IBM System/3 Model 15
3340 Direct Access Storage Facility
Reference Manual

GC21-5111-1
S3-30

Second Edition (June 1975)

This is a major revision of, and obsoletes, GC21-5111-0. Extensive changes have been made throughout and, therefore, this manual should be read in its entirety. Changes are periodically made to the information herein; before using this publication in connection with the operation of IBM systems, refer to the latest *IBM System/3 Bibliography,* GC20-8080, for the editions that are applicable and current.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A Reader's Comment form is provided at the back of this publication. If the form has been removed, comments can be addressed to IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901.

Preface

This manual contains introductory material on the IBM 3340 Direct Access Storage Facility. It is intended for users of the IBM System/3 Model 15 (System Control Program 5704-SC1) who are interested in the programming aspects of the 3340.

This manual is organized into four chapters. The first chapter provides a general description of the 3340 and the IBM 3348 Data Module and outlines the system requirements. The second chapter describes system support information and furnishes some programming considerations. The third chapter identifies the system service programs. The fourth chapter briefly describes the program products as they apply to the 3340. Information about the 3340 data module format, calculating file size, and 3340 operating instructions is included in the appendixes.

Specific information can be referred to by consulting the table of contents or the index. Coding information and examples are not discussed in this manual.

Prerequisite Knowledge

You should be familiar with the related publications, basic programming concepts, the operation of the components of the System/3 Model 15, and the application and information requirements of your organization.

Related Publications

Additional information about the 3340 Direct Access Storage Facility can be found in the following publications:

- *IBM Reference Manual for 3340 Disk Storage*, GA26-1619
- *IBM 3348 Data Module Handling Procedures*, GA26-1625

Additional information about the Model 15 can be found in the following publications:

- *IBM System/3 Model 15 Introduction*, GC21-5094
- *IBM System/3 Models 8, 10, 12, and 15 Components Reference Manual*, GA21-9236
- *IBM System/3 Model 15 Operator's Guide*, GC21-5075
- *IBM System/3 Model 15 System Control Programming Reference Manual*, GC21-5077
- *IBM System/3 Overlay Linkage Editor Reference Manual*, GC21-7561
- *IBM System/3 Model 15 System Generation Reference Manual*, GC21-7616
- *IBM System/3 Model 15 System Messages*, GC21-5076
- *IBM System/3 Model 15 System Control Programming Macros Reference Manual*, GC21-7608
- *IBM System/3 Disk Concepts and Planning Guide*, GC21-7571
- *IBM System/3 RPG II Reference Manual*, SC21-7504
- *IBM System/3 Subset American National Standard COBOL Reference Manual*, SC28-6452
- *IBM System/3 FORTRAN IV Reference Manual*, SC28-6874
- *IBM System/3 Basic Assembler Reference Manual*, SC21-7509
- *IBM System/3 Disk Sort Reference Manual*, SC21-7522
- *IBM System/3 Communications Control Program Programmer's Reference Manual*, GC21-7579
- *IBM System/3 Model 15 Communications Control Program System Reference Manual*, GC21-7620
- *IBM System/3 Communications Control Program Terminal Operator's Guide*, GC21-7580
- *IBM System/3 Model 15 Communications Control Program System Operator's Guide*, GC21-7619
- *IBM System/3 Model 15 Users Guide to Spooling*, GC21-7632

For a complete list of System/3 publications, see *IBM System/3 Bibliography*, GC20-8080.

Contents

Chapter 1. Introduction

The IBM 3340 Direct Access Storage Facility offers increased file capacity and improved performance over both the IBM 5444 and 5445 Disk Storage Drives. Two, three, or four 3340 drives can be online to the IBM System/3 Model 15. The 3340 uses the IBM 3348 Data Module (Model 70) as a storage medium.

SYSTEM REQUIREMENTS

For information concerning the minimum system configuration for the 3340 Direct Access Storage Facility and additional devices supported, see one of the following Model 15 publications:

- *IBM System/3 Model 15 Introduction,* GC21-5094
- *IBM System/3 Model 15 System Generation Reference Manual,* GC21-7616

3340 DISK STORAGE CONCEPT

The 3340 provides direct access storage for the System/3 Model 15. The 3340 is a modular, high-speed, large capacity storage facility for data base/data communications or general-purpose storage.

The 3340 drive uses the 3348 Data Module as the storage medium. The 3340 contains the necessary mechanical and electrical components to house, load, provide air filtration for, and drive the 3348 Data Module. Together, a 3340 and 3348 have:

- Less than 20 seconds start time
- 25 milliseconds average access time
- 10.12 milliseconds average rotational delay

  or

  10.8 milliseconds for planning purposes
- 885,000 bytes per second nominal data transfer rate

Unique to the 3340, when compared to the 5444 and 5445 disk storage drives, are the following basic features:

- A data module instead of a disk pack
- Defect skipping

The data module is a sealed cartridge that contains disks, a spindle, read/write heads, and access arms. The use of the data module provides the following advantages for the System/3 Model 15:

- Drive capacity is approximately 51 megabytes.
- Preventive maintenance of the heads, disks, and spindle is eliminated by reducing the exposure to outside contamination.
- Reliability is improved by dedicated read/write heads. Each head reads only the data it previously wrote.

Defect skipping allows data to be written before and after a surface defect. Thus, all of the track can be used except for that portion containing the defect. This also eliminates the access time that was formerly required to move the read/write heads to an alternate track. Only one defect per half-track can be skipped in this manner; for second and subsequent defects, an alternate track must be assigned.

Byte capacities remain at stated maximums even with known defects. The user need not be concerned with managing defect skipping since the 3340 automatically handles accessing of data on all subsequent read or write operations on that track.

The following 3340 features, available on System/370, are not available on System/3:

- Remote switch attachment
- String switch
- Fixed head feature
- Rotational position sensing

3340 CONFIGURATION

The 3340 configurations for the Model 15 include combinations of the 3340 Models A2 (control and two drives), B1 (one drive), and B2 (two drives), Figure 1-1.

One Model A2 is a prerequisite for each 3340 subsystem. The desired number of drives is obtained by attaching one 3340 Model B1 or one 3340 Model B2 to the 3340 Model A2 drive. Thus the system can have two, three, or four drives attached.

Control

The control unit, part of the 3340 Model A2, performs the following functions:

- Interprets and executes macro-orders from the storage control
- Regulates the storage control and disk storage interfaces
- Checks the integrity of data by error detection and error correction analysis
- Furnishes status to the system
- Performs diagnostic evaluation of the 3340 subsystem

Drive

Each drive of the 3340 performs the following functions:

- Responds to commands from the 3340 Model A2 (control)
- Houses, loads, provides air filtration for, and drives the 3340 data module
- Positions the access mechanism with voice-coil drive and maintains track following via an electronic servo system
- Selects the head
- Reads or writes data
- Provides safety and servicing information for subsystem and system evaluation

TIMING

The total access and data transfer time consists of access motion, head selection (negligible), rotational delay, and data transfer times.

Access Motion Time

Access motion time is the time required to position the access mechanism at the specified cylinder. If the access mechanism is already at the proper cylinder, access motion time is zero. If the access mechanism is moved, the following times are required:

Minimum	– 10 milliseconds
Average	– 25 milliseconds
Maximum	– 50 milliseconds

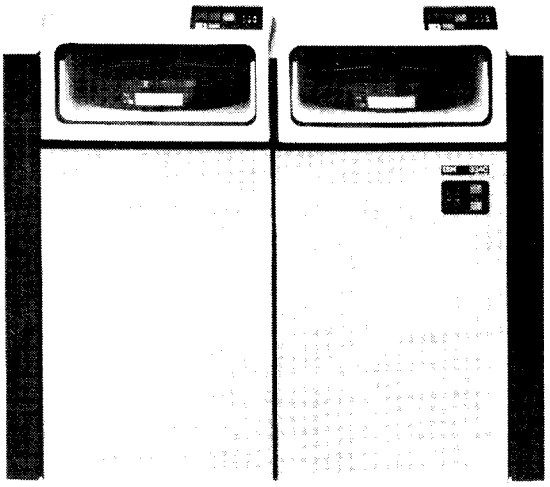
ROTATIONAL DELAY

Rotational delay is the time required for the desired record area to reach the read/write head so that data transfer can begin. The average rotational delay for a 3340 is 10.12 milliseconds. However, due to the way the 3340 is supported by System/3 programming, an average rotational delay of 10.8 milliseconds should be used for planning purposes.

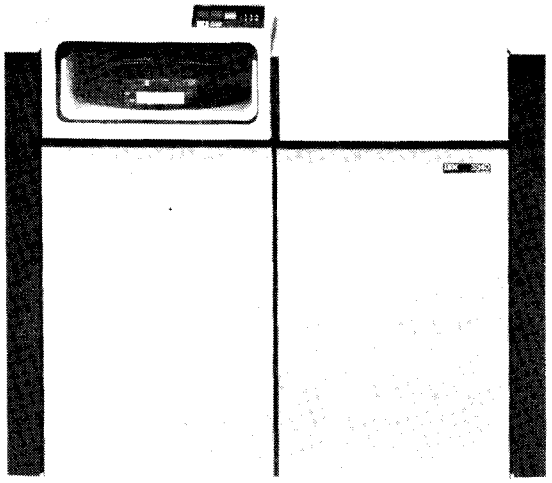
Data Transfer

Nominal data transfer rates for the disk drives are:

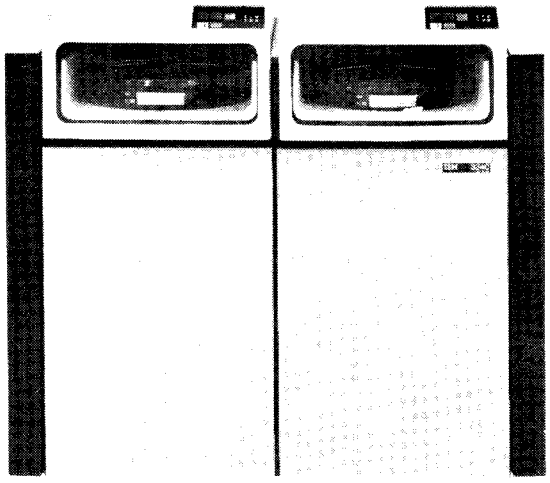
Bytes per second	– 885,000
Microseconds per byte	– 1.13



IBM 3340 Model A2 (Control and Two Drives)



IBM 3340 Model B1 (One Drive)



IBM 3340 Model B2 (Two Drives)

Figure 1-1. 3340 Configuration Models

Comparative Access Times

Figure 1-2 illustrates the access times for the 5444 and 5445 disk storage drives and the 3340.

3340 DATA MODULE

On the System/3 Model 15, the 3340 drives use the IBM 3348 Model 70 Data Module. Due to the formatting required by System/3, the capacity of a data module is 51 megabytes. This chapter describes the Model 70 data modules as used by System/3. The format of the data module is described in Appendix A.

Physical Characteristics

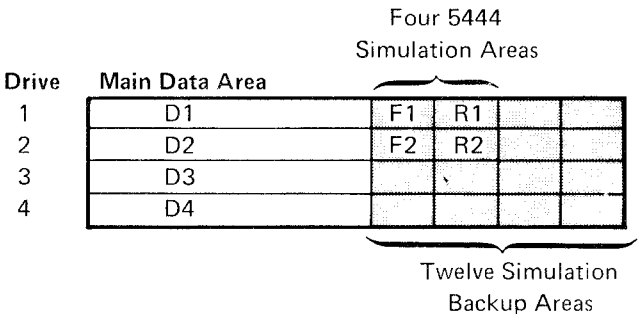
Height — 8 inches
Width — 16 inches
Maximum length — 18 inches
Shipping weight — 23 pounds

Storage Characteristics

Figure 1-3 shows the relative storage characteristics of the 5444 and the 5445 disk storage drives and the 3340 for the System/3 Model 15. The term *record* is used in this figure to indicate a physical record or sector on the disk. System/3 programming supports logical records that span physical records.

Use of the Data Module

Each 3340 data module is divided into a main data area and four simulation areas. Following is a graphic representation of these areas of a 3340 subsystem:



Main Data Area

The main data area of each module provides 40.8 megabytes of data storage. These areas are referred to as follows:

D1 = Main data area on 3340 drive 1
D2 = Main data area on 3340 drive 2
D3 = Main data area on 3340 drive 3
D4 = Main data area on 3340 drive 4

	5444 (High Speed)	5445	3340
Minimum access time	28 ms	25 ms	10 ms
Average access time	126 ms	60 ms	25 ms
Maximum access time	255 ms	130 ms	50 ms
Data transfer rate	199,000 bytes/sec	312,000 bytes/sec	885,000 bytes/sec
Rotational speed	1500 rpm	2400 rpm	2900 rpm (approx.)
Nominal rotation time	40 ms	25 ms	20.2 ms
Average rotational delay	20 ms	12.5 ms	10.1 ms

Figure 1-2. Access Times for the 5444 and 5445 Disk Storage Drives and the 3340

	5444	5445	3340[1]
Bytes per record	256	256	256
Records per track	24	20	48
Bytes per track	6,144	5,120	12,288
Tracks per cylinder	2	20	20
Bytes per cylinder	12,288	102,400	245,760
Cylinders per disk pack/data module[2]	100/200	200	207
Bytes per disk pack/data module[2]	1,228,800/ 2,457,600	20,480,000	50,872,320
Tracks per disk pack/data module[2]	200/400	4,000	4,140
Records per disk pack/data module[2]	4,800/9,600	80,000	198,720
Maximum number of disk files per pack/module	50	1,000	1,000
Maximum number of usable disk surfaces	8	80	24
Maximum number of drives	2[3]	4	4

[1] See *3348 Data Module Surface Format* in Appendix A.
[2] Excluding alternate tracks and CE tracks.
[3] The 5444 contains two packs per drive.

Figure 1-3. Characteristics of Disk Storage Drives for the System/3 Model 15

Simulation Areas

Each 3340 data module used on System/3 contains four 5444 simulation and simulation backup areas. The system uses the 5444 simulation areas in the same way it would use the 5444 disk storage drive. That is, all libraries (user programs, system programs, procedures, etc) are contained in the 5444 simulation areas.

There are four 5444 simulation and simulation backup areas for each drive (a total of 16 for all four drives). Four of these 16 areas are available for general system use:

F1 = Simulation of fixed disk on 5444 drive 1
R1 = Simulation of removable disk on 5444 drive 1
F2 = Simulation of fixed disk on 5444 drive 2
R2 = Simulation of removable disk on 5444 drive 2

The location of these four 5444 simulation areas is fixed. F1 and R1 are always on 3340 drive 1; F2 and R2 are always on 3340 drive 2. The user can prevent (via an operator control command) the simulation of F2 and R2, so that offline multivolume files can be processed on D2.

The remaining 12 simulation backup areas are accessible only by the 3340 simulation area program ($SCOPY).

The 5444 simulation areas are related to a drive rather than to a data module. The module mounted on drive 1 contains the F1 and R1 areas, and the module mounted on drive 2 contains the F2 and R2 areas. If the data module normally placed on drive 2 is placed on drive 1, the areas on drive 1 would be accessed as F1 and R1 rather than as F2 and R2.

Data Module Format

For use with the System/3 Model 15, a 3340 data module has the following format:

Cylinder	Number of Tracks	Millions of Bytes	Description
0	20	0.25	Reserved for system use
1-166	3,320	40.80	Main data area
167-168	40	0.49	Alternate tracks
169-208	800	9.83	Simulation areas
209	8	0.10	Error statistics and CE tracks
	4,188	51.47	Total

The online capacity (Figure 1-4) depends on the number of drives available to the program (a minimum of two drives is attached to the Model 15).

The 5444 simulation areas are accessible by SCP (system control program) and program products; the simulation backup areas are accessible only by the 3340 simulation area program ($SCOPY).

Number of Drives	Number of Bytes				
		Simulation Areas			
	Main Data Area	5444 Simulation Areas	Simulation Backup Areas	Other Use (Reserved)	Total
1	40,796,160	4,915,200	4,915,200	835,584	51,462,144
2	81,592,320	9,830,400	9,830,400	1,671,168	102,924,288
3	122,388,480	9,830,400	19,660,800	2,506,752	154,386,432
4	163,184,640	9,830,400	29,491,200	3,342,336	205,848,576

Figure 1-4. Online Capacity of 3340 Drives

Chapter 2. System Support

From a programming standpoint, the main data area of the 3340 is supported in the same way as for the 5445, and the 5444 simulation areas are supported in the same way as for the 5444. Highlights of the program support for the main data area are discussed in this chapter along with the considerations for using the 5444 simulation areas.

ACCESS METHODS SUPPORTED

For the main data area, sequential, indexed, and direct file organizations are supported in the same way as for the 5445.

For the 5444 simulation areas, sequential, indexed, and direct file organizations are supported in the same way as for the 5444, except that indexed files cannot be processed using the index (there is no support for processing the index). As a conversion aid, an indexed file in the 5444 simulation area can be copied to the main data area using the copy/dump program, thereby creating an indexed file in the main data area.

Files in the main data area can be single volume or multivolume files; files in the 5444 simulation area can be single volume only, or multivolume files processed as single volumes.

Figure 2-1 shows the file organizations and the access methods supported for the 3340 main data area.

Access Methods	File Organization		
	Sequential	Indexed	Direct
Load (output)	X	X	X
Input			
Consecutive	X	X	X
Random by key		X	
Random by relative record number	X	X	X
Sequential by key		X	
Sequential by limits		X	
Add			
Consecutive	X		
Random by key		X	
Sequential by key		X	
Update			
Consecutive	X		X
Random by key		X	
Random by relative record number	X		X
Sequential by key		X	
Sequential by limits		X	

Figure 2-1. File Organization and Access Methods Supported for the 3340 Main Data Area

MULTIVOLUME FILES

A file that exceeds the capacity of one data module can be continued on one or more subsequent data modules. Such files are known as multivolume files. A multivolume file can be online or offline. Online means that all the volumes (data modules) containing the file are running on drives during processing so that all the records are available for processing. Offline means that only part of the file is available for processing at any one time; the volumes must be removed and replaced with other volumes to process the entire file.

