of the defined control character as an unsolicited input from a device using the Teletype driver results in the previously defined task being scheduled as previously specified. (Note: BLDSKD cannot be scheduled in this manner) BLDSKD exits on completion of a specified action or on any error condition.

17.3.1 Set-Up Requirements

Before attempting to use BLKSKD, the operator must create a file called SCHDAD on the foreground library containing two 120-word records. This should be done with FMAIN. (Note: this is initially done by the BSCOM job stream.)

Example:

```
FMAIN
CREATE,FL,F,BLKSKD,120,2
```

17.3.2 Task Scheduling

To add, delete, or list tasks attached to control characters, BLDSKD must be scheduled through the OPCOM SCHED request from the foreground library.

Example:

```
     :SCHED,BLDSKD,5,FL,F
 BLDSKD will respond with:
     ENTER ACTION LETTER A,D OR L
```

where

A = add
D = delete
L = list

PAGE ACTION

17-6 (continued)

17.3.3 'A' Command (Attach a task to a control character)

This command is used to attach a specified task to a specified control character. BLDSKD will respond to this request with

DIRECTIVE FORMAT: LETTER,TASK,LIB,KEY,PRI

where

LETTER is the alphabetic portion of the control character to be attached to.

TASK is the task name to be attached (1 to 6 characters)

LIB is the library on which the task resides. Input may be a logical name or number.

KEY is the protection key associated with the logical unit, if any.

PRI is the priority at which the task is to be scheduled.

17.3.4 'D' Command (Delete a task)

This command is used to remove a task attachment to a specified control character. BLDSKD will respond to this request with

ENTER LETTER TO BE DELETED

where

LETTER is the alphabetic portion of the control character to be deleted.

17.3.5 L Command (List Assigned Tasks)

This command is used to list all assigned control characters and the task that have been attached to that control character. BLDSKD will respond to this request with a list in the following format:

PAGE	ACTION
17-6	(continued)

a bbbbbb

where

a is the defined control character

bbbbbb is the task attached to character a

17.4 TASK SCHEDULING

The input of any control character from a logical unit assigned to the system Teletype driver causes BLDSKD to be scheduled. If the control character input has been attached to a task, then that task is scheduled and BLDSKD exits. If the control character has not been attached, BLDSKD merely exits. Note: control G (bell), control J (line feed), and control M (carriage return) cannot be used as attachable control characters. The use of these control characters under the A command will result in an "invalid character" response from BLDSKD.

18-7 Add new section 18.9:

18.9 70-755x DISC FORMATTING

A standalone 70-755x disc formatter enables a user to format disc packs prior to generating a VORTEX II system. The formatter program is loaded into memory by the AID utility. Upon execution, the formatter outputs a prompt, "FH**". The following directive is entered to format a disc unit:

FORMAT,da,un

where

da = disc controller device address (even address only)

un = disc unit number

Upon completion of the formatting operation, the formatter outputs a prompt. The next unit may be formatted at this time. Appendix A.18.1 lists error messages which may be output by the formatter.

PAGE	ACTION			
A-7	Add the following to section A.3:			
	IO21,xxxxxxx	Memory parity error	Abort task or request	H05
	IO22,xxxxxxx	Map error	Abort task or request	H05
	IO23,xxxxxxx	Memory timeout error	Abort task or request	H05
	IO24,xxxxxxx	Rate error	Abort task or request	H05
	IO25,xxxxxxx	Memory/data bus verification error	Abort task or request	H05
	IO26,xxxxxxx	Memory access error	Abort task or request	H05
	IO27,xxxxxxx	Alternate sector partition full	Abort task or request	H05
	IO65,xxxxxxx	Alternate sector directory record read error	Abort task or request	H05
	IO66,xxxxxx	Alternate sector directory record write error	Abort task or request	H05

A-13 Add new section A.6.1:

A.6.1 RELINK

Message	Condition	Action	Possible User Action
RL01	Input error	Ignore input	Correct syntax
RL02 xxxxxx	Load module xxxxxx in error	Ignore module	Check validity of load module
RL03 xxxxxx	Invalid CRST item in module xxxxxx	Ignore module	Load module not relinkable, re-LMGEN

PAGE ACTION

A-13 (continued)

