SPERRY UNIVAC

VORTEX II
System Generation
User Guide/Programmer Reference

Mini-Computer Operations

2722 Michelson Drive
P.O. Box C-19504
Irvine, California 92713

UP-9083

W

UNIVAC®

VORTEX II SYSTEM GENERATION

User Guide/Programmer Reference

UP-9083
98A 9952 770

OCTOBER 1979

Sperry Univac is a division of Sperry Corporation

Printed in U.S.A.

PAGE STATUS SUMMARY

ISSUE: UP-9083 (98A 9952 770)

Part/Section	Page Number	Update Level
Cover		
Title Page		
PSS	1	
CR	1	
Contents	1 thru 7	
1	1 thru 4	
2	1 thru 43	
3	1 thru 10	
4	1 thru 9	
5	1 thru 13	
6	1 thru 62	
7	1 thru 19	
8	1 thru 5	
9	1 thru 3	
A	1 thru 6	
B	1 thru 9	
C	1 thru 2	

Part/Section	Page Number	Update Level

Part/Section	Page Number	Update Level

*New pages

All the technical changes are denoted by an arrow (→) in the margin. A downward pointing arrow (↓) next to a line indicates that technical changes begin at this line and continue until an upward pointing arrow (↑) is found. A horizontal arrow (→) pointing to a line indicates a technical change in only that line. A horizontal arrow located between two consecutive lines indicates technical changes in both lines or deletions

CHANGE RECORD

Change Designation	Issue Date	Change Description
Original	Oct. 1979	Original issue

Change Procedure:

When changes are made to this manual, updated pages are issued. These updated pages are either added to this manual or used to replace obsolete pages. The specific pages affected by each change are identified on the PAGE STATUS SUMMARY page.

Printed in U S A

Table of Contents

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

Table of Contents (Continued)

List of Illustrations (Continued)

List of Tables

Section 1
INTRODUCTION

1.1. GENERAL

This manual describes the system generation (SYSGEN) procedures for the SPERRY UNIVAC VORTEX II Operating System. The VORTEX II Operating System is supplied as Type Number 6301-xx.

1.2 SCOPE

This manual contains descriptive data, operating information, and reference material for the generation of a VORTEX II Operating System. The material is directed toward the experienced programmer, who is responsible for the preparation, execution, and maintenance of SYSGEN.

The information in this manual is organized as follows:

Section 1	Introduction	Overview of the manual and SYSGEN.
Section 2	Universally Executable Example	A SYSGEN example which may be executed on a typical VORTEX II System.
Section 3	System Generation Preparation	Information and tables to aid in the preparation of SYSGEN input directives.
Section 4	System Generation Procedures	Step-by-step procedures for the generation of a tailored VORTEX II Operating System.
Section 5	System Generation Functional Description	An explanation of the system generation process.
Section 6	System Generation Directives	A detailed explanation of all directives associated with SYSGEN.

Section 7	System Generation Requirements for Individual Software Modules	A guide for determining SYSGEN requirements for individual hardware or software modules.
Section 8	System Generation Requirements For Individual Hardware Devices	A guide for determining SYSGEN requirements for individual hardware devices.
Section 9	System Generation Troubleshooting	Techniques used to find causes of user errors or omissions in the system generation process.
Appendix A	Disk Pack Formatting Programs	Information on the procedures for formatting the various types of disk packs that may be used on a VORTEX system.
Appendix B	Error Messages	A description of the error messages associated with the elements detailed in this manual.
Appendix C	System Generation on Single Tape Configuration	Procedures for performing system generation on a single tape configuration.

1.3 SYSTEM GENERATION DESCRIPTION

System generation (SYSGEN) is a method by which the user can tailor the operating system software to reflect a unique hardware configuration. SYSGEN also allows the user to build a system relating to specific application requirements. Since VORTEX II is a total software system, and as such provides many integral and diverse functions, it must be tailored to provide the user with those functions applicable to his needs.

VORTEX II SYSGEN provides all the tools (directives and procedures) necessary to effect the desired and required system. SYSGEN is usually performed because of one of the following conditions:

1. An initial VORTEX II system is installed at the customer site.

2. An existing system requires a revision based upon hardware additions.

3. An existing system needs revision as a result of new or additional application requirements.

1.4 RELATED PUBLICATIONS

The following documents contain information directly related to VORTEX II system generation. As such, the user must be thoroughly familiar with the concepts presented in these documents.

VORTEX II Operating System Programmer Reference — UP-8677

"System Memo"

"VORTEX II Release Bulletin"

1.5 CONVENTIONS USED IN THIS MANUAL

The following conventions are used throughout this document:

- ␣ - indicates a required space character.
- x_2 - denotes a binary number.
  x_8 - denotes an octal number.
  x_{10} - dentes a decimal number.
- Any number with at least one leading zero denotes an octal number.
- [ ] - items enclosed in "[" and "]" are optional and are not required in the directive.
- { } - items enclosed in "{" and "}" indicate only one item may be used.
- Any reference to VORTEX implies VORTEX II.

1.6 GUIDE TO THIS MANUAL AND OTHER PERTINENT DOCUMENTS

This manual has been prepared and organized such that Sections 2, 3, and 4 contain procedures and preparatory information for the procedures. The remainder of the manual contains reference

information supporting these first sections. Guidelines for the use of this manual include:

1. Section 2 should be studied and used by any user who is responsible for SYSGEN and who has not previously performed a VORTEX SYSGEN.

2. Section 3 should be used to help the user organize the directives which are to be used in the SYSGEN procedures described in Section 4. It should also be used as a checklist to ensure that all necessary directives have been included. Sections 6 and 7 must be used with this section.

3. Section 4 must be used for generating any specific VORTEX system. Sections 5, 6, and 7, and Appendix B should be used whenever necessary in conjunction with Section 4.

4. The instructions for formatting disk packs, as detailed in Appendix A, must be used prior to performing any other SYSGEN operation, if the disk packs on the system have not already been formatted for VORTEX.

After the user has completed the procedures in Section 4, the VORTEX operating system nucleus is resident in memory and has control of system resources. Also, load modules have been created from standard and optional object modules (from the SGL tape) and have been stored on the system disk in the appropriate system libraries.

When SYSGEN is complete, the system is an online, interactive VORTEX software system. The user may proceed to the VORTEX Operating System Programmer Reference Manual (UP-8677) for instructions on how to use the system.

Section 2
UNIVERSALLY EXECUTABLE EXAMPLE

2.1 PURPOSE

The purpose of this section is to illustrate the basics of system generation by presenting an example of a system generation procedure which can be run on a V77-600 system configured for the execution of VORTEX software. This is not a canned procedure. Rather, it a limited aid to instruct the user in performing system generation procedures.

In general, this example utilizes:

- A minimum hardware configuration to support a basic VORTEX system
- Minimum software to support the hardware
- A VORTEX nucleus
- One language processor
- Communications capabilities

This section also contains information from an actual SYSGEN listing of a specific system and the applicable pages from the "System Memo" accompanying that system.

2.2 MINIMUM VORTEX II HARDWARE CONFIGURATION

The VORTEX II Operating system requires certain hardware elements to execute properly. This is termed the minimum VORTEX II system and includes the following:

- A SPERRY UNIVAC 70 series or 77 series minicomputer

-V72 -V73 -V75 -V76	70 series
-V77-200 -V77-400 -V77-600 -V77-800	77 series

- 32K words main memory

- Rotating memory device (disk or drum) on a priority interrupt module (PIM) with either a buffer interlace controller (BIC) or a block transfer controller (BTC)

- One of the following on a PIM

  - Card reader with a BIC
  - Paper tape system or a paper tape reader
  - Magnetic tape unit with a BIC

- ASR 33/35 teletypewriter or compatible CRT on a PIM

- Memory map hardware

For this example, assume a system with the following hardware:

	Model Number	Type/Feature Number
■ CPU - Model V77-600		
■ 128K words main memory		
■ real time clock		
■ card punch and dedicated BIC on PIM	70-6201	2813-xx
■ card reader and dedicated BIC on PIM	70-6200	2812-xx
■ CRT keyboard/display	70-6401	2817-xx
■ disk memory and dedicated BIC on PIM	70-7500	2822-xx
■ line printer	70-6701	
■ magnetic tape system and dedicated BIC on PIM	70-7104	F3093-xx
■ paper tape reader/punch on PIM	70-6320	
■ ASR model 35 teletype-writer on PIM	70-6104	2811-xx

	Model Number	Type/Feature Number
■ 1K words writable control store (WCS)	70-4003	F2960-xx
■ floating point processor (FPP)		
■ cache memory		
■ memory map hardware		

2.3 OVERVIEW OF THE PROCESS

Basically, this system generation (SYSGEN) procedure encompasses:

1. Formatting the disk
2. Loading the key-in loader
3. Responding to the I/O interrogation
4. Entering the system generation header
5. Entering the required input directives
6. Building the VORTEX nucleus
7. Loading the library tasks

The results produced by these steps are:

- A memory-resident operating system (the VORTEX nucleus)
- The VORTEX nucleus on disk
- Disk-resident foreground library
- Disk-resident background library
- Disk-allocated user files
- Disk-allocated miscellaneous files
- System generation listing output

2.4 SAMPLE PROCEDURE

This subsection is a step-by-step procedure for generating a minimum VORTEX II software system. This procedure is designed for execution on a V77-600 system, configured as described in paragraph 2.2.

NOTE

On any given system, there are certain parameters which are specifically system-dependent. Steps involving system-dependent information are flagged as such and notations made as to how the information can be changed for a particular system.

2.4.1 PREPARATION

1. Mount the system generation library (SGL) tape on the master magnetic tape unit.

2. Insure that all devices are on-line.

3. Insure that the system disk has been formatted using the following procedures (or the applicable disk format program described in Appendix A).

2.4.2 DISK FORMATTING PROCEDURES

Proceed as follows to format the 2822-xx system disk.

1. Load the BLD II and AID III utility programs as follows:

   a. Load the system program loader routine into memory. The reference manual for a given SPERRY UNIVAC 70 series or 77 series minicomputer system provides the procedures for execution, loading, and verification of loading the program loader routine for that system.

   b. Load 00212 into the P register.

   c. Load 07000 into the X register.

   d. Set sense switch 1.

   e. Turn on the paper tape reader specified by the program loader routine.

f. Position the BLD II program tape in the reader with the first data frame after the position-8-only punches under the high-speed reader head or over the reading station of the teletypewriter.

g. To load tape, press RUN, then START. Loading is complete when the computer changes to step mode.

h. Place the AID III program tape in the reader with the first data frame under the high-speed reader head or over the reading station of the teletypewriter.

i. Reset SENSE switch 1 and press START.

j. Loading of AID III is complete when the program outputs a carriage return, a line feed and rings the terminal bell.

2. Load the 2822 disk formatting program by placing the program tape in the reader with the first data frame under the high-speed reader head, then typing

   L1.

   on the teletypewriter.

3. Loading of the program is complete when the paper tape stops.

4. Press the RESET switch.

5. Press the P switch.

6. Press the DISPL CLR switch thus zeroing the P register.

7. Using the upper row of switches on the control panel, press the following switches:

   - Switch 9
   - Switch 3
   - Switch 6
   - Switch 5
   - Switch 3
   - Switch 2

This loads the value 001354 into the program counter.

8. Press RESET.

9. Press STEP/RUN.

10. Press START.

The formatting program begins execution and requests certain parameters to be input from the keyboard. The following requests are made. An inappropriate response causes the request to be repeated.

- INPUT BTC NUMBER

  Type a value and a carriage return. The acceptable values are octal 020, 022, 024, 026 and 070.

- INPUT DEVICE ADDRESS

  Type a value in the range from octal 014 through 017 followed by a carriage return.

NOTE

The device address and BTC number are contained within the "System Memo".

.

- INPUT VARIABLE SECTOR GAP

  Type a value and carriage return. Acceptable values are 1, 2, 3, 4, 6, 8, 12, or their equivalent octal representations. This value determines the physical location on the disk pack of sequentially addressable sectors, as such sequential transfers may be accomplished without waiting for a full revolution of the disk unit. Recommended setting is 3. Another setting may be more effective depending upon various application parameters such as number of tasks, frequency of disk transfers, and types of disk transfers.

- INPUT UNIT NUMBER

  Type the unit number of the disk followed by a carriage return. Acceptable values are 0 through 3. Up to four units can be connected to a single controller.

The formatting program performs bad-track analysis and creates and maintains a bad-track table, which is entered on each disk pack at the completion of its formatting. The bad-track table is located on sectors 0 through 2 of the first track. The table is 254 words long, starting at word 64 of sector 0. The first 64 words of sector 0 reserve the necessary space for the PST. The remaining unused words of sector 2 are filled with zeroes. Each disk I/O error will generate a ten-event retry sequence, which upon failure will set the bad-track flag within the track header. The program also sets the corresponding bit in the bad-track table. No alternate tracks are assigned.

If the first track is determined to be bad, the bad-track table may not be placed there. The program prints the error message:

FIRST TRACK BAD

and aborts formatting the current disk pack. The program returns to the keyboard interrogation routine. After the bad-track table has been written on the disk pack, the formatting program resumes the keyboard interrogation to obtain the parameters for formatting the next disk. In this way, more than one disk pack can be formatted in the same session. The formatting program may be terminated at this point when no disk packs (except those with bad first tracks) remain unformatted. If an unsafe condition (SELECT LOCK light on) occurs, reload and execute the disk formatting program. Formatting disk packs is not necessary before every VORTEX system generation.

After all requests have been answered, the system starts formatting the disk. This process requires approximately 10 minutes. When the formatting operation is complete, the prompt

INPUT BTC NUMBER

is repeated. At this time, additional disks may be formatted or the system generation procedure may proceed.

2.4.3 LOADING THE KEY-IN LOADER

For entering the key-in loader into the system, use one of the following procedures, depending upon the type of the minicomputer.

2.4.3.1 Manual Key-in Loader Procedures on the 70 Series Minicomputers and V77-600 Minicomputer

Using the control panel on the V70 series or V77-600 minicomputer, perform the following steps:

1. Press the RESET switch.

2. Press the P switch.

3. Press the DISPL CLR switch, thus zeroing the P register.

4. Press the MEM switch.

5. Using the upper row of switches on the control panel, press the following switches:

   - Switch 12
   - Switch 4
   - Switch 3

   These switches represent the value 010030

   This loads the octal value of 010030 into location 0 of the minicomputer.

6. Press the ENTER switch, loading the value into memory and incrementing the P register.

7. Press the DISPL CLR switch.

8. Repeat steps 5 through 7 (altering step 5 with the appropriate octal value) until all the values listed in Table 2-1 (or Table 2-2 if type 0870-xx Magnetic Tape Unit is used) are entered into the system.

9. Press the P switch. If the upper row of indicators reflect an octal 000031, continue to step 10. If these indicators reflect any other value, refer to the "Correction Procedures for the V70 Series and V77-600 Manual Loading Procedure".

Table 2-1. System Generation Key-in Loader (For All Tape Units Except for Type 0870-xx)

Address (octal)	Value (octal)
000000	010030
000001	001010
000002	001114
000003	040030
000004	001000
000005	000012
000006	000000
000007	006010
000010	000300
000011	050027
000012	1041xx
000013	1000xx
000014	001000
000015	000021
000016	1025xx
000017	057027
000020	040027
000021	1011xx
000022	000016
000023	1012xx
000024	100006
000025	001000
000026	000021
000027	000500
000030	177742

xx = the device address in octal of the magnetic tape on which the SGL is mounted (see "System Memo").

xxxxxx = the device address in binary of the magnetic tape on which the SGL is mounted (see "System Memo").

Table 2-2. System Generation Key-in Loader (For the Type 0870-xx Magnetic Tape Unit)

Address (octal)	Value (octal)
000000	010035
000001	001010
000002	001114
000003	040035
000004	001000
000005	000012
000006	000000
000007	006010
000010	000300
000011	050034
000012	1041zz
000013	1000yy
000014	010034
000015	1031xx
000016	006010
000017	007000
000020	1031yy
000021	1000xx
000022	1000zz
000023	1012zz
000024	000030
000025	005000
000026	001000
000027	000023
000030	1025xx
000031	050034
000032	001000
000033	100006
000034	000500
000035	177742

xx (octal)/xxxxxx (binary) = the even BIC number (see "System Memo).

yy (octal)/yyyyyy (binary) = the odd BIC number (see "System Memo").

zz (octal)/zzzzzz (binary) = the device address of the magnetic tape on which the SGL is mounted (see "System Memo").

10. Press the RESET switch.

11. Press the STEP RUN switch.

12. Press the START switch.

The key-in loader now begins executing and loading data from the SGL library on magnetic tape.

2.4.3.2 Manual Key-in Loader Procedures on the V77-200, V77-400 and V77-800

1. Turn the POWER switch on the control panel to the RESET position, then to the ON position.

2. Using the OPCOM device (virtual console), depress the P key. The system responds by displaying the current value of the P register. For example:

   P 001010

   This indicates that the P register is set to octal 001010. Note that the operator entries are underlined for clarity in this explanation only.

3. Enter a zero by depressing the 0 (zero) key and the period (.) key. The display shows:

   P 001010 0.

4. Depress the C key. The system responds by displaying the current contents of location zero. For example:

   C 102637

5. Enter the octal value 010030 by depressing the appropriate numeric keys and the comma (,) key. For example:

   C 102637 010030,

6. The system automatically increments the location of the P register by one (1) and displays the new address contents. For example:

   000001 004011

7. Enter the octal value 010010 by depressing the appropriate numeric keys and the comma (,) key. For example:

000001 004011 001010,

8. Provided that the comma is entered after each new value (as in step 7), step 6 is repeated. Repeat step 7 with the next value in Table 2-1 (or Table 2-2 if type 0870-xx Magnetic Tape Unit is used) until the octal value of 000500 and a comma is entered into location 000027. If at any time a period (or other character) is entered instead of a comma, refer to the "Correction Procedures for the V77-800 Manual Loading Procedure".

9. Enter the octal value of 177742 by depressing the appropriate numeric keys and the period (.) key. This enters the value 177742 into location 000030 and terminates the manual entering of the key-in loader data.

10. Depress the P key. The following is displayed:

P 000031

11. Depress the zero (0) key and the period (.) key:

P 000031 0.

This returns the P register to the first location (000000) of the key-in loader.

12. Depress the R key. This causes the key-in loader to begin executing and loading data from the SGL magnetic tape.

2.4.3.3 Correction Procedure for the V70 Series and V77-600 Manual Loading of the Key-in Loader

If at the end of the manual entering of the key-in loader on the V70 series and V77-600, the upper row of indicators do not reflect the value 000031_8, the following procedures allow error recovery:

1. Press the P switch, the DISPL CLR switch, the MEM switch, and the DISPL switch in this order, thus displaying the contents of address 0 (zero).

2. If the value in the upper row of indicators is equal to the octal value which should be in this address, proceed to step 3. If the value in the upper row of indicators is not equal to the value which should be in this address, proceed to step 4.

3. Press the DISPLAY switch, displaying the contents of the next address and incrementing the P register.

4. Repeat steps 2 and 3 until all memory locations are verified.

5. Press the P switch and the DISPL CLR switch.

6. Enter the octal address of the incorrect value by depressing the appropriate switches on the upper row of switches and depress the MEM switch.

7. Press the DISPL CLR switch. Enter the correct octal value into memory by depressing the appropriate switches on the upper row of switches. Depress the ENTER switch.

8. The error is now corrected. Return to step 4 in 2.4.2.1.

2.4.3.4 Correction Procedures for the V77-200, V77-400 and V77-800 Manual Loading of the Key-in Loader

If a period (or another character) is entered instead of a comma (at a time when a comma should be entered), proceed with the following:

1. Depress the P key.

2. Enter the new address into P by depressing the appropriate numeric keys and the period (.) key.

3. Depress the C key.

4. Enter the correct octal value for the address in the P register by pressing the appropriate numeric keys and the comma (,) key.

5. Return to step 6 in 2.4.2.2.

Follow the preceding procedure also to correct a value entry error. Modify step 4 by entering a period (.) instead of a comma. Return to step 10 in 2.4.2.2.

2.4.4 RESPONDING TO I/O INTERROGATION

Once the key-in loader has been entered into the system, it begins loading the data from the designated magnetic tape unit. The initial data input includes the I/O interrogation program. The following occurs:

1. The message:

   I/O INTERROGATION

   is displayed on the OPCOM device.

2. In response to this message, the user must enter all five of the following directives:

```
DIR,TY00A,01 (Use "DIR,CR00A,030" if a card reader is to
              be used)
LIB,MT00A,010
ALT,MT00A,010
LIS,LP00A,035 (Use "LIS,TY00A,01" if the system does not
               have a line printer)
SYS,D00H,016,01 (or "SYS,D00F,016,020".  Consult the
                 "System Memo".)
```

NOTE

If Type 0870 Magnetic Tape is used, enter all magnetic tape directives in the format:

```
directive,<driver name>,<device address>,<bic address>
```

These directives indicate:

- Directive input (DIR) is assigned to the OPCOM device.
- System generation library input is assigned to the master magnetic tape unit.
- The system generation library is not to be user-modified. Therefore, the user device (ALT) is assigned to the master magnetic tape unit.

- Listing output (LIS) resulting from the system generation process is assigned to the master line printer.

- The generated system (including all libraries and a copy of the nucleus) output (SYS) is assigned to the system disk.

3. If the error message SG02 or SG04, or if the system repeats the message:

   I/O INTERROGATION

   an invalid parameter for one or more of the directives has been entered. To correct the error, proceed as follows:

   - Check the syntax of the directives. Correct any errors and re-enter all five directives.

   - If a hardware device address or a BIC address is incorrect, consult the "System Memo" for the particular system. Use the information from Section 4 to correct the assignments in error. Re-enter all five directives.

2.4.5 ENTERING THE SYSTEM GENERATION HEADER

After the I/O interrogation directives have been successfully processed, the following message is displayed on the OPCOM device:

INPUT 40 CHARACTERS OF SYSGEN HEADER

A header, which is to be used as a header for the system generation listing output, must be entered. For example:

SYSTEMØGENERATIONØVORTEX,ØØ10/1/79

Note that the header can be less than, but may not exceed, 40 characters in length including space characters. The system accepts maximum of 40 characters. When the fortieth character or a carriage return is entered, the system automatically proceeds to the next event and does not recognize any additional characters.

2.4.6 ENTERING THE REQUIRED INPUT DIRECTIVES

The system is now prepared to build the VORTEX nucleus. However, before this processing can begin, the system requires specific information from the user as to the exact configuration of the nucleus. The user specifies the required information in the form of input directives.

The system displays the following message on the Operator Communication (OC) device:

```
INPUT DIRECTIVES
```

Since the OC device has been specified as the directive input (DIR) device, the system awaits user input through this device before proceeding further. The required input for this example is as follows:

NOTE

> Prior to entering any of the directives, refer to Section 4 of the "System Memo" for the particular system being generated. Validate the parameters for each directive with the specific information in the "System Memo" and change the parameters as necessary to insure that they relate to the system concerned.

1. The Memory (MRY) directive specifies:

   - The amount of memory available to the particular system.
   - The size of the foreground blank common area (used as shared storage among foreground tasks).
   - The number of pages allocated for virtual nucleus overlay (VNO) tasks.

   Enter this directive via the DIR device as follows:

   MRY,074777,512,128,V75,20,127

   The parameters in this MRY directive indicate:

- "074777" informs the system that 30.5K [32K-.5K(DCM page) - 1K(AID) = 30.5K] of memory is available for the nucleus, the AID utility is to be resident above the nucleus, and a page is to be allocated to DCM's usage.

- "512" specifies that 512 words are to be reserved for the foreground blank common.

- "128" indicates that 112K words of memory are available to VORTEX.

- "V75" indicates that this system uses a long TIDB.

- "20" informs the system that disk storage equivalent to 20 pages (7680 words) are allocated for virtual nucleus overlay tasks images on the disk.

- "127" indicates that page number 127 is the last page allocated to virtual nucleus overlay (VNO) tasks. Sufficient pages are allocated to store all the VNO tasks. The pages allocated for VNO tasks are not available to VORTEX for dymamic allocation.

2. The Clock (CLK) directive specifies:

   - The number of microseconds in the basic clock interval (hardwired).

   - The number of microseconds in the free-running counter increment period (hardwired).

   - The number of milliseconds in the user interrupt interval.

   Enter this directive via the DIR device as follows:

   CLK,100,100,20

   The parameters in the CLK directive indicate:

   - "100" is a hardwired value and indicates that there are 100 microseconds in the basic clock interval.

   - "100" is a hardwired value and indicates that there are 100 microseconds in the free-running counter increment period (this is not implemented under VORTEX, and therefore, should always be the same as the first parameter).

- "20" indicates that there are to be 20 milliseconds in the user interrupt interval. The user interrupt interval is the amount of time one task runs before another task is dispatched to run. A real time clock interrupts signals the end of an interval.

3. The Equipment (EQP) directives define the peripherals on the system. Enter one EQP directive for each peripheral in this example as follows:

NOTE

Refer to the "System Memo" for exact configuration information.

- Disk

EQP,D0C,015,1,020,15

The parameters in this directive indicate:

- The device name is D0C (D = disk, 0 = disk controller 0, and C = device feature number 2822-xx).

- "015" is the physical device address of this disk unit. This value is octal, when stated with a leading zero.

- "1" indicates that there is one disk unit attached to the controller.

- "020" is the device address of the BIC associated with this disk. This value is octal when stated with a leading zero. This disk does not use a BIC.

- "15" indicates the number of retries for I/O operations associated with this device.

- Magnetic Tape

EQP,MT0A,010,1,024,80 (unlabelled tape)
or
EQP,MT0B,010,1,022,20 (labelled tape)

The parameters in this directive indicate:

- The device name is MT0A (MT = magnetic tape, 0 = controller 0, and A = unlabelled magnetic tape).
- "010" is the physical device address of the magnetic tape unit.
- "1" indicates that 1 magnetic tape unit is connected to the controller.
- "024" is the device address of the BIC associated with this magnetic tape unit.
- "80" indicates that the I/O operation should be retried 80 times before an error or a device busy indication is received.