Online multivolume files must be multiple drives so that all data is online. D1, D2, D3, and D4 can be used for online multivolume files.

The following restrictions apply to offline multivolume files:

- D1 cannot be used for offline multivolume files because it contains the IPL (initial program load) system; therefore, it must be mounted at all times.
- If all volumes are not mounted on the same drive, they can be mounted in a sequence as specified by the // FILE OCL statement.
- D2 cannot be used for offline multivolume files unless simulation of R2 and F2 has been stopped—via a SIMULATE (SM) command. (For information concerning the SIMULATE command, see *IBM System/3 Model 15 Operators Guide,* GC21-5075.)

Multivolume files in the simulation areas can be accessed only as single volume files.

For more information concerning multivolume files, see *IBM System/3 Disk Concepts and Planning Guide,* GC21-7571.

PROCESSING LARGE INDEXED DISK FILES

When additions are made to a large indexed file, the time needed to sort the keys of the index at the end-of-job step can be excessive. This sort time can be reduced by using a work file.

The work file is used to merge the added keys into the index, and must be large enough to contain all of the keys added to the file. If the job step adds records to more than one indexed file, the work file must be large enough to contain all the keys for the file with the greatest number of additions. If possible, the work file should be located on a different drive from the indexed file whose keys are being sorted. If this is not possible, the work file should be as close as possible to the beginning of the file whose keys are being sorted. This minimizes the disk seek time.

The work file must be named $INDEX45 and be located in the main data area. To determine the number of tracks required for the work file, use the following formula:

$$\text{number of adds} \div \frac{256}{(\text{key length} + 4)} \div 48 = \text{tracks required for work file for the 3340}$$

After dividing 256 by key length +4, the remainder should be dropped. After the other divisions, round the quotient to the next highest whole number.

If the work file is not large enough to contain all the index keys, the keys are sorted in the normal manner without using the work file.

The work file can be used with multivolume files. However, it cannot be located on a pack that contains one of the offline volumes of a multivolume file. The pack containing the work file must remain online while the job is run. The work file must be RETAIN-S. If RETAIN-T or RETAIN-P is specified, the system forces it to RETAIN-S.

For small indexed files of 10 tracks or less where the sort time is negligible, a work file will not improve performance and should not be used.

To use the work file, no change is needed to your source program. Also, programs need not be recompiled to use this option; only one additional OCL FILE statement is needed.

INITIAL PROGRAM LOAD (IPL)

IPL can be performed only from drive 1. Either the F1 simulation area or R1 simulation area can be used for IPL. Therefore, the data module on drive 1 must be online at all times. Since write operations are done during IPL, the data module on drive 1 may never be read only (see *Appendix C, Operating Instructions*).

SPOOLING

Only the main data area can be used for the reader, punch, and print queues.

SYSTEM SERVICE PROGRAMS

System service programs are available for data module and file maintenance on the 3340. These programs are listed in Chapter 3. See *IBM System/3 Model 15 System Control Programming Reference Manual*, GC21-5077, for descriptions and examples of these programs.

PROGRAM PRODUCTS

Program products are used to satisfy specific application requirements. They operate under control of the Model 15 SCP. See Chapter 4 in this manual for a discussion of the available program products.

COMMUNICATIONS CONTROL PROGRAM (CCP)

CCP supports files in the main data area in the same way as for the 5445 and in the 5444 simulation area in the same way as for the 5444. The access methods supported for the 3340 apply to CCP except that no multivolume files are allowed.

OCL FILE STATEMENT

The 3340 is referenced through OCL statements at execution time. When operating in a multiprogramming environment on the Model 15, either or all drives can be addressed by both partitions, but the same file cannot be addressed by both partitions at the same time unless:

- Both partitions are using it as input only.
- One partition is using it as input and the other is using it as update.
- A file is being shared within the CCP partition. (See the *IBM System/3 Communications Control Program Programmer's Reference Manual,* GC21-7579.)

The OCL FILE statement provides information about the files on a data module so that disk system management can read and write records for user programs. Files can reside on the main data area or in 5444 simulation areas. A FILE statement must be provided for each file used by your programs. It must be contained within the set of OCL statements for each program using the 3340. Split cylinder files are not supported on the 3340.

Figure 2-2 summarizes the keywords of the FILE statement. The following sections provide additional information about the keyword parameters.

Keyword	Parameter	Keyword Required or Optional
NAME	Filename	Required
PACK	Name	Required
UNIT	Code	Required
LABEL	Filename	Optional
DATE	Date	Optional
RETAIN	Code	Optional
RECORDS or TRACKS	Number Number	Required for creating files
LOCATION	Cylinder number (main data area only)	Optional
	Cylinder number/ track number (main data area only)	Optional
	Track number (5444 simulation area only)	Optional
HIKEY (main data area only)	Highest allowed key fields	Optional
VERIFY (main data area only)	Code	Optional

Figure 2-2. Description of Parameters on the OCL FILE Statement for the 3340

Keyword Parameters for Single Volume Disk Files

NAME Parameter

The NAME parameter is required for the FILE statement. It informs disk system management of the name that your program uses to refer to the file. The filename can be any combination of characters except commas, apostrophes, or blanks. The first character must be alphabetic. The number of characters must not exceed eight.

PACK Parameter

The PACK parameter is also required for the FILE statement. It informs disk system management of the name of the main data area or 5444 simulation area that contains or will contain the file. The management routines check this name to ensure that it is the same as the name in the volume label of the main data area or the 5444 simulation area being used. This parameter can consist of from one to six characters, excluding the apostrophe, comma, and blank.

UNIT Parameter

The UNIT parameter is the last of the required parameters in the FILE statement. It supplies the location of the main data area or 5444 simulation area that contains the file. The possible codes are F1, R1, F2, R2, D1, D2, D3, and D4.

LABEL Parameter

The LABEL parameter refers to the filename by which the file is identified in the VTOC. This parameter is required only if the filename in a program differs from the filename on the main data area or 5444 simulation area. If a new file is being created and the LABEL parameter is omitted, the filename from the NAME parameter is used.

DATE Parameter

The DATE parameter is required when two or more files having the same name exist on a main data area or 5444 simulation area and a file with a particular date is desired. The creation date of the desired file is coded in the DATE parameter. If two or more files with the same name exist on a main data area or 5444 simulation area and neither the date nor the location is given, the file having the latest creation date is selected. The date must be in the form month-day-year or day-month-year as was specified at system-generation time. The date must be written as a six-digit number with three fields of two digits without punctuation, or three fields of one or two digits with the fields separated by punctuation. Any characters except numbers, apostrophes, commas, or blanks can be used as punctuation.

RETAIN Parameter

The optional RETAIN parameter indicates the classification of the file when it is created. The classifications are:

Code	Meaning
S	Scratch file. A scratch file is intended for use only by the current program and does not exist after the completion of the current program. S is also used to remove a temporary file so that its space will be available to subsequent programs.
T	Temporary file. A temporary file is one that has short term usefulness and can be over-written when this usefulness has ended.
P	Permanent file. A permanent file is one that is expected to be maintained permanently on the data module.

The file is assumed to be temporary if the RETAIN parameter is omitted at file-creation time.

RECORDS or TRACKS Parameter

Either the RECORDS or TRACKS parameter, but not both, can appear in the FILE statement. One of these is required for files being created and indicates the amount of space necessary for the file. If the file is being referenced, these parameters inform disk system management of the amount of space that was used for the file when it was created. The space requirement is specified as the number of records in the file (RECORDS) or as the number of tracks (TRACKS). When more than one file on the same main data area or 5444 simulation area has the same filename, this keyword parameter can be used to identify the desired file. Two restrictions are applicable when defining the space requirement:

- If RECORDS is used, the number can be up to six digits long and must be within the range of 1 through 999999.
- If TRACKS is used, the number can be up to four digits long and must be within the range of 1 through 3320 when the file is in the main data area or 1 through 398 when the file is in a simulation area.

LOCATION Parameter

For the main data area, the optional LOCATION parameter is used to specify the cylinder and track on which the file is to start; for a simulation area, this parameter is used to specify the track on which the file is to start. You can specify either the cylinder number or the cylinder number and the track number for the main data area. If the track number is omitted, it is assumed to be zero. For the main data area, the cylinder number must be from 1 through 166 and the track number from 0 through 19. The cylinder number and track number, when specified together, must be separated by a slash (ccc/tt). For a simulation area, the track number must be from 8 through 405.

When you are accessing an existing file, the LOCATION keyword parameter must be identical to that used in creating the file. When you are creating a file, this parameter specifies the beginning position of the file.

When two or more files on the same main data area or 5444 simulation area have the same filename, this keyword parameter can be used to identify the desired file.

VERIFY Parameter

The VERIFY parameter is used to specify verification of 3340 write operations for this file in this step. If VERIFY-YES is specified, verification takes place. If VERIFY-NO is specified, write verification is bypassed. If VERIFY is not specified, VERIFY-YES is assumed, unless RETAIN-S is coded, in which case VERIFY-NO is assumed. Verification is always done when a simulation area is accessed.

Keyword Parameters for Multivolume Files

For online multivolume files, the keyword parameters that require lists are PACK, UNIT, TRACKS, RECORDS, LOCATION, and HIKEY. These parameters require lists to describe all of the data modules containing the file. For offline multivolume files, lists are also used, but UNIT does not require a list since all the volumes can be mounted on the same drive (D2, D3, or D4).

Certain rules must be followed when indicating the lists for these parameters:

- The lists must be enclosed in quotes.
- The items in the list must be separated by commas.
- The lists, except for HIKEY, must not contain blanks.

The functions of the keyword parameters have been explained (except for HIKEY which is explained here); therefore, only the considerations for using the lists in these parameters are explained here.

PACK Parameter

The list for this parameter contains the names of the volumes in the order they are to be used.

UNIT Parameter

If the number of units specified for this parameter is less than the number of volumes specified for the PACK parameter, the file is processed as an offline multivolume file.

For online multivolume files, the unit codes must be specified in the sequence that the volumes are used (specified by the PACK parameter). Up to four units can be specified (D1, D2, D3, and D4).

For offline multivolume files, the sequence of unit codes is repeated until all volumes (specified by the PACK parameter) are processed. For example:

```
PACK-'VOL1,VOL2,VOL3,VOL4,VOL5,VOL6'
UNIT-'D3,D4'
```

Volumes VOL1, VOL3, and VOL5 are mounted on unit D3 and volumes VOL2, VOL4, and VOL6 are mounted on unit D4.

Unit D1 cannot be used for offline multivolume files. Unit D2 can be used for offline multivolume files when simulation of R2 and F2 is disabled (via a SIMULATE command).

TRACKS or RECORDS Parameter

The list for these parameters indicates the amount of space occupied by the multivolume file. The numbers in the list must correspond to the order of the names listed in the PACK parameter.

LOCATION Parameter

The list for this parameter contains the cylinder number or the cylinder number/track number parameter for the data modules you use for the file. The parameters must correspond to the order of the names in the PACK parameter. If LOCATION is specified for one volume of a multivolume file, it must be specified for all the volumes of that file.

HIKEY Parameter

The HIKEY parameter is used only for multivolume indexed files. HIKEY limits the highest key field that can be put on each data module of a multivolume file. For example, in the following HIKEY parameter, three volumes are used: HIKEY-'JONES,NICOL,ZZZZZ'. The highest key field allowed on the first volumes is JONES. This means that all the records up to and including JONES will be on this volume. Since HIKEY parameters must be in ascending order, the next volume contains all of the records with keys following JONES and including NICOL. The last volume contains all the records with names that come after NICOL and through ZZZZZ.

Chapter 3. System Service Programs

The Model 15 SCP (5704-SC1) includes a group of system service programs that are resident in a 5444 simulation area. These programs perform a variety of functions, from preparing data modules for use to reorganizing an indexed file to deleting files. The system service programs are described in *IBM System/3 Model 15 System Control Programming Reference Manual*, GC21-5077.

The following system service programs are available:

Program	Name
Alternate track assignment	$ALT
Alternate track rebuild	$BUILD
Copy/dump	$COPY
Dump/restore	$DCOPY
File delete	$DELET
File compress	$FCOMP
System history area display	$HIST
Disk initialization	$INIT
File and volume label display	$LABEL
Library maintenance	$MAINT
Recover index	$RINDX
Reassign alternate track	$RSALT
3340 simulation area	$SCOPY
1000-file VTOC conversion	$WVTOC

This page is intentionally blank.

DISK INITIALIZATION PROGRAM ($INIT)

All data modules used by programs processed under disk system management must be initialized.

This system service program performs the following functions:

- Analyzes the main data area and the 5444 simulation areas to determine if the data module is usable and whether it contains any defective tracks.
- Assigns alternate tracks for defective tracks when the data module is usable.
- Initializes a 1000-file VTOC for the main data area.
- Copies IPL information from cylinder 0 on the system data module mounted on D1 to the data module being initialized.

Note: The 5444 simulation and backup areas on a data module that have been initialized for the first time or initialized using TYPE-FORCE are unusable until the 3340 simulation area program ($SCOPY) CLEAR function has been run to assign a volume label and build a 5444 cylinder 0.

The control statements required for the disk initialization program are shown in Figure 3-2. The numbered list corresponds to the numbered items in the figure.

```
                 F1
// LOAD $INIT, { R1 }
                 F2
                 R2

// RUN                          RENAME
                                CYL0
       1                   2    PRIMARY                                            8
// UIN UNIT- { code   }  ,TYPE- { CLEAR  }   [,VERIFY-number]   [,ERASE- { YES }  ]
             { 'codes' }        FORCE                                     NO
     3    4          5                  6               7
// VOL PACK-name[,ID-characters] [,OLDPACK-name] [,NAME360-name]

// END
```

Figure 3-2 (Part 1 of 3). Control Statements Required for the Disk Initialization Program

Item	Explanation
1	The UNIT keyword parameter specifies the location of the data module you want to initialize. Any combination of a maximum of four data modules can be specified. The codes indicate the locations of the data modules: **Code** **Meaning** D1 Data module on drive 1 D2 Data module on drive 2 D3 Data module on drive 3 D4 Data module on drive 4
2	The TYPE keyword parameter specifies the type of initialization performed. The optional parameters are: • PRIMARY: The data module will be initialized only if there are no active files on the data module. Only the main data area is initialized. • CLEAR: The data module is initialized without checking for active files. Only the main data area is initialized. • FORCE: Both the main data area and the simulation areas are initialized without checking for active files. (This is invalid for D1; and for D2 if F2 and R2 are being simulated.) • CYL0: This is a fast initialization, which initializes only cylinder 0 on a previously initialized pack. • RENAME: Only the areas of the pack affected by the PACK, ID, and NAME360 parameters are initialized on a previously initialized pack. If either ID or NAME360 is omitted, the default values are used for updating. The remainder of the pack is unchanged. 1. For TYPE-CLEAR and TYPE-PRIMARY, the 5444 simulation and backup areas are initialized only if the module was not previously initialized on a System/3. Otherwise, TYPE-CLEAR and TYPE-PRIMARY initialize only the main data areas, but both the main data area and the simulation areas are checked for defective tracks and alternates are assigned if necessary. If the unit is D1, suspected defective tracks located in the 5444 simulation areas are not tested. Instead, a halt is issued indicating that the data module should be moved to another drive and $ALT run against it. If the unit is D2, a halt is issued when either F2 or R2 is being used by the other partition. 2. Individual 5444 simulation and backup areas are cleared with the 3340 simulation area program ($SCOPY).

Figure 3-2 (Part 2 of 3). Control Statements Required for the Disk Initialization Program

Item	Explanation
3	One VOL statement must be supplied for each data module specified in the UNIT parameter of the UIN statement.
4	The PACK keyword parameter entry on the VOL statement provides a data module name of up to six characters which is placed in the volume label.
5	The optional ID entry further identifies the data module. It is for your use as additional identification of the data module. The identification consists of up to 10 characters that, if specified, are placed in the volume label. If this parameter is omitted, it defaults to blanks.
6	The OLDPACK parameter specifies the name of the data module to be initialized.
7	The NAME360 parameter is an additional parameter that further identifies the data module. The parameter can be up to 44 characters long and, if specified, will be placed in the System/360 format 1 DSCB. If this parameter is omitted, it defaults to SYSTEM/3 DATA.
8	ERASE-YES, when used with TYPE-FORCE, flags and retests the tracks that are suspected as being defective, then flags them as usable if the retest results are satisfactory.

Figure 3-3 (Part 3 of 3). Control Statements Required for the Disk Initialization Program

ALTERNATE TRACK ASSIGNMENT PROGRAM ($ALT)

An alternate track is a substitute track selected to accept the contents of a defective track. Forty alternate tracks are available on the data modules for the 3340. An alternate track can be assigned to any track except cylinder 0, head 0 by this program. Anytime a program attempts to use a track that has been assigned an alternate, the alternate will be automatically used instead.

When programs encounter permanent reading or writing errors, the system automatically halts the current operation. You can then run the alternate track assignment program to test suspected defective tracks and assign alternates as needed. Some of the data might not be recovered when the alternate track assignment program is used. If the data cannot be recovered without an error condition, the record or records are read under reduced hardware checking conditions and are written on the alternate track. The record or records that contain the error are printed in a form that completely identifies the data written on the alternate tracks. This printout should be retained. It is required for running the alternate track rebuild program.

The control statements required for the alternate track assignment program are shown in Figure 3-3. The numbered list corresponds to the numbered items in the figure.

ALTERNATE TRACK REBUILD PROGRAM ($BUILD)

The alternate track rebuild program allows you to correct data that could not be transferred correctly to an alternate track. The information received from the alternate track assignment program gives the name of the data module and the number of the track and records suspected of containing incorrect data and is used for writing control statements for this program. Up to 40 alternate tracks can be corrected during one program run. When correcting alternate tracks, be sure that the defective track has been assigned an alternate before modifying the data on that track. The program reads each set of statements and moves the corrected data into its proper location within a record. Remember, these corrections are actually made on an alternate track.

