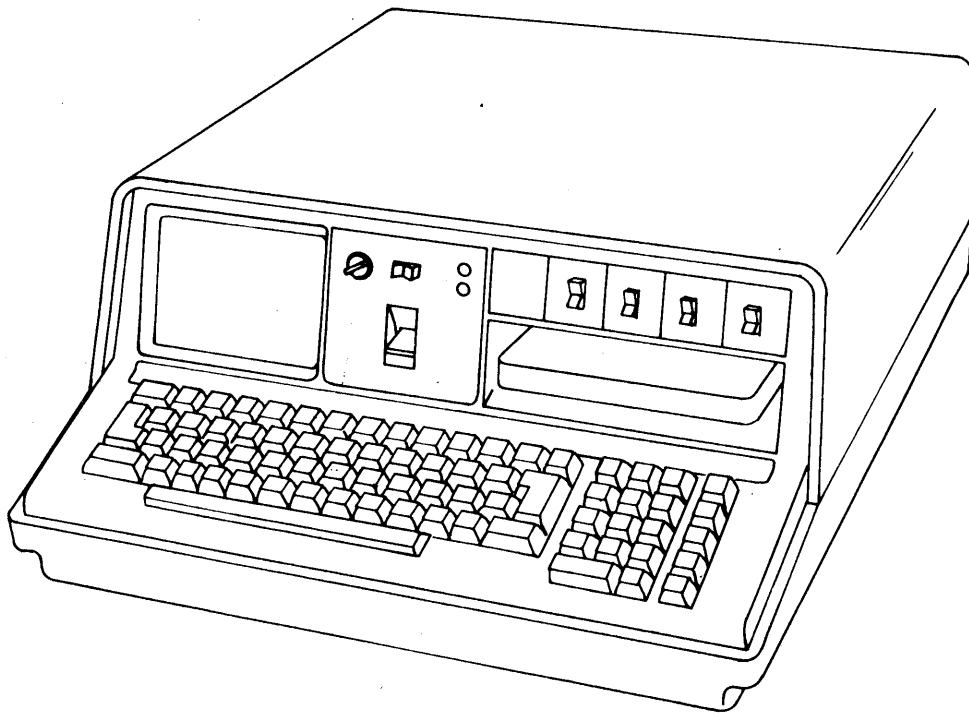




IBM 5110
Customer Support Functions
Reference Manual



5110

*IBM 5110
Customer Support Functions Reference Manual*

Preface

This manual is intended for users of the IBM 5110 who know how to operate their system and are familiar with the APL or BASIC language.

The users should read Chapter 1 for general information about the customer support functions. After becoming familiar with Chapter 1, they need only read the chapter that describes the function they want to use.

Note: This major revision *does not* apply to machines using a diskette sort IMF or the loadable diskette sort. If your machine requires a diskette sort IMF or the loadable diskette sort, use the previous documentation supplied with the machine.

Related Publications

- *IBM 5110 General Information and Physical Planning Manual, GA21-9300*
- *IBM 5110 APL Introduction, SA21-9301*
- *IBM 5110 APL User's Guide, SA21-9302*
- *IBM 5110 APL Reference Manual, SA21-9303*
- *IBM 5110 BASIC Introduction, SA21-9306*
- *IBM 5110 BASIC User's Guide, SA21-9307*
- *IBM 5110 BASIC Reference Manual, SA21-9308*
- *The IBM Diskette General Information Manual, GA21-9182*

Second Edition (October 1978)

This is a major revision of, and obsoletes, SA21-9311-0 and Technical Newsletters SN21-0280, SN21-0286, SN21-0288 and SN21-0296. Changes are periodically made to the information herein; before using this publication in connection with the operation of IBM systems, be sure you have the latest edition.

This publication contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental. Use this publication only for the purposes stated in the *Preface*.

Publications are not stocked at the address below. Requests for copies of IBM publications and for technical information about the system should be made to your IBM representative or to the branch office serving your locality.