Message	Condition	Action	Possible User Action
RL04 xxxxxx	Nucleus pointer name xxxxxx not found in CL	Abort relinking	Load module not relinkable on this system
RL05 xxxxxx	Invalid displ. found in CRST	Abort relinking	CRST is invalid, re-LMGEN

A-23 Add new section A.18.1:

A.18.1 Disc Formatter Error Messages

Message	Description
FH01	Invalid directive name
FH02	Invalid parameters
FH03	Invalid controller
FH04	Invalid unit number
FH05	Invalid request error
FH06	Unrecoverable system error (controller)
FH07	Unrecoverable disc error

H-1 Add new Appendix H:

APPENDIX H
RMD STATUS WORDS

H.1 70-76x0 RMD

Bit	Meaning if bit on
0	Unit 0 Seek Complete
1	Unit 1 Seek Complete
2	Unit 2 Seek Complete
3	Unit 3 Seek Complete
4	Selected Unit Illegal Sector
5	Selected Unit Illegal Address
6	Selected Unit Malfunction
7	Selected Unit Timing Error
8	Selected Unit Read Parity Error
9	Selected End of Track Error
10	Selected Write Protect

PAGE	ACTION
H-1	(continued)

Bit	Description
11	Selected Unit - Unit Not Ready
12	Not Used
13	Not Used
14	Not Used
15	Not Used

H.2 70-7500 and 70-7510 RMD

Bit	Meaning if Bit on
0	Timing error
1	Track stop (record not found)
2	Format error (record too long)
3	Search error (bad cyclic check)
4	Data error (bad cyclic check)
5	End of file (data length = 0)
6	Unit not on line
7	Disc pack unsafe
8	Seek error (seek incomplete)
9	Write Protect
10	Unit not selected
11	Head select error
12	Bad track flag set
13	Not used
14	Not used
15	Not used

H.3 70-7520/7530 RMD

Bit	Meaning if Bit on
0	Timing error
1	Track stop (record not found)
2	Format error (record too long)
3	Search error (bad cyclic check)
4	Data error (bad cyclic check)
5	End of cylinder (head no > 19)
6	Unit not on line
7	Disc drive unsafe
8	Seek error (seek incomplete)
9	Read Only
10	Unit not selected

PAGE	ACTION
H-1	(continued)

Bit	Meaning if Bit on
11	Head select error
12	Bad track flag set
13	Disc reserved
14	Not used
15	Not used

H.4 70-7603/7613 RMD

Bit	Description
0	Unit 0 Seek Complete
1	Unit 1 Seek Complete
2	Unit 2 Seek Complete
3	Unit 3 Seek Complete
4	Selected Unit Illegal Sector
5	Selected Unit Illegal Address
6	Selected Unit Malfunction
7	Selected Unit Timing Error
8	Selected Unit CRC Search Error
9	Selected Unit Read CRC Error
10	Selected End of Track Error
11	Selected Write Protect
12	Selected Unit - Unit Not Ready
13	Selected Unit - Header Compare Error
14	Format Error
15	Sync Byte not found

PAGE	ACTION
H-1	(continued)

H.5 70-755x RMD

H.5.1 Primary Status

Bit	Explanation
15	Unit Busy
14	Secondary status
13	End of Unit
12	End of File
11-8	0 no error 1 Memory error 2 Map error 3 Unit data error 4 Attention 5 Off line 6 Chaining interrupt 7 Memory timeout 10 Command reject 11 Rate error 12-17 Not used
7-0	Address of unit supplying status

H.5.2 Secondary Status

Word	Bit	Field	Definition
0	15-0	C	Cylinder address of the last operation
1	15-8	T	Track address of the last operation
1	7-0	S	Sector address of the last operation
2	15-10	ZERO	All zeros
2	9-0	SF	Secondary status field

PAGE	ACTION
H-1	(continued)

Bit 9 = Correctable ECC error detected
Bit 8 = Write protect error
Bit 7 = Sector flagged bad
Bit 6 = Data synchronization fail
Bit 5 = Header synchronization fail
Bit 4 = Sector search
Bit 3 = Track select error
Bit 2 = Cylinder seek error
Bit 1 = Data check error
Bit 0 = Header CRC error