The control statements required for this system service program to build alternate tracks are shown in Figure 3-4. The numbered list corresponds to the numbered items in the figure.

```
// LOAD $ALT, { F1 }
              { R1 }
              { F2 }
              { R2 }

// RUN

 1  2          3   { D1 }    4
// ALT PACK-name,UNIT- { D2 } [,VERIFY-number]
                       { D3 }
                       { D4 }

// END
```

Item	Explanation
1	An ALT statement is required for each data module processed during the current run.
2	The PACK parameter identifies the data module containing the defective tracks. The name used is the name assigned by the disk initialization program.
3	The UNIT parameter indicates the location of the data module containing the defective track. The possible codes are D1, D2, D3, and D4. The 5444 simulation areas (F1, R1, F2, R2) cannot be specified for the UNIT parameter. The 5444 simulation and backup areas are processed when D1, D2, D3, or D4 is processed. When D1 is specified, halt UA31SD is issued and the data in the 5444 simulation area is not processed. When D2 is specified, halt UA F[illegible] is issued if either F2 or R2 is being used by the other partition.
4	The VERIFY parameter, an optional parameter, specifies the number of times surface analysis is to be performed on a suspected defective track. The number specified can be from 1 to 255. The default value is 16. If an error occurs, an alternate track is assigned.

Figure 3-3. Control Statements Required for the Alternate Track Assignment Program

```
// LOAD $BUILD, {F1 / R1 / F2 / R2}

// RUN

// REBUILD PACK-name,UNIT-{D1 / D2 / D3 / D4},TRACK-location,LENGTH-number,DISP-position,data statements

// END
```

Item	Explanation
1	The REBUILD statement specifies: • The name and location of the data module being corrected • The original track and record address where the correction is to be made • The location and number of characters to be replaced in the record This control statement is required for each record being changed and must be followed by one or more data statements.
2	The PACK parameter indicates the name of the data module that contains the alternate track being modified. This is the name assigned to the data module by the disk initialization program.
3	The UNIT parameter indicates the drive on which the data module, specified by the PACK parameter, will be mounted. The possible codes are D1, D2, D3, or D4. The 5444 simulation area codes (F1, R1, F2, R2) cannot be specified for the UNIT parameter unless the PATCH statement is used. The 5444 simulation and backup areas are processed when D1, D2, D3, or D4 is processed.
4	The TRACK parameter refers to the defective track and record that contain the data being corrected. Six characters refer to this location; the first four characters identify the track number, and the last two characters identify the record number.
5	The LENGTH parameter indicates the number of characters that must be modified in the specified record. The length specified must be in multiples of 2 and can range from 2 to 256, the maximum capacity of a record. The length applies to characters that occupy consecutive positions in the record. If the characters occupy nonconsecutive positions, either replace all intervening characters or use more than one REBUILD statement.
6	The DISP parameter indicates the position in the record of the first byte of data being replaced. Starting at the position indicated, the program replaces the number of characters specified in the LENGTH parameter. The position of the first character is 1, the position of the second is 2; and so forth. The maximum DISP specification is 255. When the modified record uses a key, the key is considered part of the data area in calculating the displacement.
7	The data statements contain the information to be placed on the specified record. Replacement characters are entered on the data statements in hex form. The records are updated, two bytes at a time, by the pairs of hex characters. Pairs of characters on the data statements must be separated by commas. When more than one statement is used, the last entry on all but the last data statement must be a comma. The first blank entry indicates the last entry for all data statements.

Figure 3-4. Control Statements Required for the Alternate Track Rebuild Program

FILE AND VOLUME LABEL DISPLAY PROGRAM

The file and volume label display program is used to display any or all of the file labels on the main data area or any of the four 5444 simulation areas. This program performs the following functions:

- Displays the VTOC information for any specified filename on a given main data area or 5444 simulation area.
- Displays the VTOC information for all files in the VTOC.
- Displays the status of the main data area or 5444 simulation area on any specified unit, including the number of available alternate tracks and size and location of available space in the specified area.

The control statements required for this system service program are shown in Figure 3-5. The numbered list corresponds to the numbered items in the figure.

```
// LOAD $LABEL, { F1 | R1 | F2 | R2 }

// RUN

// DISPLAY 1 UNIT- { F1 | R1 | F2 | R2 | D1 | D2 | D3 | D4 }  2 ,LABEL- { VTOC | filename | 'filename, . . .' }  3 [,SORT-NAME]

// END
```

Item	Explanation
1	The UNIT parameter specifies the area containing the volume information to be printed.
2	The LABEL parameter indicates the information to be displayed from the specified area:

Code	Meaning
VTOC	All volume information is displayed.
filename	Only the VTOC information for the specified file is displayed.
'filename, . . .'	Only the VTOC information for the specified files is displayed. A maximum of 20 files can be displayed for one program run.

Item	Explanation
3	The SORT parameter can be specified only when LABEL-VTOC is specified. If SORT-NAME is specified, the volume information is sorted by filename into alphabetical order. This function applies only to 1000-file VTOCs and requires additional main storage for sorting.

Figure 3-5. Control Statements Required for the File and Volume Label Display Program

For the main data area, this program requires a partition size of 10K through 18K, depending on whether the SORT parameter is specified and the number of files in the VTOC. The number of VTOC entries and the storage required for each are as follows:

Number of VTOC Entries	Storage Required
300	10K
500	12K
700	14K
900	16K
1000	18K

For a 5444 simulation area display, only 8K is required and the SORT parameter cannot be used.

FILE DELETE PROGRAM

The file delete program frees space on data modules, making space available for other files. This program frees space for a single file, multiple files, or for all files on a data module in the main data area or a 5444 simulation area by removing files from the System/3 VTOC. The contents of the file are then unrecoverable.

The control statements required for this program are shown in Figure 3-6. The numbered list corresponds to the numbered items in the figure.

```
// LOAD $DELET, { F1 | R1 | F2 | R2 }

// RUN

 1    2          3
// REMOVE PACK-name,UNIT- { F1 | R1 | F2 | R2 | D1 | D2 | D3 | D4 }
    4                                  5                6
  ,LABEL- { VTOC | filename | 'filename, . . .' } [,DATE-date] [,DATA- { YES | NO } ]

 7
// FORMAT UNIT- { F1 | R1 | F2 | R2 | D1 | D2 | D3 | D4 } ,PACK-name

// END
```

Figure 3-6 (Part 1 of 2). Control Statements Required for the File Delete Program

Item	Explanation
1	The REMOVE statement removes the file entry from the VTOC and makes the space occupied by the file available.
2	The PACK parameter indicates the name of the area that contains the file being deleted. For the main data area, this is the name assigned to the data module by the disk initialization program. For the 5444 simulation areas, it is the name assigned by the 3340 simulation area program ($SCOPY).
3	The UNIT parameter indicates the area, specified in the PACK parameter, containing the files to be deleted.
4	The LABEL parameter identifies the file to be deleted. **Code** — **Meaning** VTOC — All files in the main data area or 5444 simulation area are to be deleted. filename — Only the specified file is to be deleted. 'filename, . . .' — Only the specified files are to be deleted.
5	The DATE parameter identifies a particular file to be deleted when more than one file of the same name is specified on the LABEL parameter. If the DATE parameter is omitted and there are several files with the same name, all of the files are deleted.
6	The DATA parameter indicates if the data is to be removed from a file. If the YES option is selected, the data is removed from the file. If the NO option is selected or if the parameter is omitted, the data is not removed from the file. DATA-YES should be used only if file security is required. The time needed to remove the data is much greater than the time needed to remove only the VTOC entry.
7	The FORMAT statement is used to free all allocated space which does not contain files, libraries, or system areas. This statement is used when you suspect that a system failure or an inadvertent re-IPL might have left space allocated, but not actually being used, on the data module. The UNIT parameter indicates the area containing the allocated space to be freed. The PACK parameter indicates the name of the area containing the allocated space to be freed.

Figure 3-6 (Part 2 of 2). Control Statements Required for the File Delete Program

COPY/DUMP PROGRAM

The copy/dump program performs the following 3340 related functions: copypack and copyfile.

Copypack

The copypack function:

- Copies an entire main data area to another main data area.
- Copies an entire 5444 simulation area to another 5444 simulation area. Copying an entire 5444 simulation or backup area to another 5444 simulation or backup area can be accomplished using the 3340 simulation area program ($SCOPY).

Copyfile

The copyfile function:

- Copies a file from a 5444 simulation area to another 5444 simulation area. Indexed files cannot be copied.
- Copies a file from a main data area to a 5444 simulation area. Indexed files cannot be copied.
- Copies a file from a 5444 simulation area to a main data area. Indexed files must be copied consecutively using REORG-NO. A new index is built in the main data area.

  Note: Not specifying REORG causes $COPY to default to REORG-NO.
- Copies a file from a main data area to another main data area. Sequential files can be copied consecutively and an index created, thus effectively copying a sequential file to an indexed file.
- Copies a file to or from magnetic tape, cards, or diskette. Either sequential or indexed files can be created in the main data area; only sequential files can be copied to the 5444 simulation area.
- Prints all or part of a file whether it is being copied to another file or not.
- Prints and/or copies selected records from a file based on either the relative record number or a key value.
- Recovers a destroyed file by physical address.
- Request direct output from sequential input.
- Specify a different record length for the output file.

Multivolume files are not supported for the 5444 simulation areas. Multivolume files can be processed one volume at a time (as single-volume files).

The control statements for this system service program are shown in Figure 3-7. The numbered list corresponds to the numbered items in the figure.

```
                      F1
// LOAD $COPY,      { R1 }
                      F2
                      R2

  [1]
// FILE NAME-COPYIN .......

// FILE NAME-COPYO .......

// FILE NAME-COPYP .......

// RUN

                     F1          F1
                     R1          R1
                     F2          F2
   [2]               R2          R2     [3]                 [4]
// COPYPACK FROM-  ( D1 )  ,TO- ( D1 )  [,PACKIN-packname] [,PACKO-packname]
                     D2          D2
                     D3          D3
                     D4          D4

                [6]              DISK      [7]
  [5]                            PRINT                                                    [8]                [13]
// COPYFILE  { OUTPUT } -      { BOTH }  [, { DELETE } -'position,character'] [,REORG- { YES } ] [,LENGTH-number]
               OUTPTX            FILE          OMIT                                      NO

                 [15]   F1     [16]             [17]            [18]          [19]
                FROM- { R1 }   ,CYLINDER-number,SECTOR-number,DISP-number,RECL-number
   [14]                 F2
// ACCESS    (          R2                                                               )
                        D1                       [20]
                FROM- { D2 }   ,CYLINDER-number,TRACK-number,SECTOR-number,RECL-number
                        D3
                        D4

                    [10]
  [9]           { RECORD,FROM-number[,TO-number]  }    [11]
// SELECT       { KEY,FROM-'key'[,TO-'key']       }    [,FILE-  { YES }  ]
                { PKY,FROM-'key'[,TO-'key']       }             { NO  }
  [12]
// KEY LENGTH-number,LOCATION-number
[21] // OUTDM    DATAMGMT-DIRECT
// END
```

Figure 3-7 (Part 1 of 3). Control Statements for the Copy/Dump Program

This page intentionally left blank.

Item	Explanation
1	These three FILE statements identify the input and output files. The FILE statement COPYIN is required for all COPYFILE jobs except access runs. The FILE statement COPYO is only required if the output option on the COPYFILE statement is DISK, FILE, or BOTH. The FILE statement COPYP is required only if printing on the 3284 printer. No FILE statements are required for a COPYPACK run.
2	The COPYPACK statement indicates that the content of one area is to be transferred to another. The FROM parameter identifies the location of the area being copied. The TO parameter identifies the location of the area receiving the data. COPYPACK can only be used to copy from one 5444 simulation area to another or from one main data area to another. If the FROM location contains an active $SPOOL file, the location is copied, but the $SPOOL file is deleted before end-of-job.
3	The optional PACKIN parameter identifies the area to be copied. This parameter is provided to check the area name to verify that the correct area is selected.
4	The optional PACKO parameter identifies the output area. This parameter is provided to check the area name to verify that the correct area is selected.
5	The COPYFILE statement indicates that one file is to be copied. Its parameters specify the type of copy performed.
6	The OUTPUT and OUTPTX parameters indicate that the file is to be copied, printed, or both. When the data files are being copied, the from and to files are specified as COPYIN and COPYO, respectively. If PRINT is selected, the OUTPTX parameter causes the output to be printed in hex values.
7	The optional DELETE and OMIT parameters identify records for deletion from the file by giving the identifying character and its position in the record. Position (p) is the position of the character in the records. The position specified must be less than the record length. Character (c) is the character that identifies the records. The DELETE parameter cannot be used for direct files. If the DELETE parameter is used, the deleted records are printed. If the OMIT parameter is used, the deleted records are not printed. If neither parameter is used, all of the records in the input file are copied.
8	The optional REORG parameter permits reorganization of the file as it is being copied. The YES option applies only to indexed files. If the YES option is specified, the output file is reorganized putting the records in ascending order by their key fields. If the NO option is specified or if the entry is not included, either the file is not an indexed file or the records in the output file have the same sequence as in the input file. In either case, the input file is not affected.
9	The optional SELECT statement allows selected records to be printed and/or copied to an output file. If the SELECT statement is not specified and the OUTPUT parameter on the COPYFILE statement is PRINT or BOTH, all the records are selected.
10	The RECORD parameters use relative record numbers to identify the records to be selected. Relative record numbers identify a record's location in the file with respect to other records in the file. The FROM parameter specifies the first record to be selected. The TO parameter indicates the last record to be selected. The program assumes that the number of the last record in the file is the TO number if the TO parameter is omitted. The KEY and PKY parameters apply to selecting part of an indexed file. The FROM parameter gives the key of the first record to be selected. The TO parameter gives the key of the last record to be selected. The program assumes that the last key in the index is the TO key if the TO parameter is omitted. The records selected are those which have keys between or equal to the FROM and TO limits. The PKY parameter applies to an indexed file containing packed keys.

Figure 3-7 (Part 2 of 3). Control Statements for the Copy/Dump Program

Item	Explanation
11	The optional FILE parameter allows only selected records to be copied. If FILE-YES is specified, selected records are copied to the file named in the COPYO FILE statement. If FILE-NO is specified, OUTPUT-BOTH or OUTPUT-PRINT must be specified. If OUTPUT-BOTH is specified, selected records are printed and the entire file is copied to the file named in the COPYO FILE statement. If OUTPUT-PRINT is specified, selected records are printed only.
12	The optional KEY statement is used to create an indexed file from a sequential file. LENGTH and LOCATION specify the key length and start location of the key in the record.
13	The optional LENGTH parameter specifies the length of the output records. If LENGTH parameter is not used, the output record length will be the same as the input record length.
14	The ACCESS statement indicates that data on a disk pack which has no VTOC entry is to be copied and the location of this data is specified by physical address.[1]
15	The FROM parameter indicates the drive on which the data is located.
16	The CYLINDER parameter indicates the starting cylinder number of the data. For simulation areas, this number may be 0-202. For main data areas, the cylinder number may be 0-167.
17	The SECTOR parameter indicates the starting sector number of the data. For simulation areas, this number may be 0-47. For the main data areas, the sector number may be 1-48.
18	The DISP parameter indicates the displacement into the sector of the first record to be copied (recovered). It may be a number between 0-255.
19	The RECL parameter indicates the length of the data records in bytes.
20	The TRACK parameter indicates the starting track number of the data. It is used only with recovery from main data areas and may be 0-19.
21	The OUTDM statement indicates that direct output is requested from sequential input.

[1] Select record with FILE-YES must be used with all access runs.

Figure 3-7 (Part 3 of 3). Control Statements for the Copy/Dump Program

DUMP/RESTORE PROGRAM ($DCOPY)

The $DCOPY program is used to dump or restore a 3340 data module to or from magnetic tape or diskette for backup.

If BACKUP-3741 is specified in the control statement, the simulation area or the IPL records and a simulation area are dumped or restored from the 3741. If no BACKUP parameter is specified, the backup is to tape. Backup to 3741 takes approximately 11 full diskettes. When operating the 3741, specify a record length of 128.

If the program is loaded and executed, it will perform one of the following:

- Copy the main data area of a 3340 data module onto magnetic tape. (Note that if the area to be copied contains an active $SPOOL file, the area is copied, but the $SPOOL file is deleted before end-of-job.)
- Copy a 5444 simulation area of a 3340 data module onto magnetic tape or diskette
- Copy the 3340 IPL area and its associated 5444 simulation area of a 3340 data module onto magnetic tape or diskette
- Restore the main data area of a 3340 data module from magnetic tape
- Restore a 5444 simulation area of a 3340 data module from magnetic tape or diskette
- Restore the 3340 IPL area and its associated 5444 simulation area of a 3340 data module from magnetic tape or diskette

The control statements and parameters required for this system service program to function are shown in Figure 3-8. The numbered list corresponds to the numbered items in the figure.

```
// LOAD $DCOPY, { F1 | R1 | F2 | R2 }

 1
// FILE NAME-BACKUP,UNIT- { T1 | T2 | T3 | T4 } ,REEL { NL | nnnnnn }

// RUN

 2                                                3
// COPYPACK { TO | FROM } { F1 | R1 | F2 | R2 | D1 | D2 | D3 | D4 } [,PACK-name] [, SYSTEM - { YES | NO } ] [,BACKUP- { TAPE | 3741 } ]

// END
```

Item	Explanation
1	This FILE statement identifies the tape file used as a backup copy. The FILE statement is not necessary if BACKUP-3741 is specified. (For a more detailed description of the file, see the *IBM System/3 Model 15 System Control Programming Reference Manual,* GC21-5077.)
2	The COPYPACK statement indicates that the information from one main data area, or the 3340 IPL area, and/or the 5444 simulation area is to be transferred to tape or that the content of one main data area, or the 3340 IPL area, and/or the 5444 simulation area is to be restored from tape. The FROM parameter indicates the location of the area being copied. The TO parameter indicates the location of the area receiving the copy. The PACK parameter indicates the name of the area being used.
3	The SYSTEM parameter indicates whether or not the 3340 IPL area associated with the 5444 simulation area should be copied to or from magnetic tape. This parameter is used when restoring distribution tapes from the program library. When SYSTEM-YES is specified, the 3340 IPL area and its associated 5444 simulation area is copied to or from magnetic tape. When SYSTEM-NO is specified, the program proceeds as if the parameter was not included in the control statement. If SYSTEM-YES is specified, the valid codes that may be used with the TO and FROM parameters are R1, F1, R2, or F2. If no SYSTEM parameter is specified, the default is SYSTEM-NO.