This publication could contain technical inaccuracies or typographical errors. Use the Reader's Comment Form at the back of this publication to make comments about this publication. If the form has been removed, address your comments to IBM Corporation, Publications, Department 245, Rochester, Minnesota 55901. IBM may use and distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Contents

CHAPTER 1. INTRODUCTION	1	CHAPTER 9. LABEL DISPLAY	43
Overview	1	Operation	43
Diskette Initialization	1	Display Tape Labels	43
Copy Functions	2	Display Diskette Labels	44
Recovery Functions	2	Display Format	46
Label Display	2	CHAPTER 10. DISKETTE RECOVERY	49
Diskette Compress	3	Operation	49
Loader	3	Volume Label Recovery	50
Diskette Sort	3	Header Label Recovery	51
Operation	3	Data File Recovery	52
File Names	5	CHAPTER 11. TAPE DATA RECOVERY	53
Rules for File Names	5	Operation	53
CHAPTER 2. LOADER	7	CHAPTER 12. TAPE HEADER RECOVERY	55
Operation	7	Operation	55
CHAPTER 3. DISKETTE INITIALIZATION	9	CHAPTER 13. DISKETTE SORT FEATURE	57
Operation	10	Introduction to Sort	58
The INIT Command	12	Files	60
Hints	15	Input File	61
Diskette Format	16	Output File	62
Additional File Labels	17	Work File	63
Diskette Exchange Files	18	Control File	64
CHAPTER 4. DISKETTE-TO-DISKETTE COPY AND		Statements	67
IMAGE COPY	19	FILE Statement	67
Operation of Diskette-to-Diskette Copy	19	Sort Group Name Statement	68
One-Diskette-Drive Copy	20	Command Statement	69
Two-Diskette-Drive Copy	22	SORT Statement	69
Hints	23	MASK Statement	70
Operation of Image Copy	24	ALTS Statement	72
One-Diskette-Drive Copy	24	WORK Statement	74
CHAPTER 5. TAPE-TO-DISKETTE COPY	29	Operation	75
Operation	29	Control File	75
Hints	30	Keyboard Entry	76
File Types Copied	31	Stop	78
CHAPTER 6. DISKETTE-TO-TAPE COPY	33	Procedure File	78
Operation	33	Hints	78
Hints	34	Additional Information	79
File Types Copied	35	Performance Considerations	79
CHAPTER 7. TAPE-TO-TAPE COPY	37	Work File Storage Requirements	79
Operation	37	APPENDIX A: SORT WORK FILE SIZE CHARTS	83
One-Tape-Drive Copy	37	APPENDIX B. GLOSSARY	91
Two-Tape-Drive Copy	39	APPENDIX C. ERROR CODES	93
Hints	40	INDEX	107
CHAPTER 8. DISKETTE COMPRESS	41		
Operation	41		
Hints	42		

Chapter 1. Introduction

The functions described in this manual, except for sort, are supplied by IBM on a tape cartridge or diskette. The Diskette Sort feature is a hardware option and can be ordered as a feature. The following is a list of the functions, the file names, and the associated tape and diskette file numbers.

File Number	File Name	Description
1	LOADER	Loader
2	INITIAL	Diskette initialization
3	DDCOPY	Diskette-to-diskette copy
4	TDTCOPY	Tape-to-diskette copy
4	TDTCOPY	Diskette-to-tape copy
5	TTCOPY	Tape-to-tape copy
6	COMPRESS	Diskette compress
7	LDISPLAY	Label display
8	DRECOVER	Diskette recovery
9	TRECOVER	Tape data recovery
10	HRECOVER	Tape header recovery
11	IMF	IMF file
12	IMAGCOPY	Image copy

You can use one of the copy functions to copy the customer support functions from the IBM-supplied tape cartridge or diskette to one of your own. You can use any file number you want when you copy the customer support functions. The file numbers given in this manual refer to the IBM-supplied tape cartridge or diskette.