■ Teletypewriter

EQP,TY0A,01,1,0,0

The parameters in this directive indicate:

- The device name is TY0A (TY = teletypewriter, 0 = teletypewriter controller 0, and A = an ASR-33 or ASR-35 or a compatible CRT device).
- "01" is the physical device address of the teletypewriter, which is the OC device for the system.
- "1" indicates that there is one teletypewriter connected to the controller.
- There is no BIC associated with this device.
- Retries are not possible.

■ Line Printer

EQP,LP0A,035,1,026,0

The parameters in this directive indicate:

- The device name is LP0A (LP = line printer, 0 = line printer controller 0, and A = type number 2819-xx line printer).

- "035" is the physical device number of the line printer.

- "1" indicates that there is one line printer connected to this controller.

- "026" is the physical device address of the BIC associated with this device.

- Retries are not possible.

■ Data Communications Multiplexer

EQP,MX0A,076,1,0,0

The parameters in this directive indicate:

- The device name is MX0A (MX = data communications multiplexor (DCM), 0 = data communications multiplexor controller 0, and A = feature number F3000-xx).

- "076" is the physical device number of the data communications multiplexor.

- "1" indicates that there is one DCM connected to this device address (070). Additional DCM hardware must have different device addresses, and therefore, separate EQP directives.

- "0" indicates that there is no BIC associated with this device.

- "0" indicates that retry is not possible. Retry is handled by the operating system data communications software and is not user selectable during system generation.

■ Card Punch

EQP,CP0A,031,1,022,0

The parameters in this directive indicate:

- The device name is CP0A (CP = card punch, 0 = card punch controller 0, A = type number 2813-xx card punch).

- "031" is the physical device number of the card punch.

- "1" indicates there is one card punch connected to this controller.

- "022" is the physical device address of the BIC associated with this device.

- "0" indicates retries are not possible.

■ Card Reader

EQP,CR0A,030,1,026,0

The parameters in this directive indicate:

- The device name is CR0A (CR = card reader, 0 = card reader controller 0, A = type number 2812-xx card reader).

- "030" is the physical device number of the card reader.

- "1" indicates there is one card reader connected to this controller.

- "026" is the physical device address of the BIC associated with this device.

- "0" indicates that retries are not possible.

■ Paper Tape Reader/Punch

EQP,PT0A,037,1,026,0

The parameters in this directive indicate:

- The device name is PT0A (PT = paper tape reader/punch, 0 = paper tape reader/punch controller 0, and A = feature number F3082-xx, F3083-xx, F3084-xx or model number 70-6320 paper tape reader/punch).

- "037" is the physical device number of the paper tape reader/punch.

- "1" indicates there is one paper tape reader/punch connected to this controller.

- "026" is the physical device address of the BIC associated with this device.

- "0" indicates that no retries are possible.

■ CRT keyboard/display

EQP,CT0A,02,1,0,0

The parameters in this directive indicate:

- The device name is CT0A (CT = CRT keyboard/display, 0 = CRT keyboard display controller 0, and A = type number 2817-xx, 2818-xx, 2838-xx, 2839-xx or 2840-xx CRT keyboard/display).

- "02" is the physical device number of the CRT keyboard/display.

- "1" indicates there is 1 CRT keyboard/display connected to this controller.

- "0" indicates there is no BIC associated with this device.

- "0" indicates no retries are possible.

4. The Priority Interrupt Module (PIMs) directives define the interrupt system architecture by specifying:

   ■ The number of PIMs in the system

   ■ The interrupt levels to be enabled at system initialization

   ■ The interrupts to be manipulated by the interrupt handler software within the system.

   Enter the PIMs for this example as follows:

■ Disk

PIM,01,TBD0C,01,0

The parameters in this directive indicate:

- "02" specifies PIM number 0 and line number 1 within PIM number 0.

- "TBD0C" is the name of the task which handles the interrupt from this particular PIM.

- "01" is the event word for this interrupt.

- "0" indicates that an interrupt from the disk is handled by a common interrupt handler.

PIM,006,TBD0C,01,0

The parameters in this directive are the same as those in the preceding directive with the following exceptions.

- "006" indicates PIM number 0 and line number 6 within PIM number 0.

NOTE

The preceding example is for disk type 2822-xx. For systems with other model disks, one of the following may be used:

```
PIM,00,DMDI0H,01,2   (for types 2825-xx, 2842-xx,
                      2826-xx, 2843-xx, and 8433-xx)

PIM,00,TBD0E,01,0    (for types 2823-xx and 2824-xx)

PIM,00,TBD0F,01,0    (for F3094-xx and F3096-xx)
```

- Line Printer

PIM,003,TBLP0A,01,0

The parameters in this directive indicate:

- "003" specifies PIM number 0 and line number 3 within PIM number 0.

- "TBLP0A" is the name of the task which handles the interrupt. "LP0A" specifies the line printer defined in the EQP directive.

- "01" is the event word for this interrupt. In this case, it indicates that the BIC activity is complete.

- "0" indicates that this interrupt from the line printer is to be handled by a common interrupt handler.

■ Magnetic Tape

PIM,012,TBMT0A,01,0

The parameters in this directive specify:

- "012" specifies PIM number 1 and line number 2 within PIM number 1.

- "TBMT0A" is the name of the task which handles the interrupt. "MT0A" specifies magnetic tape.

- "01" is the event word for this interrupt. In this case, it indicates that the BIC activity is complete.

- "0" indicates that this interrupt is to be handled by a common interrupt handler.

■ Teletypewriter

PIM,026,TBTY0A,01,0

The parameters in this directive indicate:

- "026" specifies PIM number 2 and line number 6 within PIM number 2.

- "TBTY0A" is the name of the task which handles this interrupt. "TY0A" specifies the teletypewriter.

- "01" is the event word for this interrupt. In this case, it indicates that the read buffer is ready.

- "0" indicates that this interrupt is to be handled by a common interrupt handler.

PIM,027,TBTY0A,02,0

The parameters in this directive are the same as those in the preceding PIM directive for the teletypewriter, with the following exceptions:

- "027" specifies PIM number 2 and line number 7 within PIM number 2.

- "02" is the event word for this interrupt. In this case, it indicates that the write buffer is ready.

NOTE

Both PIM directives for the teletypewriter must always be entered.

- Data Communications Multiplexer (DCM)

  Six PIM directives must be specified for each DCM defined by an EQP directive. All six for this example are as follows:

```
PIM,060,C52LIP,000,2
PIM,061,C52LIP,001,2
PIM,062,C52LIP,002,2
PIM,063,C52LIP,004,2
PIM,064,C52LIP,003,2
PIM,065,C52CIH,005,2
```

  The parameters in these directives specify:

  - "060" through "065" indicate a PIM number of 6 in all cases. The "0" through "5" designations indicate the PIM line number for each type of DCM interrupt.

  - "C52LIP" is the name of the interrupt handler which services the first five interrupt types. "C52CIH" is the name of the interrupt handler which services the control interrupt.

  - Each event word (the third parameter) has a unique meaning which defines the interrupt type as follows:

    1) "000" = the input byte count is zero
    2) "001" = the output byte count is zero
    3) "002" = an input line error has occurred
    4) "004" = a control character has been detected
    5) "003" = a status change has occurred
    6) "005" = a control interrupt

  - "2" indicates that directly connected interrupt handlers are to service these interrupts.

- Card Reader

PIM,03,TBCR0A,01,0

The parameters in this directive indicate:

- "03" specifies PIM number 0 and line number 3 within PIM number 0.
- "TBCR0A" is the name of the task which handles the interrupt. "CR0A" specifies the card reader.
- "01" is the event word for this interrupt. In this case, it indicates that the BIC activity is complete.
- "0" indicates that this interrupt is to be handled by a common interrupt handler.

- Card Punch

PIM,05,TBCP0A,01,0

The parameters in this directive indicate:

- "05" specifies PIM number 0 and line number 5 within PIM number 0.
- "TBCP0A" is the name of the task which handles the interrupt. "CP0A" specifies the card punch.
- "01" is the event word for this interrupt. In this case, it indicates that the BIC activity is complete.
- "0" indicates that this interrupt is to be handled by a common interrupt handler.

- CRT Keyboard/display

PIM,024,TBCT1A,01,0

The parameters in this directive indicate:

- "024" specifies PIM number 2 and line number 2 within PIM number 2.
- "TBCT1A" is the name of the task which handles this interrupt. "TBCT0A" specifies the CRT keyboard/display.

- "01" is the event word for this interrupt. In this case, it indicates that the read buffer is ready.

- "0" indicates that this interrupt is to be handled by a common interrupt handler.

PIM,025,TBCT1A,02,0

The parameters in this directive are the same as those in the preceding PIM directive for the CRT keyboard/display, with the following exceptions:

- "025" specifies PIM number 2 and line number 3 within PIM number 2.

- "02" is the event word for this interrupt. In this case, it indicates that the write buffer is ready.

NOTE

Both PIM directives for the CRT keyboard/display must always be entered.

- Paper Tape Reader/Punch

PIM,015,TBPT0A,01,0

The parameters in this directive indicate:

- "015" specifies PIM number 1 and line number 5 within PIM number 1.

- "TBPT0A" is the name of the task which handles the interrupt. "PT0A" specifies the paper tape reader/punch defined in the EQP directives.

- "01" is the event word for this interrupt. In this case it indicates character ready.

- "0" indicates that this interrupt from the paper tape reader/punch is to be handled by a common interrupt handler.

5. The Partition (PRT) directives specify the size of each partition on a disk. One PRT directive is required for every partition on every disk connected to the system. For this example, there is one controller with one disk drive associated with that controller. Therefore enter the following directives:

PRT,D00A,5,C	(Operating system: LUN = 1, logical unit name = CL)
PRT,D00B,60,F	(Foreground library: LUN = 106, logical unit name = FL)
PRT,D00C,150,E	(Background library: LUN = 105, logical unit name = BL)
PRT,D00D,100,D	(Object module library: LUN = 104, logical unit name = OM)
PRT,D00E,20,S	(Checkpoint file: LUN = 101, logical unit name = CU)
PRT,D00F,60,B	(System work file: LUN = 102, logical unit name = SW)
PRT,D00G,40,*	(GO unit: LUN = 9, logical unit name = GO)
PRT,D00H,340,*	(System scratch, processor output: LUN = 8,10, logical unit name = SS,PO)
PRT,D00J,8,*	(For Writable Control Store: LUN = 116, logical unit name = WC)
PRT,D00L,260,*	(Binary output: LUN = 7, logical unit name = BO)
PRT,D00M,220,*	(User work area: LUN = 23, logical unit name = SM)
PRT,D00N,200,*	User work area: LUN = 24, logical unit name = AP)
PRT,D00Q,120,*	(VORTEX Libraries: LUN = 189)

The parameter in these PRT directives indicate:

- "D00A" through "D00O" describe the disk and the partitions. "D" indicates a disk; "0" indicates the disk controller number, in this case zero; "0" indicates the number of the disk drive on the controller, in this case zero; "A" through "O" specify the identifying letter assigned to each partition. For the type 2825-xx, 2842-xx, 2826-xx, 2843-xx, and 8433-xx moving head disks, the highest number allowed is 63. For all other disk types, the highest number allowed is 20. Note that the letters A through T can be substituted for the numbers 1 through 20, if desired. If letters are used to identify partitions, the parentheses are omitted; for example, D00A.

- The second parameters in each directive specifies the decimal number of tracks in each partition.

- The third parameters in each directive specifies the protection code for each partition. The asterisk (*) specifies that the particular partition is unprotected. All one letter codes are protection keys, which must be specified by the user when the partition is accessed. The letters themselves have no further significance.

- This example does not utilize all the space on the disk. The user may add more PRT directives as needed.

6. The Assign (ASN) directive equates the logical unit number and the device name assignments with the physical devices. These assignments allow the user to easily access the physical devices.

For this example, enter the following ASN directives:

```
ASN,1:OC=CT10          (OPCOM device)
ASN,2:SI=CT10          (system input)
ASN,3:SO=CT10          (system output)
ASN,5:LO=LP00          (list output)
ASN,6:BI=MT00          (binary input)
ASN,7:BO=D00L          (binary output)
ASN,8:SS=D00H          (system scratch)
ASN,9:GO=D00G          (GO unit)
ASN,10:PO=D00H         (processor output)
ASN,11:DI=TY00         (debugging input)
ASN,12:DO=TY00         (debugging output)
ASN,18:MT=MT00         (magnetic tape drive)
ASN,23:SM=D00M         (user work area)
ASN,24:AP=D00N         (user work area)
ASN,189:TP=D00O        (VORTEX Libraries)
```

The parameters in these ASN directives indicate:

- The first parameter is the logical unit number which is assigned to the specified device.
- The second parameter is a two-letter logical unit name which is assigned to the specified device and correlated with the logical unit number.
- The third parameter is the four-character physical device name obtained from the "System Memo".

The result of the assign directive is to correlate a logical number (and optionally, a logical name) with the physical device.

7. The Define External (DEF) directive defines special conditions relating to specific peripherals in the system. For this example, enter the following:

DEF,V$PTSZ,200

This directive allocates a 200 word area for system patches (PTSZ = patch size). This area is named V$PSTR (PSTR = patch start area).

8. The Virtual Overlay Task Definition (VOL) directive specifies the tasks which are to be declared virtual nucleus overlays (VNOs). These tasks are generally I/O controller tasks. For this example, enter the following:

VOL,VZDC	(disk controller tasks)
VOL,VZMTA	(magnetic tape controller task)
VOL,VZLPA	(line printer controller task)
VOL,V$TYA	(teletypewriter controller task)
VOL,VZFMA	(file maintenance psuedo driver task)
VOL,VZCRA	(card reader controller task)
VOL,VZCPA	(card punch controller task)
VOL,VZPTA	(paper tape reader/punch controller task)

When required, the selected task is loaded into the overlay area in the nucleus and executed. This load is accomplished using the memory mapping hardware; no disk transfers occur.

9. The CPU directive specifies the type of computer which is being used to generate and execute the VORTEX software system. For this example, enter the following directive:

CPU,1

(since the computer is the V77-600 with software operating system nucleus)

10. Individual Software Modules

- Intertask Communications

EQP,IT0E,0,1,0,0

This directive is considered an equipment specification. However, since this is a software module and thus not a true device or even a psuedo I/O controller, it does not require a controller table. The parameters in this directive indicate:

- The software module name is "IT0E" for intertask communications.

- "1" indicates that there is only one IT0E module.

- All other parameters are zero because this is not an I/O controller or device.

DEF,VI$MXQ,024000

This directive specifies that the mailbox queue is to contain 24000 elements.

DEF,VI$NPG,4

This directive specifies that 4 physical pages are to be used for the internal pool.

■ Writable Control Store

DEF,V$$WCS,2

This DEF directive is used to select a WCS module. The value 2 in this directive selects the FORTRAN accelerator (with Floating Point Processor) or with commercial firmware.

ASN,116:WC=D00J

This directive is required to associate logical unit 116 and the name WC with physical device D00J.

EQP,WCS,072,1,0,0

The parameters in this directive indicate:

- The device name is WCS.

- "072" is the physical device number of the WCS.

- "1" indicates there is one WCS connected to this controller.

- "0" indicates there is no BIC associated with this device.

- "0" indicates that no retries are possible.

■ Simultaneous Peripheral Output Overlap Subsystem (SPOOL)

Two EQP directives are required for SPOOL files. These directives are:

```
EQP,SP0A,0,1,0,0
EQP,SP2A,0,1,0,0
```

These directives indicate that SPOOL files numbered 0 and 2 will be used with the SPOOL subsystem.

```
PRT,D00I,380,S
```

This directive creates a partition for the holding of output data for SPOOL. The partition has the name D00I, has 380 tracks, and has a protection key of S.

■ VZFMA File Maintenance Driver

```
EQP,FM0A,0,1,0,0
```

This directive causes a software module to be included in the VORTEX system. Since this is a pseudo I/O driver and requires a controller table, the EQP directive is used rather than the MOD directive. The parameters in this directive indicate:

- The software module name is "FM0A" for the file maintenance pseudo driver.

- "1" indicates there is only one FM0A module.

- All other parameters are zero since this is not an I/O controller nor a device.

```
ASN,115:FM=FM00
```

This directive equates logical unit 115 and the name FM with the physical device FM00.

- VTAM

  In addition to the directives

  EQP,MX0A,076,1,00

  and

  MRY,074777,512,112,V75,20,127

  previously entered, VTAM requires the following directives:

  DEF,V$LCW0,075000

  This directive specifies that 075000 is the address at which the DCM line control block is wired. This information is obtained from the "System Memo".

  The "0" in V$LCW0 corresponds to the fact the multiplexer used with this system is wired to controller number 0. Refer to the EQP,MX0A,076,1,0,0 directive for the system data communication multiplexer that was previously entered.

  DEF,V$POLL,0

  This directive is required to specify the polling mode used for the system.

  ASN,182:MC=MX00

  This directive specifies that logical unit number 182 also has the name "MC". In addition, this logical unit is assigned to the physical device MX00.

- Buffer Interlace Controller (BIC)

  Since this system has BIC device addresses and trap locations that differ from the default values, the following directives are required:

  DEF,!BIC20,0102
  DEF,!BIC22,0112

  These directives specify that BIC device address 20 has the trap location 0212 and BIC device address 22 has the trap location 0112.

- Memory Dump Routine (DSYSTM)

  To include DSYSTM in this system, the following directive is required:

  EQP,RM0C,0,1,0,0

  This directive specifies that the DSYSTM driver is to be included in the system and the model code of the disk which is to contain the dump files is "C".

  PRT,D00K,432,Z

  This directive creates a partition on device D00K which will be used to contain the dump image files. The partition has 432 tracks and a protection key of Z.

  ASN,190:DS=D00K

  This directive assigns logical unit 190 to the partition D00K. In addition, this directive specifies the name "DS" as an alternate name for logical unit 190.

11. To delete the nucleus version of the object modules V$RERF and V$RERS, the following directives are required:

    DEL,V$RERF
    DEL,V$RERS

    For this example, V$RERS will be included at a later time from the object module tape.

12. The End (EDR) directive terminates the input directives, and therefore, must be the last directive entered. Enter this directive as follows for this system:

    EDR,S,20,44,29,L,1

    The parameters in this EDR directive indicate:

    - "20" is the maximum number of partitions on the system disk.
    - "44" is the number of lines per page for the listing output (LO) device, with a typical value of 44 for the line printer and 61 for the teletypewriter.

- "29" designates that the 029 keypunch (Hollerith code) is used for punched cards.
- "L" indicates that map information is to be listed.
- "1" is used as the last parameter for all VORTEX systems except systems with model A or B disks. For systems with these model disks, the last parameter may be "0" if bad track analysis is to be performed.
- "S" specifies that a standard (full) system generation is to be performed.

2.4.7 BUILDING THE VORTEX NUCLEUS

Once the input directives have been entered and processed, the system builds the VORTEX nucleus. At this point in the system generation procedure, the user is not involved. The processing taking place is completely transparent to the user. The outcome of this processing is a generated, application-oriented VORTEX nucleus, which is both memory-resident and stored on the disk.

Completion of the system generation procedure of the VORTEX nucleus is indicated by the following message:

```
VORTEX II PHYSICAL MEMORY ALLOCATION

PAGES(OCTAL)   ALLOCATED TO

        0      PAGE 0 SYSTEM DATA
    1 - 36     UNALLOCATED
   37 - 72     NUCLEUS PROGRAM MODULE
   72 - 75     NUCLEUS TABLE MODULE
       75      GLOBAL FCB PAGE
       75      FOREGROUND BLANK COMMON
  100 - 171    UNALLOCATED

VORTEX SYSTEM READY TO LOAD LIBRARY TASKS

IO10, FILINT
EX02, FILINT                    Standard Messages
IO10, BTTASK
EX02, BTTASK

                                Displayed for non micro-
IO10, WCSRLD                    VORTEX systems that have
EX02, WCSRLD                    included the "DEF,WCS"
                                directive.
```

Note that after the building of the VORTEX nucleus and prior to the loading of the library tasks, certain system utility routines are generated and are made available for execution to facilitate the remainder of the system generation processing. These routines include:

- File Maintenance (FMAIN)
- Job Control Processor (JCP)
- Rotating Memory Analysis and Initialization (RAZI)
- Patch Program (PATCH)
- Load-module Generator (LMGEN)
- Operator Communication Routine (OPCOM)

Detailed information concerning these utilities is contained within the SPERRY UNIVAC VORTEX II Operating System Programmer Reference Manual(UP-8677).

2.4.8 LOADING THE LIBRARY TASKS

When the VORTEX nucleus has been generated and is executing, and after the necessary utility routines have been made available for execution, the following message is displayed:

VORTEX SYSTEM READY TO LOAD LIBRARY TASKS

The user should then enter the date and time using the operator console commands as described in the VORTEX II Operating System Programmer Reference Manual. For this example, enter:

;TIME,1330
;DATE,10/01/79

or other appropriate time and date values.

The user must then proceed as follows:

1. Enter:

   /SFILE,MT,1

   This causes the magnetic tape to move forward one file mark. The magnetic tape is now positioned at the beginning of the first file of the library tasks.

2. Enter:

/ASSIGN,SI,MT

This instructs the system to assign the system input (SI) logical device to the magnetic tape. It also flags the system to begin displaying information concerning the first object module (OM) file.

3. At this point the system displays:

- the name of the first OM file (BSCOM)
- pertinent information concerning this file
- the number of files to skip if this module is to be excluded from the library (2).

4. If this module (BSCOM) is to be included in the OM library, proceed to step 6. If this module is to be eliminated from the library, enter the following:

/SFILE,MT,2

Proceed to step 8.

NOTE

This is an example for illustrative purposes. The module BSCOM must always be included in the system. It is used here because it is the first OM file at this point on every tape.

5. If the module is to be included in the system, enter:

/ASSIGN,SI,MT

This instructs the system to include BSCOM as part of the OM library. The system then processes this module and places it on the disk.

6. The system displays the following:

time /FINI

The user now enters:

/SFILE,MT,1

This positions the magnetic tape to the next file on tape.

7. Enter:

/ASSIGN,SI,MT

8. Return to step 3. Execute steps 3 through 7 until the last OM has been read from magnetic tape. Refer to table 2-3 for a list of those modules to skip and those to include for this example.

9. When the last module has been read from the magnetic tape, enter:

/REW,MT
/FINI

This rewinds the magnetic tape and ends the system generation process.

NOTE

Because some VORTEX utilities are coded in FORTRAN, the FORTRAN runtime support libraries must be loaded, even if the FORTRAN compiler is not included in the system. The following object modules must be loaded.

1. BSCOM

2. Either V$RERS or V$RERF (if the nucleus versions of these object modules have been deleted)

3. One of the following math libraries:

- SFTOM
- ACLOM
- FPPOM
- FPAOM

Table 2-3. VORTEX Object Module Library (Magnetic Tape) (For This Example Only)

File Contents	Include Module	Skip Module
BSCOM (Basic object module job stream)	X	
V$RERS (FORTRAN run time routine with WCS)	X	
V$RERF (FORTRAN run time routine with WCS)		X
SFTOM (Software math object module)	X	
ACLOM (Floating point firmware and accelerator object module)		X
FPPOM (Floating point processor object module)		X
FPAOM (Floating point processor with enhanced firmware object module)		X
VTXTSK (VORTEX system tasks library job stream)	X	
FORTRAN (FORTRAN compiler library job stream)	X	
VSORT (Background SORT utility library job stream)	X	
RPGIV (RPG IV compiler library job stream)		X
VSORT (Foreground SORT utility library job stream)	X	
WCS (V77-WCS and V77-WCS library job stream)	X	
WCS (V70-WCS support library job stream)		X

Table 2-3. VORTEX Object Module Library (Magnetic Tape) (Continued)

File Contents	Include Module	Skip Module
(V70 firmware loading job stream)		X
FMUTIL (FM utility job stream)	X	
COMSY (Source editor job stream)	X	
VTAM (VTAM job stream)	X	

2.5 INFORMATION FROM AN ACTUAL SYSGEN

Figure 2-1 consists of two pages from the "System Memo" associated with a specific system.

When correlating these figures, note especially the use of the information in Figure 2-1 in the following directives:

- Equipment specification (EQP)
- PIM interrupt assignments (PIM)
- Logical unit assignments (ASN)

Device Address Assignments

DA	DEVICE	DA	DEVICE
		040	PIM 1 IA 100-117
01	Teletypewriter	041	PIM 2 IA 120-137
02		042	PIM 3 IA 140-157
03	CRT 70-6402	043	
04		044	All PIM Enable/Disable
05		045	MP
06		046	Memory Map
07		047	RTC
010	MT	050	DAC
011		051	DOM
012		052	
013		053	
014		054	
015	Disk	055	
016		056	
017		057	
020	BIC 1	060	ADC
021		061	MUX
022	BIC 6	062	
023		063	
024	BIC 3	064	
025		065	
026	BIC 4	066	
027		067	
030	card Reader	070	BIC 5
031	Card Punch	071	
032		072	WCS Page 1 and 2
033		073	
034		074	
035	Line Printer	075	
036		076	DCM
037	Paper Tape System	077	Console

Figure 2-1. Hardware Configuration Information from a "System Memo"

PIM PRIORITY ASSIGNMENTS

PIM #1

Device Address 40
Interrupt Address 100-117
Priority Levels:

0. BIC 1 Comp
1. BIC 2 Comp
2. BIC 3 Comp
3. BIC 4 Comp
4. BIC 5 Comp
5. BIC 6 Comp
6. Disk Seek Comp
7.

PIM #2

Device Address 41
Interrupt Address 120-137
Priority Levels:

0.
1.
2. MT Motion Comp
3.
4.
5. PTS Char Ready
6.
7.

PIM #3

Device Address 42
Interrupt Address 140-157
Priority levels:

0.
1.
2.
3.
4. CRT read
5. CRT write
6. TTY read
7. TTY write

BIC ASSIGNMENTS

BIC #1

Device Address 20,21
Controller Assignments:

1. Disk DA 15
2.
3.
4.
5.

BIC #2

Device Address _____
Controller Assignments:

1.
2.
3.
4.
5.

BIC #3

Device Address 24,25
Controller Assingments:

1. MT DA 10
2.
3.
4.
5.

BIC #4

Device Address 26,27
Controller Assignments:

1. Card Reader DA 30
2. Paper Tape System DA 37
3. Line Printer DA 35
4.
5.

BIC #5

Device Address 70,71
Controller Assignments:

1.
2.
3. CRT DA 03
4.
5.