Figure 3-8. Control Statements Required for the Dump/Restore Program

3340 SIMULATION AREA PROGRAM ($SCOPY)

The 3340 simulation area program has the following six functions:

COPYAREA: Copies the entire contents of one 5444 simulation or backup area to another 5444 simulation or backup area.

CLEAR: Clears all the data from a 5444 simulation or backup area and builds a simulated cylinder 0 (optionally gives volume ID and owner ID).

NEWNAME: Changes the name (volume ID) of a 5444 simulation or backup area.

NAMES: Prints/displays the name (volume ID) of each available 5444 simulation area, 5444 backup area, and main data area.

MOVE: Copies the entire contents of one 5444 simulation or backup area to another 5444 simulation or backup area, clears the area from which the contents were copied, and builds a simulated 5444 cylinder 0 in the area copied from.

COPYIPL: Copies IPL records from one 3340 data module to another 3340 data module.

The use of any of these functions requires that the 5444 area referenced be dedicated to the partition executing $SCOPY, except for the FROM area in the COPYAREA function and the NAMES function. The data module, on which the 5444 area is being referenced, cannot be dedicated to the other partition.

Four contiguous areas of 10 cylinders each (starting at cylinder 169) are reserved on each of the four 3340 drives to simulate 5444 drives. Each simulated 5444 requires 10 cylinders; therefore, on a system with four 3340 drives, there are sixteen 5444 simulated areas. The first two areas on D1 are reserved for F1 and R1; the first two areas on D2 are reserved for F2 and R2. These four areas are accessible via normal data management (except multivolume and indexed files) and 5444 system service programs (except $INIT or $ALT). $SCOPY provides access to these and the other 12 possible areas for maintenance purposes.

The 5444 simulation areas are designated as follows:

Area	Start (CCC/HH/RR)	End (CCC/HH/RR)
A – First 5444 simulation or backup area	169/00/01	178/19/48
B – Second 5444 simulation or backup area	179/00/01	188/19/48
C – Third backup area	189/00/01	198/19/48
D – Fourth backup area	199/00/01	208/19/48

The following diagram shows the location of data modules and backup areas on the 3340.

	Cyl 0	Cyl 1-166	Cyl 167-168	Cyl 169-178	Cyl 179-188	Cyl 189-198	Cyl 199-208	Cyl 209 heads 1-7
Drive 1	System area	Main data area D1	Alternate tracks	Simulation area A D1A F1 area	Simulation area B D1B R1 area	Simulation area C D1C	Simulation area D D1D	Error logging and CE tracks
Drive 2	System area	Main data area D2	Alternate tracks	Simulation area A D2A F2 area if simulation is on	Simulation area B D2B R2 area if simulation is on	Simulation area C D2C	Simulation area D D2D	Reserved
Drive 3 (if supported)	System area	Main data area D3	Alternate tracks	Simulation area A D3A	Simulation area B D3B	Simulation area C D3C	Simulation area D D3D	Reserved
Drive 4 (if supported)	System area	Main data area D4	Alternate tracks	Simulation area A D4A	Simulation area B D4B	Simulation area C D4C	Simulation area D D4D	Reserved

The control statements required for this system service program are shown in Figure 3-9. The numbered list corresponds to the numbered items in the figure.

```
// LOAD $SCOPY, { F1 | R1 | F2 | R2 }

// RUN

// COPYAREA FROM- [1] { D1 | D2 | D3 | D4 } { A | B | C | D } [2] ,TO- { D1 | D2 | D3 | D4 } { A | B | C | D } [3] [4] [5]
,PACK-name,AREA-name[,TONAME-name]

[6]
[,SYSTEM- { YES | NO } ]

// CLEAR FROM- [7] { D1 | D2 | D3 | D4 } { A | B | C | D } [8] [9] [10] [11] [12]
,PACK-name[,AREA-name] [,CLRNAME-name] [,ID-name] [,TYPE- { CHECK | FORCE } ]

// NEWNAME TO- [13] { D1 | D2 | D3 | D4 } { A | B | C | D } [14] [15] [16]
,PACK-name,AREA-name,TONAME-name

[17]
// NAMES [ { PRINT | DISPLAY | BOTH } ]

// MOVE FROM- [18] { D1 | D2 | D3 | D4 } { A | B | C | D } [19] ,TO- { D1 | D2 | D3 | D4 } { A | B | C | D } [20] [21] [22] [23]
,PACK-name,AREA-name[,TONAME-name] [,ID-name]

[24]
[,SYSTEM- { YES | NO } ] [25] [,CLRNAME-name]

// COPYIPL FROM- [26] { D1 | D2 | D3 | D4 } [27] ,TO- { D2 | D3 | D4 } [28] ,PACK-name

// END
```

Item	Explanation

1 The FROM parameter identifies the data module and the simulation or backup area being copied.

Code	Meaning	Code	Meaning
D1	3340 unit D1	A	First simulation or backup area
D2	3340 unit D2	B	Second simulation or backup area
D3	3340 unit D3	C	Third simulation or backup area
D4	3340 unit D4	D	Fourth simulation or backup area

D1A is the area code for F1.
D1B is the area code for R1.
D2A is the area code for F2 if simulation is on (via a SIMULATE command).
D2B is the area code for R2 if simulation is on (via a SIMULATE command).

Figure 3-9 (Part 1 of 3). Control Statements Required for the 3340 Simulation Area Program

Item	Explanation
2	The TO parameter identifies the data module and the simulation or backup area receiving the copy (see item 1 for an explanation of the codes).
3	The PACK parameter identifies the name of the data module containing the simulation or backup area receiving the copy. This is the name assigned by the disk initialization program ($INIT).
4	The AREA parameter identifies the name of the simulation or backup area being copied.
5	The TONAME parameter (up to six characters) is used to change the name of the simulation or backup area receiving the copy. If the TONAME parameter is omitted, the name of the simulation or backup area being copied is used.
6	The SYSTEM parameter is used to copy IPL information. If SYSTEM-YES is specified, the IPL information from cylinder 0 of the system data module on drive 1 is copied to cylinder 0 of the data module receiving the copy. If SYSTEM-NO is specified, the IPL information is not copied. If no parameter is specified, SYSTEM-NO is assumed.
7	The FROM parameter identifies the data module and the simulation or backup area being cleared (see item 1 for an explanation of the codes).
8	The PACK parameter specifies the name of the data module containing the simulation or backup area being cleared. This is the name assigned by the disk initialization program ($INIT).
9	The AREA parameter specifies the name of the simulation or backup area being cleared. This parameter cannot be specified if the area has no assigned name. The AREA parameter must be specified as PID001 in order to clear an area used for distribution of programs from the IBM program library/PID.
10	The CLRNAME parameter (up to six characters) specifies the name to be given to the area being cleared. If no parameter is specified, the name of the area is the name previously defined. If no name has been previously defined, CLRNAME must be specified. The name can be any combination of standard System/3 characters except apostrophes, blanks, and commas (due to their delimiter function). (See *Appendix E* for a list of standard System/3 characters.) Its length must not exceed six characters. The following are valid area names: 0, F0001, 012, A1B9, ABC.
11	The ID parameter enables you to include a maximum of 10 characters, in addition to the area name, to further identify a disk. The characters can be any combination of standard System/3 characters (see Appendix E) except apostrophes, blanks, and commas (due to their delimiter function). The information is strictly for pack identification. (It is not used by the system for checking purposes.) If no parameter is specified, the owner ID area in the volume label is blank.
12	The TYPE parameter specifies the type of clear to be done. The optional parameters are: • CHECK: Check for active files or libraries and halt if any are found. • FORCE: Clear the area without checking for active files or libraries. (All libraries and data files are deleted.) If no parameter is specified, the TYPE-CHECK is assumed.

Figure 3-9 (Part 2 of 3). Control Statements Required for the 3340 Simulation Area Program

Item	Explanation
13	The TO parameter identifies the data module and the simulation or backup area being renamed (see item 1 for an explanation of the codes).
14	The PACK parameter specifies the name of the data module containing the simulation or backup area being renamed. This is the name assigned by the disk initialization program ($INIT).
15	The AREA parameter specifies the existing name of the simulation or backup area being renamed.
16	The TONAME parameter (up to six characters) specifies the new name being given to the simulation or backup area. (See item 10 for explanation of valid names.)
17	The PRINT parameter indicates that all online simulation or backup area names are to be printed on the device assigned as the system print device (// PRINTER). The DISPLAY parameter indicates that all online simulation or backup area names are to be displayed on the display screen. The BOTH parameter indicates that all online simulation or backup area names are to be printed and displayed. If no parameter is specified, the simulation or backup area names are printed. If an area is unavailable or in use by the other partition, its volume ID will be left blank and an exception line will be built giving the reason. During a display, a second screen will be provided for this exception output.
18	The FROM parameter identifies the data modules and the simulation or backup area being moved (see item 1 for an explanation of the codes).
19	The TO parameter identifies the data module and the simulation or backup area receiving moved information (see item 1 for an explanation of the codes).
20	The PACK parameter identifies the name (assigned by $INIT) of the data module containing the simulation or backup area receiving the moved information.
21	The AREA parameter specifies the name of the simulation or backup area being moved.
22	The TONAME parameter (up to six characters) is used to change the name of the simulation or backup area receiving the moved information. If no parameter is specified, the name of the simulation or backup area being moved is used. (See item 10 for explanation of valid names.)
23	The ID parameter (up to 10 characters) specifies the owner ID that will be given to the area from which information was moved. If no parameter is specified, the owner ID in the volume label is left blank. *Note:* Using a COPYAREA or MOVE, the receiving area is assigned the owner ID of the area being copied from. (See item 11 for explanation of valid names.)
24	The SYSTEM parameter is used to move IPL information. If SYSTEM-YES is specified, the IPL information from cylinder 0 of the system data module on drive 1 is moved to cylinder 0 of the data module receiving the moved information. If SYSTEM-NO is specified, the IPL information is not moved. If no parameter is specified, SYSTEM-NO is assumed.
25	The CLRNAME parameter (up to six characters) is used to assign a name to the area from which the information has been moved. If no parameter is specified, the area is cleared and the name previously assigned is used. (See item 10 for explanation of valid names.)
26	The FROM parameter identifies the data module containing the IPL records to be copied.
27	The TO parameter identifies the data module receiving the IPL records.
28	The PACK parameter identifies the name (assigned by $INIT) of the data module receiving the IPL records.

Figure 3-9 (Part 3 of 3). Control Statements Required for the 3340 Simulation Area Program

The following are examples of control statements used to perform specific functions of the 3340 simulation area program.

```
// CLEAR FROM-D3C,PACK-D3D3D3,
   CLRNAME-D3CD3C,ID-BACKUPF1
```

Explanation: After a check for active files and libraries (default is TYPE-CHECK), the third backup area on drive D3 is cleared. It is given a volume ID of D3CD3C and an owner ID of BACKUPF1. This is an example of the CLEAR that is to be run after the entire data module has been initialized by $INIT.

```
// CLEAR FROM-D1C,PACK-D1D1D1,
   AREA-PID001,CLRNAME-D1CD1C,TYPE-FORCE
```

Explanation: After verification that the volume ID on the third area of drive D1 is PID001, the area is cleared and given a volume ID of D1CD1C. The owner ID is all blanks and the check for active files and libraries is bypassed. This is an example of the control statement needed to clear an area containing programs from the IBM program library/PID.

```
// COPYAREA FROM-D1A,AREA-F1F1F1,TO-D4C,
   PACK-D4D4D4
```

Explanation: After verification that the volume ID of area D1A is F1F1F1, the area (D1A) is copied to the third backup area on drive D4. The entire simulation area is copied including cylinder 0, the volume ID, and the owner ID if it was present on D1A.

```
// MOVE FROM-D1B,AREA-R1R1R1,TO-D4D,
   PACK-D4D4D4,TONAME-BKUPR1
```

Explanation: The entire R1 simulation area on drive D1 is copied to the fourth backup area on drive D4, and the D4D area is given a volume ID of BKUPR1 and an owner ID of the R1 area if one exists. After the copy is complete, the R1 simulation area is cleared of all data, its owner ID field is blank, and it retains its volume ID of R1R1R1. The R1 simulation area is now ready to be the receiving area of a COPYAREA or another MOVE.

```
// COPYIPL FROM-D2,TO-D3,PACK-D3D3D3
```

Explanation: The IPL (initial program load) records and the 3340 micro code are copied from cylinder 0 of the data module on drvie D2 to cylinder 0 of the data module on drive D3. A check is made before the copy to ensure that the volume ID of the data module on drive D3 is D3D3D3.

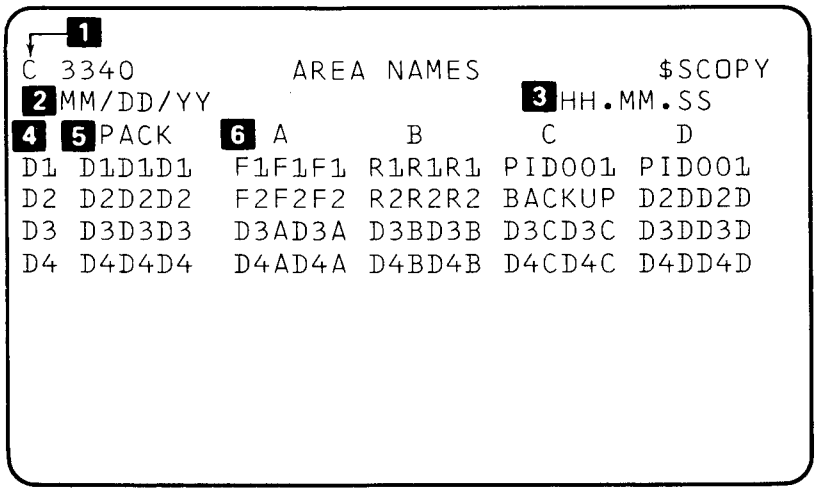
```
// NAMES BOTH
```

Explanation: This control statement enables you to print on the system print device the volume ID and ID of all online and available 3340 data modules and 5444 simulation and backup areas. All simulation and backup areas on a data module are considered by the 3340 simulation area program as unavailable if the data module is dedicated to the other partition, if the other partition has a roll-in pending, or if the data module has not been initialized by System/3 $INIT. This control statement also prints an exception line, if needed, giving the reason for any unavailable 5444 simulation or backup areas.

This statement displays the volume ID of all online and available 3340 data modules and 5444 simulation and backup areas on the 3277 display station (see Figure 3-10). It also displays an exception screen, if needed and requested by the operator, giving the reason for any unavailable 5444 simulation or backup areas (see Figures 3-11 and 3-12).

```
// NEWNAME TO-D4B,AREA-R1R1R1,
   PACK-D4D4D4,TONAME-BACKUP
```

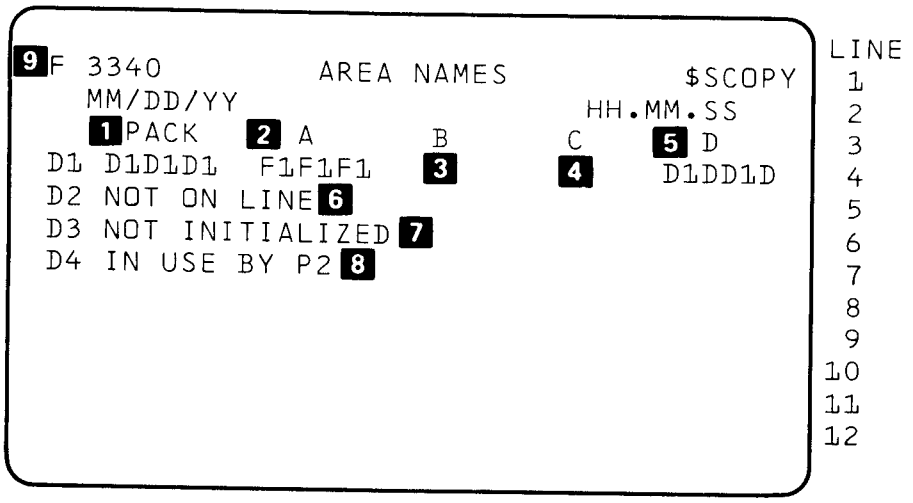
Explanation: After verifying that the name (volume ID) of the 3340 data module on D4 is D4D4D4 and that the name (volume ID) of the second backup area on D4 is R1R1R1, the name (volume ID) of the backup area is changed from R1R1R1 to BACKUP.



Item	Explanation
1	Operator response C – Cancel display and return to read the next control statement. F – Page forward to next display. B – Page backward to previous display. The first display screen will have a C in this position if there is no subsequent display. It will have an F in this position to indicate that a subsequent display containing exception information is available. The last exception output page will have a C in this position.
2	The system date.
3	The system time (if the time-of-day option is supported).
4	The unit code of the 3340.
5	The volume ID of the 3340 data module.
6	The volume ID of the 5444 simulation or backup area.

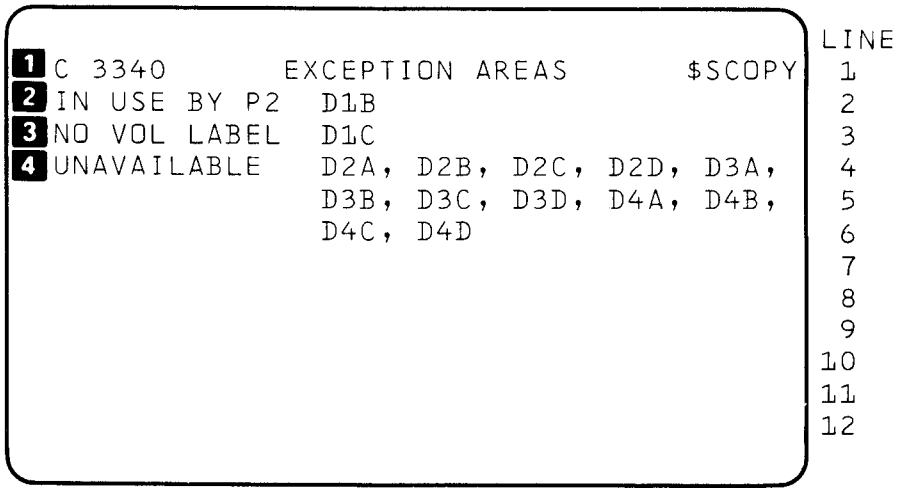
Note: The lines for D3 and D4 appear only if the system supports D3 and D4.