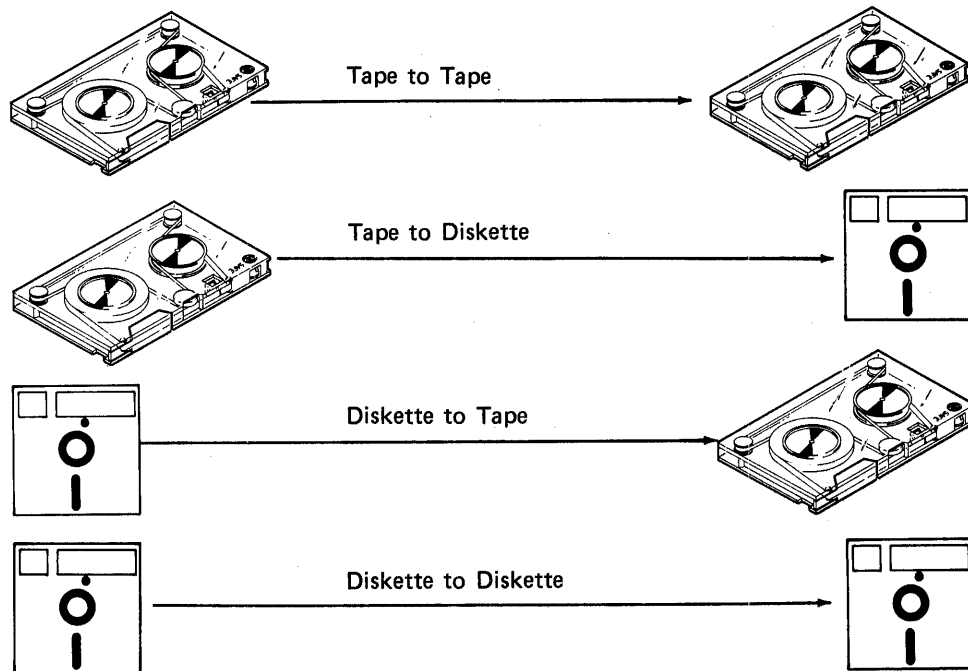
OVERVIEW

Diskette Initialization

The diskette initialization function formats a diskette. The number of bytes in a sector and the number of sectors in a track determine the format. A diskette must be formatted before it can be used. You may also use the diskette initialization function to change the format of a diskette. Changing the format can increase the capacity of a diskette and sometimes can improve the performance of programs that use a diskette file. Any data on a diskette is destroyed by the diskette initialization function.

Copy Functions

Four different copy functions are available. They provide for all possible combinations of copying between tapes and diskettes.



The copy functions enable you to make another copy of a file for security reasons or to use in testing a program. You should maintain an extra copy (backup) of all your important files. These copies would help you retain the data in the files should the original files be accidentally lost or destroyed. To test your programs, you should use a copy of the file and not the original file. In this way, if the program should fail, you would not have to recreate or correct your original file.

Recovery Functions

The recovery functions allow you to correct most read errors on tape and diskette data files, tape and diskette header labels and diskette volume labels.

Label Display

The label display function displays and prints the diskette and tape labels. This printout is useful when a read error occurs on the label portion of a file. It is a good idea to get a printout of labels on all your files occasionally so that you have a record of the contents of each diskette or tape.

Diskette Compress

The diskette compress function moves all files on the diskette as close to cylinder 0 as possible. When you free files and then mark new files on a diskette, space can be left between files. For example, you could free a 20K file in the middle of your diskette and then mark a 16K file. The 4K of space would not be used unless you marked a 4K file. The diskette compress function moves all the available space to the end of the diskette.

Loader

The loader function is used to load certain feature programs, such as the Serial I/O Adapter feature, into the system. The loader function is also used to maintain the language part of your system. Internal machine fixes (IMFs) are supplied by IBM to correct known problems and can be loaded into the machine with the loader.

Diskette Sort

The Diskette Sort feature of the IBM 5110 allows you to sort diskette record I/O files. This feature can be used to change the order of the records in a file so that a special report can be easily generated. The sort may also be used to put the input records for a program in order so that processing time is reduced. The Diskette Sort feature produces two types of output: standard (in which the records are sorted and written on the output file) and address out (in which the relative record number (addresses) of the sorted records are written on the output file).

OPERATION

The customer support functions on the tape or diskette are loaded into the machine with the link command. The syntax of the link command for APL is)LINK (device file number) [file name] and for BASIC it is LINK (file number), ['file name',] (device). After a function is loaded, it is automatically executed. The Diskette Sort feature is resident (if the feature is installed) within the machine and is executed when a)SORT (for APL) or UTIL SORT (for BASIC) command is executed. For BASIC, the system device is changed to the default value by all the functions except sort. The Diskette Sort feature does not change the system device when it is executed from a procedure file.

When the customer support functions are loaded, the ATTN key does not function the same as it does in the language mode. The ATTN key still clears the input line, but it will not stop the function when it is executing.

Display messages are provided to assist you in using the functions. Each time a message is displayed you must enter some information or just press the EXECUTE key to have the function continue. Anytime a function (except for sort) is waiting for you to respond to a message, you can enter a * (asterisk) to allow you to restart or terminate the function. The * must be entered in position 1 of the input line. For sort you can enter STOP in positions 1 through 4 of the input line to end the sort.