BIC #6

Device Address 22,23
Controller Assignments:

1. Card Punch DA 31
2.
3.
4.
5.

Figure 2-1. Hardware Configuration Information from a "System Memo" (Continued)

Section 3
SYSTEM GENERATION PREPARATION

3.1 GENERAL

The purpose of this section is to guide the user in the preparation of system generation (SYSGEN) input directives. Sections 2, 6, and 7 contain supporting information for and should be used in conjunction with this section.

3.2 PREPARATION OF DIRECTIVES

All input directives should be prepared in advance of performing the SYSGEN. Table 3-1 lists all the hardware drivers according to their software model code and type/feature numbers. For example:

CRnA (Card Reader)

2812-xx

"CRnA" is the software model code for the card reader and "2812-xx" is the card reader's identifying type number. Entries are made under the name of each directive and across from each device as follows:

- "N/A" indicates "not applicable".
- "Refer to" means that the references to paragraph numbers, section numbers, or tables are where detailed information concerning the indicated device/directive combination can be located.
- "XXnm" is the device mnemonic in the form of "XX", the controller number in the form of "n", and the software model code in the form of "m".
- "TBXXnm" is the name of the task handling the interrupt associated with the indicated device.
- "VOL,taskname" is the form of the VOL directive to be used in conjunction with the indicated device. "taskname" is the name of the VNO task associated with the device.

Table 3-2 lists pseudo drivers and spooler names and their applicable directives.

Table 3-1. Directive/Device Cross Reference

SOFTWARE MODEL CODE OR NAME	DIRECTIVES					
	ASN	DEF	EQP	PIM	PRT	VOL
BIC (Buffer Interlace Controller)	N/A	!BICxx Refer to: 7.9 Table 7-1	N/A	N/A	N/A	N/A
CInA	Refer to: 6.3.1	None	CInm Refer to: 6.3.10	N/A	N/A	N/A
COnA	Refer to: 6.3.1	None	COnm Refer to: 6.3.10	N/A	N/A	N/A
CPnA (Card Punch) 2813-xx	Refer to: 6.3.1 2.4.5 Table 6-1	None	CPnm Refer to: 6.3.6 Table 6-4	TBCP0A Refer to: 6.3.10	N/A	VOL,VZCPA Refer to: 6.3.12 Table 6-8
CRnA (Card Reader) 2812-xx	Refer to: 6.3.1 2.4.5 Table 6-1	None	CRnm Refer to: 6.3.6 Table 6-4	TBCR0A Refer to: 6.3.10	N/A	VOL,VZCRA Refer to: 6.3.12 Table 6-8
CTnA/CTnB (CRT) 2817-xx/ 2818-xx/ 2838-xx/ 2839-xx/ 3840-xx	Refer to: 6.3.1 2.4.5 Table 6-1	None	CTnm Refer to: 6.3.6 Table 6-4	TBCT0A/TBCT0B 2 PIMs Refer to: 6.3.10	N/A	VOL,V$TYA VOL,V$TYB Refer to: 6.3.12 Table 6-8
DnA F3098x	Refer to: 6.3.1 Table 6-2	None	DnA Refer to: 6.3.6 Table 6-4	TBD0A Refer to: 6.3.10	Refer to: 2.4.5 6.3.11 Table 6-5	VOL,VZDA Refer to: 6.3.12 Table 6-8
DnB 70-7600 70-7610	Refer to: 6.3.1 Table 6-2	None	DnB Refer to: 6.3.6 Table 6-4	TBD0B Refer to: 6.3.10	Refer to: 2.4.5 6.3.11 Table 6-5	VOL,VZDB Refer to: 6.3.12 Table 6-8

Table 3-1. Directive/Device Cross Reference (Continued)

SOFTWARE MODEL CODE OR NAME	DIRECTIVES					
	ASN	DEF	EQP	PIM	PRT	VOL
DnC (Disk) 2822-xx	Refer to: 2.4.5 6.3.1 Table 6-2	None	DnC Refer to: 6.3.6 Table 6-4	TBD0C Refer to: 6.3.10	Refer to: 2.4.5 6.3.11 Table 6-5	VOL,VZDC Refer to: 6.3.12 Table 6-8
DnD 70-7510	Refer to: 6.3.1 Table 6-2	None	DnD Refer to: 6.3.6 Table 6-4	TBD0D Refer to: 6.3.10	Refer to: 2.4.5 6.3.11 Table 6-5	VOL,VZDD Refer to: 6.3.12 Table 6-8
DnE (Disk) 2823-xx/ 2824-xx 70-7530	Refer to: 2.4.5 6.3.1 Table 6-2	None	DnE Refer to: 6.3.6 Table 6-4	TBD0E Refer to: 6.3.10	Refer to: 2.4.5 6.3.11 Table 6-5	VOL,VZDE Refer to: 6.3.12 Table 6-8
DnF (Disk) F3094-xx/ F3096-xx/ F3310-xx	Refer to: 2.4.5 6.3.1 Table 6-2	V$DSKn,m Refer to: 6.3.4 8.4	DnF Refer to: 6.3.6 Table 6-4	TBD0F Refer to: 2.4.5 6.3.10	Refer to: 2.4.5 6.3.11 Table 6-5	VOL,VZDF Refer to: 6.3.12 Table 6-8
DnH (Disk) 2825-xx/ 2842-xx/ 2826-xx/ 2843-xx/ 8433-xx	Refer to: 6.3.1 Table 6-2	None	DnH/ER0H Refer to: 6.3.6 Table 6-4	DMDI0H Refer to: 2.4.5 6.3.10	Refer to: 2.4.5 6.3.11 Table 6-6	VOL,VZDH VOL,VZERH VOL,V$DI
DnJ F3097-xx	Refer to: 6.3.1 Table 6-2	None	DnJ Refer to: 6.3.6 Table 6-4	TBD0J Refer to: 6.3.10	Refer to: 2.4.5 6.3.11 Table 6-5	VOL,VZDJ Refer to: 6.3.12 Table 6-8
LPnA/LPnB (Printer) 2819-xx/ 2920-xx/ 0786-xx 70-6701	Refer to: 2.4.5 6.3.1 Table 6-1	None	LPnm Refer to: 2.4.5 6.3.6 8.7 Table 6-4	TBLP0A/ TBLP0B Refer to: 2.4.5 6.3.10	N/A	VOL,VZLPA VOL,VZLPB Refer to: 2.4.5 6.3.12 Table 6-8

Table 3-1. Directive/Device Cross Reference (Continued)

SOFTWARE MODEL CODE OR NAME	DIRECTIVES					
	ASN	DEF	EQP	PIM	PRT	VOL
LPnD 70-6602	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0D Refer to: 6.3.10	N/A	VOL,VZLDD Refer to: 6.3.12 Table 6-8
LPnE 70-6603	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0E Refer to: 6.3.10	N/A	VOL,VZLPE Refer to: 6.3.12 Table 6-8
LPnG 70-6603	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0G Refer to: 6.3.10	N/A	VOL,VZLPG Refer to: 6.3.12 Table 6-8
LPnH 70-7702	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0H Refer to: 6.3.10	N/A	VOL,VZLPH Refer to: 6.3.12 Table 6-8
LPnJ 70-66xx	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0J Refer to: 6.3.10	N/A	VOL,VZLPH Refer to 6.3.12 Table 6-8
LPnK	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0K Refer to: 6.3.10	N/A	VOL,VZLPK Refer to: 6.3.12 Table 6-8
LPnL	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0L Refer to: 6.3.10	N/A	VOL,VZLPL Refer to: 6.3.12 Table 6-8
LPnM	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0M Refer to: 6.3.10	N/A	VOL,VZLPM Refer to: 6.3.12 Table 6-8

Table 3-1. Directive/Device Cross Reference (Continued)

SOFTWARE MODEL CODE OR NAME	DIRECTIVES					
	ASN	DEF	EQP	PIM	PRT	VOL
LPnN	Refer to: 6.3.1 Table 6-1	V$SWnm Refer to: 8.5	LPnm Refer to: 6.3.6 Table 6-4	TBLP0N Refer to: 6.3.10	N/A	VOL,VZLPN Refer to: 6.3.12 Table 6-8
MTnA/MTnB (Magnetic Tape) F3088-xx/ F3089-xx/ 0870-xx	Refer to: 2.4.5 6.3.1 Table 6-1	None	TYnm Refer to: 2.4.5 6.3.6 Table 6-4	TBMT0A/ TBMT0B Refer to: 2.4.5 6.3.10	N/A	VOL,VZMTA VOL,VZMTB Refer to: 2.4.5 6.3.12 Table 6-8
MXnA (Data Communications Multiplexor) F3000-xx 70-521x	Refer to: 2.4.5 6.3.1 7.11	V$LCWn V$POLL Refer to: 2.4.5 6.3.4 7.11	MXnm Refer to: 2.4.5 6.3.6 Table 6-4 7.11	C52LIP - 5 PIMs C52CIH - 1 PIM Refer to: 2.4.5 6.3.10 7.11	N/A	N/A
PTnA (Paper Tape Reader/ Punch) F3084-00/ F3082-00/ F3083-00 70-6320	Refer to: 6.3.1 Table 6-1	None	PTnm Refer to: 6.3.6 Table 6-4	TBPT0A Refer to: 6.3.10	N/A	VOL,VZPTA Refer to: 6.3.12 Table 6-8
TYnA/TYnB (Teletypewriter) 2810-xx/ 2811-xx	Refer to: 2.4.5 6.3.1 Table 6-1	None	TYnm Refer to: 2.4.5 6.3.6 Table 6-4	TBTY0A/ TBTY0B 2 PIMs Refer to: 2.4.5 6.3.10	N/A	VOL,V$TYA VOL,V$TYB Refer to: 2.4.5 6.3.12 Table 6-8
WCS (Writable Control Store) F2960-xx 70-4002	Refer to: 7.4	V$$WCS TAPEDA TAPEBA WCSAD Refer to: 7.4	WCS Refer to: 7.4	N/A	N/A	N/A

Table 3-2. Pseudo Drivers/Spooler Names Cross Reference

SOFTWARE MODEL CODE OR NAME	DIRECTIVES					
	ASN	DEF	EQP	PIM	PRT	VOL
FM0A (File Maintenance Pseudo Driver)	Refer to: 6.3.1 7.10	N/A	FM0A Refer to: 2.4.5 6.3.7 7.10	N/A	N/A	VOL,VZFMA Refer to: 2.4.5 6.3.12 7.10
SD0A (Multitask Spooler	Refer to: 6.3.1 7.8	N/A	SD0A Refer to: 6.3.7 7.8	N/A	N/A	N/A
SPnA (Peripheral Spooler)	Refer to: 6.3.1 7.7	N/A	SPnA Refer to 6.3.7 7.7	N/A	N/A	N/A

3.3 SINGLE USE DIRECTIVES

The directives listed here are used only once during any given SYSGEN. These directives are:

- Clock (CLK) specification - Required
- CPU (CPU) directive - Required
- End (EDR) SYSGEN - Required
- Memory (MRY) specification - Required
- No memory (NMRY) directive - Optional

Each of these directives is detailed in Section 6.

3.4 SYSGEN CHECKLIST

Table 3-3 is a checklist for use in the preparation of the input directives. Once these directives have been prepared, the user may proceed to Section 4.

Table 3-3. SYSGEN Directive Checklist

SOFTWARE MODEL CODE	DIRECTIVES										
	ASN	CLK	CPU	DEF	EDR	EQP	MRY	NMRY	PIM	PRT	VOL
BIC (Buffer Interlace Controller)											
CInA (Process I/O)											
COnA (Process I/O)											
CPnA (Card punch) 2813-xx											
CRnA (Card Reader) 2812-xx											
CTnA/CTnB (CRT) 2817-xx/2818-xx/ 2838-xx/2839-xx/ 2840-xx											
DnA (Disk) F3098-xx											
DnB (Rotating Memory) 70-7600 70-7610											
DnC (Disk) 2822-xx											
DnD (Rotating Memory) 70-7510											
DnE (Disk) 2823-xx/2824-xx 70-7530											

Table 3-3. SYSGEN Directive Checklist (Continued)

SOFTWARE MODEL CODE	DIRECTIVES										
	ASN	CLK	CPU	DEF	EDR	EQP	MRY	NMRY	PIM	PRT	VOL
DnF (Disk) F3094-xx/ F3096-xx/ F3310-xx											
DnH (Disk) 2825-xx/2842-xx/ 2826-xx/2843-xx/ 8433-xx											
DnJ (Diskette) F3097-xx											
LPnA/LPnB (Printer) 2819-xx/2820-xx/ 0786-xx/70-6701											
LPnD (Statos-31) 70-6602											
LPnE (Statos-31) 70-6603											
LPnG (Statos-31) 70-6603											
LPnH (Statos-31) 70-7702											
LPnJ (Statos-33) 70-66xx											
LPnK (Statos-41)											
LPnL (Statos-41)											
LPnM (Statos-42)											
LPnN (Statos-42)											

Table 3-3. SYSGEN Directive Checklist (Continued)

SOFTWARE MODEL CODE	DIRECTIVES										
	ASN	CLK	CPU	DEF	EDR	EQP	MRY	NMRY	PIM	PRT	VOL
MTnA/MTnB (Magnetic Tape) F3088-xx/ F3089-xx/0870-xx											
MXnA (Data Communications Multiplexor) F3000-xx											
PTnA (Paper Tape Reader/Punch) F3084-00/ F3082-00/ F3083-00/70-6320											
TYnA/TYnB (Teletypewriter) 2810-xx/2811-xx											
WCS (Writable Control Store) F2960-xx											
Non-device Dependent Directives											
FM0A (File Maintenance Pseudo Driver)											
SD0A (Multi-task Spooler)											
SPnA (Peripheral Spooler)											

Section 4
SYSTEM GENERATION PROCEDURES

4.1 INTRODUCTION

Prior to using the instructions in this section, the user should read the previous sections in this manual and become thoroughly familiar with the sample system generation (SYSGEN) in Section 2. This section merely extends the example so that any VORTEX configuration can be generated.

This section consists of a series of general steps or guidelines for generating VORTEX systems. Applicable references to specific details presented elsewhere in this manual are given whenever required.

It is important to note that all user input information should be fully prepared before any attempt is made to use the steps in this section.

4.2 EQUIPMENT PREPARATION

The following should be checked before beginning the SYSGEN process:

1. CPU power

   - Ensure that the power switches at the rear of the CPU cabinet are switched to their ON positions.

   - Ensure that the power switch on the front of the cabinet is switched to its ON position.

   - Reset the CPU using the RESET function switch on the control panel.

2. Ensure that power is on to all peripherals to be used for SYSGEN. Also ensure that all peripherals are online.

3. Ensure that all disk packs to be used are appropriately formatted. For formatting details, refer to Appendix A.

4.3 SYSTEM GENERATION (SYSGEN)

This subsection lists the steps required for generating a tailored VORTEX system.

4.3.1 INITIALIZATION

1. Mount the appropriate system generation library (SGL) media (tape, disk or cards) onto the SGL input unit. (For details concerning the SGL media, refer to the "Release Bulletin" for the particular system being generated).

2. Enter the key-in loader for the hardware configuration to be used. The loaders are detailed in Section 2 and Table 4-1.

3. When the key-in loader is executed, it begins to input from the SGL input unit.

4.3.2 I/O INTERROGATION

1. The message:

   IO INTERROGATION

   is displayed to the user.

2. Respond by entering all five (5) SYSGEN I/O directives as follows:

   - DIR,<driver name>,<device address>,[bic address]
   - LIB,<driver name>,<device address>,[bic address]
   - ALT,<driver name>,<device address>,[bic address]
   - LIS,<driver name>,<device address>,[bic address]
   - SYS,<driver name>,<device address>,<bic address>

   Refer to Section 6 for details concerning these directives.

3. The system next displays:

   INPUT 40 CHARACTERS OF SYSGEN HEADER

4. Enter a header with a maximum of 40 characters (including blanks or spaces) into the system. This header is displayed at the top of each page of the SYSGEN listing.

Table 4-1. System Generation Key-In Loaders

Address	Card Reader	RMD* (70-76x0)	RMD F3094-xx F3096-xx (70-76x3)	All Magnetic Tape Devices Except Type 8070-xx	Magnetic Tape Type 0870-xx
000000	010054	010064	010064	010030	010035
000001	001010	140066	140066	001010	001010
000002	001114	001010	001010	001114	001114
000003	040054	001114	001114	040030	040035
000004	001000	001000	001000	001000	001000
000005	000012	000012	000012	000012	000012
000006	000000	000000	000000	000000	000000
000007	006010	006010	006010	006010	006010
000010	000300	000300	000300	000300	000300
000011	050053	050065	050065	050027	050034
000012	1002zz	1004zz	1004zz	1041zz	1041zz
000013	002000	1002zz	010063	1000zz	1000yy
000014	000046	010063	110072	001000	010034
000015	1025zz	110072	1031zz	000021	1031xx
000016	002000	1031zz	1002zz	1025zz	006010
000017	000046	101uzz	101dzz	057027	007000
000020	1026zz	000023	000023	040027	1031yy
000021	004044	001000	001000	1011zz	1000xx
000022	004444	000017	000017	000016	1000zz
000023	057053	1025zz	1025zz	1012zz	1012zz
000024	005001	150071	150071	100006	000030
000025	040053	001016	001016	001000	115000
000026	004450	000012	000012	000021	001000
000027	002000	1000yy	1000yy	000500	000023
000030	000046	1003zz	005000	177742	1025xx
000031	1026zz	010064	010064		050034
000032	004044	110072	110072		001000
000033	004450	1031zz	1031zz		100006
000034	002000	010065	010065		000500
000035	000046	1031xx	1031xx		177742
000036	1022zz	120070	120070		
000037	057053	005012	005012		
000040	040053	1031yy	1031yy		
000041	067053	1000xx	1000xx		
000042	040053	1000zz	1000zz		
000043	001000	1014zz	1014zz		
000044	000013	000043	000043		
000045	1011zz	1025zz	1025zz		
000046	000000	150071	150071		
000047	1016zz	001016	001016		

Table 4-1. System Generation Key-In Loaders (Continued)

Address	Card Reader	RMD* (70-76x0)	RMD F3094-xx F3096-xx (70-76x3)	All Magnetic Tape Devices Except Type 8070-xx	Magnetic Tape Type 0870-xx
000050	100006	000012	000012		
000051	001000	060065	060065		
000052	000045	040064	040064		
000053	000500	010064	010064		
000054	177742	140067	140067		
000055		001016	001016		
000056		100006	100006		
000057		050064	050064		
000060		040063	040063		
000061		001000	001000		
000062		100006	100006		
000063		000001	000001		
000064		000001	000001		
000065		000500	000500		
000066		000037	000037		
000067		000060	000060		
000070		000074	000074		
000071		007760	007760		
000072		0v0000	ww0000		

* = Rotating Memory Device

xx = The even BIC number of this device (see "System Memo")

yy = The odd BIC number of this device (see "System Memo")

zz = Device address of the device used to read the SGL media (see "System Memo")

v = RMD unit number in unit select instruction (V=0 for unit 0, V=1 for unit 1)

ww = Drive (bits 15-14)/platter (bit 13) (i.e., platter 1, drive 00-02)

Note: To SGEN VORTEX II VNO from magnetic tape or cards, change bootstrap location 2 from 1106 to 1114; to SGEN VORTEX II VNO from RMD, change location 3 from 1106 to 1114.

4.3.3 INPUT DIRECTIVES

1. The system displays:

   INPUT DIRECTIVES

2. The user must respond by entering all required directives for the specific system being generated. Table 4-1 lists the input directives, their general formats, and whether they are required or optional. For a detailed explanation of each directive, refer to Section 6.

3. Input directives may be entered manually through the OPCOM keyboard, on punched cards via the card reader, on paper tape via the paper tape reader, or from magnetic tape via the magnetic tape device, depending upon the specific system configuration. Refer to Appendix C for procedures describing SYSGEN on a system with one tape drive and no card reader.

4. Once all directives have been input and the final EDR directive has been recognized by the system, generation of the nucleus begins.

5. The SYSGEN process proceeds internally and automatically until the nucleus and basic system utility routines have been generated.

6. The system displays:

   VORTEX PHYSICAL MEMORY ALLOCATION

PAGES (OCTAL)	ALLOCATED TO
0	PAGE 0 SYSTEM DATA
1 - 11	UNALLOCATED
12 - 63	NUCLEUS PROGRAM MODULE
63 - 73	NUCLEUS TABLE MODULE
73	GLOBAL FCB PAGE
74	FOREGROUND BLANK COMMON
100 - 157	UNALLOCATED
160 - 177	VIRTUAL OVERLAY TASKS
200 - 337	UNALLOCATED

NOTE

The values in this display depend upon the individual system.

Table 4-2. Input Directive Summary

Input Directive	General Format	Req'd	Opt
ADD	ADD,<name>		X
ASN	ASN,<lun>[:luname]=<devl>	X	
CLK	CLK,<clock>,<counter>,<interrupt>	X	
CPU	CPU<x>		X
DEF	DEF,<name>,<value>		X
DEL	DEL,<name>		X
EDR	EDR,<type>,<part>,<list>,<kpun>,<map> [,<analysis>]	X	
EQP	EQP,<name>,<address>,<units>,<bic add>, <retr>[,<alg>][,<mul>]	X	
LAD	LAD,<name>		X
LDE	LDE,<name>		X
LRE	LRE,<name>		X
MOD	MOD,<name>,<address>,<units>,<bic add>, <retr>],<alg>][,<mul>]	X	
MRY	MRY,<mem>,<yyy>,<size>,[V75],<npages>,<lpg>	X	
NMRY	NMRY,<pageno>,<numb>		X
PIM	PIM,<p>,<q>,<r>,<s>	X	
PRT	PRT,<Dcup>,<s>,<k>	X	
REP	REP,<name>		X
VOL	VOL,<taskname>		X

4.3.4 SYSTEM LIBRARY PROCESSING

1. The following message is displayed:

```
VORTEX SYSTEM READY TO LOAD LIBRARY TASKS
```

2. Enter the time and date using the operator console commands as described in the VORTEX II Operating System Programmer Reference Manual. For example:

```
;TIME,1330
;DATE,10/01/79
```

3. Enter from the OC device:

```
/SFILE,MT,1
/ASSIGN,SI,MT
```

These entries position the magnetic tape to the next file (the first system library routine) and assign the system input file to the SGL magnetic tape.

4. At this point the system automatically begins displaying information concerning each system library routine on the SGL tape. For example, the following shows the display information for the first library routine "BSCOM":

```
20:14:17/JOB,BSCOM.  LOAD THE BASIC OM LIBRARY TASKS
20:14:17/KPMODE,1
/C,  EXECUTE--TO LOAD THE BASIC OM LIBR ROUTINES (OM IS
/C,  INITIALIZED) THIS JOB STREAM MUST BE EXECUTED PRIOR
/C,  TO ANY SUBSEQUENT JOBS
/C,  TO BYPASS JOB STREAM SKIP 2 FILES
20:14:18/ASSIGN,SI,OC
```

All of the above is automatically displayed by SYSGEN. If this particular routine is not to be included in the system, the user must skip two (2) files by entering the following:

```
/SFILE,MT,2
```

NOTE

BSCOM must be included in all VORTEX systems. This is an example only.

To include the BSCOM routine in the particular system being generated, the following must be entered:

```
/ASSIGN,SI,MT
```

Note that in the example displayed by the SYSGEN routines, SI is automatically reassigned to OC (the OPCOM device). This is performed so that the user can enter the response. In turn, if the user wishes to include the routine in the system, he must reassign SI back to the magnetic tape unit. The user must perform this after the "SFILE" command if the routine is to be skipped.

5. After each routine in included or skipped, enter:

```
/SFILE,MT,1
/ASSIGN,SI,MT
```

   to receive instructions and information about the next SGL routine.

6. This system display/user response interchange continues until all routines on the SGL tape (or tapes) have been acknowledged either through a rejection (skip) or through an inclusion (assign).

7. When the last system library routine on any SGL tape has been processed, the user must enter:

```
/REW,MT
/FINI
```

If another tape is mounted, the user must enter:

```
/SFILE,MT,1
/ASSIGN,SI,MT
```

This restarts the processing of system library routines.

NOTE

Optional software tapes are processed at this time in the same way the required system library software is processed. The tapes are mounted and the user proceeds as described in the preceding Step 3.

4.3.5 USER LIBRARIES CREATION

Once the system libraries have been created and processed, system generation for the VORTEX operating system is complete. However, as an extension to the SYSGEN process, user libraries still require creation and processing. This is a continual process using the VORTEX Load Module Generator procedures. As such, it is explained in detail in the VORTEX II Operating System Reference Manual (UP-8677).

At this point, the user should return to Section 1 for information concerning additional requirements of certain modules.

Section 5
SYSTEM GENERATION FUNCTIONAL DESCRIPTION

5.1 GENERAL

This section contains a functional explanation of the actual system generation process, including the flow of data through the system. The information in this section is directed toward the experienced user who is familiar with the VORTEX operating system.

5.2 SYSTEM GENERATION COMPONENTS

The system generation process makes use of the following components:

1. Input directives - Commands issued by the user to the system generation routines which specify and define the system environment.