Figure 3-10. Example of Output from a // NAMES Display



Item	Explanation
1	Drive D1 was available; therefore, the volume ID is displayed.
2	5444 simulation area D1A (F1) was available; therefore, its volume ID is displayed.
3	5444 simulation area D1B (R1) was in use by the other partition or had no volume ID; therefore, its volume ID is not displayed.
4	5444 backup area D1C was in use by the other partition or it had no volume ID; therefore, its volume ID is not displayed.
5	5444 backup area D1D was available; therefore, its volume ID is displayed.
6	3340 unit D2 is not ready; therefore; its volume ID and the volume IDs of the 5444 simulation and backup areas are not displayed.
7	The 3340 data module on unit D3 is not properly initialized; therefore, its volume ID and the volume IDs of the 5444 backup areas are not displayed.
8	The 3340 data module on unit D4 is in use by the other partition; therefore, its volume ID and the volume IDs of the 5444 backup areas are not displayed.
9	The F (forward code) appears indicating that a subsequent display containing exception information is available.

Figure 3-11. Example of // NAMES Display with Exception Output



Item	Explanation
1	The C (cancel code) appears indicating that this is the last display page.
2	5444 simulation area D1B (R1) is in use by the other partition.
3	5444 backup area D1C did not have a valid volume label.
4	The designated 5444 simulation and backup areas were not available to $SCOPY.

Figure 3-12. Example of Exception Output with // NAMES Display Statement

SYSTEM HISTORY AREA DISPLAY PROGRAM

The system history area display program allows you to print the contents of the SHA (system history area). The SHA is a special area on the system pack that contains the following information:

- OCL statements read by the system
- OCL diagnostics issued
- Control statements for the system service programs read by the system
- OCC entered by the operator
- OCC diagnostics issued
- System messages issued
- Operator responses to system messages
- CRT display written to the history area by the operator

The control statements required for this system service program are shown in Figure 3-13. The numbered list corresponds to the numbered items in the figure.

```
// LOAD $HIST, { F1 }
               { R1 }
               { F2 }
               { R2 }

// RUN

// PRINT HISTORY- { CURRENT }
                  { ALL     }

// END
```

Item	Explanation
1	The HISTORY parameter allows you to obtain a printed copy of the SHA. If HISTORY-ALL is specified, the entire SHA is printed. If HISTORY-CURRENT is specified or the PRINT statement is not specified, only SHA entries not previously printed are listed.

Figure 3-13. Control Statements Required for the System History Area Display Program

LIBRARY MAINTENANCE PROGRAM

The library maintenance program has five functions: allocate, copy, delete, modify, and rename.

Allocate

The allocate function:

- Creates (reserves space for) libraries, SWA (scheduler work area), SHA (system history area), and rollout/rollin area
- Changes the sizes of libraries and SHA
- Deletes libraries
- Reorganizes libraries

Note: Remember that libraries cannot exist in the main data area; only R1, F1, R2, and F2 can contain libraries that may be referenced by the library maintenance program.

The control statements required for the allocate function are shown in Figure 3-14. The numbered list corresponds to the numbered items in the figure.

Copy

The copy function:

- Copies or replaces library entries from reader to library
- Copies or replaces library entries from file to library
- Copies entries from library to file
- Copies entries from library to library
- Prints entries from library
- Punches entries from library
- Prints and punches entries from library
- Prints entries from a directory

```
// LOAD $MAINT, { F1 | R1 | F2 | R2 }

// RUN

// ALLOCATE TO- 1 { F1 | R1 | F2 | R2 } [,SOURCE- { nnn | R } ] 2 [,OBJECT- { nnn | R } ] 2

3 [,SYSTEM- { YES | NO } ] 4 [,HISTORY-nnn] 5 [,DIRSIZE-nnn] 6 [,WORK- { F1 | R1 | F2 | R2 | D1 | D2 | D3 | D4 } ]

// END
```

Figure 3-14 (Part 1 of 2). Control Statements Required for the Allocate Function of the Library Maintenance Program

Item	Explanation
1	The TO parameter indicates the location of the 5444 simulation area that contains, or will contain, the library. If the program use involves both libraries, the libraries must be in the same 5444 simulation area. The TO parameter cannot be the same 5444 simulation area from which the library maintenance program or system is loaded.
2	The SOURCE and OBJECT parameters identify the library and function to perform. You can specify either or both parameters.

Parameter	Meaning
SOURCE-nnn OBJECT-nnn	If the 5444 simulation area contains no library, the parameter means create a library. The nnn is the number of tracks you want to assign to the library. If the 5444 simulation area contains a library, the parameter means change the library size. The nnn is the number of tracks you want to assign to the library.
SOURCE-0 OBJECT-0	Delete the library.
SOURCE-R OBJECT-R	Reorganize the library.

3	The optional SYSTEM parameter allows you to create a system pack. If SYSTEM-YES is specified, a scheduler work area (SWA) and a system history area (SHA) are created to make the 5444 simulation area a system pack. If SYSTEM-NO is specified or the parameter is omitted, these areas are not created and the 5444 simulation area is not a system pack.
4	The optional HISTORY parameter allows you to assign the number of tracks for the system history area. The minimum size is two tracks; the maximum is the number of tracks in the available area. If this parameter is not specified, the SHA is assigned two tracks.
5	The optional DIRSIZE parameter allows you to specify the size of the object library directory. Up to nine tracks can be specified. If the DIRSIZE parameter is omitted, the SYSTEM parameter is used to determine the directory size. If SYSTEM-YES is coded, three tracks are assigned to the directory. If SYSTEM-NO is specified or the SYSTEM parameter is omitted, one track is assigned to the directory.
6	The optional WORK parameter indicates the location of the 5444 simulation area or main data area that contains a work area. Library entries are temporarily stored in the work area while the program moves and reorganizes libraries. The WORK parameter is not required for a compress in place.

Figure 3-14 (Part 2 of 2). Control Statements Required for the Allocate Function of the Library Maintenance Program

The control statements required for the copy function are shown in Figure 3-15. The numbered list corresponds to the numbered items in the figure.

```
                      ( F1 )
// LOAD $MAINT,       { R1 }
                      { F2 }
                      ( R2 )

// FILE (OCL statement required for library-to-file or file-to-library)

// RUN

Reader-to-library:

          1          2            ( S )        3          4   ( F1 )        5          ( T )
// COPY FROM-READER,LIBRARY-      { P } ,NAME-xxxxxx,TO-      { R1 }   [,RETAIN-  { P }   ]
                                  { O }                       { F2 }              ( R )
                                  ( R )                       ( R2 )
  6
library entry

// CEND

File-to-library:

          1        4   ( F1 )    7              5          ( T )       8           ( 80 )
// COPY FROM-DISK,TO-  { R1 } ,FILE-x . . . x, [RETAIN-    { P }  ] [,RECL-        { 96 }  ]
                       { F2 }                              ( R )
                       ( R2 )

          2      ( S )   3
// COPY LIBRARY- { P } ,NAME-xxxxxx in the file
                 { O }
  6              ( R )
library entry } in the file

// CEND } in the file

Library-to-file:

          1   ( F1 )   4            7               8        ( 80 )
// COPY FROM- { R1 } ,TO-DISK,FILE-x . . . x[,RECL-          { 96 }  ]
              { F2 }
              ( R2 )

  9         2      ( S   )
// ENTRY LIBRARY-  { P   }   3    ( xxxxxx    )
                   { O   } ,NAME- { ccccc.ALL }
                   { R   }        ( ALL       )
                   ( ALL )
 10
// NEND
```

Figure 3-15 (Part 1 of 4). Control Statements Required for the Copy Function of the Library Maintenance Program

Library-to-library:

```
// COPY FROM- (1) { F1 | R1 | F2 | R2 }  (2) ,LIBRARY- { S | P | O | R | ALL }  (3) ,NAME- { xxxxxx | ccccc.ALL | ALL | SYSTEM }  (4) ,TO- { F1 | R1 | F2 | R2 }  [,RETAIN- { T | P | R } ]
   [,NEWNAME- { xxxxxx | ccccc } ]  (11)
```

Library-to-printer-and/or card:

```
// COPY FROM- (1) { F1 | R1 | F2 | R2 }  (2) ,LIBRARY- { S | P | O | R | ALL | SYSTEM }  (3) ,NAME- { xxxxxx | ccccc.ALL | ALL | DIR }  (4) ,TO- { PUNCH | PRINT | PRTPCH }
[,OMIT- { xxxxxx | ccccc.ALL } ]  (12)

// END
```

Item	Explanation

1 The FROM parameter indicates the location of the library entries to be copied, printed, or punched.

Code	Meaning
F1, R1, F2, R2	Location of library containing the entries
READER	Entry is to be read from the system input device
DISK	Entry resides in a data file in a main data area or 5444 simulation area. The file must be described by a FILE statement.

2 The LIBRARY parameter indicates the type of entries in the copy function.

Code	Meaning
S	Source statements (source library)
P	OCL procedures (source library)
O	Object programs (object library)
R	Routines (object library)
ALL	All types of entries (S, P, O, R)
SYSTEM	Only system directory entries

Figure 3-15 (Part 2 of 4). Control Statements Required for the Copy Function of the Library Maintenance Program

Item	Explanation
3	The NAME parameter identifies the entries involved in the copy function. **Code** — **Meaning** xxxxxx — Name of library entry. ccccc.ALL — Only those entries beginning with the indicated characters of the type indicated in the LIBRARY parameter. You can use up to five characters. ALL — All entries of the type indicated in the LIBRARY parameter. SYSTEM — System programs that make up the minimum system and IPL information contained on the 5444 simulation area cylinder 0 are copied. DIR — Directory entries for all library entries of the type indicated in the LIBRARY parameter are involved in copy use. If LIBRARY-ALL is specified, system directory entries are also printed.
4	The TO parameter indicates the location of the library to contain the copies of the entries or whether the entries are to be printed, punched, both, or copied to a file. **Code** — **Meaning** F1, R1, F2, R2 — Location of library to contain copies. PRINT — Entries are printed. PUNCH — Entries are punched. PRTPCH — Entries are printed and punched. DISK — Entries are copied to a data file in a main data area or 5444 simulation area. The file must be described by a FILE statement.
5	The optional RETAIN parameter indicates the designation of the TO entry. **Code** — **Meaning** T — Temporary. If replacing an entry, halt before replacing entry. P — Permanent. If replacing an entry, halt before replacing entry. R — Permanent. If replacing an entry, do not halt before replacing entry. If RETAIN is not specified, T is assumed except when copying library-to-library where LIBRARY-ALL, NAME-ALL or LIBRARY-O, NAME-SYSTEM are specified. In these cases, P is assumed.

Figure 3-15 (Part 3 of 4). Control Statements Required for the Copy Function of the Library Maintenance Program

Item	Explanation
6	Library entry being replaced or added. The // CEND statement must follow the source or object entry being placed in the source or object library.
7	The FILE parameter identifies the file for a file-to-library or library-to-file copy. The filename must match the filename on the FILE statement.
8	The optional RECL parameter specifies the size of the records for a file-to-library or library-to-file copy. Only 80- or 96-column card-image records are allowed. If this parameter is omitted, 96 is assumed.
9	The // ENTRY statement identifies the library entry to be copied to a file. Any number of these statements can precede the // NEND statement.
10	The // NEND statement must follow the // ENTRY statements to terminate the copy to file.
11	The optional NEWNAME parameter indicates the name you want to use for entries being put in the library. If this parameter is omitted, the name in the NAME parameter is used.
	Code — **Meaning** xxxxxx — Name to be used for entry. ccccc — Beginning characters of names for entries. The number of characters (up to five) must be the same number as in the NAME-ccccc.ALL parameter.
12	The optional OMIT parameter indicates which entries are to be omitted when printing directory entries.
	Code — **Meaning** xxxxxx — Omit the entry specified by xxxxxx. ccccc.ALL — Omit all entries with the beginning characters ccccc.

Figure 3-15 (Part 4 of 4). Control Statements Required for the Copy Function of the Library Maintenance Program

Delete

The delete function deletes temporary and permanent entries from a library.

The control statements required for the delete function are shown in Figure 3-16. The numbered list corresponds to the numbered items in the figure.

Modify

The modify function:

- Reserializes a source library entry
- Lists the statements in a source library entry
- Removes statements from a source library entry
- Replaces source library statements
- Inserts statements into a source library entry

```
// LOAD $MAINT, { F1 }
                { R1 }
                { F2 }
                { R2 }

// RUN

// DELETE FROM- 1 { F1 } 2 ,LIBRARY- { S   } 3 ,NAME- { xxxxxx } 4 [,RETAIN- { T } ]
                  { R1 }               { P   }        { ccccc.ALL }          { P }
                  { F2 }               { O   }        { ALL       }
                  { R2 }               { R   }
                                       { ALL }

// END
```

Item	Explanation
1	The FROM parameter indicates the location of the library that contains the entries to be deleted.
2	The LIBRARY parameter indicates the type of entries to be deleted.

Code	Meaning
S	Source statements (source library)
P	OCL procedures (source library)
O	Object programs (object library)
R	Routines (object library)
ALL	All types of entries (S, P, O, R)

3 The NAME parameter identifies the entries to be deleted.

Code	Meaning
xxxxxx	Name of the library entry.
ccccc.ALL	Entries that have names beginning with the indicated characters (up to five) of the type indicated in the LIBRARY parameter.
ALL	All entries of the type indicated in the LIBRARY parameter. NAME-ALL cannot be used with LIBRARY-ALL.

4 The optional RETAIN parameter indicates the designation of the entries to be deleted.

Code	Meaning
T	Temporary
P	Permanent

If RETAIN is omitted, T is assumed.

Figure 3-16. Control Statements Required for the Delete Function of the Library Maintenance Program

The control statements required for the modify function are shown in Figure 3-17. The numbered list corresponds to the numbered items in the figure.

```
                   ⎧ F1 ⎫
// LOAD $MAINT,    ⎨ R1 ⎬
                   ⎪ F2 ⎪
                   ⎩ R2 ⎭

// RUN

     1     2             3  ⎧ F1 ⎫         4              5         ⎧ F1 ⎫
// MODIFY NAME-xxxxxx,FROM- ⎨ R1 ⎬ ,LIBRARY- ⎧ S ⎫ ,WORK- ⎨ R1 ⎬
                            ⎪ F2 ⎪           ⎩ P ⎭        ⎪ F2 ⎪
                            ⎩ R2 ⎭                        ⎩ R2 ⎭

     6          7                   8                    9
  [,RESER- ⎧ YES  ⎫ ] [,LIST- ⎧ YES ⎫ ] [,SEQFLD- ⎧ xxyy ⎫ ] [,INCR- ⎧ nnnnn ⎫ ]
           ⎨ NO   ⎬           ⎩ NO  ⎭             ⎩ 9296 ⎭          ⎩ 10    ⎭
           ⎩ ONLY ⎭
  10
// REMOVE FROM-n . . . n,TO-n . . . n
  11
// REPLACE FROM-n . . . n,TO-n . . . n

Statements to replace those removed
  12
// INSERT AFTER-n . . . n

Statements to be inserted
  13
// CEND

// END
```

(Defaults underlined in the figure: NO for RESER, NO for LIST, 9296 for SEQFLD, 10 for INCR.)

Item	Explanation
1	The MODIFY statement initiates the modification by describing the source library entry.
2	The NAME parameter identifies the name of the entry being modified. This is the name that identifies the entry in the library directory.
3	The FROM parameter indicates the location of the library that contains the entry to be modified.
4	The LIBRARY parameter indicates the type of library entry to be modified. **Code** — **Meaning** S — Source statements (source library) P — OCL procedures (source library)
5	The WORK parameter indicates the location of the 5444 simulation area containing the space to be used as a work area.

Figure 3-17 (Part 1 of 2). Control Statements Required for the Modify Function of the Library Maintenance Program

Item	Explanation
6	The optional RESER parameter indicates whether reserialization should be done when the entry is placed back in the source library. **Code** — **Meaning** YES — Reserialization is to be done. NO — Reserialization is not to be done. ONLY — Reserialization only; no other maintenance is done. When this is coded, no REMOVE, REPLACE, INSERT, or CEND statements can be entered. If this parameter is omitted, NO is assumed.
7	The optional LIST parameter allows you to list the source library entry as the modified entry is placed back in the source library. If this parameter is omitted, NO is assumed.
8	The optional SEQFLD parameter identifies the start and end positions of the field that contains the sequence number. The sequence number can be up to eight digits long. The start position is entered first (xx), then the end position (yy). If this parameter is omitted, 9296 is assumed.
9	The optional INCR parameter indicates the increment value for the sequence field if reserialization (RESER-YES or RESER-ONLY) is specified. The value can be up to five digits. If this parameter is omitted, 10 is assumed.
10	The REMOVE statement deletes all statements between and including the FROM and TO sequence numbers.
11	The REPLACE statement replaces all statements between and including the FROM and TO sequence numbers. The statements replacing the statements being removed immediately follow the REPLACE statement.
12	The INSERT statement inserts the supplied statements after the statement identified by the AFTER parameter. The statements to be inserted immediately follow the INSERT statement.
13	The CEND statement must follow the control statements to terminate the modify function.

Figure 3-17 (Part 2 of 2). Control Statements Required for the Modify Function of the Library Maintenance Program

Rename

The rename function changes the names of library entries.

The control statements required for the rename function are shown in Figure 3-18. The numbered list corresponds to the numbered items in the figure.