All customer support functions use the system's read/write storage (workspace for APL or workarea for BASIC). Therefore, any data you have in the machine should be saved before a function is used. Since IMFs and feature programs also use part of the system's read/write storage, at times there may not be enough room to load the customer support functions with the IMFs and feature programs loaded.

Several of the functions use keyword statements followed by parameters. The parameters must be entered in the order shown. Each parameter must be separated from the next by a comma. When a parameter is omitted, the comma must still be entered. For example:

Parameter 1 , Parameter 2 , Parameter 3
or
Parameter 1 , , Parameter 3 (when Parameter 2 is omitted)

When tape or diskette files are used, the file number must be specified. The file name need not be used. If a file name is specified for a diskette input file, the system checks to make sure the file name agrees with the file number on the diskette.

The customer support functions use a one-digit entry to indicate the tape and diskette drive. The entry used by the customer support functions and the equivalent APL or BASIC entry are:

Utility Entry	APL Entry	BASIC Entry	Physical Device
1	1	E80	Built-in tape
2	2	E40	Auxiliary tape
1	11	D80	Left drive, first unit
2	12	D40	Right drive, first unit
3	13	D20	Left drive, second unit
4	14	D10	Right drive, second unit

FILE NAMES

Certain rules must be followed when file names are used. The rules depend on the type of file, whether it is on tape or diskette, and the language that will use the file.

The term *nondata* as used in the following chart means that the files contain programs (an APL save file, for example). A simple name, as used in the chart, consists of up to 8 alphameric characters. The first character of the name must be uppercase alphabetic (A-Z). The other characters of the name may be alphameric (A-Z, 0-9). For example, DATA, HISTORY, and WEEK21 are all simple names. A complex name, as used in the chart, consists of two or more simple names separated by a period. Up to 17 characters, including periods, are allowed in a complex name. For example, PAYROLL.DATA, WEEK21.DATA, and DATA.TEST.R21.A are all complex names.

Rules for File Names

Diskette Files

APL (nondata)	Simple name.
APL (data)	Simple or complex name.
APL (basic exchange)	Simple name.
BASIC	Simple or complex name.
BASIC (basic exchange)	Simple name.

Tape Files

APL (nondata)	Up to 11 characters. Valid characters are alphameric (A-Z, 0-9) and underscored alphabetics. The first character cannot be numeric (0-9).
APL (data)	Up to 17 characters. All characters except the right parenthesis are valid.
BASIC	Up to 17 characters. All characters are valid.

Each of the remaining chapters in this manual describes a customer support function. You need to read only the chapter that covers the function you want to use. The error codes are shown in Appendix C.

The loader function is used to load feature programs and to maintain the language part of your system. The internal machine fixes (IMFs) supplied by IBM to correct a problem are loaded into the system with the loader. You must load IMFs before you load any feature programs.

OPERATION

The loader function is loaded via the link command and is on file 1 on the IBM-supplied tape cartridge or diskette. The file name is LOADER. After the function is loaded, the following is displayed:

```
LOADER FUNCTION
```

```
OPTIONS:
```

1. LOAD FROM TAPE
2. LOAD FROM DISKETTE

```
ENTER OPTION NUMBER AND PRESS EXECUTE
```

Enter a 1 if you are loading from tape, or a 2 if you are loading from a diskette. After the EXECUTE key is pressed, the following is displayed if you are loading from tape:

```
ENTER TAPE DRIVE NUMBER AND PRESS EXECUTE
```

```
ENTER FILE NUMBER AND PRESS EXECUTE
```

Or if you are loading from diskette the following is displayed:

```
ENTER DISKETTE DRIVE NUMBER AND PRESS EXECUTE
```

```
ENTER FILE NAME OR NUMBER AND PRESS EXECUTE
```

Enter the drive number (1 or 2 for tape; 1, 2, 3, or 4 for diskette) you will be using, then press the EXECUTE key.

To load the feature program from tape, enter the file number of the tape file. To load the feature program from diskette, enter either the file number or the file name of the diskette file. The feature program is loaded when the EXECUTE key is pressed. After the program is loaded, the following is displayed:

feature name LOADED

OPTIONS:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

ENTER OPTION NUMBER AND PRESS EXECUTE

The procedure to load an IMF is similar to loading a feature program. When the loader function determines that the tape or diskette file contains an IMF and not a feature program, a special message is displayed. How to use the loader after this message is displayed is explained in *IBM 5110 Language Support Maintenance Information Manual, SY31-0581*.