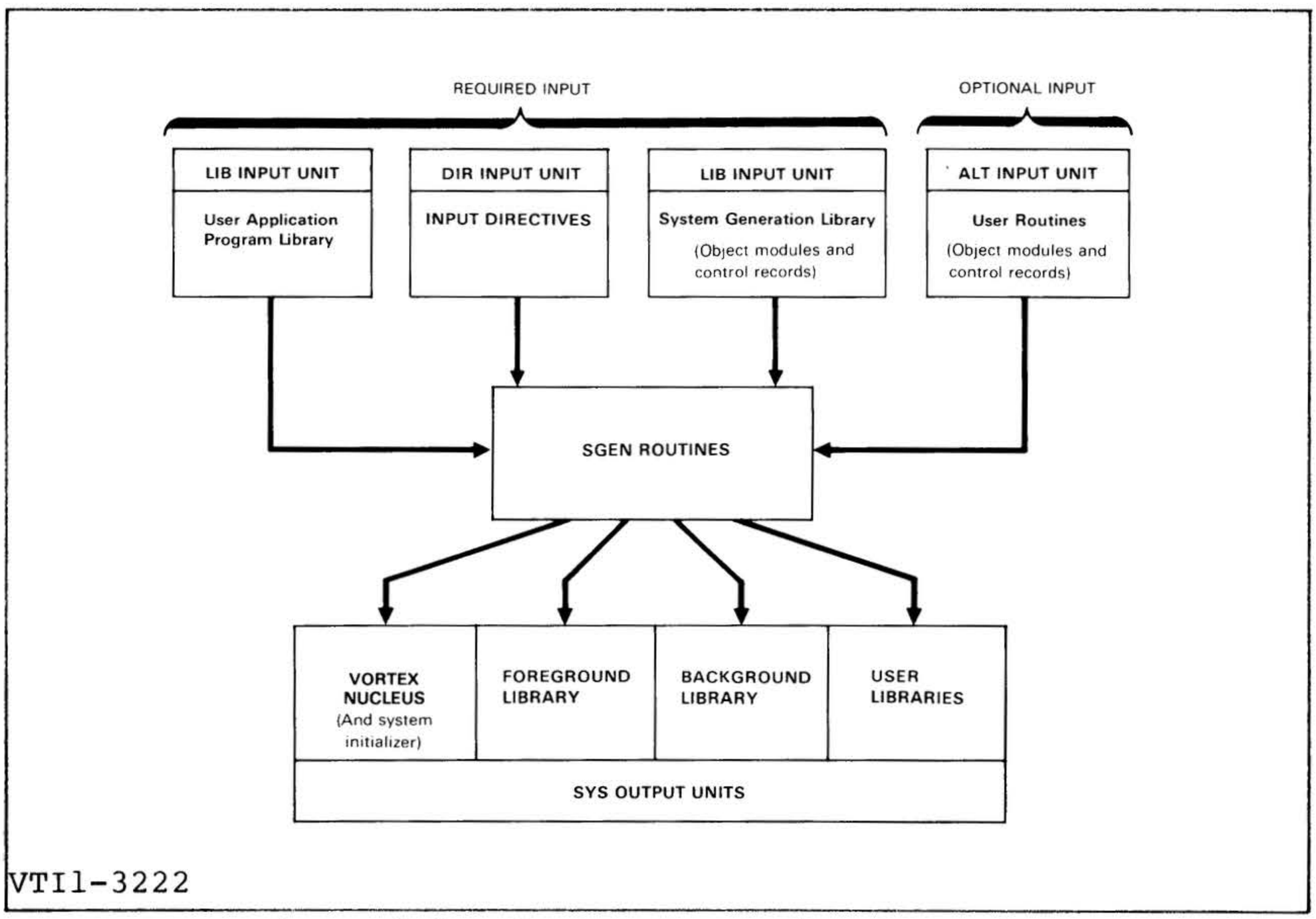


Figure 5-1. System Generation Data Flow

2. System library - Collection of files which consist of:

   a. System generation routines (SGEN) - The master routines which perform the system generation procedures. These routines are used solely for system generation. Once this process is complete, these routines no longer reside in the system.

   b. System generation library (SGL) - The master data processed by SGEN. This library consists of system programs (in object module form) and control records (in alphanumeric form) which are used to create:

      - VORTEX nucleus
      - foreground library
      - background library
      - object module library

The SGL is further described in Paragraph 5.3.1.

3. User library input - Collection of user (application) programs, which form the disk-resident (system-resident) libraries.

During various phases of the system generation process, other routines are used to aid in system generation. However, these routines are not part of the system generation components and should not be confused with those enumerated in the preceding paragraphs.

5.3 SYSTEM GENERATION LIBRARY

The System-generation library (SGL) is a collection of system programs (in object-module form) and control records (in alpha-numeric form) from which a VORTEX system is constructed.

In the case of punched cards or of magnetic tape, the SGL occupies contiguous records, beginning with the first record of the punched cards or magnetic tape.

In the case of disk pack, the SGL occupies contiguous records beginning with the second track. Track 0 contains the partition-specification table (PST) that specifies one partition extending from the second track (track 1) to the end of device.

NOTE

The SGL and the VORTEX system cannot be on the same disk pack during system generation.

The SGL is divided into five functional parts each separated by CTL control records. Each of the five functional parts is described in the following paragraphs and pictured in Figure 5-2.

5.3.1 SGL - PART 1

Part 1 of the SGL consists of the following items:

- VORTEX bootstrap loader
- I/O interrogation routine
- SGEN relocatable loader
- Basic I/O control routine
- Library of peripheral drivers

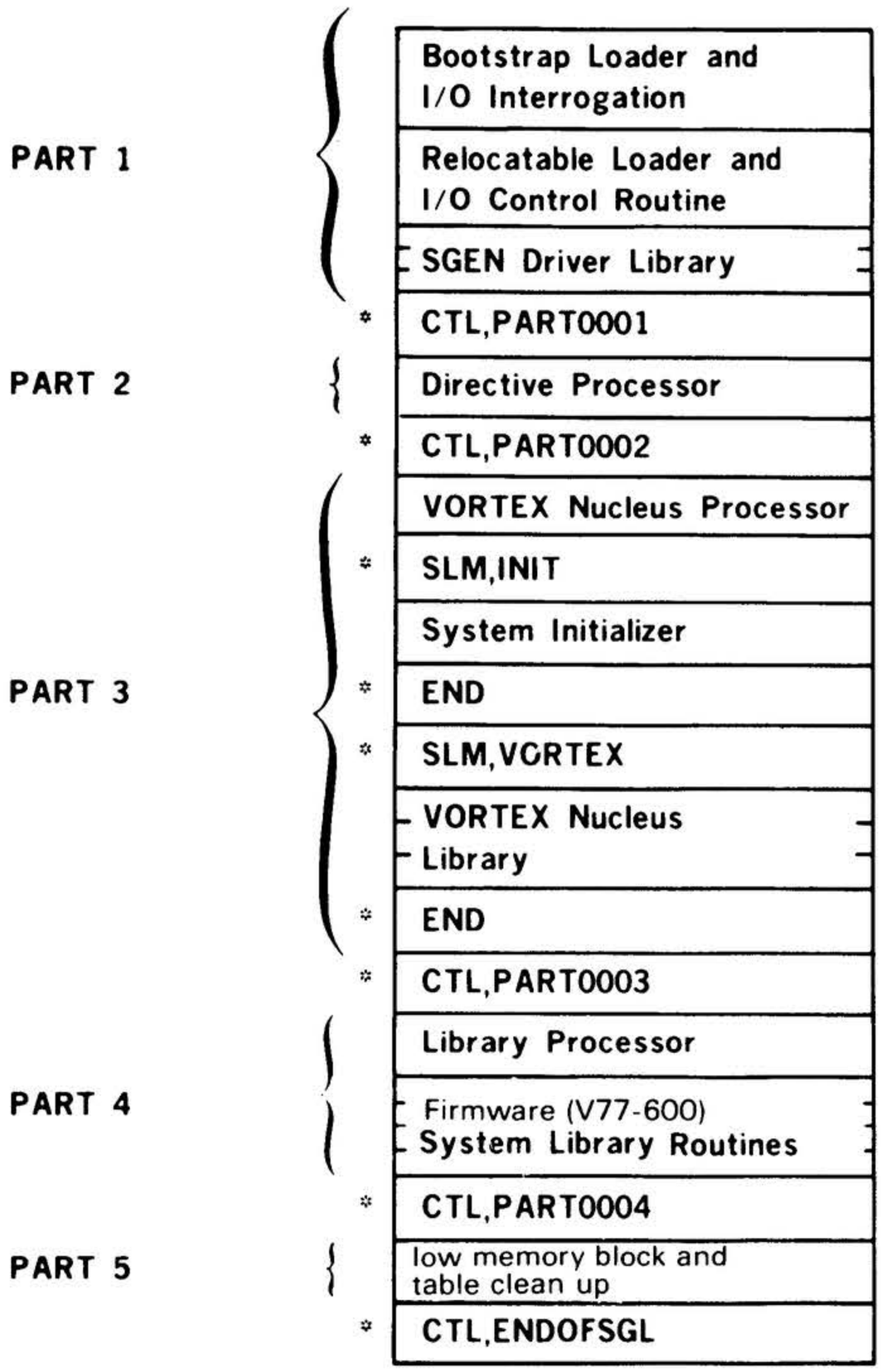
Part 1 of the SGL consists entirely of object modules. It is loaded with the device sensitive key-in loader. The device sensitive key-in loader also serves the bootstrap loader as a read-next-record routine. Because the bootstrap loader calls the key in loader to read records, the bootstrap loader/interrogator is itself device insensitive.

Part 1 of the SGL is terminated by the control record CTL,PART0001.

5.3.2 SGL - PART 2

Part 2 of the SGL contains the directive processor. After the directive processor is itself read into memory, it obtains all subsequent SYSGEN input from the DIR and OC devices.

Part 2 of the SGL is terminated by the control record CTL,PART0002.



NOTE:

* = Alphanumeric control record

Figure 5-2. System Generation Library

5.3.3 SGL - PART 3

Part 3 of the SGL consists of all the system routines and control records required to build the VORTEX nucleus. The following routines are contained in Part 3 of the SGL:

- VORTEX nucleus processor
- SLM control record
- System initializer routines
- END control records
- SLM control records
- VORTEX nucleus routines
- Control record CTL,21
- Nucleus program module

5.3.4 SGL - PART 4

Part 4 of the SGL consists of all the system routines and control records required to build load module libraries on the system disk. The library processor converts these inputs into load modules, catalogs them and enters these load modules into the foreground, background and user libraries. The library processor is followed by groups of control records and object modules. Each group of control records and object modules is referred to as a load module package (LMP).

The library processor also creates an eight page firmware file named WCSIMG on the partition assigned to logical unit 116. This file is used by the microprogram utility to maintain an up to date image of the contents of the Writable Control Store (WCS). (See Writable Control Store and Floating-Point Processor section of the VORTEX II Operating System Programmer Reference Manual for additional information.) The file WCSIMG is created only if the system in use contains a WCS and is not a V77-600 (CPU 1 or CPU 3).

Part 4 of the SGL is terminated by the control record CTL,PART0004.

5.3.5 SGL - PART 5

Part 5 of the SGL contains the memory block and table clean up portion of SGEN. The routines contained in Part 5 of the SGL set final values in the VORTEX page 0 constants and the page allocation table.

Part 5 of the SGL and the SGL itself is terminated by the record CTL,ENDOFSGL.

5.4 OVERVIEW OF THE SYSTEM GENERATION PROCESS

The portion of the initial software which processes the input data and tailors the system is termed the VORTEX system generation component (SGEN). As illustrated in Figure 5-1, the SGEN routines receive input from the designated devices, process this input according to standard procedures and system and user-specified directives, and output the results to the designated devices.

Basically, the system generation process is accomplished in five phases. These phases are usually transparent to the user, unless modification of the system generation library is required. The five phases are:

- I/O interrogation (Phase 1)
- Input directive processing (Phase 2)
- Building the VORTEX nucleus (Phase 3)
- Building the system libraries (Phase 4)
- Resident task configuration (Phase 5)

Each of these phases is detailed in following subsections.

5.4.1 I/O INTERROGATION (PHASE 1)

Phase 1 of system generation uses the following routines from the SGL tape:

NOTE

The key-in loader must already be in the system before this phase can be effected.

1. VORTEX bootstrap loader
2. I/O interrogation
3. SGEN relocatable loader
4. Basic I/O control routine
5. Library of peripheral drivers

These routines are in object module format. They are originally loaded into the system via the device-sensitive key-in loader (see Section 2). This loader reads the required data from the SGL magnetic tape (LIB device).

During Phase 1 processing, two requests to the user are displayed on the device connected to physical device address 01 (teletypewriter or CRT). The first requests the I/O devices which are to be used by SGEN during system generation only.

IO INTERROGATION

The user responds with:

1. The library (LIB) unit assignment
2. User routine (ALT) unit assignment
3. Input directive (DIR) unit assignment
4. The generated VORTEX system output (SYS) unit assignment
5. The system generation listing output (LIS) unit assignment

These 5 assignments may be input by the user in any order. The SYSGEN routines continue to request the device assignments until valid assignments have been made for all 5 units.

The input directives are loaded next and a bad-track analysis is performed. The system then begins loading the VORTEX drivers.

NOTE

There are no expected halts in the process except for the I/O interrogation display. The most common causes for failure at this point are:

1. An incorrect bootstrap routine,

2. The LIB unit (SGL input) is not at the beginning of the file,

3. There is a wrong track or density (magnetic tape SGL),

4. A required peripheral is off-line,

5. Key-in errors have been made during the I/O interrogation input (particularly in the case of the letter O and zero),

6. Failure to include all disks in the SYS directive.

INPUT 40 CHARACTERS OF SYSGEN HEADER --.

The user responds with a header of up to 40 characters.

Control record CTL,PART0001 on the SGL officially terminates Phase 1 of system generation. However, this record is not logged, and therefore, is transparent to the user. The next system/user communication interface indicates to the user that Phase 1 has terminated and Phase 2 is currently being processed.

5.4.2 INPUT DIRECTIVE PROCESSING (PHASE 2)

Basically, Phase 2 of system generation processing specifies the architecture of the VORTEX system based upon user-supplied information that is collected and stored for later use when the system is actually constructed. These directives permit the design of systems enveloping the entire range of VORTEX applications.

At the beginning of Phase 2, the directive processing routine is input and given control. It then obtains all input from the DIR and OPCOM devices and stores this data appropriately.

During Phase 2, "CPU,n" directive is scanned to determine the minicomputer model on which this VORTEX system is to execute. This allows the tailoring of VORTEX for the specific CPU environment in which it is to execute.

During Phase 2, the following message appears on the OPCOM device:

INPUT DIRECTIVES

In response, the user must input all directives necessary for the particular system via the DIR device. These directives are detailed in Section 6. If any input is required from the OPCOM device, this is usually in the form of directive correction from the user. The user may either enter corrected directive or "C" to continue. During this phase, the system may request input from the ALT device. The use of the ALT device is detailed under "Library Modification Directives".

The EDR directive must be the last directive entered during Phase 2. This directive terminates the input directive processing and also terminates Phase 2 of system generation.

Bad track or disk device partitioning analysis is performed following input of the EDR directive. After this process is complete, Phase 2 is terminated. Internally, Phase 2 is terminated via the control record CTL,PART0002 on the SGL.

5.4.3 BUILDING THE VORTEX NUCLEUS AND BASIC LIBRARIES (PHASE 3)

This phase consists of gathering object modules and control records from the system generation library (SGL) tape and from the user input, and constructing the VORTEX nucleus from this data. The SGL items are input via the LIB device and user items via the ALT device (refer to "Library Modification Directive), according to the rules established by the input directives.

The major components of Phase 3 include:

- VORTEX nucleus processor - SGEN load modules for processing of the nucleus modules.
- SLM control record - A start of load module record for indicating the beginning of the system initializer part of this process.
- System initializer routines - Object modules which are to be converted into the system initializer.

- END control record - A record for indicating the termination of the system initializer processing.

- SLM control record - A start of load module record indicating the beginning of the VORTEX nucleus processing portion of system generation.

- VORTEX nucleus routines - Control records and object modules which are to be converted into the VORTEX nucleus.

- END control record - A record for terminating the VORTEX nucleus processing.

- Control record CTL,21 - The record which specifies the end of the nucleus table module. All user data and programs to be included in this module must precede the CTL,21 control record. The user data, which may be included in this module, includes tables associated with software modules.

- Nucleus program module - All programs contained on the SGL tape between the CTL,21 and CTL,PART0003 control records are part of the nucleus program module. Included here are user programs themselves.

In addition to the preceding routines and control records, included in Phase 3 following the CTL,PART0003 control record are:

- Basic library processor - The module which builds load modules from basic object modules and prepares them to be used as necessary in Phase 4 and Phase 5 of system generation.

- Basic system library routines - Basic routines to be used by SGEN in Phases 4 and 5 as necessary. These routines include:

  - OPCOM (operator communication package)

  - FMAIN (file maintenance routine)

  - RAZI (rotating memory analysis and initialization routine)

  - PATCH (system patch routine)

- JCP (job control processor)

- LMGEN (load module generator)

During Phase 3, the following is output to the system list (LIS) device:

1. RMD partitioning list

2. Core resident library map

3. Non-scheduled (VNO) tasks

4. Load module information for the basic system library routines

5. The control record CTL,PART0004

6. VORTEX physical memory allocation

The listing of the VORTEX physical memory allocation indicates the end of Phase 3 of system generation. At this point, the nucleus is ready for processing and is present both in memory and on the disk. Also, the basic VORTEX library routines (support tasks) are also on the disk in the appropriate libraries.

5.4.4 BUILDING THE SYSTEM LIBRARY (PHASE 4)

This phase consists of the routines, control records, and data required to construct load module libraries on the system disk. Object modules (data) are input from the SGL tape (LIB device), processed and converted into load modules, and output to the load module libraries on the system disk (SYS device).

The major components of Phase 4 include:

- Library processor - That portion of the system generation processing routines which converts object modules into load modules. The library processor inputs the object modules and control records from the LIB device, converts them into load modules, catalogs them, and enters them into the system foreground and background libraries.

- Control records - Records which impart information, but are not directly processed as part of the data.

- Object modules - Programs in binary format which do not yet contain all the information necessary for execution under control of VORTEX; that is, they are not yet load modules. Phase 4 fashions them into load modules.

The beginning of Phase 4 is marked by display of the following message on the LIS device:

VORTEX SYSTEM READY TO LOAD LIBRARY TASKS

The user then positions the SGL tape to the first library record via the OPCOM device and assigns the system input logical device (SI) to magnetic tape. The library processor then begins reading object modules and control records from the SGL tape, automatically reassigns SI (system input) to the OPCOM device, and outputs messages to the user. Generally, these messages inform the user of the module currently being processed, ask whether or not the user wishes to include this module in this particular system, and acts upon the user's response. If the module is to be included, it is processed appropriately, created into a load module, and placed on the appropriate library on the disk. If there are any further questions concerning the module, the system outputs requests and messages to the user until all information is obtained. If the module is not to be included in this system, the user is directed to skip the stated number of files, thus placing the SGL tape at the next module to be processed.

The file maintenance (FMAIN) routine is used by SGEN to create files on the disk and to load modules onto the files. The load module generator (LMGEN) routine is used to form load modules from the object modules which are input from the SGL tape and from the FMAIN created files on the disk. These two routines are used continuously by SGEN until all the system object modules (which are to be included in this system) have been processed, formed into load modules, and placed in the appropriate library on the system disk.

Phase 4 may be concerned with additional LIB tapes. If this is the case, as each tape ends, it should be rewound, dismounted, the next tape mounted, and the new tape positioned to the first record. When all SGL tapes have been used, Phase 4 is completed.

5.4.5 RESIDENT TASK CONFIGURATION (PHASE 5)

Phase 5, resident task configuration, is also called the application program phase. That is, at the completion of Phase

4, the VORTEX operating system, as provided by Sperry Univac, is complete and operational. Phase 5, therefore, is a continuous process of creating, adding to, deleting from, updating, and changing user application programs. Phase 5 is not limited to the time of system generation. It can occur at any time and does not require the user to return to earlier phases of system generation.

For complete details concerning the building of user programs and the subsequent creation of object modules and then load modules, refer to the sections describing File Maintenance and Load Module Generation in the VORTEX II Operating System Programmer Reference Manual.

Phase 5 also encompasses maintenance of the resident task configuration, that is, partial system generation. This is covered in the VORTEX II Operating System Programmer Reference Manual (UP-8677).

Section 6
SYSTEM GENERATION DIRECTIVES

6.1 GENERAL

This section includes a description and explanation of those directives (instructions to the system) which are used during an initial system generation process. This does not, by definition, include the directives used for system maintenance.

The directives detailed in this section are:

- I/O interrogation directives
  - ALT (user device specification)
  - DIR (directive input device specification)
  - LIB (system generation library input device specification)
  - LIS (system generation listing output device specification)
  - SYS (generated system output device specification)
- Input directives
  - ADD (SGL addition)
  - ASN (device assignments)
  - CLK (real time clock specification)
  - CPU (CPU model specification)
  - DEF (external definitions)
  - DEL (delete module)
  - EDR (end directive input)
  - EQP (define peripheral equipment)
  - LAD (SGL load module addition)
  - LDE (SGL load module deletion)

- LRE (SGL load module replacement)
- MOD (define peripheral equipment with controller table link)
- MRY (memory specification and allocation)
- NMRY (unavailable memory specification)
- PIM (priority interrupt module assignments)
- PRT (disk partition allocation)
- REP (SGL control records and object module replacement)
- VOL (virtual overlay module specification)

The directives are presented in alphabetical order to facilitate referencing. The recommended or required order of their input into the system is: the MRY directive is entered first and the EDR directive entered last. No special order is required for the remaining directives.

Each directive is presented according to:

1. Functional definition (purpose)
2. Basic format
3. Description and usage
4. Example

Refer to Section 2 for a working example of the use of these directives.

6.2 I/O INTERROGATION DIRECTIVES

The I/O interrogation directives are the first set of directives entered by the user. These directives are entered at the beginning of the system generation process. The I/O interrogation directives specify the peripherals which are to be used by the system generation routines to generate a VORTEX operating system. They are in effect only for the duration of the system generation process, and have no meaning after the process is complete.

All five directives must be entered in response to the following message:

I/O INTERROGATION

Once the message is output, no other prompt is given to the operator. When all five directives have been successfully input, the system automatically resumes processing.

The directives may be entered in any order. Each directive must be terminated with a carriage return function. If any directive, prior to the fifth one, is entered incorrectly, it may be re-entered correctly.

6.2.1 ALT DIRECTIVE

Purpose:

This directive allows the specification of an alternate input device for user modifications, deletions, or additions to the nucleus modules or to the object module library.

Format:

The general format for the ALT directive is:

ALT,<driver name>,<device name>[,bic address]

where:

driver name
: Consists of the 2-letter drive mnemonic, the controller number, the unit number, and the model code letter. Permissable entries are of the form MTxxm, PTxxm, and CRxxm.

device address
: Is the hardware address of the device to be assigned to ALT. Refer to the "System Memo" for device address assignments.

bic address
: Represents the hardware address of the buffer interlace controller (BIC) associated with this device. This parameter is optional.

Description:

Input from the ALT device must be in binary object format or in the form of a Hollerith TDF directive. The ALT directive is detailed according to its usage.

For an initial system generation, modifications and additions normally are not performed. Therefore, this device assignment should be the same as the LIB directive (see 6.2.3).

Example:

```
ALT,MT00A,010
```

This example specifies that the ALT input is to be assigned to the magnetic tape (MT), controller number 0, unit number 0, and model code A. Its physical (hardware) device address is 010_8.

NOTE

Any of the directives ALT, DIR or LIB which specify the type 0870 magnetic tape unit must include the BIC address parameter.

6.2.2 DIR DIRECTIVE

Purpose:

This directive specifies the device from which all "input directives" are to be entered into the system.

Format:

The general format for the DIR directive is:

```
DIR,<driver name>,<device address>[,bic address]
```

where:

driver name
: Consists of the 2-letter driver mnemonic, the controller number, the unit number, and the model code letter. Magnetic tape, teletypewriter/CRT, paper tape, or card reader are the only acceptable DIR devices. Permissable driver names are of the form MTxxm, TYxxm, PTxxm, or CRxxm.

device address
: Is the hardware address of the device to be assigned to DIR.

bic address
: Represents the hardware address of the buffer interlace controller (BIC) associated with this device. This parameter is optional and is used for magnetic tape units only.

Description:

Once this directive is assigned and accepted by the system, all input directives (excluding the I/O interrogation directives) are expected to be input from the DIR device. Normally, for the initial system generation, the DIR device is the card reader or the teletypewriter or CRT. For system generation maintenance, the original DIR input may be stored on the system generation library tape and input thereafter from magnetic tape, whenever a system generation update is required.

Example:

```
DIR,TY00A,01
```

The DIR input is assigned to the teletypewriter (TY), controller number 0, unit number 0, and model code A. Its physical (hardware) device address is 01.

6.2.3 LIB DIRECTIVE

Purpose:

This directive specifies the device from which the VORTEX libraries are to be input.

Format:

The general format of this directive is:

```
LIB,<driver name>,<device address>[,bic address]
```

where:

driver name
: Consists of the 2-letter driver mnemonic, the controller number, the unit number, and the model code letter. Magnetic tape, card reader, and disk are the only acceptable LIB devices. Permissable driver names are of the form MTxxm, CRxxm or Dxxm.

device address
: Is the hardware address of the device to be assigned to LIB.

bic address
: Represents the hardware address of the buffer interlace controller (BIC) associated with this device. This parameter is optional.

Description:

The LIB directive specifies the device from which the system generation libraries are to be input. These libraries consist of:

1. The nucleus (system generation) library
2. The object module (OM) library
3. The optional software library

As explained in Section 5, each of these libraries is input into the system via the LIB device during one of the phases of system generation, processed by the system generation routines, tailored by the user DIR input information, and output to the disk for resident storage.

These libraries are normally supplied on magnetic tape to the user at the time of equipment delivery. For information concerning the modification of this tape, refer to Library Modification Directives.

Example:

```
LIB,MT00A,010
```

LIB is assigned to the magnetic tape (MT), controller number 0, unit number 0, and model code A. Its physical (hardware) device address is 01.

6.2.4 LIS DIRECTIVE

Purpose:

This directive allows the user to specify the device to which the list information as a result of the system generation process is to be output.

Format:

The general format for this directive is:

LIS,<driver name>,<device address>[,bic address]

where:

driver name
: Consists of the 2-letter driver mnemonic, the controller number, the unit number, and the model code letter. The only devices which may be assigned to LIS are the teletypewriter and the line printer. Permissable driver names are of the form TYxxm or LPxxm.

device address
: Is the hardware address of the device to be assigned to LIS.

bic address
: Represents the hardware address of the buffer interlace controller (BIC) associated with this device. This parameter is optional.

Description:

This directive specifies the device to which the listing information is to be output during the system generation process. The following information appears on the LIS device:

1. Error messages.

2. Load map of each load module (each module from the libraries on the system generation library tape).

3. Directive input through the DIR device.

4. Partition table for each system disk.

To suppress listing output during the system generation process, see the EDR directive.