1000-FILE VTOC CONVERSION PROGRAM

This program is used to convert a 50-file VTOC (volume table of contents) on a 5445 to a 1000-file VTOC. The program can also be used to convert a 1000-file VTOC on a 5445 to a 50-file VTOC, assuming that there are 50 or fewer files in the VTOC. For a description of these functions, see *IBM System/3 Model 15 System Control Programming Reference Manual*, GC21-5077.

```
// LOAD $MAINT, {F1|R1|F2|R2}

// RUN

// RENAME [1] FROM- {F1|R1|F2|R2} [2] ,LIBRARY- {S|P|O|R} [3] ,NAME- {xxxxxx|ccccc.ALL} [4] ,NEWNAME- {xxxxxx|ccccc}

// END
```

Item	Explanation
1	The FROM parameter identifies the location of the library that contains the entry to be renamed.
2	The LIBRARY parameter identifies the type of library entry to be renamed.

Code	Meaning
S	Source statements (source library)
P	OCL procedures (source library)
O	Object programs (object library)
R	Routines (object library)

3 The NAME parameter identifies the entry to be renamed.

Code	Meaning
xxxxxx	Current name of the entry to be renamed. This is the name that identifies the entry in the library directory.
ccccc.ALL	Entries that have names beginning with the indicated characters (up to five) of the type indicated in the LIBRARY parameter.

4 The NEWNAME parameter indicates the new name of the entry to be renamed.

Code	Meaning
xxxxxx	New name to be given to entry.
ccccc	Beginning characters (up to five) of names to be given to entries.

Figure 3-18. Control Statements Required for the Rename Function of the Library Maintenance Program

For any disk with a 1000-file VTOC (5445 or 3340), this program can be used to compress the entries in the VTOC. All file entries are moved to the beginning of the VTOC, thus decreasing the time required to locate a particular file.

The control statements required to do this compression are shown in Figure 3-19. The numbered list corresponds to the numbered item in the figure.

```
// LOAD $WVTOC, {F1|R1|F2|R2}

// RUN

// COMPRESS PACK-name,UNIT- {D1|D2|D3|D4}

// END
```

Item	Explanation
1	The PACK parameter (PACK-name) tells the program the name of the data module on which the VTOC is to be reformatted. The 1000-file VTOC conversion program compares the name in the PACK parameter with the one on the data module to ensure that they match. In this way the program ensures that it is using the right data module.
2	The UNIT parameter (UNIT-code) tells the program the location of the data module on which the VTOC is to be compressed. Codes for the possible locations are as follows:

Code	Meaning
D1	Data module on 3340 drive 1
D2	Data module on 3340 drive 2
D3	Data module on 3340 drive 3
D4	Data module on 3340 drive 4

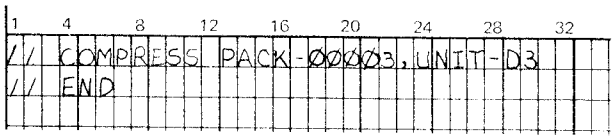
Figure 3-19. Control Statements Required to Compress File Entries in the VTOC Using the 1000-File VTOC Conversion Program

The 1000-file VTOC conversion program compresses a 1000-file capacity VTOC. For improved performance in file access time, all file entries are moved to the beginning of the volume table of contents.

Note: All active files are retained as active after the VTOC is compressed. The program is also used to convert a 50-file VTOC to a 1000-file VTOC for a 5445. For a description of this function, see *IBM System/3 Model 15 System Control Programming Reference Manual*, GC21-5077.

The // COMPRESS PACK-name, UNIT-code control statement moves all active file directory entries in a 1000-file capacity disk to the beginning of the VTOC.

The following example shows the control statements needed to compress a 1000-file VTOC.



Explanation:

- The data module containing the VTOC to be compressed is named 00003 (PACK-00003 in the COMPRESS statement).
- The removable data module containing the VTOC to be compressed is on drive 3 (UNIT-D3).

Reassign Alternate Track Program ($RSALT)

When it is necessary to transport a 3340 data module from System/3 to System/360 or System/370, the reassign alternate track program ($RSALT) must be used prior to using the DOS/OS initialization program.

On a 3340 data module initialized on System/3, there are 40 alternate tracks on cylinders 167 and 168. On a System/360 or System/370 3340 data module, there are 24 alternate tracks on cylinders 208 and 209. $RSALT moves the System/3 alternate tracks from cylinders 167 and 168 to cylinders 208 to 209. Consequently, there is a restriction that if a 3340 data module initialized on System/3 has more than 24 defective primary tracks, it cannot be initialized by System/360 or System/370.

The control statements required for this system service program are shown in Figure 3-20. The numbered list corresponds to the numbered items in the figure.

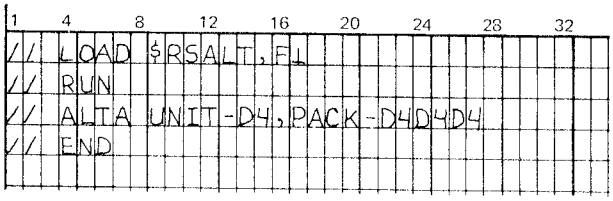
Note: Data interchange is not supported between the System/3 and 360/370 systems, so this program cannot be used for that purpose. System/3 data existing on the data module prior to running $RSALT will be lost.

```
// LOAD $RSALT, { F1 }
                { R1 }
                { F2 }
                { R2 }

// RUN

// ALTA UNIT- { code    }  ,PACK- { name    }
              { 'codes' }         { 'names' }
              { ALL     }

// END
```

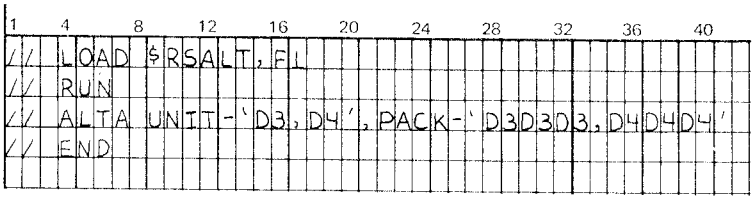
Item	Explanation
1	The UNIT parameter (UNIT-code) tells the location of the 3340 data module that you want to modify. The program can modify up to three data modules during a program run. The form of the UNIT parameter depends on the number of data modules that you want modified. • For one data module, use UNIT-code. • For three data modules, use UNIT-ALL. • For two or three data modules, use UNIT-'code, code, code'. The codes indicate the location of the data modules: **Code** — **Meaning** D2 — Modify data module on 3340 drive 2 D3 — Modify data module on 3340 drive 3 D4 — Modify data module on 3340 drive 4 ALL — Modify all of the preceding data modules $RSALT cannot be run on UNIT-D2 if the 5444 simulation areas (R2 and F2) are active. The operator control command, SIMULATE OFF, can be used prior to executing $RSALT.
2	The PACK parameter (PACK-name) tells the program the name of the 3340 data module to be modified. The parameter length must not exceed six characters. It can contain any of the standard System/3 characters except apostrophes, commas, or leading or embedded blanks. The first PACK parameter applies to the first disk listed in the UNIT parameter. The second PACK parameter applies to the second disk parameter, and so on. If UNIT-ALL is specified, the PACK parameter names must be specified as if the UNIT parameter had been 'D2, D3, D4'. The reassign alternate track program compares the name in the PACK parameter with the name on the data module to ensure that they match. If the names do not match, the program halts with an error message. In this way, the program ensures that it is using the right data module.

Figure 3-20. Control Statements Required for Reassign Alternate Track Program

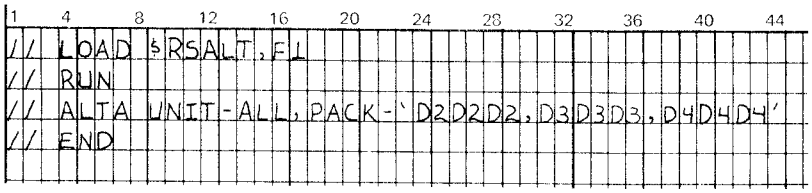
The following illustration shows examples of control statements required to execute the $RSALT program.



Explanation: The 3340 data module on drive 4 is to be modified to System/360 – System/370 format.



Explanation: The 3340 data modules on drives 3 and 4 are to be modified to System/360 – System/370 format.



Explanation: The 3340 data modules on drives 2, 3, and 4 are to be modified to System/360 – System/370 format.

RPG II

Using RPG II (Program Number 5704-RG1), you can process files on the data modules. The file description specifications define the control information and device entry for the data module to the RPG II compiler. Columns 40-46 of this form identify the input/output device for the file.

The main data area is referenced by specifying DISK40 or DISK45 in columns 40-46. If device independent data management is used, these columns are left blank. At execution time, you can assign the device independent files to a 3340 by using a FILE OCL statement. Sequential, direct, and indexed file processing are supported for the main data area. Device independent data management supports only sequential files.

The 5444 simulation area is referenced by specifying DISK in columns 40-46 of the file description specifications. Sequential and direct file processing are supported for the simulation areas. However, index processing and multivolume processing are not supported for the 5444 simulation areas.

Work files for the compiler can reside on either the main data area or a 5444 simulation area.

SUBSET ANS COBOL

Using the subset ANS COBOL compiler and library program product (Program Number 5704-CB1), you can process files on the data modules. The ASSIGN clause is used to assign files to the 3340. The main data area is referenced by specifying 3340 or 5445; the 5444 simulation area is referenced by specifying 5444.

Sequential, indexed, and direct files are supported on the main data area. Only single-volume sequential and direct files are supported on the 5444 simulation areas.

Work files for the compiler can reside in either the main data area or a 5444 simulation area.

FORTRAN IV

Using FORTRAN IV (Program Number 5704-F01), you can process files on the data modules. The main data area is referenced by specifying 5445; the simulation area is referenced by specifying 5444. Work files for the compiler can reside in either the main data area or a 5444 simulation area.

BASIC ASSEMBLER

The SCP macros that presently support the 5445 are available to assist the programmer in using the 3340 main data area. Work files for the assembler can reside on either the main data area or a 5444 simulation area.

DISK SORT

During the sorting process, the disk sort program (Program Number 5704-SM1) requires a minimum of three files: input, output, and work. These files can reside in either the main data area or a 5444 simulation area of a 3340. The input file contains the records to be sorted. The output file contains the sorted records. The work file is an area on the data module which is set aside as a work area. If the work file is a multivolume file, all of the volumes are described on the OCL FILE statement. The parameters for the FILE statement are described in Chapter 2 of this manual. The location of the work file can also be determined automatically. Directed automatic work file allocation uses the SWITCH statement to determine the location of the work file.

Appendix A. 3348 Data Module Surface Format

The IBM 3348 Data Module (Model 70) contains six data surfaces. Each data surface has two read/write heads and is divided into 700 concentric data bands called tracks (Figure A-1). Each of the two read/write heads on a data surface is associated with a track. There are 350 access positions for the 700 tracks.

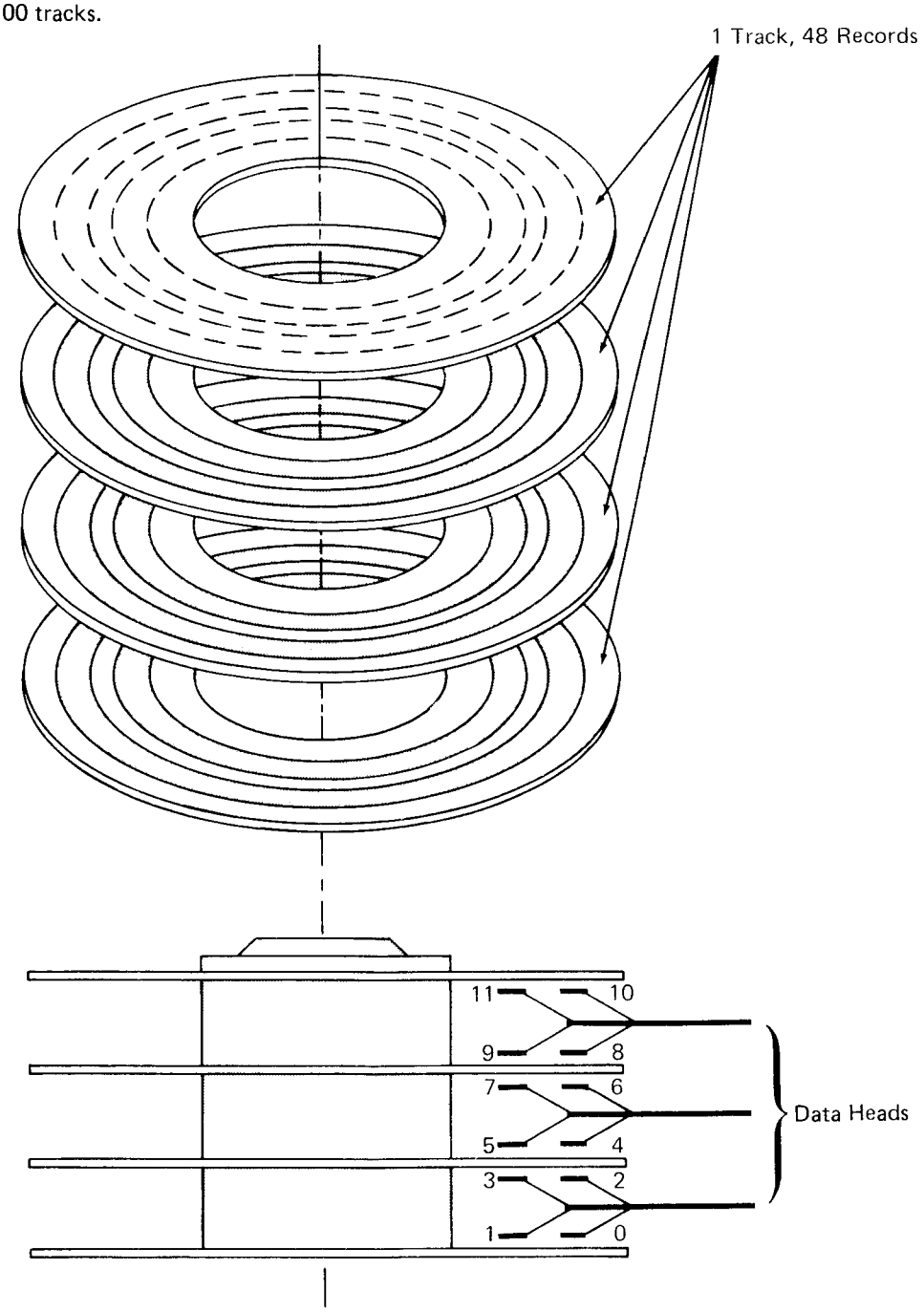


Figure A-1. Data Surface Track Layout

On the 5444 and 5445 disk drives, the tracks that can be accessed without repositioning the read/write heads comprise a physical cylinder. Since the 3340 data module has 12 read/write heads on the access mechanism, there would be 12 tracks per physical cylinder. However, to be consistent with the 5445 addressing technique—when cylinders are mentioned in connection with the 3340 as used on System/3—the cylinders are logical cylinders, not physical cylinders, and there are 20 tracks per logical cylinder. Therefore, to access all 20 tracks of this logical cylinder the access mechanism must be moved. Five 3340 access positions (physical cylinders) are equal to three System/3 logical cylinders or 60 tracks.

In this and other System/3 publications, *track* is used to refer to a logical, not physical, 3340 track; and *cylinder* is used to refer to a logical 3340 cylinder.

Each track contains 48 records, except cylinder 0, which is formatted differently. Each record contains 256 bytes. (A record on the 3340 is similar to a sector on a 5444, and is sometimes referred to as a sector in the System/3 publications and programming support.) A disk address referred to as CCHHR has the same meaning for the 3340 as for the 5445 (CC = cylinder number, HH = head [track] number, R = record number).

Figure A-2 shows the difference between the 3340 as used on the System/3 and as used on the System/370, assuming a physical record length of 256 bytes.

A record (sector) can contain one or more logical records, and a logical record can span one or more records (sectors). Thus, using IBM programming support, a logical record can range from 1 through 32,768 bytes.

At one position of the access mechanism, the following can be accessed:

12 tracks
576 records (sectors)
147,456 bytes

Description	System/370	System/3
Access positions	350 positions	350 positions
One access position	2 cylinders	0.6 cylinders
Six surfaces (4 disks)	1 cylinder	6 tracks
Two read/write heads per surface	2 cylinders	12 tracks
Five access positions	10 cylinders	3 cylinders
Addressable cylinders per module	700 cylinders	210 cylinders
Addressable tracks per cylinder	12 tracks	20 tracks
Byte capacity per track	5,120 bytes	12,288 bytes
Byte capacity per cylinder	61,440 bytes	245,760 bytes
Byte capacity per data module (approximate)	43 Mb[1]	51 Mb

[1] This assumes System/3 record length and one physical block per physical track. The 3348 Model 70 used on a System/370 can store up to 70 megabytes. The 3348 Data Module Model 35 and Model 70F cannot be used on System/3.

Figure A-2. A Comparison of the 3340 as Used on the System/370 with the 3340 as Used on the System/3

Appendix B. Calculating File Size

The factors that control file size are explained in the *IBM System/3 Disk Concepts and Planning Guide,* GC21-7571. The structure of a file on the 3340 main data area (sequential, direct, or indexed) is virtually identical to the structure of a file on the 5445. The 3340 contains 48 records per track and 166 cylinders of main data area per pack while the 5445 contains 20 records (sectors) per track and 200 cylinders per pack.

Following is a list of similarities between the 5445 and the 3340:

- Index entries are calculated using the following formula: key length + 4.
- Index and data areas are contained on whole tracks.
- Indexed files with indexes of 16 tracks or more have space reserved for a track index.
- Spanned records are used.
- The index area must have at least one extra record (sector) for a delimiter or at least three extra records (sectors) if additions will be made to the file.
- There are 20 tracks per cylinder (3980 tracks on the 5445; 3320 tracks of main data area on the 3340).

The following tables and discussion include the specific information necessary to evaluate file size on the 3340.

DATA AREA TRACK REQUIREMENTS

Sequential, direct, and indexed data areas are all structured the same. Figure B-1 shows the number of tracks needed to store a given number of logical records, using various logical record lengths.