Chapter 3. Diskette Initialization

The diskette initialization function is used to initialize a diskette. The following are some reasons you may want to use this function:

- The diskette was exposed to a strong magnetic field, and the data can no longer be read.
- A defect has occurred in one or two cylinders, and you want to assign one or two of the spare cylinders to replace the defective cylinders.
- You want to change the format of the diskette (number of sectors per track and bytes per sector) to change the diskette capacity.
- You want to increase the area of a Diskette 2D type of diskette used for data set labels.

Diskette initialization does the following:

- Creates and writes the ERMAP label. This label is used to record the location of any defective cylinders. Up to two defective cylinders are allowed on each diskette. If more than two cylinders are defective, the diskette should not be used.
- Creates and writes a volume label on the diskette.
- Sets aside space for the data set labels.
- Allocates a file (file 1).
- Writes the ID field for each sector.
- Writes data and then reads it back on all tracks. This is done to locate any defective areas on the diskette. When a defective area is found, one of the spare cylinders is automatically assigned to replace the defective one.
- Writes the volume identifier and owner identifier on the diskette.

OPERATION

Diskette initialization is loaded via the link command and is on file 2 on the IBM-supplied tape cartridge or diskette. The file name is INITIAL. After the function is loaded, the following is displayed:

DISKETTE INITIALIZATION FUNCTION.

OPTIONS:

1. DESCRIPTION OF FUNCTION.
2. RETURN TO SYSTEM.
3. ENTER INITIALIZATION COMMAND.

ENTER OPTION NUMBER IN POSITION ONE AND PRESS EXECUTE TO CONTINUE.

When a 3 is entered, the following is displayed:

INIT VOL-ID, OWNER-ID, FORMAT, SEQUENCE NUMBER

ENTER APPROPRIATE INITIALIZATION COMMAND AND PRESS EXECUTE.

INIT _

An entry of 1 provides 3 displays that define the operation of the function. The following is displayed when a 1 is entered:

DISKETTE INITIALIZATION FUNCTION.

DISKETTE INITIALIZATION REFORMATS THE DISKETTE. ALL DATA AND PROGRAM FILES WILL BE ERASED. THEREFORE, BEFORE INITIATING THE DISKETTE INITIALIZATION FUNCTION, ENSURE THERE IS NO DATA IN ANY FILE ON THIS DISKETTE THAT WILL EVER BE USED AGAIN.

PRESS EXECUTE TO CONTINUE.

After the EXECUTE key is pressed, the first of three displays explaining the initialization command appears. Press the EXECUTE key after the first two displays. Enter the required parameters before pressing the EXECUTE key for the third display. The three displays are:

INIT COMMAND HAS THIS FORMAT:

INIT VOL-ID,OWNER-ID,FORMAT,SEQUENCE NUMBER

VOL-ID CAN BE UP TO SIX ALPHAMERIC CHARACTERS. ENTER COMMA IF DEFAULT VALUE (VOLID) IS DESIRED.

OWNER-ID CAN BE UP TO 14 ALPHAMERIC CHARACTERS. ENTER COMMA IF DEFAULT VALUE (OWNERID) IS DESIRED.

PRESS EXECUTE TO CONTINUE.

Press the EXECUTE key to get the second display.

FORMAT IS 1, 2, 3, 4, 5, 6, 7, 8 OR 9:

1, 2, AND 3 EQUAL DISKETTE SECTOR SIZES 128, 256, AND 512, RESPECTIVELY FOR DISKETTE 1.

4, 5, AND 6 EQUAL DISKETTE SECTOR SIZES 128, 256, AND 512, RESPECTIVELY FOR DISKETTE 2.

7, 8, AND 9 EQUAL DISKETTE SECTOR SIZES 256, 512, AND 1024, RESPECTIVELY FOR DISKETTE 2D.

FORMATS 1, 4, AND 7 ARE EXCHANGE FORMATS.

PRESS EXECUTE TO CONTINUE.

Press the EXECUTE key to get the third display.

SEQUENCE NUMBER IS OPTIONAL. DEFAULT IS SEQUENTIAL SECTOR SEQUENCE. THIS FIELD IS A TWO DIGIT FIELD THAT CAN BE USED TO SPECIFY A SECTOR SEQUENCE INCREMENT THAT RESULTS IN A NONSEQUENTIAL SECTOR SEQUENCE ON THE DISKETTE.