Example:

LIS,LP00A,035

LIS is assigned to the line printer (LP), controller number 0, unit number 0, and model code A. Its physical (hardware) device address is 035_8.

6.2.5 SYS DIRECTIVE

Purpose:

This directive specifies the unit to which the result of the system generation (the tailored VORTEX system) is to be output.

This directive specifies the disk units onto which the VORTEX system will be generated. The nucleus of the VORTEX system will be placed on the first disk specified. The SYS directive permits up to 16 disk devices to be specified.

Format:

The general format for the SYS directive is:

```
SYS,<driver>,<device1>,<bic1>;<driver2>,<device2>,<bic2>;...
        <driver16>,<device16>,<bic16>
```

where:

driver
: Is a disk driver name in the form Dcum (c= controller number, u= unit number, m= model code).

device
: Is the hardware device address of the corresponding driver. An even device address should be used for the following disk devices:

  - 2825-xx
  - 2842-xx
  - 2826-xx
  - 2843-xx

bic
: Is the address of the applicable BIC or BTC. This parameter is mandatory and must be 1 for the disk devices mentioned under the device parameter.

Description:

The SYS device is the peripheral onto which the tailored (generated) VORTEX software system is to be stored. Normally, the SYS device is at least one disk.

Once the VORTEX system has been generated and is stored on the SYS device, the nucleus can be loaded directly into the system from this device. In addition, any other programs or data on the SYS device can be loaded into memory under the control of the VORTEX nucleus. Refer to the VORTEX II Operating System Programmer Reference Manual for information concerning the reloading of the nucleus and loading of programs and data into memory.

Every disk in the system (those specified in the EQP directives) must be included in the SYS directive. The system disk must be specified first. For example:

```
SYS=D00C,015,070;D10B,016,020
```

A colon at the end of the line indicates that the SYS directive continues on the next input line. The colon may be used as a replacement for the semicolon but not for the comma. Any additional lines of the SYS directive must begin with the driver parameter. SYS= must not be used on the additional SYS directive input lines.

Each disk must have an EQP directive and at least one PRT directive. The system disk must have at least seven PRT directives.

The device name for the SYS or LIB directive for a model F disk must be formed as follows:

```
DcddpF
```

where:

c
: Is the controller designation (0 through 3).

dd
: Represents the drive number. Permissible values are 00 through 11.

p
: Is 0 for fixed platter and 1 for removable platter.

The unit numbers need not be consecutive. Note that the system disk is always unit 0 regardless of the platter.

Example:

```
SYS,D00H,014,01
```

SYS is assigned to the disk (D), controller number 0, unit number 0, and model code H. Its physical (hardware) address is 014_8 and its BIC physical address is defined as 01_8 to satisfy the format requirements. However, the model H disk does not use a BIC.

NOTE

Each disk must have been formatted before system generation is started. Refer to Appendix A for disk formatting procedures.

6.3 INPUT DIRECTIVES

As shown in the example in Section 2, after the I/O interrogation directives and the system generation header have been input, the system displays the following message on the OC device:

INPUT DIRECTIVES

This subsection includes those input directives which may or must be entered in response to this message. However, not all possible responses are presented herein. For those relating to additions or modifications to the system generation library, refer to "Library Modification Directives".

All directives given in response to the preceding message must be input through the DIR device. Once all directives have been input, system generations processing continues as detailed in Section 5.

NOTE

Device assignments and associations as described in this subsection do not relate to those described in the I/O interrogation directive subsection (6.2). Therefore, these explanations should be studied independently of the preceding subsection.

CAUTION

If the DIR device specified is a card reader, the input directives must be prepared using 026 keypunch characters. If an 029 keypunch is used, the following character conversions must be made:

026 keypunch character	Equivalent 029 keypunch character
=	#
:	'
(	%
)	<

This manual uses 026 keypunch characters to show the format of SYSGEN directives.

6.3.1. ASN (ASSIGN) DIRECTIVE

Purpose:

The ASN (assign) directive equates the logical unit number (and, optionally, the logical unit name) with the physical device name.

Format:

The general format for the ASN directive is:

```
ASN,<lun1>[,:luname1]=<dev1>[,<lun2>[,:luname2]=<dev2>,]...
    [<lunn>[,:lunamen]=<devn>]
```

where:

lun
: (logical unit number) is a software identification number which defines a device without being directly related to the physical characteristics of the device (that is, without being directly related to the physical device address). The parameter lun may be a value of 1 through 100 or 107 through 255.

luname (optional)
: Is a 2-letter name optionally assigned to the specified device and associated with the logical unit number.

devn
: Is the physical device name which is associated with the logical unit number and the logical unit name.

The ASN directive may not extend over multiple lines. Additional ASN statements must be used if more device assignments are to be made than fit on one line.

Description:

Logical unit numbers, and optionally logical unit names, are assigned to physical devices for ease of programming and using the system. Effectively, a LUN is assigned, not to the physical device itself, but to the controller associated with that device. This permits the programmer to issue an I/O call to a physical device using a LUN or logical device name, and eliminates the necessity for knowledge concerning the physical characteristics of the device.

The following are a list of considerations and special notations when using the ASN directive:

1. Using the ASN directive, a physical device can be assigned to more than one logical unit, but a logical unit number or name can be assigned to only one physical device.

2. In the VORTEX software system, logical unit numbers 1 through 12 have preassigned logical unit names (see Table 6-1) and require physical device assignments via the ASN directive.

3. Each partition on the disk is normally assigned a unique LUN. However, a few assignments are inherent in the VORTEX system, as shown in Table 6-2. These preassigned disk partitions do not require ASN directives. However, each does require a PRT directive.

4. It is advisable to include extra logical unit numbers which are assigned to dummy units. These logical unit numbers can then be used as temporary assignments through OPCOM and JCP, whenever necessary. For example:

```
ASN,39:DD=DUM
ASN,53:BP=DUM
```

Table 6-1. Logical Units, Preset Names, and Permissible Peripheral Device Assignments

Logical Unit	Preset Logical Name*	Permissible Physical Unit: Teletype-writer or CRT	Disk or MT	Line Printer	Other Output (CP,PT)	Other Input (PT,CR)
1	OC	X				
2	SI	X	X			X
3	SO	X				
4	PI	X	X			X
5	LO	X	X	X	X	
6	BI		X			X
7	BO		X		X	
8	SS		X			
9	GO		X			
10	PO		X			
11	DI	X				X
12	DO	X		X		

* Refer to Table 6-3.

5. If a new assignment specifies the same logical unit as a previous assignment, the old assignment is replaced and is no longer valid.

6. All logical units, except the preassigned logical units (Table 6-1), for which device assignments are not explicitly made are considered dummy devices.

Table 6-2. Optional and Pre-Set Logical Unit/Disk Partition Relationship

Logical Unit Name*	Logical Unit Number	Partition Name	Protection Key	Minimum VORTEX Sector Allocation	Refer to Note Number
Pre-Set Logical Unit/Disk Partition Relationships					
CL	103	D00A	C	025	1
FL	106	D00B	F	0106	
BL	105	D00C	E	01135	
OM	104	D00D	D	0417	
CU	101	D00E	S	0310	2
SW	102	D00F	B	0310	3
optional	116	optional	none		4
Optional Logical Unit/Disk Partition Relationships					
GO	9	D00G	none	0310	5
SS	8	D00H	none	varies	
PO	10	D00H	none	0515	6
BI	6	D00I	none	varies	
BO	7	D00I	none	varies	

Notes:

1. There are 12 entries per 2 sectors. Number of sectors equals number of entries + 6.

2. The CU file must be as large as the background task's largest part in central memory at one time. The value 24K is assumed in this table.

Table 6-2. Optional and Pre-Set Logical Unit/Disk Partition Relationship (Continued)

3. The SW file must be as large as the largest single task including overlays. The value of 24K is assumed in this table.

4. This file must reside on the system disk and be at least 160 sectors.

5. The GO file must be somewhat larger than the largest task run in load-and-go mode. The value 24K is assumed in this table. If the system is foreground only or all tasks to be run will be entered in libraries before execution, this partition may be eliminated.

6. The PO file must be large enough for the source images of the largest task to be assembled or compiled. Source images are stored 3 card images per sector. In this table, 1000 cards are assumed. If the source images are to be stored on magnetic tape, this partition may be eliminated.

Restrictions: An attempt to change any of the preset logical unit name/number or name/number/partition relationships causes an error to be flagged.

* Refer to Table 6-3.

Table 6-3. Logical Name Designation Definitions

Mnemonic Designation	Meaning
OC	Operator Communication Device
SI	System Input Unit
SO	System Output Unit
PI	Processor Input Unit
LO	List Output Unit
BI	Binary Input Unit
BO	Binary Output Unit
SS	System Scratch Unit
GO	GO Unit
PO	Processor Output Unit
DI	Debugging Input Unit
DO	Debugging Output Unit
CL	Nucleus Entry Library
FL	Foreground Library
BL	Background Library
OM	Object Module Library
CU	Checkpoint Unit
SW	System Work Unit

Example:

Specify physical device assignments for logical units 1-12, 107, 108, 180 and 181. The last two units have in addition to their numbers, two-character names.

```
ASN,1=TY00,2=CR00,3=TY01,4=CR00
ASN,5=LP00,6=MT00,7=D00I,8=D00G
ASN,9=D00H,10=D00G,11=TY00,12=LP00
ASN,107=LP00,108=CR00
ASN,180:S6=MT00,181:S8=MT01
```

Example:

```
ASN,1:OC=TY00
```

By system definition, logical unit 1 is the OC device. Therefore, the teletypewriter is assigned as the OC device with a logical unit number of 1 and a logical unit name of OC.

6.3.2 CLK (CLOCK) DIRECTIVE

Purpose:

This directive allows the user to specify all the parameters related to the real time clock.

Format:

The general format for this directive is:

```
CLK,<clock>,<counter>,<interrupt>
```

where:

clock
: Specifies the number of microseconds in the basic clock interval. This function is hardwired. Therefore, to correctly enter this value, refer to the "System Memo" accompanying the particular system. The standard value is 100.

counter
: Specifies the number of microseconds in the free-running counter increment period. This value is stored in V$FREE but is not supported by VORTEX. The value for this parameter must be the same as that for "clock" in the preceding parameter.

interrupt
: Specifies the number of milliseconds in the user interrupt interval. The user interrupt interval is the amount of time one task runs before another task is dispatched. A real time clock interrupt signals the end of an interval. This value must be between 5 and 50.

If the value of interrupt is not a multiple of 5 milliseconds, it is increased to the next multiple of 5 milliseconds. For example, if interrupt is 31, the interrupt interval will be set to 35 milliseconds.

For models V77-200 and V77-400 minicomputers running on 60 cycle power, the CLK and accompanying DEF directive must be:

```
CLK,100,100,15
DEF,V$CKCT,3
```

For models V77-200 and V77-400 minicomputers running on 50 cycle power, the CLK and accompanying DEF directive must be:

```
CLK,100,100,20
DEF,V$CKCT,3
```

Example:

```
CLK,100,100,20
```

This example indicates that there are 100 microseconds in the basic clock interval. Since the free-running counter increment period is not supported, its parameter is also 100 microseconds. The user interrupt interval is set to 20 milliseconds. This is an optimum time frame for a VORTEX system.

6.3.3 CPU DIRECTIVE

Purpose:

This optional directive allows the user to specify the type of central processing unit (CPU) on which this VORTEX system is to execute, and any special conditions relating to that CPU. One CPU directive is required.

Format:

The general format of this directive is:

```
CPU,x
```

where:

x

Represents one of the following:

1 = for all V72 through V76 CPUs and V77-600 CPU with software operating system nucleus

2 = V77-200 or V77-400 CPUs

3 = V77-600 CPU with firmware operating system nucleus (micro-VORTEX)

4 = V77-800 CPU with software operating system nucleus.

5 = V77-800 CPU with the Floating Point Processor

Example:

```
CPU,3
```

This specifies that the VORTEX system which is being generated is to execute on a V77-600 CPU with firmware operating system.

6.3.4 DEF (EXTERNAL DEFINITION) DIRECTIVE

Purpose:

This directive allows the user to define special conditions relating to specific peripherals in the system.

Format:

The general format of this directive is:

```
DEF,<name>,<value>
```

where:

name
: Is the name directly attached to an absolute value.

value
: Is the absolute value associated with "name" that is loaded into the system generation loader tables and the CL library.

Description:

Because of the nature of this directive, it is defined specifically for each case in which it is used. Refer to Section 7 and to other directives in this section.

NOTE

Modules processed by either SGEN or the VORTEX Load-Module Generator can reference any names defined by the DEF directive.

6.3.5 EDR (END) DIRECTIVE

Purpose:

The EDR directive must be the last system generation directive. It specifies all special system parameters and terminates the system generation directive input.

Format:

The general format of the EDR directive is:

EDR,<type>,<part>,<list>,<kpun>,<map>,[<anal>]

where:

type
: Is S for a standard (full) system generation or N for a nucleus only system regeneration. A standard system generation specifies that the entire VORTEX system is to be regenerated and all partitions are to be initialized (existing files will be lost). A nucleus only system regeneration specifies that only the nucleus image and SGEN generated load modules are to be recreated. Existing file directories are not destroyed in a nucleus only system regeneration.

part
: Is the maximum number (7 through ~~20~~ 62) of partitions per spindle on disk, except for disk types 2825-xx, 2842-xx, 2826-xx, 2843-xx, and 8433-xx, which can have a maximum of 63 partitions per spindle.

list
: Is the number of lines per page for the list output (LO) device, with typical values of 44 for the line printer and 71 for the teletypewriter.

kpun
: Specifies which translate table is used by the card reader and card punch I/O drivers. 26 should be used for the 026 keypunch (Hollerith code) and 29 for the 029 keypunch.

map
: Is L if map information is to be listed or 0 if map information is to be suppressed.

anal

Is 0, blank, or omitted if a complete bad track analysis is to be performed on all disk devices, or 1 if the bad track tables from the last system generation are to be reused. The value 1 may be entered only when an analysis has been made from a previous system generation.

NOTE

If the SGL is on a slave disk, bypass the bad track analysis by setting <anal> = 1.

A nucleus only system regeneration is subject to the following restrictions:

- The <mem> parameter on the MRY directive must be the same as in the previous system generation.
- The SGEN cataloged load modules must be the same size (in sectors) as the previous SGEN versions and must reside in their original sectors.
- All existing non-SGEN created load modules must be relinked using the RELINK program. (Refer to the VORTEX II Operating System Programmer Reference Manual for a description of RELINK.)
- All disk devices must be partitioned in the same manner as in the previous system generation.

The nucleus only system regeneration is not generally usable between different versions of the VORTEX operating system. A nucleus only system regeneration is normally used to change a nucleus component or add a nucleus component to identical revisions of a VORTEX system.

Example:

```
EDR,S,40,44,26,L,0
```

The parameters in this example indicate:

- A standard (full) system generation is to be performed.
- Disks have a maximum of 40 partitions per spindle.

- The LO (list output) device prints a maximum of 44 lines per page.
- The 026 (Hollerith code) keypunch is to be used.
- Map information is to be listed.
- A complete bad track analysis is to be performed on all disk devices.

6.3.6 EQP (PERIPHERAL EQUIPMENT DEFINITION) DIRECTIVE

Purpose:

The EQP directives provides the user with the means of defining the peripherals on the system. One EQP directive must be stated for each peripheral controller and pseudo driver.

Format:

The general format of this directive is:

```
EQP,<name>,<address>,<units>,<bic add>,<retr>,[<alg>],[<mul>]
```

where:

name
: Is the mnemonic for a peripheral controller.

address
: Is the physical address of the device controller (01 through 077 inclusive).

units
: Is the quantity (1 through 4 inclusive) of the peripheral devices attached to the controller. (1 through 8 are allowed for disk types 2825-xx, 2842-xx, 2826-xx, 2843-xx, and 8433-xx.)

bic add
: Is the physical address of the BIC or BTC, if one is used. This should be zero if neither a BIC nor a BTC are used.

retr
: Is the number (0 to 63) of the retries to be attempted by the I/O driver when an error is encountered.

<alg>
Is the I/O algorithm value ($0 \leq alg \leq 1$) as a decimal fraction. (Refer to "Adding an I/O Driver to the System File" in the VORTEX II Operating System Programmer Reference Manual for an explanation of the I/O algorithm.) This is an optional parameter and is not needed unless a change is to be made in the algorithm value. If this parameter is used on non-process I/O controller tables, the controller table must contain CTIOA as an entry name.

<mul>
Is the multiplexer address. This parameter applies only to process I/O drivers.

Description:

This directive, in conjunction with the other input directives, defines the external subsystem to VORTEX. The EQP directive details information concerning the individual peripherals attached to the system. The following are rules and considerations which are helpful when using the EQP directive:

Acceptable mnemonics for the "name" parameter are:

- CI process input
- CO process output
- CPnm card punch
- CRnm card reader
- CTnm CRT device
- Dnm disk
- ER0H error handling driver for disk types 2825-xx, 2842-xx, 2826-xx, 2843-xx, and 8433-xx (pseudo driver)
- FM0A file maintenance pseudo driver
- LPnm line printer
- MTnm magnetic tape unit
- Mxnm communication multiplexer

- PTnm paper tape reader/punch

- RM0m system dump driver (pseudo driver)

- SD0A multitask spooler

- SPnA spool unit

- TYnm teletypewriter

- WCS writable control store

In the preceding mnemonics, "n" represents the controller number (0, 1, 2, or 3) and "m" represents the model code (refer to Table 6-4).

- Controller tables are arranged according to the priority level of their task identification blocks (TIDBs). On any given level, the tables are arranged in the input sequence of the corresponding EQP directives. Device specification tables (DST) entries are unsorted.

- The following order is suggested for entry of the EQP directives for peripheral controllers:

  1. Disks (system disk first)

  2. Operator communication (OC) device

  3. Magnetic tape units

  4. Other units

- Some peripherals require further definition through the use of the DEF directive. Refer to 6.3.4 and Section 8 for information concerning the DEF directive and when it should be used.

Table 6-4. Software Model Codes, Model Numbers and Type/Feature Numbers for VORTEX peripherals

Software Model Code	Model Number	Type/Feature Number	Description
CInA			Process I/O
COnA			Process I/O
CPnA	70-6201	2813-xx	Card punch. 35 cards/min
CRnA	70-6200	2812-xx	Card reader. 300 or 600 cards/min
CTnA	70-6401	2817-xx 2818-xx 2838-xx 2839-xx 2840-xx	CRT keyboard/display
DnA	70-770x	F3098-xx	Fixed head disk
DnB	70-7600 70-7610		Rotating memory
DnC	70-7500	2822-xx	Moving head disk (22 megabyte)
DnD	70-7510		Rotating memory
DnE	70-7520	2823-xx	Moving head disk (92 megabyte)
	70-7530		Rotating memory
		2824-xx	Moving head disk (186 megabytes)
DnF	70-7603	F3094-xx	Moving head disk (9.3 megabytes)
	70-7613	F3096-xx	Moving head disk (4.6 megabytes)
		F3310-xx	Moving head disk (9.3 megabytes)
DnH	70-755x	2825-xx	Moving head disk (9.3 megabytes)

Table 6-4. Software Model Codes, Model Numbers and Type/Feature Numbers for VORTEX peripherals (Continued)

Software Model Code	Model Number	Type/Feature Number	Description
		2826-xx	Moving head disk (116.3 megabytes)
		2842-xx	Moving head disk (61.2 megabytes)
		2843-xx	Moving head disk (232.6 megabytes)
		8433-xx	Moving head disk (200 megabytes)
DnJ	70-762x	F3097-xx	Disk memory; flexible diskette
LPnA	70-6701		Line printer
	70-6710	2819-xx	Line printer
	70-6721	2820-xx	Line printer
		0786-xx	Line printer
LPnB	70-6701		Line printer with 24 character print buffer
LPnD	70-6602		Statos-31 printer/ plotter
LPnE	70-6603		Statos-31 with 041 option
LPnG	70-6603		Statos-31 with 042 option
LPnH	70-7702		Statos-31 with 041, 051 and 052 options
LPnJ	70-66xx		Statos-33
LPnK			Statos-41 with 145 character generator
LPnL			Statos-41 with 144 character generator

Table 6-4. Software Model Codes, Model Numbers and Type/Feature Numbers for VORTEX peripherals (Continued)

Software Model Code	Model Number	Type/Feature Number	Description
LPnM			Statos-42 with 146 character generator
LPnN			Statos-42 with 144 character generator
MTnA	70-7100	F3088-xx	Magnetic tape; 9 track, 800 bpi, 25 ips
	70-7107	0870-xx	Magnetic tape; 9 track, 800/1600 bpi, 75 ips
	70-7102	F3089-xx	Magnetic tape; 9 track, 800 bpi, 37.5 ips
	70-7103	F3089-xx	Slave unit with 70-7102
MXnA	70-520x 70-521x	F3000-xx	Data communication multiplexer
PTnA	70-6320		Paper tape reader/ punch
		F3082-00	Paper tape reader
		F3083-00	Paper tape punch
		F3084-00	Paper tape reader/ punch
TYnA	70-6104	2811-xx	ASR 35 teletypewriter
	70-6101	2810-xx	ASR 33 teletypewriter
WCS	70-4002		Writable control store
	70-4003	F2960-05	Writable control store
	70-4004	F2960-06	Writable control store

NOTE: Other peripheral devices can be added to the system by creating an EQP directive with a unique physical unit name for the device. A controller table with the same name is then added to the VORTEX nucleus by an ADD directive.

- Disk types 2825-xx, 2842-xx, 2826-xx, 2843-xx and 8433-xx (software model code H) require at least two EQP directives. One directive is the standard disk directive for the Model H disk. An EQP is required for each controller. Also required is a special EQP directive for the software model code H disk error routine, VZERH. This is a dummy EQP directive used to select the error module and thus only one such directive is needed, regardless of the number of controllers. For example, a system containing two of these disk types on device addresses 012 and 014 require the following:

```
EQP,D0H,012,1,0,10

EQP,D1H,014,1,0,10

EQP,ER0H,0,1,0,0
```

Example:

```
EQP,D0F,016,1,020,10
```

The parameters in this example indicate:

- The device name is D0F (D = disk, 0 (zero) = disk controller address, and F = the software model code of the device).
- "016" is the physical device address of the disk unit.
- "1" indicates that there is one disk unit attached to the controller.
- "020" is the device address of the BIC associated with this disk.
- "10" indicates the number of retries for the I/O operation associated with this device, if an error or busy occurs.

6.3.7 MOD DIRECTIVE

Purpose:

The MOD directive performs the same function as the EQP directive except that this directive is used to define peripherals on the system which do not have a controller table.

Format:

The general format of the MOD directive is:

MOD,<name>,<address>,<units>,<bic add>,<retr>[,<alg>][,<mul>]

where:

The parameters in the MOD directive are the same as those in the EQP directive.

6.3.8 MRY (MEMORY) DIRECTIVE

Purpose:

The memory directive allows the user to define all parameters necessary to configure the memory for a particular VORTEX system in the required manner. One MRY directive is required.

Format:

The general format for the MRY directive is:

MRY,<mem>,<yyy>,<size>[,V75],<npages>,<lpage>

where:

mem

Is the highest address (octal) of the memory available to the VORTEX system. This parameter must be at least 027777 (12K).

yyy

Is the size (in words) of the foreground blank common area. This parameter must be 0 or positive. It is recommended that this value be a multiple of 512 (one page).

size
: Is the total physical memory available to the VORTEX system in units of 1024 (1K) words. The permissable range of values for this parameter is 32 through 1024.

,V75
: If present, indicates that the system to be generated is an extended register system (V77, V75, or V76) and a long task identification block (TIDB) is to be used.

npages
: Is a value which specifies the number of 512 word pages that are allocated to virtual nucleus overlay (VNO) task storage on the disk. The value of "npages" is converted to the equivalent number of sectors and added to the nucleus image size on the system disk. This parameter may affect the track number of the initial disk partition. The value of this parameter is for disk storage space only. It does not affect the amount of memory available to the user.

NOTE

If the value of the "npages" parameter is too small, part of the VNO disk image will be lost and another SYSGEN must be performed. No error message is displayed should this occur.

<lpage>
: Indicates the page number of last page in memory which is to be allocated to virtual nucleus overlay tasks. This value must be greater than 64, but less than or equal to (2*<size>)-1. Virtual nucleus overlay (VNO) tasks normally occupy consecutively numbered physical pages ending with the page specified by <lpage>. However, the sequential assignment of physical pages to VNO tasks may be altered by the presence of "unavailable" pages (refer to the NMRY directive). If the page specified by the <lpage> is within the unavailable pages, assignment of pages to VNO begins immediately below the unavailable pages. If the page specified by <lpage> is greater than the highest unavailable page, but insufficient to provide the number of consecutive pages specified by "npages", then pages above and immediately below the unavailable pages are assigned to VNO tasks.

Description:

The "format" explanation in the preceding paragraph details the information which may be entered into the system via the MRY directive.

Example:

```
MRY,074777,512,1024,V75,26,127
```

This example illustrates the following:

- 30.5K of memory is available for the nucleus, AID (1K) is to be retained in memory, and a page (0.5K) is allocated to the DCM for its use.
- 512 words are to be reserved for the foreground blank common.
- 1024K words of memory are available to VORTEX.
- This system is an extended register system and will use a long TIDB.
- The disk equivalent of 26 pages of memory are allocated for virtual nucleus overlay tasks.
- Page number 127 is the last memory page allocated to virtual nucleus overlay tasks. Sufficient pages are allocated to store all VNO tasks.

6.3.9 NMRY (NO MEMORY) DIRECTIVE

Purpose:

This directive allows the user to "hide" physical pages of memory from the VORTEX operating system.

Format:

The general format of this directive is:

```
NMRY,<pageno(1)>,<numb(1)>[;pageno(2),numb(2)]...
[;pageno(n),numb(n)]
```

where:

pageno(n)
Specifies the first physical page number of a block of pages which are to be made unavailable to VORTEX. "pageno" can be any number from 64 through the highest numbered page in the system.

numb(n)
Specifies the number of consecutive pages, starting at "pageno", which are to be made unavailable to VORTEX.