NUMBER OF TRACKS[1]

Number of Logical Records	Logical Record Length 50		Logical Record Length 64		Logical Record Length 100		Logical Record Length 128		Logical Record Length 256	
	5445	3340	5445	3340	5445	3340	5445	3340	5445	3340
500	5	3	7	3	10	5	13	6	25	11
1000	10	5	13	6	20	9	25	11	50	21
1500	15	7	19	8	30	13	38	16	75	32
2000	20	9	25	11	40	17	50	21	100	42
2500	25	11	32	14	49	21	63	27	125	53
3000	30	13	38	16	59	25	75	32	150	63
3500	35	15	44	19	69	29	88	37	175	73
4000	40	17	50	21	79	33	100	42	200	84
4500	44	19	57	24	88	37	113	47	225	94
5000	49	21	63	27	98	41	125	53	250	105
5500	54	23	69	29	108	45	138	58	275	115
6000	59	25	75	32	118	49	150	63	300	125
6500	64	27	82	34	127	53	163	68	325	136
7000	69	29	88	37	137	57	175	73	350	146
7500	74	31	94	40	147	62	188	79	375	157
8000	79	33	100	42	157	66	200	84	400	167
8500	84	35	107	45	167	70	213	89	425	178
9000	88	37	113	47	176	74	225	94	450	188
9500	93	39	119	50	186	78	238	99	475	198
10,000	98	41	125	53	196	82	250	105	500	209
10,500	103	43	132	55	206	86	263	110	525	219
11,000	108	45	138	58	215	90	275	115	550	230
11,500	113	47	144	60	225	94	288	120	575	240
12,000	118	49	150	63	235	98	300	125	600	250
12,500	123	51	157	66	245	102	313	131	625	261
13,000	127	53	163	68	254	106	325	136	650	271
13,500	132	55	169	71	264	110	338	141	675	282
14,000	137	57	175	73	274	114	350	146	700	292
14,500	142	60	182	76	284	119	363	152	725	303
15,000	147	62	188	79	293	123	375	157	750	313
15,500	152	64	194	81	303	127	388	162	775	323
16,000	157	66	200	84	313	131	400	167	800	334
16,500	162	68	207	86	323	135	413	172	825	344
17,000	167	70	213	89	333	139	425	178	850	355
17,500	171	72	219	92	342	143	438	183	875	365
18,000	176	74	225	94	352	147	450	188	900	375
18,500	181	76	232	97	362	151	463	193	925	386
19,000	186	78	238	99	372	155	475	198	950	396
19,500	191	80	244	102	381	159	488	204	975	407
20,000	196	82	250	105	391	163	500	209	1000	417
21,000	206	86	263	110	411	171	525	219	1050	438
22,000	215	90	275	115	430	180	550	230	1100	459
23,000	225	94	288	120	450	188	575	240	1150	480
24,000	235	98	300	125	469	196	600	250	1200	500
25,000	245	102	313	131	489	204	625	261	1250	521
26,000	254	106	325	136	508	212	650	271	1300	542
27,000	264	110	338	141	528	220	675	282	1350	563

[1] The number of tracks for data records does not include indexes. For the 3340, this is the main data area, not the 5444 simulation area.

Figure B-1 (Part 1 of 2). Disk Requirements for Data Records

NUMBER OF TRACKS[1]

Number of Logical Records	Logical Record Length 50		Logical Record Length 64		Logical Record Length 100		Logical Record Length 128		Logical Record Length 256	
	5445	3340	5445	3340	5445	3340	5445	3340	5445	3340
28,000	274	114	350	146	547	228	700	292	1400	584
29,000	284	119	363	152	567	237	725	303	1450	605
30,000	293	123	375	157	586	245	750	313	1500	625
31,000	303	127	388	162	606	253	775	323	1550	646
32,000	313	131	400	167	625	261	800	334	1600	667
33,000	323	135	413	172	645	269	825	344	1650	688
34,000	333	139	425	178	665	277	850	355	1700	709
35,000	342	143	438	183	684	285	875	365	1750	730
36,000	352	147	450	188	704	293	900	375	1800	750
37,000	362	151	463	193	723	302	925	386	1850	771
38,000	372	155	475	198	743	310	950	396	1900	792
39,000	381	159	488	204	762	318	975	407	1950	813
40,000	391	163	500	209	782	326	1000	417	2000	834
41,000	401	167	513	214	801	334	1025	428	2050	855
42,000	411	171	525	219	821	342	1050	438	2100	875
43,000	420	175	538	224	840	350	1075	448	2150	896
44,000	430	180	550	230	860	359	1100	459	2200	917
45,000	440	184	563	235	879	367	1125	469	2250	938
46,000	450	188	575	240	899	375	1150	480	2300	959
47,000	459	192	588	245	918	383	1175	490	2350	980
48,000	469	196	600	250	938	391	1200	500	2400	1000
49,000	479	200	613	256	958	399	1225	511	2450	1021
50,000	489	204	625	261	977	407	1250	521	2500	1042
75,000	733	306	938	391	1465	611	1875	782	3750	1563
100,000	977	407	1250	521	1954	814	2500	1042	5000	2084
125,000	1221	509	1563	652	2442	1018	3125	1303	6250	2605
150,000	1465	611	1875	782	2930	1221	3750	1563	7500	3125
175,000	1709	713	2188	912	3418	1425	4375	1823	8750	3646
200,000	1954	814	2500	1042	3907	1628	5000	2084	10,000	4167

[1] The number of tracks for data records does not include indexes. For the 3340, this is the main data area, not the 5444 simulation area.

Figure B-1 (Part 2 of 2). Disk Requirements for Data Records

INDEX AREA TRACK REQUIREMENTS

Indexed files also require tracks for the file index. The following chart shows how many keys can be contained on one track of the file index.

Key-Length	Number of Keys Per Index Track 5445	3340
1	1020	2448
2	840	2016
3	720	1728
4	640	1536
5	560	1344
6	500	1200
7	460	1104
8	420	1008
9	380	912
10	360	864
11	340	816
12	320	768
13	300	720
14	280	672
15	260	624
16	240	576
17	240	576
18	220	528
19	220	528
20	200	480
21	200	480
22	180	432
23	180	432
24	180	432
25	160	384
26	160	384
27	160	384
28	160	384
29	140	336

Since there is one key for each logical data record in the file, the number of tracks required for the file index is easily calculated for a given number of logical records.

The index area also includes a disk track index if the file index requires 16 or more tracks. Figure B-2 shows the file sizes supported by one track of the disk track index for various key lengths.

Key-Length	File Capacity (Number of Logical Records) 5445	3340
1	1,040,400	5,992,704
2	705,600	4,064,256
3	518,400	2,985,984
4	409,600	2,359,296
5	313,600	1,806,336
6	250,000	1,440,000
7	211,600	1,218,816
8	176,400	1,016,064
9	144,400	831,744
10	129,600	746,496
11	115,600	665,856
12	102,400	589,824
13	90,000	518,400
14	78,400	451,584
15	67,600	389,376
16	57,600	331,776
17	57,600	331,776
18	48,400	278,784
19	48,400	278,784
20	40,000	230,400
21	40,000	230,400
22	32,400	186,624
23	32,400	186,624
24	32,400	186,624
25	25,600	147,456
26	25,600	147,456
27	25,600	147,456
28	25,600	147,456
29	19,600	112,896

Figure B-2. File Capacity for One Track of Disk Track Index

TRACK USAGE FOR INDEXED FILES

When loading an indexed file, you specify either the number of records in the file or the number of tracks. When you specify the number of tracks, the system determines how the specified space is to be split among data tracks, file index tracks, and disk track index tracks. Figure B-3 illustrates how the system splits an area on the 3340 when the TRACKS parameter is used in the OCL statement, and illustrates the maximum record capacities for various size indexed files. Remember to use the smaller of the two numbers: *number of keys,* or *number of logical data records.*

Number of Tracks	Key Length	Logical Record Length	Disk Track Index	Number of File Index Tracks	Number of Data Tracks	Number of Keys	Number of Logical Data Records
5	5	64		1	4	1,316	768
5	5	128		1	4	1,316	384
5	5	256		1	4	1,316	192
5	10	64		1	4	846	768
5	10	128		1	4	846	384
5	10	256		1	4	846	192
10	5	64		2	8	2,660	1,536
10	5	128		1	9	1,316	864
10	5	256		1	9	1,316	432
10	10	64		2	8	1,710	1,536
10	10	128		1	9	846	864
10	10	256		1	9	846	432
50	5	64		7	43	9,380	8,256
50	5	128		4	46	5,348	4,416
50	5	256		2	48	2,660	2,304
50	10	64		9	41	7,758	7,872
50	10	128		5	45	4,302	4,320
50	10	256		3	47	2,574	2,256
100	5	64		13	87	17,444	16,704
100	5	128		7	93	9,380	8,928
100	5	256		4	96	5,348	4,608
100	10	64	1	18	81	15,534	15,552
100	10	128		10	90	8,622	8,640
100	10	256		6	94	5,166	4,512
500	5	64	1	63	436	84,644	83,712
500	5	128	1	34	465	45,668	44,640
500	5	256	1	18	481	24,164	23,088
500	10	64	1	91	408	78,606	78,336
500	10	128	1	50	449	43,182	43,104
500	10	256	1	27	472	23,310	22,656
1,000	5	64	1	125	874	167,972	167,808
1,000	5	128	1	67	932	90,020	89,472
1,000	5	256	1	35	964	47,012	46,272
1,000	10	64	1	182	817	157,230	156,864

Figure B-3 (Part 1 of 2). Sample Record Capacities of Indexed Files

Number of Tracks	Key Length	Logical Record Length	Disk Track Index	Number of File Index Tracks	Number of Data Tracks	Number of Keys	Number of Logical Data Records
1,000	10	128	1	100	899	86,382	86,304
1,000	10	256	1	53	946	45,774	45,408
2,000	5	64	1	250	1,749	335,972	335,808
2,000	5	128	1	134	1,865	180,068	179,040
2,000	5	256	1	69	1,930	92,708	92,640
2,000	10	64	1	364	1,635	314,478	313,920
2,000	10	128	1	200	1,799	172,786	172,704
2,000	10	256	1	106	1,893	91,566	90,864
3,000	5	64	1	375	2,624	503,972	503,808
3,000	5	128	1	200	2,799	268,772	268,704
3,000	5	256	1	104	2,895	139,748	138,960
3,000	10	64	1	546	2,453	471,726	470,976
3,000	10	128	1	300	2,699	259,182	259,104
3,000	10	256	1	158	2,841	136,494	136,368
3,320	5	64	1	415	2,904	557,732	557,568
3,320	5	128	1	222	3,097	298,340	297,312
3,320	5	256	1	115	3,204	154,532	153,792
3,320	10	64	1	604	2,715	521,838	521,280
3,320	10	128	1	332	2,987	286,830	286,752
3,320	10	256	1	175	3,144	151,182	150,912

Figure B-3 (Part 2 of 2). Sample Record Capacities of Indexed Files

CORE INDEX

The core index is a table of pointers to the index tracks of an indexed file. The core index is built just before your COBOL or RPG II program is executed and contains, at most, one entry for each index track. Figure B-4 shows the number of bytes of main storage required for a core index that provides the most efficient random processing of an indexed file, using key length and number of logical records as variables.

Key Length	Number of Logical Records (in 1000s) 2	5	8	10	15	20	50
4	12	24	36	42	60	84	198
5	14	28	42	56	84	105	266
6	16	40	56	72	104	136	336
7	18	45	72	90	126	171	414
8	20	50	80	100	150	200	500
9	33	66	99	121	187	242	605
10	36	72	120	144	216	288	696
11	39	91	130	169	247	325	806
12	42	98	154	196	280	378	924
13	45	105	180	210	315	420	1,050
14	48	128	192	240	368	480	1,200
15	68	153	221	289	425	561	1,377
16	72	162	252	324	486	630	1,566
17	76	171	266	342	513	665	1,653
18	80	200	320	280	580	760	1,900
19	84	210	336	399	609	798	1,995
20	110	242	374	462	704	924	2,310

Figure B-4. Core Index Sizes in Bytes for 3340 Single Volume Indexed Files

CALCULATING FILE SIZES – SUMMARY

This section contains step-by-step explanations of some common calculations of file sizes.

Determining the Number of Tracks in a Sequential or Direct File

1. Number of logical records x logical record length = number of characters

2. $\frac{\text{Number of characters (from step 1)}}{\text{12,288 (number of characters/track)}}$ = number of tracks (round to the next highest whole number)

Determining the Number of Tracks in an Indexed File

To determine the number of data tracks in an indexed file, the following two steps should be used:

1. Number of logical records x logical record length = number of characters

2. $\frac{\text{Number of characters (from step 1)}}{\text{12,288 (number of characters/track)}}$ = number of data tracks (round to the next highest whole number)

The following four steps should then be followed to determine the number of file index tracks in an indexed file:

1. Key length + 4 = index length

2. $\frac{\text{256 (number of characters/record)}}{\text{Index length (from step 1)}}$ = number of entries per record (drop remainder)

3. $\frac{\text{Number of logical records}}{\text{Number of entries per record (from step 2)}}$ = number of records (round to the next highest whole number; then, add one record for a delimiter, and two or more additional records if you plan to add logical records to the file later)

4. $\frac{\text{Number of records (from step 3)}}{\text{48 (number of records/track)}}$ = number of index tracks (round to the next highest whole number)

Determining the Number of Tracks of Disk Track Index

If an indexed 3340 file has more than 15 index tracks (from step 4 in the preceding calculation), the file will have a disk track index in addition to the file index. The following two steps should be used to determine the number of tracks needed for the disk track index:

1. $\frac{\text{Number of index tracks (greater than 15)}}{\text{Number of entries per record (from step 2 of the preceding calculation)}}$ = number of records (round to the next highest whole number)

2. $\frac{\text{Number of records (from step 1)}}{\text{48}}$ = number of disk track index tracks (round to the next highest whole number)

The total number of tracks in a 3340 indexed file can be determined by adding the number of data tracks, the number of file index tracks, and the number of disk track index tracks.

Converting Cylinder/Track to Track Number

To convert cylinder/track to track number, multiply cylinder number by the number of tracks on each cylinder and add track number.

Examples:

5444

6/1 = cylinder/track
6 x 2 + 1 = 13
13 = track number

5445

6/1 = cylinder/track
6 x 20 + 1 = 121
121 = track number

3340

6/1 = cylinder/track
6 x 20 + 1 = 121
121 = track number

Converting Track Number to Cylinder/Track

To convert track number to cylinder/track, divide track number by the number of tracks on a cylinder. The quotient is the cylinder and the remainder is the track.

Examples:

5444

13 = track number
13 ÷ 2 = 6 (remainder 1)
6/1 is the cylinder/track

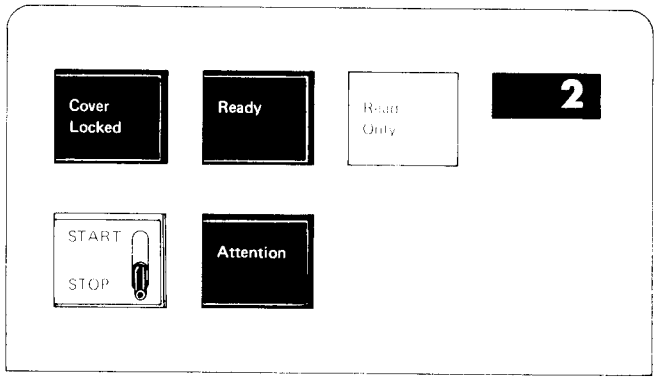
5445

121 = track number
121 ÷ 20 = 6 (remainder 1)
6/1 is the cylinder/track

3340

121 = track number
121 ÷ 20 = 6 (remainder 1)
6/1 is the cylinder/track

3340 OPERATOR PANEL



Ready Indicator

The Ready indicator lights when the data module is properly inserted and ready for operation.

START/STOP Switch

With the switch in the START position, the cover is locked, the data module is loaded, and the heads are moved to track 0, provided that:

1. The drive power is on.
2. The data module is in place.
3. The cover is closed and latched.

When the switch is placed in the STOP position, the data module is unloaded and the cover is unlocked. The data module will not unload while the system is communicating with the drive or when an Attention is pending.

Read Only Indicator

The Read Only indicator lights when the write function is inhibited after the data module is loaded. Write is inhibited by enabling the read only function on the data module (see *Read Only Function* in this appendix).

Cover Locked Indicator

The Cover Locked indicator lights when the drive cover is locked.

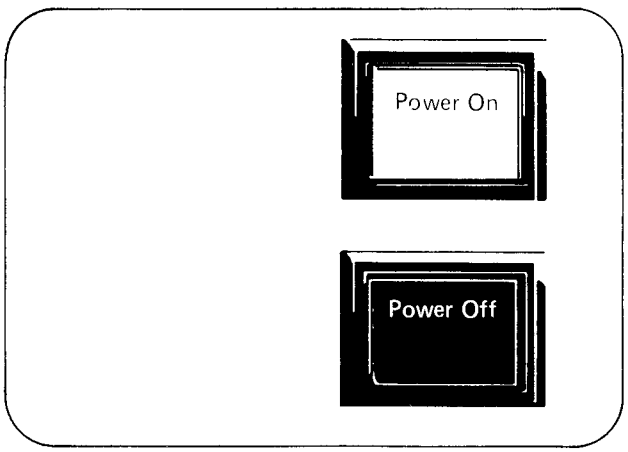
Attention Pushbutton

Operating the Attention pushbutton causes the drive to perform a rezero operation:

1. Read/write heads are moved to track 0.
2. DM (data module) Attention is signaled to the controller.
3. Drive is placed online and CE mode is reset (after completion of CE maintenance).

3340 POWER PANEL

3340 Model A2



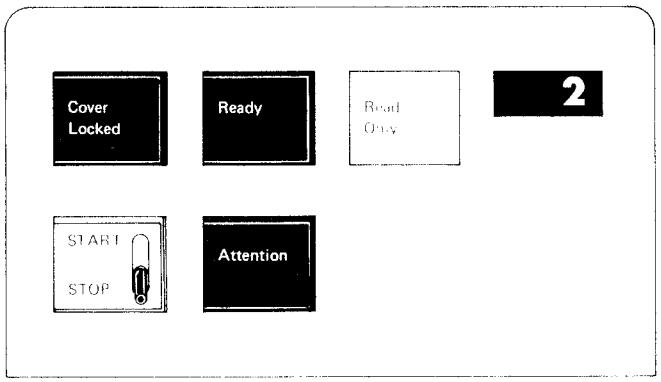
Power On Pushbutton

Operation of the Power On pushbutton allows ac power to be applied to the 3340 provided subsystem power is present. During a system power-up operation, it is not necessary to press this pushbutton.