Description:

This directive specifies the physical pages of memory which are to be made unavailable for use by the VORTEX operating system. This directive is strictly optional. Often, it is used to reserve an area in memory for the storage of a utility or a loader program, so that VORTEX cannot overlay or destroy the contents of that area, or it is used for shared memory.

Example:

```
NMRY,64,64;192,64
```

This example declares pages 64 through 127 and pages 192 through 255 to be unavailable to the VORTEX operating system.

6.3.10 PIM (PRIORITY INTERRUPT MODULE) DIRECTIVE

Purpose:

The PIM directive permits the user to define the architecture of the interrupt subsystem. This directive is required for every peripheral controller and/or its associated BIC.

Format:

The general format for the PIM directive is:

```
PIM,<p(1)>,<q(1)>,<r(1)>,<s(1)>[;p(2),q(2),r(2),s(2)]...
[;p(n),q(n),r(n),s(n)]
```

where:

p(n)
: Is an interrupt line number, comprised of three octal digits. The first digit is the PIM number (PIM 1 = 0; PIM 2 = 1, etc.); the second is the line number within the PIM (line number 0 through 7). The two digits must be preceded by a zero (for example, 012).

q(n)
: Is a 1 to 6 character name of the task handling the interrupt. The format of the task name is TBxxxx (where xxxx is the hardware code name defined in the EQP directives). If s(n)=2, this parameter is the interrupt processor entry name.

r(n)
: Is the event word (octal) for this particular interrupt. This parameter is wholly dependent upon the peripheral device. Refer to "RMD Status Words" in the VORTEX II Operating System Programmer Reference Manual and Table 7-2 in this manual.

s(n)
: Is 0 if the interrupt is to be serviced by a common interrupt handler; is 1 if the interrupt is directly connected (option 1); and, is 2 if the interrupt is directly connected (option 2). Refer to "Real Time Programming" in the VORTEX II Operating System Programmer Reference Manual for a description of these interrupts.

Description:

The PIM directive defines the interrupt subsystem architecture by specifying the number of PIMs in the system, the interrupt levels to be enabled at system initialization, and the interrupts to be manipulated by interrupt handlers coded by the user. The following shows the relationship of interrupts to PIM numbers:

Memory Address of Interrupt	PIM Number
0100 - 0117	0
0120 - 0137	1
0140 - 0157	2
0160 - 0177	3
0200 - 0217	4
0220 - 0237	5
0240 - 0257	6
0260 - 0277	7

If an interrupt line is to use a common interrupt handler, a TIDB is generated for the related interrupt-processing routine which can be either in the VORTEX nucleus or in the foreground library. If an interrupt line is to have a direct connection, the interrupt-processing routine must be added to the nucleus. Failure to do so results in an error. See Appendix B for error messages.

NOTES

1. The only interrupt used by the magnetic tape I/O driver is the motion complete interrupt.

2. For a teletypewriter or teletypewriter compatible CRT, the interrupt event word r(n) must be set to 01 for input interrupt or 02 for output interrupt.

Typical examples of common interrupt handlers being referenced in PIM directives are:

BIC complete on disk type 2822-xx:	PIM,000,TBD0C,01,0
Card Reader:	PIM,001,TBCR0A,01,0
Seek Complete:	PIM,002,TBD0C,01,0
Paper Tape:	PIM,005,TBPT0A,01,0
TTY Read, PIM 2, Line 6:	PIM,016,TBTY0A,01,0
TTY Write, PIM 2, Line 7:	PIM,017,TBTY0A,02,0

For specialized information concerning the relationship of PIMs to specific software modules, refer to Section 7.

Example:

```
PIM,000,TBD0F,01,0
```

This example defines a PIM for a disk in the following manner:

- The interrupt is received on PIM #0, Line #0.

- The task which handles the interrupt from this particular PIM and line is TBD0F.
- The event word indicates that this interrupt signifies BIC activity complete.
- The interrupt is handled by a common interrupt handler.

6.3.11 PRT (PARTITION) DIRECTIVE

Purpose:

The PRT (partition) directive allows the user to specify the size of each partition on a disk. This directive is required for every 32K of disk storage.

Format:

The general format for the PRT directive is:

$$\text{PRT,Dcu}\left\{\begin{matrix}\text{p}\\(\text{n})\end{matrix}\right\},\text{s(1),k(1)[;Dcu}\left\{\begin{matrix}\text{p}\\(\text{n})\end{matrix}\right\},\text{s(2),k(2)]}$$
$$\ldots\text{[;Dcu}\left\{\begin{matrix}\text{p}\\(\text{n})\end{matrix}\right\},\text{s(n),k(n)]}$$

where:

Dcu $\left\{\begin{matrix}\text{p}\\(\text{n})\end{matrix}\right\}$
: Is the name of the disk partition.

c
: Is the controller number (1, 2 or 3).

u
: Is the unit number (1, 2 or 3).

p
: Is the letter assigned as the partition identification (A through T).

(n)
: Is the number assigned as the partition identification (1 through 20). This value must be enclosed in parentheses. For the type 2825-xx, 2826-xx, 2842-xx, 2843-xx and 8433-xx disks, (n) may be (1) through (63).

Description:

The partition directive allows the specification of the size of each partition on a disk. Partition directives are required for every partition on every disk in the system. The system disk must have a minimum of 6 and a maximum of 20 (maximum of 63 for disk types 2825-xx, 2842-xx, 2826-xx, 2843-xx and 8433-xx) defined partitions. Each disk on the system must have at least one PRT directive associated with it.

Disk partitioning permits the disk to be accessed as several logical unit. That is, each partition is treated as a separate logical unit.

NOTE

Tracks of a disk device not mentioned in a PRT directive are not accessible to the operating system.

The following should be considered when using the PRT directive:

- The partition specifications for a disk device may be entered in any order but must comprise a contiguous group. For example, the sequence D00A, D00C, D00B, D00D is valid but the sequence D00A, D00C, D00D, D00E will cause an error (there is no D00B).

- If the LIB unit is a disk device, the PRT directives for the LIB unit are ignored and the existing partition specification table (PST) for the LIB unit is used. However, even though the PRT directives for the LIB unit are ignored, the LIB unit should have at least one PRT directive. The RAZI program may be used to format the disk on the LIB unit after system generation. If the SGL on the LIB unit is to be saved, the disk on the LIB unit must be replaced with a scratch disk pack prior to executing RAZI for that unit. Additional information regarding the RAZI program may be found in the VORTEX II Operating System Programmer Reference Manual.

- Logical units 101 through 106 have preassigned protection codes. At no time attempt to change these codes. Table 6-5 gives the preset protection codes and their corresponding logical units.

- Tables 6-6 and 6-7 list the sizes of the disks supported by the VORTEX software system.

Table 6-5. Preassigned Protection Codes

Logical Unit Number	Preassigned Protection Code
101	S
102	B
103	C
104	D
105	E
106	F

Table 6-6. Sizes of Disks Supported by VORTEX

Type/ Feature No.	Tracks	Sectors	Words (16-Bit) Per Sector
2822-00	4060	24	120
2823-00	8120	48	120
2824-00	8120	48	120
F309x-xx	406	48	120

Table 6-7. Types 2825-xx, 2842-xx, 2826-xx, 2843-xx and Feature F3097-xx Disks Formatted Sizes

Type or Feature Number	Size	Hex Type	Bytes/ Sector	Sectors/ Track	Tracks/ Cylinder	Cylinders/ Disk	Bytes/ Disk
2825-xx	40 *	0	240	62	5	411	30,578,400
	Megabyte	2	480	35	5	411	34,524,000
2842-xx	80 *	4	240	62	5	823	61,231,200
	Megabyte	6	480	35	5	823	69,132,000
2826-xx	150 *	8	240	62	19	411	116,197,920
	Megabyte	A	480	35	19	411	131,191,200
F3097-xx	200 *	D	240	39	19	815	144,939,600
	Megabyte	F	480	23	19	815	170,954,400
	300 *	C	240	62	19	823	232,678,560
	Megabyte	E	480	35	19	823	262,701,600

*These numbers are used for classification purposes only and do not represent actual byte counts.

- The total number of tracks of all partitions plus the VORTEX nucleus must not exceed the disk's track capacity. (Note that the nucleus resides only on a disk on controller zero.) The size of the nucleus is equal to the memory size divided by the product of the number of sectors per track and 120. For this calculation, memory never exceeds 32K. For example, to calculate the number of tracks for a system with a type 2822-xx disk, perform the following:

$$\frac{32K}{62(120)} = \frac{32768}{7440} = 4.4$$ (Round up to 5 tracks for the nucleus)

  Or, to calculate the number of tracks for a 32K system with an F3094-xx or F3096-xx disk:

$$\frac{32K}{48(120)} = \frac{32768}{5760} = 5.5$$ (Round up to 6 tracks)

  To this value is added the number of sectors for VNO tasks as computed from the "npages" parameter in the MRY directive.

- The first partition specified by a PRT directive is listed in the bad-track table as the first sequential track number outside the nucleus area.

- If partitions are to contain object modules (such as the OM library or GO partitions), the following should be calculated:

$$\frac{b}{2(z)}$$

where:

b
: Is the number of 60-word binary records.

z
: Is the number of sectors per track.

- If partitions are to contain load modules (in the background and foreground library), the following should be calculated:

$$\frac{k}{120(z)}$$

where:

k
: Is the amount of memory required for a module.

z
: Is the number of sectors per track.

Example:

```
PRT,D00A,3,C
```

This example assigns the first three tracks to D00A (the operating system) with a protection code of C. Therefore, to access these tracks, the user must specify the C protection code.

6.3.12 VOL (VIRTUAL OVERLAY TASK DEFINITION) DIRECTIVE

Purpose:

This directive permits the user to specify the tasks which are to be designated as virtual nucleus overlays (VNOs).

Format:

The general format of the VOL directive is:

VOL,<taskname(1)>[,taskname(2)]...[,taskname(n)]

where:

taskname(n)
Is a 1 to 8 alphanumeric character name to be associated with a specific task. This task is to be designated as a VNO. The name must match the name in the first record of one of the object modules on the system generation library (SGL) tape.

Description:

This directive specifies the tasks which are to be designated as virtual nucleus overlays (VNOs). VNO tasks are loaded into memory by the VORTEX system initializer. Immediately before a VNO task executes, its physical page locations are output to the mapping hardware. The VNO task is then executed in the same manner as other nucleus-resident tasks.

NOTE

The error message SG50 is displayed during system generation as a notification that a task which could have been defined as a VNO task was not so defined and thus, the task is resident in the first 32K words of the nucleus.

The following restrictions apply to VNO tasks:

1. DCBs, FCBs, and I/O buffers used by VNO tasks must not be part of the task. They must reside in map 0 of logical memory during any I/O transfers.

2. Jumps to VNO tasks may not be performed unless the task has been mapped into map 0 of logical memory.

3. A VNO task may not exceed 31 pages or the capacity of the nucleus (one page = 512 words).

4. VNO tasks and the corresponding taskname must be attached to a resident TIDB. (Refer to the TDF directive for additional information.)

VNO tasks are generally I/O handlers.

Table 6-8 gives the correspondences between VNO tasks and VOL directive tasknames.

Example:

```
VOL,VZMTA
```

This example designates the magnetic tape handler to be a VNO task.

Table 6-8. VNO/VOL Correspondences

VNO Task	VOL Directive Name	Size
DnC	VZDC	4 pages
DnE	VZDE	3 pages
DnF	VZDF	3 pages
DnH	VZDH	6 pages
ER0H	VZERH	1 page
V$DI	V$DI	6 pages
TYnA	V$TYA	3 pages
CTnA	V$TYA	3 pages
TYnB	V$TYB	3 pages
CTnB	V$TYB	3 pages
MTnA	VZMTA	2 pages
MTnB	VZMTB	4 pages
LPnA	VZLPA	1 page
LPnB	VZLPB	1 page
CRnA	VZCRA	1 page
PTnA	VZPTA	2 pages
CPnA	VZCPA	1 page
FM0A	VZFMA	4 pages

The following tasks are not I/O drivers but should be used as VNO tasks to conserve nucleus memory space:

IT0E	VZITE	5 pages
V$SERV	V$SERV	3 pages

6.4 NUCLEUS/LIBRARY MODIFICATION DIRECTIVES

The directives described in this paragraph allow the user to modify the nucleus image and basic load module libraries at system generation time.

6.4.1 ADD (SGL ADDITION) DIRECTIVE

Purpose:

The ADD directive allows a user to add control records or object modules from the ALT device to the SGL.

Format:

The general format of the ADD directive is:

```
ADD,<name(1)>[,<name(2)>,]...[,<name(n)>]
```

where:

name(n)
: Is the name of the SGL control record or object module after which the new items are to be added. This parameter must be the name of a SGL control record or object module that will be included in the system as determined by the SGEN directives.

Description:

The ADD directive specifies the SGL control records and object modules after which new control records and/or object modules from the ALT device are to be added.

When the name of the SGL item specified by <name(n)> is read from the SGL, the item specified by <name(n)> is processed. After <name(n)> is processed, the message

```
ADD AFTER <name(n)>
READY
```

is printed on the OC device. The following responses may be entered after this prompt:

- If the response

  ALT

  is entered on the OC device, SGEN reads the next object module or alphanumeric record from the ALT unit, adds it to the nucleus image and prints the prompt

  READY

  on the OC device.

- If the response

  ALT, <name>

  where:

  name

  Is a one to six alphanumeric character name representing the title name of the module to be added.

  is entered, the ALT unit will be searched from its current position for the module specified by name. The parameter name can specify an object module name or a TDF record name. If an end-of-file is encountered prior to finding the module specified by name, an SG08 diagnostic occurs.

  To cause rewinding of the ALT unit prior to each search, set Sense Switch 1 prior to entering this response.

- If the response

  LIB

  is entered, processing of the SGL from the DIR unit will continue.

Any of these responses may be entered after the prompt

READY

is printed on the OC device.

Example:

Add three object modules from the ALT device after the object module named V$IOC.

```
ADD,V$IOC
```

When this directive is processed, the following dialog will be printed on the OC device. For clarity, items input by the user are shaded.

```
ADD AFTER V$IOC
READY
ALT
READY
ALT
READY
ALT
READY
LIB
```

Example:

Specify that the module CTTY1C is to be added to the SGL after the module CTTYOB.

```
ADD,CTTYOB
```

After the module CTTYOB is read from the SGL, the following dialog takes place. User inputs are shaded.

```
ADD AFTER CTTYOB
READY
ALT,CTTY1C
READY
LIB
```

6.4.2 DEL (DELETE) DIRECTIVE

Purpose:

This directive allows the user to delete nucleus control records and object modules that would normally be included in the nucleus.

Format:

The general format of this directive is:

DEL,<name(1)>[,name(2)]...[,name(n)]

where:

name(n)
Is the name of a control record or an object module that is to be deleted.

Description:

This directive allows modules to be deleted from the nucleus. It is entered with the rest of the input directives.

When the name of the specified item is read from the SGL tape, the item is skipped and processing continues with the next control record or object module.

Example:

DEL,DMEMRY

This example causes the panic memory dump program DMEMRY to be deleted from the nucleus.

6.4.3 LAD (LIBRARY ADDITION) DIRECTIVE

Purpose:

The LAD directive is used to add load module packages from the ALT unit to the basic load module libraries during system generation.

Format:

The general format of this directive is:

LAD,<name(1)>[,<name(2)>]...[,<name(n)>]

where:

name(n)
Is the name of the load module package (specified in the SLM directive) after which the new items from the ALT unit are to be added.

Description:

The LAD directive allows a user to add a new load module package to the basic load module libraries after an existing load module package.

When the load module package specified by <name(n)> is read from the SGL tape, it is processed, and the message:

```
ADD AFTER <name(n)>
```

is printed on the OC device. After this message is printed, the user may enter the response:

```
ALT
```

or

```
LIB
```

These responses are described with the ADD directive.

Example:

```
LAD,PROG1,PROG2
```

This example specifies that new load module packages are to be added after the existing load module packages PROG1 and PROG2.

6.4.4 LDE (LIBRARY DELETION) DIRECTIVE

Purpose:

The LDE directive is used during system generation to delete load module packages from the basic load module libraries.

Format:

The general format of this directive is:

```
LDE,<name(1)>[,<name(2)>]...[,<name(n)>]
```

where:

name(n)
: Is the name of a load module package (as specified in an SLM directive) that is to be deleted from the SGL.

Description:

This directive allows the user to delete load module packages from the basic load module libraries.

When the name of a load module package specified in the LDE directive is read from the SGL, SGEN skips the load module package. Processing then continues with the following load module package.

Example:

```
LDE,PROG1,PROG2
```

This example specifies that during system generation, all load module packages named PROG1 and PROG2 are to be deleted.

6.4.5 LRE (LIBRARY REPLACEMENT) DIRECTIVE

Purpose:

This directive is used to specify the load module packages which are to be replaced with new load module packages during system generation.

Format:

The format of this directive is:

```
LRE,<name(1)>[,<name(2)>]...[,<name(n)>]
```

where:

name(n)
: Is the name of a load module (as specified in the SLM directive) which is to be replaced.

Description:

The LRE directive allows the replacement of existing load module packages with new load module packages from the ALT device.

When the name of a load module specified in the LDE directive is read from the SGL, the load module is skipped, and the message

```
REPLACE name(n)
READY
```

is printed on the OC unit. After this message is printed, the response

LIB

or

ALT

may be entered on the OC device. These responses are described with the ADD directive.

Example:

LRE,BLMGEN

This example specifies that the load module package BLMGEN is to be replaced during system generation.

6.4.6 REP (SGL REPLACEMENT) DIRECTIVE

Purpose:

The REP directive is used to specify the control records and object modules which are to be replaced with new control records and/or object modules during nucleus generation.

Format:

This directive has the format:

REP,<name(1)>[,<name(2)>]...[,<name(n)>]

where:

name(n)
Is the name of the control record or object module that is to be replaced.

Description:

The REP directive allows the user to specify control records and/or object modules which are to be replaced with new object modules or control records during nucleus generation. The input for the new object modules or control records is read from the ALT unit.

When the name of an item specified in the REP directive is read from the SGL, the item is skipped, and the message

```
REPLACE <name(n)>
READY
```

is printed on the OC device. After this message is printed, the user may enter the response

```
ALT
```

or

```
LIB
```

or

```
ALT,<name>
```

These responses are described with the ADD directive.

Example:

```
REP,V$IOC,V$TYB
```

This example specifies that the object modules V$IOC and V$TYB are to be replaced during system generation.

6.5 BUILDING AND MODIFYING THE VORTEX NUCLEUS

6.5.1 GENERAL

This paragraph briefly explains the procedures and control directives used to build the VORTEX nucleus. The control directives described in this paragraph are read from the SGL when the VORTEX nucleus is built. These control directives are also read from the ALT device when adding or replacing load modules during phase 3 of system generation.

The information contained in this paragraph should be used to prepare the input that is read from the ALT device when load modules are added or replaced (using the ADD or REP directives) during phase 3 of system generation.

6.5.2 BUILDING THE VORTEX NUCLEUS

If a full system generation has been specified by the EDR directive, the nucleus processor is loaded when directive processing is complete. The nucleus processor then reads control records and object modules from the SGL and, according to the specifications made by the SYSGEN directives, either ignores

items read from the SGL or incorporates items read from the SGL into the VORTEX nucleus.

The control records that are used to build the VORTEX nucleus are:

- SLM - Start Load Module
- TDF - Build task identification block
- MEM - Default extra memory pages
- END - End of nucleus library

6.5.3 SLM (START LOAD MODULE) DIRECTIVE

Purpose:

This directive is used to indicate the beginning of a load module.

Format:

The general format of the SLM directive is:

```
SLM,name
```

where:

name
: Is the name of the load module which follows the SLM directive.

Description:

The SLM directive indicates the beginning of a load module, system initializer, or the VORTEX nucleus.

Example:

```
SLM,VORTEX
```

This example indicates the beginning of the VORTEX nucleus.

```
SLM,V$TDF
```

This example indicates the beginning of a load module with the name of V$TDF.

6.5.4 TDF (BUILD TASK IDENTIFICATION BLOCK) DIRECTIVE

Purpose:

The TDF directive specifies all parameters necessary to build a task identification block (TIDB) in the VORTEX nucleus.

Format:

The TDF directive has the format:

```
TDF,<name>,<exec>,<ctrl>,<stat>,<level> [,V75] [,<taskname>]
```

where:

name
: Is the name given to the TIDB for linking purposes. This parameter must be from 1 to 6 alphanumeric characters.

exec
: Is the name associated with the execution address of the task. This parameter must be from 1 to 6 alphanumeric characters in length.

ctrl
: Is the name of the controller table required for teletypewriter and CRT processing tasks, or 0 for any other task. If this parameter is not 0, it must be from 1 to 6 alphanumeric characters.

stat
: Is the octal value of the 16 bit binary status word. The meaning of the individual bits in the status words is shown in Table 6-9.

levl
: Is the priority level of the related task.

V75
: If present, indicates that the system is an extended register system (V77, V76 or V75) and that a long TIDB is to be used.

taskname
: Is a 1 to 8 alphanumeric name associated with a VNO task. Refer to the description of the VNO directive for additional information.

Description:

The TDF directive specifies all the parameters necessary to build a TIDB in the VORTEX nucleus. This directive causes a resident TIDB to be created for the task specified by "name". The task for which the TIDB is to be created may or may not be a resident task as specified by "stat". (Refer to the description of the TSK directive for the generation of resident tasks without a resident TIDB.

NOTES

1. If the taskname parameter is specified, the VNO task and the associated TIDB must be resident.

2. A TDF record on magnetic tape or disk must be 60 words long and blank filled.

Example:

```
TDF,TIDFTS,PTSK,0,05400,10,V75
```

This example defines a foreground resident task named FTSK at which will execute on boot at priority level 10.

Table 6-9. TIDB Status-Word Bits

Bit	When Set Indicates	Explanation
15	Interrupt suspended	The task is suspended during the processing of a higher-priority task. The contents of volatile registers are stored in TIDB words 12-16 (interrupt stack).

Table 6-9. TIDB Status-Word Bits (Continued)

Bit	When Set Indicates	Explanation
14	Task suspended	The task is suspended because of I/O or because it is waiting to be activated ay an interrupt, time delay, or another task. The task is activated whenever this bit is zero, or if TIDB word 3 has an interrupt pending and the task expects the interrupt.
13	Task aborted (or exited if under micro-VORTEX)	The task is not activated. All stacked I/O is aborted, but currently active I/O is completed.
12	Task exited (or task under nucleus control of micro-VORTEX)	The task is not activated. All stacked and currently active I/O is completed.
11	TIDB resident	The TIDB (drivers, task-interrupt processors, resident tasks, and time-scheduled tasks) is resident and not released when the task is aborted or exited.
10	Task resident	The task is resident and not released when aborted or exited.
9	Foreground task	The task is a foreground task (priority 2-31).
8	Check-point flag	Set: may be check-pointed by a lower priority task. Reset: may not be check-pointed by a lower priority task.
7	Task scheduled by time increment	The task will be loaded when a specified time interval is reached.

Table 6-9. TIDB Status-Word Bits (Continued)

Bit	When Set Indicates	Explanation
6	Time delay active	The clock decrements the time counter that, upon reaching zero, clears bit 14, thus resuming the task.
5	Task check-pointed	The background task is check-pointed and suspended. I/O is not activated.
4	Error in task	The task contains an error that will cause an error message to be output.
3	Task interrupt expected	A task interrupt is expected.
2	Overlay task	The task contains overlays.
1	Task-schedule this task	The scheduling task is suspended until the scheduled task exits or aborts.
0	Task searched, allocated and loaded	The task is loaded in memory and is ready for execution.

6.5.5 END DIRECTIVE

Purpose:

This directive is used to indicate the end of the system initializer or the VORTEX nucleus.

Format:

The END directive has the format:

```
END
```

Example:

Indicate the end of the system initializer.

```
END
```

6.6 SYSTEM LIBRARY AND RESIDENT TASK CONFIGURATION

6.6.1 GENERAL

This paragraph briefly describes the procedures and control directives used to build the VORTEX system library and resident tasks. The control directives described in this paragraph are read from the SGL when the VORTEX system libraries and resident tasks are built. These control directives are also read from the ALT device when adding items to the system library (with the LAD directive) or replacing items in the system library (with the LRE directive) during phase 4 of system generation.

The information contained in this paragraph should be used to prepare the input that is read from the ALT device when library items are added or replaced.

6.6.2 BUILDING THE SYSTEM LIBRARIES AND RESIDENT TASK CONFIGURATION

The library generator is loaded when nucleus processing is complete.

Library processing consists of the reading of control records and object modules from the SGL and construction of the system library from these inputs. The only manual operations required to build the system library are the replacement and addition of load modules.

The control records used to build the system libraries and perform resident task configuration are:

- SLM Start Load Module
- TID Task identification block specification
- OVL Overlay
- ESB End of segment
- MEM Add extra memory blocks (background tasks only)
- END End load module generation

6.6.3 LOAD MODULES

A load module is a complete task or operation that can be executed by the VORTEX system in foreground or background. Load modules themselves reside in the foreground or background library or in user libraries.

Load modules are constructed from sets of binary object modules interspersed with alphanumeric control records. The control records indicate the beginning and end of the data that is to be incorporated into the load module. The control records also specify certain parameters that are required to construct the load module. The group of binary object modules and control records used to construct a load module is called a load module package (LMP). Each LMP begins with a SLM control record and ends with an END control record. An LMP also includes all the control records and binary object modules between the SLM and END control records.

Figure 6-1 shows a LMP without overlays. Figure 6-2 shows a LMP with overlays.

6.6.4 SLM (START LOAD MODULE PACKAGE) DIRECTIVE

Purpose:

The SLM directive is used to indicate the start of an LMP.

Format:

The format of the SLM directive is:

```
SLM,name
```

where:

name
: Is the name of the load module package that begins with the SLM directive.

Description:

The SLM directive indicates the start of an LMP. All control directives and binary object modules which are between the SLM directive and the next END directive are considered to be part of the LMP with the name specified by "name".