Power Off Pushbutton

Operation of the Power Off pushbutton removes ac power from the 3340 when system power is up. During a system power-down operation, it is not necessary to press this push-button.

Note: Do not use Power On/Power Off switches to load or unload the data module.

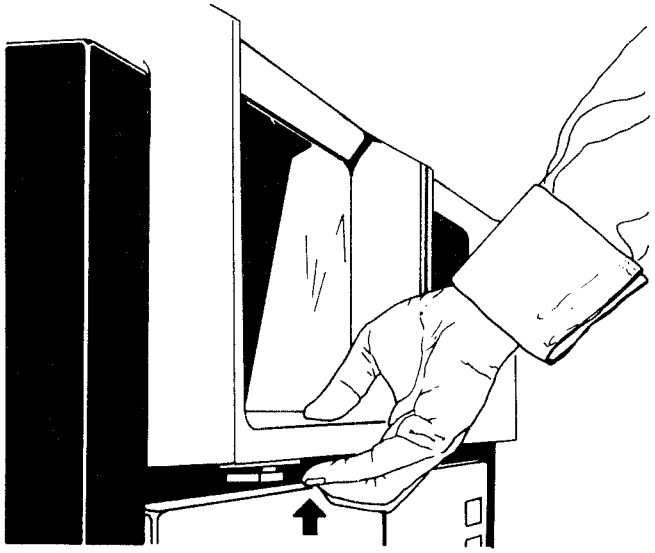


Operator Panel

DATA MODULE LOADING

1. Place the START/STOP switch in the STOP position.
2. Open the top cover.
3. With the data module doorway facing the rear of the drive, lower the data module into the drive shroud recess until it is seated.
4. Close the top cover.
5. Place the START/STOP switch in the START position. The Cover Locked indicator will light indicating the start of the load sequence.
6. The data module is automatically loaded. After 20 seconds, the Ready indicator will light to show the drive is ready. This also forces a pack change device end interrupt to the storage control.

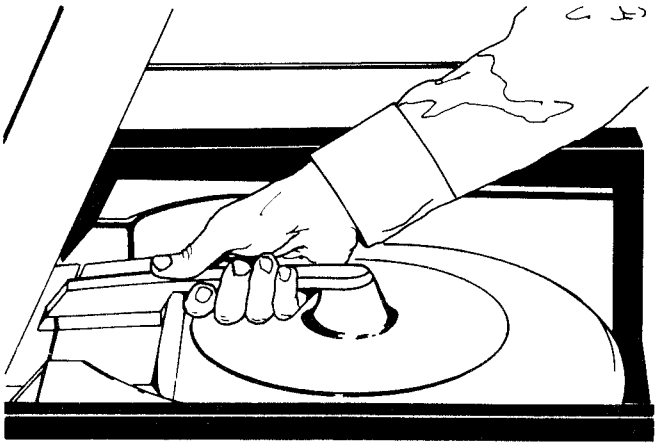
Note: Correct loading of the data module depends on: 1) proper alignment of the data module in the drive shroud recess, and 2) the 3340 top cover being fully closed.



Opening Cover

DATA MODULE UNLOADING

1. Place the START/STOP switch in the STOP position.
2. When the Cover Locked indicator goes out, open the top cover.
3. Lift the data module from the drive.



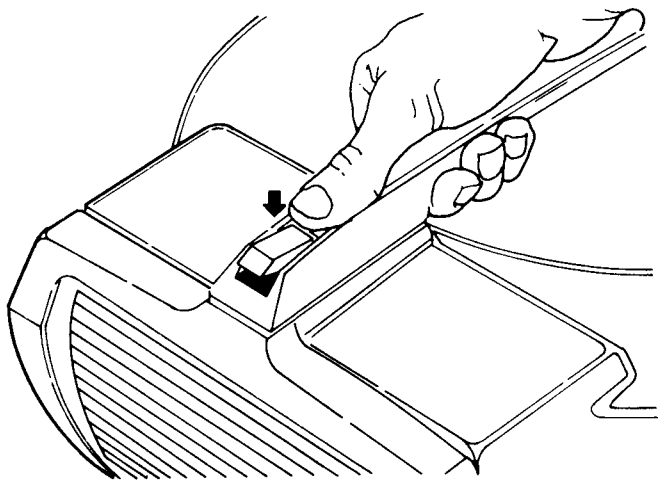
Load/Unload Data Module

READ ONLY FUNCTION

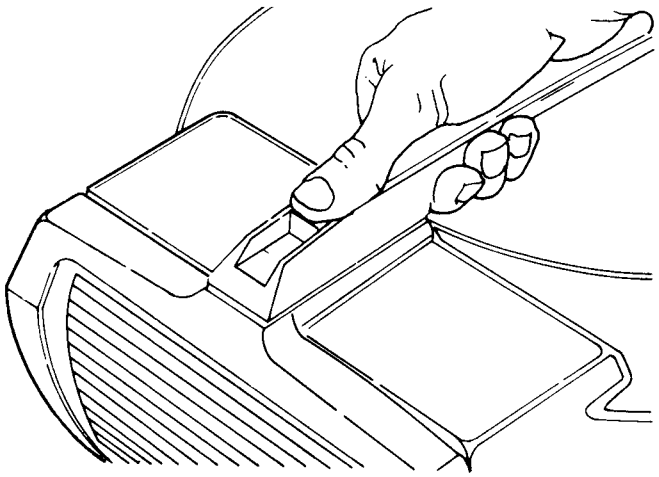
The means to protect previously written data modules is provided by the read only function. The following procedures show how to enable or disable the read only function.

Enable Read Only Function

1. Press down on the IBM logo inset of the handle.

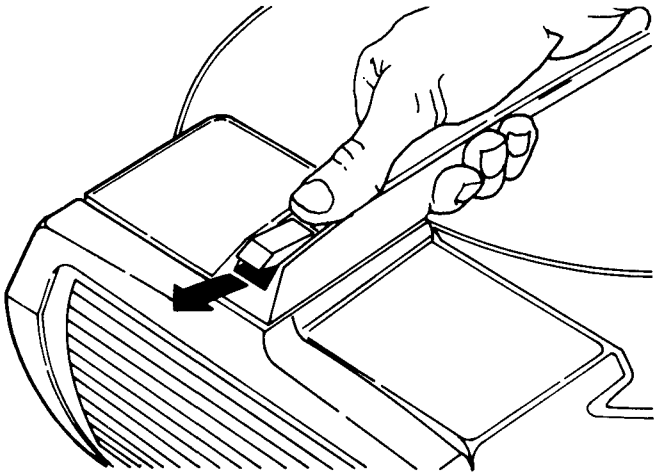


2. Turn inset 180° and snap into place until the hollow side of the inset is exposed as shown below.

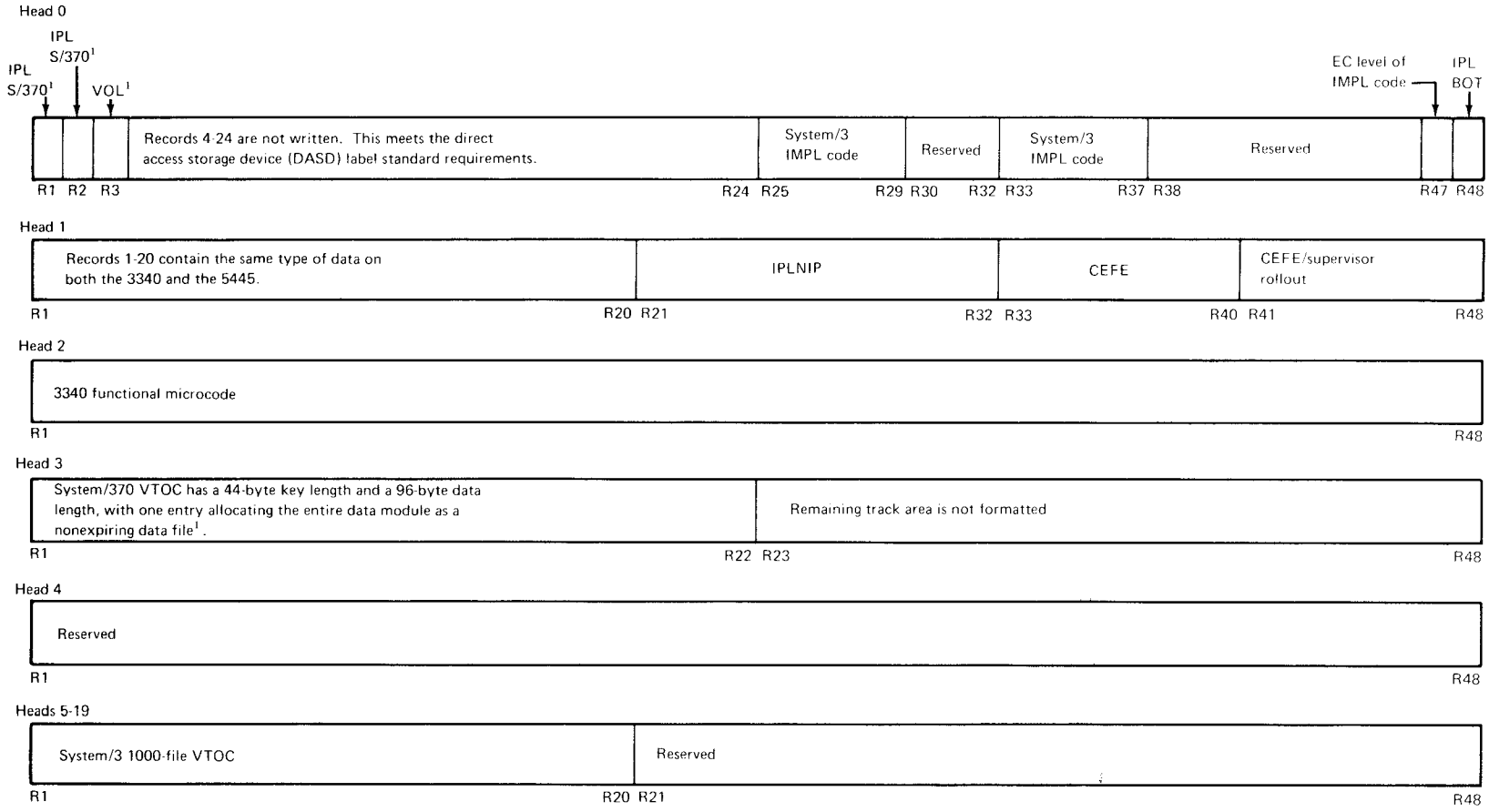


Disable Read Only Function

1. Press down on the IBM logo inset of the data module handle.



2. Turn inset 180° and snap into place so that the solid side of the inset is exposed as shown above.



[1] These areas are written in count-key-data format (standard data format) readable to System/3 and System/370. Other areas are written in compressed data format.

Appendix E. IBM System/3 Standard Character Set

Character	Hexadecimal Equivalent
Blank	40
¢	4A
.	4B
<	4C
(	4D
+	4E
|	4F
&	50
!	5A
$	5B
*	5C
)	5D
;	5E
¬	5F
- (minus)	60
/	61
,	6B
%	6C
_ (underscore)	6D
>	6E
?	6F
:	7A

Character	Hexadecimal Equivalent
#	7B
@	7C
' (apostrophe)	7D
=	7E
"	7F
A	C1
B	C2
C	C3
D	C4
E	C5
F	C6
G	C7
H	C8
I	C9
}	D0
J	D1
K	D2
L	D3
M	D4
N	D5
O	D6
P	D7

Character	Hexadecimal Equivalent
Q	D8
R	D9
S	E2
T	E3
U	E4
V	E5
W	E6
X	E7
Y	E8
Z	E9
0	F0
1	F1
2	F2
3	F3
4	F4
5	F5
6	F6
7	F7
8	F8
9	F9

access time: The time interval between the instant that transfer of data to/from the 3340 is requested and the instant that transfer begins.

alternate track: A track designated to contain data in place of a defective track.

data module: A sealed cartridge that contains disks, a spindle, read/write heads, and access arms. The IBM 3348 Model 70 Data Module is used on the 3340.

data surface: One side of a disk used for storage of data. The 3340 data module has six data surfaces.

defect skipping: The ability to write data before and after a defect on a disk surface. This ability of the 3340 reduces the chance of an alternate track being assigned and thus eliminates the access time required to move the read/write heads to the alternate track.

logical cylinder: Twenty tracks on the 3340 data module when used on the System/3.

main data area: The main area of the 3340 used for storage of files. Each data module has one main data area that can store 40.8 million bytes of data.

physical cylinder: All the tracks that can be accessed without repositioning the access mechanism. The 3340 data module has 12 tracks per physical cylinder.

read/write head: The electromagnet used to read, write, or erase data on a disk surface. On the 3340, there are two read/write heads for each data surface.

record: A fixed area of a track. Each record contains 256 bytes of data. There are 48 records per track on the 3340 data module. These records are similar to sectors on the 5444 Disk Storage Drive.

rotational delay: The time required for the desired record area to reach the read/write head so that data transfer can begin.

simulation area: An area on the 3340 data module that simulates a 5444 disk storage drive. There are four simulation areas on each 3340 data module.

track: The area accessible to one read/write head at one access position. There are 700 tracks on each data surface of a 3340 data module. Because there are two heads per data surface, two tracks on each surface are accessible at each access position.

READER'S COMMENT FORM

IBM System/3 Model 15
3340 Direct Access Storage Facility
Reference Manual

GC21-5111-1

YOUR COMMENTS, PLEASE . . .

Your comments assist us in improving the usefulness of our publications; they are an important part of the input used in preparing updates to the publications. All comments and suggestions become the property of IBM.

Please do not use this form for technical questions about the system or for requests for additional publications; this only delays the response. Instead, direct your inquiries or requests to your IBM representative or to the IBM branch office serving your locality.

Corrections or clarifications needed:

Page *Comment*

Due to the current paper shortage, we will not send a reply to your comments unless you check the box below.

I would like a reply. ☐

Name _____

Address _____

- Thank you for your cooperation. No postage necessary if mailed in the U.S.A.

GC21-5111-1

Fold

Fold

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

IBM Corporation
General Systems Division
Development Laboratory
Publications, Dept. 245
Rochester, Minnesota 55901

Fold

Fold

International Business Machines Corporation
General Systems Division
5775D Glenridge Drive N.E.
Atlanta, Georgia 30301
(USA Only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

Cut Along Line

IBM System/3 Model 15 3340 Direct Access Storage Facility ...erence Manual (S3-30) Printed in U.S.A. GC21-5111-1

IBM® Technical Newsletter

This Newsletter No.	GN21-5410
Date	24 September 1976
Base Publication No.	GC21-5111-1
File No.	S3-30
Previous Newsletters	GN21-5366

IBM System/3
3340 Direct Access
Storage Facility
Reference Manual

This technical newsletter, a part of version 5, modification 00 of the IBM System/3 Model 15 System Control Program (Program Number 5704-SC1), provides replacement pages for the subject publication. These replacement pages remain in effect for subsequent versions and modifications unless specifically altered. Pages to be inserted and/or removed are:

3-3, 3-4
3-4.1, 3-4.2 (added)
3-5, 3-6
3-9, 3-10
3-10.1, 3-10.2 (added)
3-11, 3-12
3-12.1, 3-12.2

Changes to text and illustrations are indicated by a vertical line at the left of the change.

Summary of Amendments

Miscellaneous updates

Note: Please file this cover letter at the back of the manual to provide a record of changes.

IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901

Printed in U.S.A.

Technical Newsletter

This Newsletter No. GN21-5580
Date 26 August 1977

Base Publication No. GC21-5111-1
File No. S3-30

Previous Newsletters GN21-5366
GN21-5410

IBM System/3 Model 15
3340 Direct Access Storage Facility
Reference Manual

This technical newsletter, a part of version 06, modification 00 of the IBM System/3 Model 15 System Control Program 5704-SC1, provides replacement pages for the subject publication. These replacement pages remain in effect for subsequent versions and modifications unless specifically altered. Pages to be inserted and/or removed are:

iii, iv
v, vi
2-1, 2-2
2-3 through 2-6 (deleted)
3-1, 3-2
3-3 through 3-36 (deleted)
B-1, B-2
B-5, B-6

Changes to text and illustrations are indicated by a vertical line at the left of the change.

Summary of Amendments

OCL FILE statement information and system service program descriptions have been removed from this manual. Refer to *IBM System/3 Model 15 System Control Programming Reference Manual*, GC21-5077 for complete OCL and system service programs information for the 3340.

Note: Please file this cover letter at the back of the manual to provide a record of changes.

IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901

Printed in U.S.A.

IBM

Technical Newsletter

This Newsletter No. GN21-5366
Date 5 December 1975

Base Publication No. GC21-5111-1
File No. S3-30

Previous Newsletters None

IBM System/3 Model 15
3340 Direct Access
Storage Facility
Reference Manual

This technical newsletter, a part of version 04, modification 00 of the IBM System/3 Model 15 System Control Program (Program Number 5704-SC1), provides replacement pages for the subject publication. These replacement pages remain in effect for subsequent versions and modifications unless specifically altered. Pages to be inserted and/or removed are:

1-1, 1-2
1-5, 1-6
2-3, 2-4
3-3, 3-4
3-9 through 3-12
3-12.1, 3-12.2 (added)
3-13, 3-14
3-19, 3-20
3-23 through 3-26
3-31, 3-32
A-1, A-2
B-1 through B-4

Changes to text and illustrations are indicated by a vertical line at the left of the change.

Summary of Amendments

- Updates 3340 simulation area program
- Miscellaneous updates

Note: Please file this cover letter at the back of the manual to provide a record of changes.

IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901

Printed in U.S.A.

GC21-5111-1

IBM System/3 Model 15 3340 Direct Access Storage Facility Reference Manual (S3-30) Printed in U.S.A. GC21-5111-1

IBM

International Business Machines Corporation
General Systems Division
5775D Glenridge Drive N.E.
Atlanta, Georgia 30301
(USA Only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)