*	SLM,name1
*	TID,name2,...
	Object Modules Comprising the Root Segment
*	ESB
*	END

* = Alphanumeric control record

Figure 6-1. Load Module Package for Module Without Overlays

*	SLM,name1
*	TID,name2,...
	Object Modules Comprising the Root Segment
*	ESB
*	OVL,name3,...
	Object Modules Comprising the First Overlay Segment
*	ESB
*	OVL,name4,...
	Object Modules Comprising the Second Overlay Segment
	Object Modules Comprising the nth Overlay Segment
*	ESB
*	END

* = Alphanumeric control record

Figure 6-2. Load Module Package for Module With Overlays

6.6.3 LOAD MODULES

A load module is a complete task or operation that can be executed by the VORTEX system in foreground or background. Load modules themselves reside in the foreground or background library or in user libraries.

Load modules are constructed from sets of binary object modules interspersed with alphanumeric control records. The control records indicate the beginning and end of the data that is to be incorporated into the load module. The control records also specify certain parameters that are required to construct the load module. The group of binary object modules and control records used to construct a load module is called a load module package (LMP). Each LMP begins with a SLM control record and ends with an END control record. An LMP also includes all the control records and binary object modules between the SLM and END control records.

Figure 6-1 shows a LMP without overlays. Figure 6-2 shows a LMP with overlays.

6.6.4 SLM (START LOAD MODULE PACKAGE) DIRECTIVE

Purpose:

The SLM directive is used to indicate the start of an LMP.

Format:

The format of the SLM directive is:

```
SLM,name
```

where:

name
: Is the name of the load module package that begins with the SLM directive.

Description:

The SLM directive indicates the start of an LMP. All control directives and binary object modules which are between the SLM directive and the next END directive are considered to be part of the LMP with the name specified by "name".

*	SLM,name1
*	TID,name2,...
	Object Modules Comprising the Root Segment
*	ESB
*	END

* = Alphanumeric control record

Figure 6-1. Load Module Package for Module Without Overlays

*	SLM,name1
*	TID,name2,...
	Object Modules Comprising the Root Segment
*	ESB
*	OVL,name3,...
	Object Modules Comprising the First Overlay Segment
*	ESB
*	OVL,name4,...
	Object Modules Comprising the Second Overlay Segment
	Object Modules Comprising the nth Overlay Segment
*	ESB
*	END

* = Alphanumeric control record

Figure 6-2. Load Module Package for Module With Overlays

Example:

```
SLM,ABC
```

This directive indicates the start of a LMP with the name of ABC.

6.6.5 TID (TIDB SPECIFICATION) DIRECTIVE

Purpose:

The TID directive is used to specify the parameters necessary to build the TIDB required for each load module.

Format:

The TID directive has the format:

```
TID,name,mode,ovly,lun
```

where:

name
: Is the name of the task for which the load module is to be generated. This parameter must be from 1 to 6 alphanumeric characters in length.

mode
: Is 1 if the task is a background task or 2 if the task is a foreground task.

ovly
: Is the number of overlay segments of the task. This parameter must be 0 if the task has no overlay segments.

NOTE

The ovly parameter may not have a value of 1.

lun
: Is the logical unit number onto which the task is to be cataloged.

Description:

The TID directive is used to specify the parameters necessary to build a TIDB for a task. Once a TID directive is input and

processed, object modules which make up the task are input, processed, and output to the logical unit specified by "lun". This process is continued until an ESB directive is read.

Examples:

```
TID,PROG1,1,0,105
```

This example specifies that a TIDB is to be generated for a task with the name of PROG1. The task has no overlays and is cataloged on logical unit 105.

```
TID,PROG2,2,4,106
```

This example specifies that a TIDB is to be generated for a task with the name of PROG2. The task has four overlay segments and is to be cataloged on logical unit number 106.

6.6.6 OVL (OVERLAY) DIRECTIVE

Purpose:

The OVL directive is used to indicate the beginning of an overlay segment.

Format:

The format of the OVL directive is:

```
OVL,name
```

where:

name
: Is the name of the overlay segment. This parameter must be from 1 to 6 alphanumeric characters in length.

Example:

```
OVL,SINE
```

This example specifies the beginning of the overlay segment with the name SINE.

6.6.7 ESB (END SEGMENT) DIRECTIVE

Purpose:

This directive is used to indicate the end of a segment.

Format:

The ESB directive has the format:

```
ESB
```

Description:

The ESB directive indicates the end of a root segment, overlay segment or load module. This directive indicates all object modules comprising the segment have been loaded and processed.

When this directive is read, the CL library, which was generated during nucleus processing, is searched to satisfy all undefined externals.

Example:

Indicate the end of an overlay segment.

```
END
```

6.6.8 MEM (MEMORY) DIRECTIVE

Purpose:

The MEM directive is used to specify the default number of extra memory blocks to be attached to the background task specified in the TDF directive.

Format:

The format of the MEM directive is:

```
MEM,n
```

where:

n
: Is the number of extra memory pages to be attached to the background task specified in the TID directive.

Description:

The MEM directive is an optional directive used to specify the number of extra memory pages to be attached to the background task specified in the TID directive. The MEM directive, if used, must appear after the last ESB directive and before the END directive.

Example:

```
MEM,2
```

This example specifies that 2 extra memory pages are to be attached to a background task.

6.6.9 END DIRECTIVE

Purpose:

The END directive is used to indicate the end of load module generation.

Format:

This directive has the form:

```
END
```

Example:

Specify the end of load module generation.

```
END
```

Section 7
SYSTEM GENERATION REQUIREMENTS FOR INDIVIDUAL SOFTWARE MODULES

7.1 GENERAL

This section is a guide for determining the system generation requirements for individual VORTEX software modules. Only those modules with unique requirements are addressed herein. Note that the modules in this section are not necessarily optional.

This information should aid the user in the preparation of the input directives and in the acceleration of the entire system generation (SYSGEN) process. The input directives are detailed in Section 6.

NOTE

This section does not explain the modules themselves; it only explains their unique SYSGEN requirements. For anexplanation of the function of each module, refer to the VORTEX II Operating System Programmer Reference Manual.

7.2 INTERTASK COMMUNICATION MODULE WITH EXTENDED CAPABILITIES (ITE)

This module requires the following directives at SYSGEN time:

1. A modified equipment directive must be entered to define the ITE:

   MOD,IT0E,0,1,0,0

2. DEF directives are required to define special parameters:

   . DEF,VI$MXQ,n

where:

n

Is the number of elements in the mailbox queue. The value is placed in bits 9 throughl5 and must be octal. The recommended value for n is 024000.

. DEF,VI$NPG,m

where:

m

Is the number of physical pages to be used for the internal pool. The recommended value for m is 4.

7.3 INTERTASK COMMUNICATION MODULE-BASIC VERSION (ITC)

The ITC module requires the following directive to be included at system generation time:

EQP,IT0C,0,1,0,0

This directive causes the entry name of the individual reentrant subroutines (VI$MBX,VI$FRE,VI$PST,VI$CPY,VI$FND, and VI$AWK) to be placed in the core library at system generation time.

7.4 WRITABLE CONTROL STORE (WCS) AND MICRO-VORTEX

If this feature is to be included in the system, the following directives must be entered at SYSGEN time:

1. The appropriate WCS module must be selected:

DEF,V$$WCS,n

where:

n

= 1 for FORTRAN accelerator (without Floating Point Processor) only, or with commercial firmware;

= 2 for FORTRAN accelerator (with Floating Point Processor) only, or with commercial firmware;

= 3 for commercial firmware only.

2. A logical unit number (lun) must be associated with an unprotected disk partition if WCS is included in the system:

ASN,116=D00n

where:

n
is an unkeyed partition on the system disk that contains at least 160 sectors and is not assigned to GO, PO, SS, BI, BO, or CU. This partition is to contain the WCS image. On a model C disk (typenumber 2822-xx) this partition must begin within the first 32K sectors.

3. For a V77-600 system with a software nucleus or a V77-800 system with WCS, the following must be used:

EQP,WCS,da,p,0,0

where:

da
Is the device address of the WCS.

p
Is the number of pages of WCS.

4. For V77-600 systems with firmware (micro-VORTEX--CPU = 3), the following directives must be used in addition to number 1 and in place of numbers 1 and 3.

a) MOD,WCS,da,#p,0,0

where:

da
Is the device address of the WCS.

p
Is the number of pages of WCS.

b) MOD,WC0m,0,1,0,0

where:

m

Is the model code of the disk driver containing the system image (models C, D, E, F, and H only). This directive defines the system resident disk to the firmware nucleus. It also defines the driver which is used to load the operating system nucleus into the WCS at bootstrap time and during any power restart phases (that is, following a power fail). Permissable mode codes and their corresponding type/feature numbers are as follows:

Model Code	Model No.	Type/Feature
C	70-7500	2822-xx
D	70-7510	
E	70-7520	2823-xx
	70-7530	
		2824-xx
F	70-7603	F3094-xx
	70-7613	F3096-xx
		F3310-xx
H	70-755x	2825-xx
		2826-xx
		2842-xx
		2843-xx
		8433-xx

c) DEF,TAPEDA,da

where:

da

Is the device address of the magnetic tape which is to be used for emergency WCS reload purposes.

d) DEF,TAPEBA,ba

where:

ba
Is the BIC address of the magnetic tape which is to be used for emergency WCS reload purposes.

e) DEF,WCSAD,da

where:

da
Is the device address of the WCS. This directive specifies backup information.

f) DEF,V$$WCS,n

where:

n
= 0 for micro-VORTEX firmware in pages 1 and 2.

= 1 (default) for ACLOM firmware in page 1, COBOL firmware in page 2, and micro-VORTEX firmware in pages 3 and 4.

= 2 for FPA0M firmware in page 1, COBOL firmware in page 2, and micro-VORTEX firmwarein pages 3 and 4.

= 3 for COBOL firmware in page 1 only (not valid for micro-VORTEX).

Refer to Table 7-1 for clarification of firmware combinations and corresponding permitted values of the parameter "n".

Table 7-1. DEF,V$$WCS,n Directive

Firmware Combination	Permitted Values of "n"				Minimum Required Number of WCS Pages
	0	1	2	3	
Micro-VORTEX only	Yes	No	No	No	2
FORTRAN without FPP only	No	Yes	No	No	1
FORTRAN with FPP only	No	No	Yes	No	1
Comercial only (CPU≠3)	No	No	No	Yes	1
FORTRAN without FPP and Commercial	No	Yes	No	No	2
FORTRAN WITH FPP and Commercial	No	No	Yes	No	2
FORTRAN without FPP and Commercial and Micro-VORTEX	No	Yes	No	No	4
FORTRAN with FPP and Commercial and Micro-VORTEX	No	No	Yes	No	4
V$RERS	Yes	Yes	Yes	Yes	0
V$RERF	No	No	Yes	No	1
SFTOM	Yes	Yes	Yes	Yes	0
ACLOM	No	Yes	No	No	1
FPPOM	Yes	Yes	Yes	Yes	0
FPAOM	No	No	Yes	No	1

7.5 DATA COMMUNICATIONS (VTAM)

To properly generate the data communication modules, the user must know the exact number of data communication multiplexers (DCMs) associated with the particular system being generated. Then the following directives can be used.

1. The memory directive on a VTAM system must reflect the fact that each DCM uses 512 words of memory for hardware control words. In addition, the <memory> parameter of the MRY directive must be a multiple of 512 (for example, 74000, or 75000).

   Example:

   The following directive may be used for a 32K system, saving AID III and BLD II. The LCB is at 75000 and VTAM is to be included in the system:

   ```
   MRY,074777,200.....
   ```

2. EQP directives and DEF directives are required for each data communication multiplexer (DCM) and terminal control module (TCM). The first DCM must be unit 0.

   Example:

   ```
   EQP,MX0A,070,1,0,0
   EQP,MX1A,071,1,0,0
   ```

   These EQP directives define two DCMs. Note that all hardware addresses and device addresses depend entirely upon the specific configuration of the computer system. Refer to the "System Memo" accompanying each particular system.

3. Six priority interrupt modules (PIMs) must be defined for each DCM. For VORTEX II, the interrupt connection parameter (s) should be 2. (Refer to Section 6 for a description of the PIM directive.)

```
PIM,070,C52LIP,000,2
PIM,071,C52LIP,001,2
PIM,072,C52LIP,002,2
PIM,073,C52LIP,004,2
PIM,074,C52LIP,003,2
PIM,075,C52CIH,005,2
PIM,060,C52LIP,010,2
PIM,061,C52LIP,011,2
PIM,062,C52LIP,012,2
PIM,063,C52LIP,014,2
PIM,064,C52LIP,013,2
PIM,065,C52CIH,015,2
```

These PIM directives define twelve priority interrupt modules for two DCMs (six PIMs per DCM). MY0A uses the interrupt locations for PIM7 and MX1A uses the interrupt locations for PIM6. Note that all hardware addresses and device addresses depend entirely upon the specific configuration of the computer system. Refer to the "System Memo" for specific configuration information.

The PIM directives for a DCM define directly connected interrupts. The names of the programs servicing the directly connected interrupts are listed in Table 7-2.

Table 7-2. Directly Connect Interrupts

Event Word Value	Interrupt Description	Directly Connected Interrupt Servicing Routine Name
0x0	input byte count = 0	C52LIP
0x1	output byte count = 0	C52LIP
0x2	line error	C52LIP
0x4	control character detected	C52LIP
0x3	status change	C52LIP
0x5	control	C52CIH

The value of the event word parameter (r) in the PIM directive (Section 6) is taken from Table 7-2. The value x is the number of the DCM being described in the PIM directive. For example, in a system using only one DCM, x = 0 in all six PIM directives. In a system using two DCM's the first DCM would be described by six PIM directives with x = 0, and the second by six PIM directives with x = 1.

4. VORTEX requires a DEF directive to define the address of the line control block (LCB) for each DCM. This directive has the form:

```
DEF,V$LCWn,<addr>
```

where:

n
: Is a numeric value corresponding to MXna (that is, V$LCW0 = the first DCM, V$LCW1 = the second DCM, etc.).

<addr>
: Is the address at which the DCM line control block (LCB) is wired. Refer to the "System Memo" for this value.

Example:

```
DEF,V$LCW0,075000
DEF,V$LCW1,074000
```

5. A DEF directive is required to specify if high speed polling is to be enabled in the software. This directive has the form:

```
DEF,V$POLL,n
```

where:

n
: Is 1 if the system is using the BSC polling mode and the system contains the E-3714 BSC LAD. If n is 0, this is not the case.

NOTE

Any BCS lines which are to be polled (i.e., IBM 3270 slave lines) must have the "high speed poll option" installed on the LAD.

6. If the terminal control module is required, use the following directive:

```
EQP,TCOA,00,1,0,0
```

Omission of this directive will cause the TCM components to be deleted.

7. An ASN directive is required to associate each DCM and terminal controller with the physical device.

```
ASN,27:MC=MX00    (First DCM)
ASN,26:MD=MX10    (Second DCM)
ASN,28:TT=TC00    (Terminal Controller)
```

7.6 PRONTO

A disk partition with a size of at least 1600 sectors must be assigned to logical unit number 189 for PRONTO object and source modules:

```
ASN,189=Dcup
```

where:

c
: Is the controller number (0,1,2, or 3).

u
: Is the unit number (0 through 7).

p
: Is the partition letter as specified in the PRT directive.

The following disk partitions should contain space in addition to what is required by the basic VORTEX system. The following lists the logical unit names, the amount of space required by VORTEX, and the purpose this space is required:

LUN	SIZE REQUIRED	PURPOSE
BL	150 sectors (maximum)	background tasks
FL	970 sectors (maximum)	load modules
OM	750 sectors (maximum)	object modules
189	1600 sectors (maximum)	object and source modules

7.7 SIMULTANEOUS PERIPHERAL OUTPUT OVERLAP SUBSYSTEM (SPOOL)

The following SYSGEN directives should be considered prior to generating the peripheral spooler subsystem as part of the system:

1. To realize greater efficiency from the SPOOL subsystem, at least two disks should be used. Basically, one is dedicated to the VORTEX system and includes PO files. The other is for the SPOOL files and additional storage. An EQP directive is required for each SPOOL file to be created. For example:

   ```
   EQP,SPnA,0,1,0,0
   ```

   where:

   n
   Is the SPOOL unit number.

   Ommission of this directive results in the deletion of all SPOOL modules.

2. The following changes to the preset logical unit numbers and names are permissable when this spooler subsystem is generated:

   ```
   ASN,5=SP00      (list output)

   ASN,7=SP10      (punch card binary output)

   ASN,xx=SP20     (magnetic tape)
   ```

Each of the SPnn assignments is then associated with one of the logical unit numbers 180 through 187 to which the spooled data is output, such as:

```
ASN,180:S0=LP00

ASN,181:S1=CP00

ASN,182:S2=MT00
```

3. The disk partition created for the SPOOL files must be assigned to logical unit 107, with the logical name of SX and the protection key of S. An example of this is:

```
ASN,107:SX=D00I
PRT,D00I,380.5
```

where:

Dnna
: Represents the name of the partition, for example:

  D00L

4. The peripheral spooler subsystem requires a partition for the holding of the output data until the designated peripheral is ready for this data. The spool partition must have a protection key of S and correspond to logical unit 107. An example of the format for the spool partition is:

```
PRT,Dnna,t,S
```

where:

t
: Is the number of tracks in the partition. This number may be in octal or decimal.

Dnna
: Is the name of the partition. This corresponds to the ASN directive for logical unit number 107.

To determine the optimum size for SPOOL output partitions, the following should be calculated:

(a - p) t = output line = sectors

where:

a
: Is the maximum number of lines per minute a program (application, language processor, etc.) can output if not bound by the printer speed.

b
: Is the printer speed in lines per minute.

p
: Is the maximum time (in minutes) a program should be engaged in output operations.

For example, if an assembly is expected to generate 1000 lines per minute for 10 minutes and the printer speed is 600 lines per minute, the following is performed to calculate the number of sectors required to spool this output:

a = 1000, p = 600 and t = 10

1000 - 600 = 400

400(10) = 4000 lines of output will be output to the SPOOL file.

5. The system generation procedures do not permit the actual creation of spool files. The SPOOL files must be created using the FMAIN utility. Refer to "Simultaneous Peripherial Output Overlap (SPOOL)" in the VORTEX II Operating System Programmer Reference Manual.

NOTES

The SPOOL files can be of any size. If the SPOOL file becomes to full, the OC device displays IO. This is only an indicator and can be ignored. However, it is occurs frequently, the files sizes should be increased.

If the system contains the multitask pooler(MSPOOL) it cannot contain SPOOL.

7.8 MULTITASK SPOOLER (MSPOOL)

If the multitask spooler is to be generated as part of the system, its driver (SD00) requires several assignments as follows:

1. An EQP directive is required for each MSPOOL file to be created. For example:

```
EQP,SD0A,0,1,0,0
```

2. Logical unit 107 must be assigned to an unprotected partition. This partition is the first spool file partition. The spooler uses this partition and any subsequent unkeyed partitions on the specified disk unit for spool space as needed. Therefore, the selection of the particular partition should be based upon the estimated spool space required (the spooler uses one sector per print line) and the amount of disk space to be free from spooler use. Note that spool print files can coexist with other files on the same partition.

```
ASN,107=D00J
```

3. Logical unit 108 and 109 must be assigned to DUM:

```
ASN,108=DUM,109=DUM
```

4. Assign directives may be included for logical units 110 through 123, as required. 110 is used by the print task number 1 (S$SP01), 111 is used by print task number 2 (S$SP02), etc. They must be assigned to a dummy device. For example, the following directives must be included for a system using spool print tasks S$SP01, S$SP02, and S$SP04:

```
ASN,110=DUM

ASN,111=DUM

ASN,113=DUM
```

7.9 BIC (BUFFER INTERLACE CONTROLLER)

In system configurations where the BIC device address and trap locations (specified in the System Memo) deviate from the default values (see Table 7-3), a DEF directive for each deviating BIC is required. The DEF directive format is:

```
DEF,!BICxx,y
```

where:

xx
: Is the device address.

y
: Is the trap location.

Table 7-3. BIC Default Trap Locations

BIC Device Address	Default Trap Location
20	100
22	102
24	104
26	106
70	110
72	112
74	114
50	116
52	120
54	122
56	124
60	126
62	130
64	132
66	134
76	136

For example, the BICs for a system are at device addresses 20, 22, 24, 26, 72, 74, and 50. The BICs are on PIM 0, line 0 through 6. The first 4 BICs trap at their default locations. However, the last three BICs are to be redefined as follows:

```
DEF,!BIC72,0110

DEF,!BIC74,0112

DEF,!BIC50,0114
```

Note that BICs are not required to be on consecutive lines. Table 7-3 lists the BIC default trap locations.

7.10 VZFMA FILE MAINTENANCE DRIVER

An equipment directive and an assign directive are required for the file maintenance driver (VZFMA). These directives must always be entered as follows:

```
EQP,FM0A,0,1,0,0

ASN,115:FM=FM00
```

NOTE

VORTEX systems which use the multitask spooler (MSPOOL), FMUTIL, TSS or any other task which makes a VORTEX file maintenance driver request (a V$FILE call) must include these directives.

7.11 PATCH

If the PATCH program is to be included in the VORTEX system, a DEF directive must be used at system generation time to create a reserved area at the bottom of the VORTEX nucleus region. This reserved area is used to accommodate system patches. The format of the DEF directive is:

```
DEF,V$PTSZ,nn
```

where:

nn
Is the number of words to be reserved.

7.12 VORTEX II MEMORY DUMP ROUTINE (DSYSTM)

The VORTEX II system generation routines require the following directives to include the memory dump routine DSYSTM.

- An EQP directive is required for the DSYSTM driver. This directive must have the form:

  EQP,RM0x,0,1,0,0

  where:

  x
  Is the model code of the disk device which is to contain the dump files.

- A PRT directive which specifies the disk partition which is to contain the dump image files. This partition must have a protection code of Z. This directive must have the format:

  PRT,Dcup,nnn,Z

  where:

  c
  Is the controller number of the disk.

  u
  Is the unit number of the disk.

  p
  Is the letter identifier of the partition.

  nnn
  Is the partition size in tracks. This number partition should contain at least 1.5 * (5*k) sectors where k is the number of 512 word pages in the system.

- Lun 190 must be assigned to the partition which is to contain the dump image files. The ASN directive which performs this function must have the format:

  ASN,190,Dcup

  where:

  c
  Is the controller number of the disk as specified in the PRT directive.

  u
  Is the unit number of the disk as specified in the PRT directive.

  p
  Is the partition letter as specified in the PRT directive.

7.12.1 DSYSTM POST SYSGEN REQUIREMENTS

The following post system generation requirements must be met to support the system memory dump:

- The foreground program FILINT and the background program DSPMEM must be cataloged by standard VORTEX II job streams.

- The dump image files must be created using the FMAIN directive:

  CREATE,190,Z,IMAGEx,512,n

  where:

  x
  Is the image file number, 0-9.

  n
  Is the number of 512-word records required.

File numbers must be continguous (i.e., 0, 1, 2, 3) and begin with zero. The number of records should be at least 1-1/2 times the number of pages in the system.

- The following patch must be made to the VORTEX nucleus to call DSYSTM when an EX20 through EX24 error occurs.

```
.B
%=V$PSTR+n                          (where n is the next
                                     available patch
                                     location)
.A                                   (or .P)
%,CALL,DSYSTM,JMP,V$DISP
V$DYS,JMP,%
```

7.13 DMEMRY

DMEMRY is part of the VORTEX II SGL as a nucleus resident task. DMEMRY is included in the VORTEX nucleus unless the following directive is specified:

```
DEL,DMEMRY
```

7.14 FORTRAN REENTRANT RUNTIME

If the FORTRAN reentrant runtime nucleus versions V$RERF (used only if FPAOM is loaded) and V$RERS (used with any other OM library) are not required, both must be deleted. If only one of V$RERF or V$RERS is required, the module not needed must be deleted. Delete these modules by using the directives:

```
DEL,V$RERF
```

and

```
DEL,V$RERS
```

Section 8
SYSTEM GENERATION REQUIREMENTS FOR INDIVIDUAL HARDWARE DEVICES

8.1 GENERAL

This section is a guide for determining individual requirements for individual hardware devices used with the VORTEX system. Only those modules with inidividual requirements are discussed in this section.

This information will aid the user in the preparation of the input directives and the acceleration of the entire system generation (SYSGEN) process. The inputs are described in Section 6.

8.2 TYPE 2825-xx, 2842-xx, 2826-xx, 8433-xx (70-755x) DISK

The type 2815-xx, 2842-xx, 2826-xx, 2843-xx, and 8433-xx disks require an EQP directive for each controller. In addition, an EQP directive is required for the model code E error routine VZERH. This EQP directive is a dummy EQP directive used to select the error module. Only one such EQP directive is required regardless of the number of controllers.

Example:

For a system containing two 2825-xx, 2842-xx, 2826-xx, 2843-xx, or 8433-xx disks on device addresses 012 and 014, the following directives are required:

```
EQP,D0H,012,1,0,10
EQP,D1H,014,1,0,10
EQP,ER0H,0,1,0,0
```

Although ER0H does not use a BIC, a BIC device address of 1 is used in the EQP,ER0H,0,1,0,0 directive. This is necessary since the EQP directive requires a non-zero value for the BIC device address.

The 2825-xx, 2826-xx, 2844-xx, 2843-xx, and 8433-xx disks require one PIM directive for each controller. The PIM directive connects the interrupt acknowledge interrupt with the directly connected interrupt handler V$DMPnH where n is the disk controller number.

Example:

A dual drive type 2825-xx, 2826-xx, 2842-xx, 2843-xx, or 8433-xx disk on one controller requires the directives:

```
PIM,p(1),DMDIOH,1,2
```

where:

p(1)
: Is the interrupt line number.

8.3 V77-400 WITH V77-600 COMPATIBLE CLOCK OPTION

For the model V77-400 minicomputer with the V77-600 compatible clock option, the following DEF directive is required:

```
DEF,V$CKCT0
```

8.4 F3094-xx AND F3096-xx (70-7603, 70-7613) DISKS

When using feature numbers F3094-xx and F3096-xx disks, a special DEF directive must be included for each equipment (EQP) directive used for these disks:

```
DEF,V$DSKx,y
```

where:

x
: Is the controller number (0 through 3).

y
: Is a bit pattern in bits 0 through 7. Bit(n) corresponds to platter(n). The bit is set if the corresponding platter is part of a dual platter driver.

For example:

a) A system contains two F309x-xx controllers with the following drives attached:

   - Controller 0 has one dual unit and three single units;
   - Controller 1 has two dual units, one single unit, and one dual unit.

The corresponding directives are:

```
EQP,D0F,016,5,020,5
DEF,V$DSK0,3
EQP,D1F,017,7,022,5
DEF,V$DSK1,0157
```

b) A system contains one F3094-xx drive and controller (dual platter):

```
EQP,D0F,016,2,020,5
DEF,V$DSK0,3
```

c) A system contains one F3096-xx drive and controller (singleunit):

```
EQP,D0F,016,1,020,5
DEF,V$DSK0,0
```

8.5 STATOS

In addition to the STATOS EQP directive of the form EQP,LPnm, VORTEX system generation requires the following directive:

```
DEF,V$SWnm,c
```

where:

n
: Is controller number 0, 1, or 2.

m
: Is a STATOS model code D, E, G, H, J, K, L, M, N.

c
: Is a width code, defined as follows:

Without 1-line buffer	With 1-line buffer (31-51 option)
0 = 8-1/2 inches	4 = 8-1/2 inches
1 = 11 inches	5 = 11 inches
2 = 14-7/8 inches	6 = 14-7/8 inches
3 = 22 inches	7 = 22 inches

8.6 TELETYPEWRITER AND CRT -- MODEL CODE B

The model code B teletypewriter and CRT driver (TYB) is an alternate version of model code A teletypewriter and CRT driver.

The TYB version performs special functions required by the VORTEX II Time Sharing Subsystem (TSS) and the console logging program. In addition to providing all the functions of TYA, TYB contain an additional write mode (designated MODE5) and the ability of activating a special TSS task when a control-C character is entered.

To include this driver in the system enter the directive:

```
EQP,TYnB
```

or

```
EQP,CTnB
```

where:

n
: Is the controller number 0-7.

8.7 LINE PRINTER -- MODEL CODE B

Systems using the line printer driver model code B require a DEF directive to define the size in characters of the first print buffer. The format of the directive is:

```
DEF,V$LPcB,x
```

where:

c
: Is the controller number of the line printer.

x
: Is the printer buffer size in characters.

An EQP directive indicating a printer buffer greater than 24 characters is also required. This directive has the format:

```
EQP,LPnB
```

where:

n
Is the controller number of the line printer.

8.8 CPU MODELS V77-200, V77-400 AND V77-800

The model V77-200, V77-400 and V77-800 CPUs require the following directives:

- The V75 parameter must be included with the MRY directive for V77-200, V77-400, and V77-800 CPUs only.
- If 60 cycle power is to be used, the following directives must be included:

```
CLK,100,100,15
DEF,V$CKCT,3
```

- If 50 cycle power is used, the following directives must be included:

```
CLK,100,100,20
DEF,V$CKCT,0
```

Section 9
SYSTEM GENERATION TROUBLESHOOTING

9.1 GENERAL

This section describes some of the techniques that may be used to find the causes of user errors or omission in the system generation process.

9.2 SYSTEM GENERATION DRIVER LOCATION

The start of the physical unit table is at memory location 0604. The physical unit table contains a two word entry for each system peripheral. Bits 0 through 5 of this two word entry contains the device address. The second word of this entry contains the address of the device specification table (DST) for the peripheral. The STATUS routine in the driver for a device is pointed to by the DST address for the device + 7. The STATUS routine usually begins at the first location of the driver.

To locate the driver for a device, find the device address in the physical unit table for the peripheral. In the case of master/slave units, several device addresses for the peripheral will be in the physical unit table. Any one of these device addresses can be used. The DST address + 7 is the address containing the driver address for the peripheral.

9.3 TRAPPING THROUGH THE SYSTEM GENERATION PROCESS

The system generation procedures consist of four overlays:

- Directive Processor
- Nucleus Processor
- Library Processor
- Resident Task Processor

To trap in any of the four overlays, set sense switch 2 on. When the next overlay is loaded, a jump to AID (physical address 076000) is executed.

The AID utility may then be used as described in the "70 Series Utility Programs Programmer Reference" UP-8684 (98A 9952 47x).

To continue execution of the overlay, execute the following AID command:

To continue with:	Execute the command:
Directive Processor	G3010.
Nucleus	G3010.
Library Processor	G3002.
Resident Task Processor	G3002.

To complete execution of the overlay and not trap in the next system generation component, turn sense switch 2 off.

Memory may be dumped during execution of the system generation using the following procedures:

1. After AID is entered, place the starting dump address in register A (R1) and the ending dump address in register B (R2).

2. Execute the AID command:

   G3004.

9.4 TRAPPING THROUGH SYSTEM BOOTSTRAP

Use the following procedure to trap through the system bootstrap and initializer. This procedure is useful for making patches to the nucleus.

1. Load the system bootstrap but do not start execution.

2. Change the bootstrap location containing 0600 to the address of AID.

3. Execute the modified bootstrap using the AID command:

   G1130.

4. At this point, the system initializer has been loaded but has not executed. The initializer loads starting at memory location 0600.

NOTE

Initializers have a name of the form I$Da where "a" is the model code for the system RMD.

5. Load the nucleus but do not activate the clock and priority interrupt modules by setting a trap at the location of the label V$SCV in the appropriate initializer then continuing execution at the start of the initializer by executing the AID command:

Txxxxxx,600

where:

xxxxxx
Is the address of the label V$SCV.

6. The nucleus is now loaded, but VORTEX is not activated. Patches may be made to the nucleus at this time.

7. To activate the VORTEX system, execute the AID command:

Gxxxxxx.

where:

xxxxxx
Is the address of the label V$SCV.

Appendix A
DISK PACK FORMATTING PROGRAMS

A.1 INTRODUCTION

This appendix contains information on the procedure for formatting the various types of disk packs that may be used on a VORTEX system. All disk packs which are to be used for system generation (SYSGEN) must be formatted prior to the start of the SYSGEN procedure. After the system has been generated, refer to the VORTEX Operating System Programmer Reference Manual for formatting new or additional disk packs.

The disk formatting programs described in this appendix may be loaded into the system via the AID or BLD programs. The programs are detailed in the V70 Series Architecture Reference Manual (UP-8634).

A.2 TYPE NO. 2822-xx DISK PACK FORMATTING PROGRAM

Before VORTEX can be prepared on a 2822-xx disk pack or any 2822-xx disks can be used under VORTEX, the disk packs must be formatted. The formatting program forms 120-word sectors, which are grouped 24 per track. The program also examines the disk pack for bad tracks.

The formatting program operates in a stand-alone mode. It may be loaded and executed with either AID or BLD. Execution begins at location 01354_8. Upon execution, the formatting program requests some parameters to be input from the keyboard. The following requests are made. An inappropriate response causes the request to be repeated.

INPUT BTC NUMBER

Type a value and a carriage return. The acceptable values are octal 020, 022, 024, 026 and 070. Refer to the "System Memo" to obtain this value.

INPUT DEVICE ADDRESS

Type a value in the range from octal 014 through 017 followed by a carriage return. Refer to the "System Memo" to obtain this value.

INPUT VARIABLE SECTOR GAP

Type a value and carriage return. Acceptable values are 1, 2, 3, 4, 6, 8, 12, or their equivalent octal representations. This value determines the physical location on the disk pack of sequentially addressable sectors, as such sequential transfers may be accomplished without waiting for a full revolution of the disk unit. Recommended setting is 3. Another setting may be more effective depending upon various application parameters such as number of tasks, frequency of disk transfers, and types of disk transfers.

INPUT UNIT NUMBER

Type unit number followed by a carriage return. Acceptable values are 0 through 3. Up to four units can be connected to a single controller. Refer to the "System Memo" to obtain this value.

In addition, the formatting program performs bad-track analysis and creates and maintains a bad-track table, which is entered on each disk pack at the completion of its formatting. The bad-track table is located on sectors 0 through 2 of the first track. The table is 254 words long, starting at word 64 of sector 0. The first 64 words of sector 0 reserve the necessary space for the PST. The remaining unused words of sector 2 are filled with zeroes. Each disk I/O error will generate a ten-event retry sequence, which upon failure will set the bad-track flag within the track header. The program also sets the corresponding bit in the bad-track table. No alternate tracks are assigned.

If the first track is determined to be bad, the bad-track table may not be placed there. If this is the case, the program prints the error message:

FIRST TRACK BAD

and aborts formatting the current disk pack. The program then returns to the keyboard interrogation routine. After the bad-track table has been written on the disk pack, the formatting program resumes the keyboard interrogation to obtain parameters for formatting the next disk. In this way, more than on disk pack can be formatted in the same session. The formatting program may be terminated at this point when no disk packs (except those with bad first tracks) remain unformatted. If an unsafe condition (SELECT LOCK light on) occurs, reload and execute the program. Formatting disk packs is not necessary before every VORTEX system generation.

A.3 FEATURE NO. F309x-xx DISK PACK FORMATTING PROGRAM

Each F309x-xx disk pack requires formatting before any input or output operation can be performed on it. The formatter forms 120-word sectors which are grouped 48 per track. The program also performs a bad-track analysis.

The formatter (p/n 92A0205-030) operates under the MAINTAIN III executive. For instructions on loading from magnetic tape, cards or paper tape, see the MAINTAIN III Manual (UP-8672). Execution begins at location 500. Some parameters are requested from the keyboard. Inappropriate responses cause the request to be repeated. All inputs are terminated by periods.

INPUT BIC NUMBER

Enter an even value in the range octal 020 through 076.

INPUT DEVICE ADDRESS

Enter a value in the range octal 014 through 017.

INPUT UNIT

Enter a value in the range 0 through 7. This must be the physical unit number of the format:

uup

where:

uu
Is unit number 00 - 03.

p
Is 0 for fixed platter.
Is 1 for removable platter.

NOTE

System disk is always 000 regardless of the platter.

INPUT KNOWN BAD TRACKS

Enter octal track numbers in the range 0 through 0625, separated by commas and terminated by a period. If there are no known bad tracks, input only a period.

In addition, the F309x-xx formatting program performs bad-track analysis and creates and maintains a bad-track table, which is entered on each disk pack at the completion of its formatting. The bad-track table is located on sector 0 of the first track. The table is 26 words long, starting at word 64 of sector 0. The first 64 words of sector 0 reserve the necessary space for the PST. The remaining unused words of sector 0 are filled with zeroes. Each disk I/O error will generate a five event retry sequence which, upon failure, will set the corresponding bit in the bad-track table. No alternate tracks are assigned.

If the first track is determined to be bad, the bad-track table may not be placed there. The program prints the error message:

FIRST TRACK BAD

and aborts formatting the current disk pack. The program returns to the keyboard interrogation routine. After the bad-track table has been written on the disk pack, the formatting program resumes the keyboard interrogation to obtain parameters for formatting the next disk. In this way, more than one disk pack can be formatted in the same session. The formatting program may be terminated at this point when no disk packs (except those with bad first tracks) remain unformatted. Formatting disk packs is not necessary before every VORTEX system generation.

A.4 TYPE NO. 282x-xx DISK PACK FORMATTING PROGRAM

Each 282x-xx disk pack requires formatting before any input or output operation can be performed on it. The disk pack formatting program forms 120 word sectors which are grouped 48 per track. The program also performs a bad track analysis.

The formatter operates under the MAINTAIN III executive. For instructions on loading from magnetic tape, cards, or paper tape, see the MAINTAIN III Manual (UP-8672). Execution begins at location 500. Some parameters are requested from the keyboard. Inappropriate responses cause the request to be repeated. All inputs are terminated by periods.

INPUT BTC NUMBER

Enter an even value in the range octal 020 through 076.

INPUT DEVICE ADDRESS

Enter a value in the range octal 014 through 017.

INPUT UNIT

Enter a value in the range 0 through 3.

INPUT KNOWN BAD TRACKS

Enter octal track numbers in the range 0 through 017667, separated by commas and terminated by a period. If there are no known bad tracks, input just a period.

In addition, the formatting program performs bad track analysis and creates and maintains a bad-track table, which is entered on each disk pack at the completion of its formatting. The bad-track table is located at sector 0 of the first track. The table is 508 words long, starting at word 64 of sector 0. The first 64 words of sector 0 reserve the necessary space for the PST. Each disk I/O error will generate a three event retry sequence, which upon failure will set the corresponding bit in the bad-track table. No alternate tracks are assigned.

To bypass a full bad-track analysis, set SSW1. This option should not be used on new packs which require a full verification.

If the first track is determined to be bad, the bad-track table may not be placed there. The program prints the error message:

FIRST TRACK BAD

and aborts formatting the current disk pack. The program returns to the keyboard interrogation routine. After the bad-track table has been written on the disk pack, the formatting program resumes the keyboard interrogation to obtain parameters for formatting the next disk. In this way, more than one disk pack can be formatted in the same session. The formatting program may be terminated at this point when no disk packs (except those with bad first tracks) remain unformatted. Formatting disk packs is not necessary before every VORTEX system generation.

A.5 TYPE 2825-xx, 2842-xx, 2826-xx, 2843-xx, AND 8433-xx DISK PACK FORMATTING PROGRAM

A standalone disk formatter for disk types 2825-xx, 2842-xx, 2826-xx, 2843-xx, and 8433-xx enables a user to format disk packs

prior to generating a VORTEX 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 disk unit:

```
FORMAT,da,un
```

where:

da
: Is the disk controller device address (even address only).

un
: Is the disk unit number.

Upon completion of the formatting operation, the formatter outputs a prompt. The next unit may be formatted at this time.

NOTE:

The AID Utility is detailed in the V70 Series Architecture Reference Manual (UP-8634).

Appendix B
ERROR MESSAGES

B.1 GENERAL

This appendix contains the error messages associated with the system generation elements detailed in this manual.

B.2 SYSTEM GENERATION

System generation error messages are divided into three categories:

- Record input errors - Errors in input records which are found before processing.
- Memory errors - Compatibility errors between allocated memory and a portion of the VORTEX nucleus.
- System loading and linking errors - Errors that prevent normal loading or linking of system components.

These error message types are detailed in Table B-1.

Table B-1. System Generation Error Messages

Message	Condition	Action	User Response
RECORD INPUT ERRORS - error in input record found before processing			
SG00	Read error (I/O)	System waits for corrective input	1) Check for illegal data under current mode, i.e., binary in ASCII record, non-binary in binary record. 2) Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.
SG01	Syntax error in SYSGEN directive	System waits for corrective input	1) Check spelling, delimiters, and parameters. 2) Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.
SG02	Invalid or missing parameter in SYSGEN directive	System waits for corrective input	1) Check spelling, delimiters, and parameters. 2) Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.
SG03	Syntax error in control record	System waits for corrective input	Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.
SG04	Invalid or missing parameter in control record	System waits for corrective input	1) Check spelling, delimiters, and parameters. 2) Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.
SG05	Binary object checksum error	System waits for corrective input	1) Check the module for an error in the checksum value (word 2). 2) Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.

Table B-1. System Generation Error Messages (Continued)

Message	Condition	Action	User Response
RECORD INPUT ERRORS - error in input record found before processing			
SG06	Binary-object sequence error	System waits for corrective input	1) Check the module for an error in the checksum value (word 2) or sequence number (word 1, bits 0-7). 2) If needed, list binary records using IOUTIL. 3) Correct input record by entering it on SO or indicate the record is ready for rereading by entering C on SO.
SG07	Binary-object record code error	System waits for corrective input	1) Check source for erroneous EOF or END or similar directive.
SG08	Unexpected end of file, end of device, or beginning of device	System waits for corrective input	1) Check the module for an error in the checksum value (word 2) or sequence number (word 1, bits 0-7). 2) Correct input record by entering it on SO or indicate the record is ready for rereading by entering C on SO.
SG09	Improper ordering of load-module-package control records	System waits for corrective input	1) Correct input record by entering it on SO or indicate the record is ready for rereading by entering C on SO.
OUTPUT ERRORS - errors in the attempt to perform I/O on an RMD or listing unit			
SG10	RMD I/O error in directive processor. Pack not formatted correctly	System waits for corrective input	Restart component by entering C on SO. Repositioning is automatic for magnetic tape or disk. For cards, reload card deck.
SG11	RMD I/O error in nucleus processor	System waits for corrective input	Restart component by entering C on SO. Repositioning is automatic for magnetic tape or disk. For cards, reload card deck.
SG12	RMD I/O error during library generation	System waits for corrective input	Restart component by entering C on CO. Repositioning is automatic for magnetic tape or disk. For cards, reload card deck.

Table B-1. System Generation Error Messages (Continued)

Message	Condition	Action	User Response
OUTPUT ERRORS - errors in the attempt to perform I/O on an RMD or listing unit			
SG13	RMD I/O error during resident-task generation	System waits for corrective input	Restart component by entering C on CO. Repositioning is automatic for magnetic tape or disk. For cards, reload card deck.
SG14	First track on RMD bad (unable to write PST/bad-track table)	System waits for corrective input	Restart component by entering C on SO. Repositioning is automatic for magnetic tape or disk. For cards, reload card deck.
SG15	Write error on listing device	System waits for corrective input	Restart component by entering C on SO. Repositioning is automatic for magnetic tape or disk. For cards, reload card deck.
SYSTEM GENERATION PROCESSING ERRORS - errors preventing the correct functioning of system generation			
SG21	Loading error in directive processor	System waits for indicated user response	Restart the component by entering C on SO. (Repositioning is automatic for magnetic tape and disk. For cards, reload the entire deck. SGEN finds the required component.
SG22	Loading error in nucleus processor	System waits for indicated user response	Restart the component by entering C on SO. (Repositioning is automatic for magnetic tape and disk. For cards, reload the entire deck. SGEN finds the required component.
SG23	Loading error in library processor/ resident task configurator	System waits for indicated user response	Restart the component by entering C on SO. (Repositioning is automatic for magnetic tape and disk. For cards, reload the entire deck. SGEN finds the required component.)
SG24	Stacks exceed available memory	System waits for corrective action	1) If MEM request is ok or cannot be increased, decrease foreground common, empty TIDBs, re-entry stack size, peripherals drivers, etc. by restarting the SYSGEN process. 2) Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.

Table B-1. System Generation Error Messages (Continued)

Message	Condition	Action	User Response
SYSTEM GENERATION PROCESSING ERRORS - errors preventing the correct functioning of system generation			
SG25	Incomplete system definition (mising directive) or conflicting ASN directive (duplicate logical unit names). May be caused by: 1) EQP directive refers to a BIC address that is not in the default table and is not defined via a "BICxx" directive. 2) Model F disk does not have a "DEF,V$DSKx" directive describing the platter configuration. 3) A TIDB of a controller table for a Model H disk has been deleted. 4) The CL partition is not on the system disk	System waits for indicated corrective action	1) Check spelling, delimiters, and parameters. 2) Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.
SG26	Disk error (too many sectors allocated, or nonsequential partition assignments)	System waits for indicated user response	1) Check spelling, delimiters, and parameters. 2) Check FRM directive and total number of tracks specified in PRT directive. 3) Correct input record by entering it on SO or indicate that it is ready for rereading by entering C on SO.
SG27	Error while loading	System halts	Check spelling, delimiters, etc. of I/O interrogation directives. Restart SYSGEN.

Table B-1. System Generation Error Messages (Continued)

Message	Condition	Action	User Response
SYSTEM GENERATION PROCESSING ERRORS - errors preventing the correct functioning of system generation			
SG28,xx	Error while loading SGEN component xx = 05 - checksum 06 - sequence 07 - record 21 - other in SGEN1 22 - other in SGEN2 23 - other in SGEN3 24 - other in SGEN4	System waits for indicated corrective action	1) Check the module for the indicated error. 2) Restart component by entering C on SO. (Repositioning is automatic for magnetic tape. For cards, reload the entire deck. SGEN will find the correct component.)
SG29	Last partition listed begins on a bad track	System waits for indicated corrective action	Modify PRT directive(s) so that the bad track is not the first track in the partition.
MEMORY ERRORS - compatibility errors between allocated memory and a position of the VORTEX system			
SG30	Size of nucleus larger than that of defined foreground area	System waits for indicated corrective action	1) If the MEM request is ok or cannot be increased, decrease the foreground common, number of empty TIDBs, reentry stack size, number of peripheral drivers, etc. by restarting SYSGEN. 2) Restart component by entering C on SO. (Repositioning is automatic for magnetic tape. For cards, reload the entire deck. SGEN will find the correct component.)

Table B-1. System Generation Error Messages (Continued)

Message	Condition	Action	User Response
MEMORY ERRORS - compatibility errors between allocated memory and a position of the VORTEX system			
SG31	Load module literal pool overflow	Current load module processing terminated, system continues	1) Check $LIT and $IAP values from the load module map. 2) Correct indicated module for the next SYSGEN or add corrected module with LMGEN after SYSGEN completes.
SG32	Size of load module larger than defined memory area	Current load module processing terminated, system continues	1) Recode task using overlays. 2) Correct indicated module for the next SYSGEN or add corrected module with LMGEN after SYSGEN completes. 3) If MEM request is ok or cannot be increased, decrease foreground common, empty TIDBs, reentry stack size, peripheral drivers, etc. by restarting the SYSGEN Process.
SG33	Invalid definition of common during load module generation	Current load module processing terminated, system continues	1) If MEM request is ok or cannot be increased, decrease foreground common, empty TIDBs, reentry stack size, peripheral drivers, etc. by restarting the SYSGEN process. 2) Correct indicated module for the next SYSGEN or add corrected module with LMGEN after SYSGEN completes.
SG34	Number of overlays input not the same as specified by TID control record	Current load module processing terminated, system continues	1) Check spelling, delimiters, and parameters. 2) Correct indicated module for the next SYSGEN or add corrected module with LMGEN after SYSGEN completes.
SG35	Insufficient space in two largest disk partitions to to contain VNO task object modules	Nucleus processing terminated	1) Restart SYSGEN from the beginning. 2) Increase track allocation for the two largest disk partitions in PRT directives.

Table B-1. System Generation Error Messages (Continued)

Message	Condition	Action	User Response
SYSTEM LOADING AND LINKING ERRORS - errors preventing normal loading and linking of system components			
SG40	Loader code error in library processor	Current load module processing terminated, system continues	1) Check module for a sequence number error (word 1, bits 0-7) or a checksum value error (word 2). 2) Correct indicated module for the next SYSGEN or add corrected module with LMGEN after SYSGEN completes.
SG41	Loading program contains no entry name	Current load module processing terminated, system continues	1) Check module for a sequence number error (word 1, bits 0-7) or a checksum error (word 2). 2) Correct indicated module for the next SYSGEN or add corrected module with LMGEN after SYSGEN completes.
SG42	Unsatisfied external in library processor	Current load module processing terminated, system continues	1) Examine map for missing externals and make necessary program changes. 2) Correct indicated module for the next SYSGEN or add corrected module with LMGEN after SYSGEN completes.
SG43	No execution address found in root segment or overlay	Processing continues. Address defaults to to the first location of the program	Check for an execution label on the END statement of the source. Note that this is a normal diagnostic for FORTRAN overlays.
SG44	Loader code error in nucleus processor (i.e., indirect or literal in foreground task)	System waits for indicated corrective action	1) Check module for a sequence number error (word 1, bits 0-7) or a checksum error (word 2). 2) Restart component by entering C on SO. (Repositioning is automatic for magnetic tape. For cards, reload the entire deck. SGEN will find the correct component.)

Table B-1. System Generation Error Messages (Continued)

Message	Condition	Action	User Response
SYSTEM LOADING AND LINKING ERRORS - errors preventing normal loading and linking of system components			
SG45	Unsatisfied external in nucleus processor	System waits for indicated corrective action	1) Examine map for missing externals and make necessary program changes. 2) Restart component by entering C on SO. (Repositioning is automatic for magnetic tape. For cards, reload the entire deck. SGEN will find the correct component.)
SG46	System peripheral assigned to more than one logical unit class	System waits for indicated corrective action	Restart the component by entering C on SO. (Repositioning is automatic for magnetic tape and disk. For cards, reload the entire deck. SGEN finds the required component.)
SG47	Number of pages specified on EQP, WCS does not agree with option on DEF,V$$WCS	System waits for indicated corrective action	1) Correct appropriate SYSGEN directives as indicated. 2) Restart component by entering C on SO. (Repositioning is automatic for magnetic tape and disk. for cards, reload the entire deck. SGEN finds the required component.)
SG48	Hardware status error from WCS	Restart system generation	Check hardware for indicated problem.
SG50	Task TIDB set for VNO, but no VOL directive for task (A task could have been defined on a VOL directive, but was not.)	Task loaded as non-VNO (This is a notification not a fatal error)	1) Specify task using VOL directive. 2) Remove taskname from TDF directive. 3) Ignore warning message and leave as a non-VNO task.

Appendix C
SYSTEM GENERATION ON A SINGLE TAPE CONFIGURATION

System Generation (SYSGEN) on a system with one tape drive and no card reader can be accomplished by using one tape drive, one tape reel containing the system generation library (SGL) and one tape reel containing the SYSGEN directives. Maintaining SYSGEN directives on tape with an existing VORTEX system eliminates the necessity for punching 026 Hollerith code on an 029 keypunch machine or on a system without a card reader, entering SYSGEN directives from the OC device.

To use the same tape drive for both the SGL and SYSGEN directives, proceed as follows:

1. At I/O interrogation time, specify that the tape drive is to be used for both the DIR device and LIB device.

2. Mount the tape containing the SGL on the tape drive.

3. When the system displays:

   ENTER 40 CHARACTERS OF SYSGEN HEADER

   rewind and dismount the SGL tape.

4. Mount the tape containing the SYSGEN directives and place the tape unit on-line.

5. Enter the system generation header. After this header is entered, the system reads the SYSGEN directives.

6. The SYSGEN routines search for the next part of the SGL. If the SYSGEN directive tape is still mounted, an error message is printed and the SYSGEN procedures are suspended.

7. Dismount the SYSGEN directive tape, remount the SGL tape and place the tape unit on-line.

8. Resume the SYSGEN procedures by entering the character "C" followed by a carriage return on the SO device.

NOTES

1. The SYSGEN directives may be placed on the SGL tape by including the directives immediately after V$SGEN1 and prior to the control record CTL,PART0002.

2. SYSGEN directives placed on tape must be 80 character card image records.