INIT VOL-ID,OWNER-ID,FORMAT,SEQUENCE NUMBER

ENTER APPROPRIATE INITIALIZATION COMMAND AND PRESS EXECUTE.

INIT ...

Enter the parameters for the INIT command, then press the EXECUTE key.

The INIT Command

The INIT (initialization) command specifies the volume identification, owner identification, format, and sector sequence.

Syntax

INIT [vol-id],[owner-id],format[,sequence-number]

where:

INIT is the command and must be in positions 1 through 4.

vol-id, (optional) is the volume identifier field. A maximum of 6 alphameric characters is allowed in this field. If this field is left blank and only the comma is entered, the default name of VOLID is used as the volume identifier.

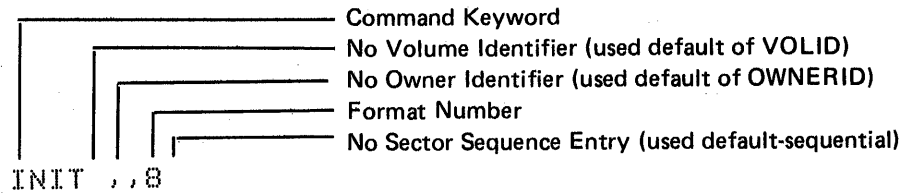
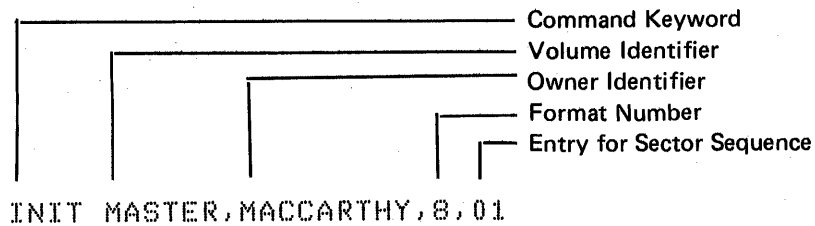
owner-id, (optional) is the owner identifier field. A maximum of 14 alphameric characters is allowed in this field. If this field is left blank and only the comma is entered, the default name of OWNERID is used as the owner identifier.

format, specifies the format for the diskette. The valid entries are 1 through 9. See *Diskette Format*, which follows, for more information on the formats.

sequence-number

(optional) specifies a sector numbering sequence other than sequential. The number in this field is used as a fixed value when the sector numbers are incremented on the diskette. See *Diskette Format* for more information on sector sequence. If this field is left blank, the default value of 01 (sequential sector numbering) is assumed.

The following are examples of the INIT command.



When format 7, 8, or 9 is specified for the INIT command, the following is displayed:

ADDITIONAL FILE LABELS CAN BE PLACED ON THIS DISKETTE. THE NUMBER OF FILE LABELS CAN BE FROM 71 TO 1007. ENTER ONE OF THE FOLLOWING AT POSITION ONE AND PRESS EXECUTE.

OPTIONS:

BLANK	ALLOTS 71 FILES	4	ALLOTS 487 FILES	7	ALLOTS 799 FILES
1	ALLOTS 175 FILES	5	ALLOTS 591 FILES	8	ALLOTS 903 FILES
2	ALLOTS 279 FILES	6	ALLOTS 695 FILES	9	ALLOTS 1007 FILES
3	ALLOTS 383 FILES				

Additional space for file labels may now be specified. The valid entries are blank and digits 1 through 9. The number of labels and the diskette space used by each entry follows.

Entry	Number of Labels	Cylinders Number(s) Used
blank	71	0
1	175	0 and 1
2	279	0 through 2
3	383	0 through 3
4	487	0 through 4
5	591	0 through 5
6	695	0 through 6
7	799	0 through 7
8	903	0 through 8
9	1007	0 through 9

After the EXECUTE key is pressed, the following is displayed:

```
INSERT DISKETTE TO BE INITIALIZED IN DESIRED DRIVE.  ENTER ONE
DIGIT DRIVE NUMBER TO BE USED IN THE FIRST POSITION AND
PRESS EXECUTE TO CONTINUE.
```

Insert the diskette to be initialized in a drive. Enter the one-digit diskette drive number (1, 2, 3, or 4) for that diskette drive. Then press the EXECUTE key. After initialization is complete, and if no defective cylinders were found, the following is displayed:

```
DISKETTE INITIALIZATION COMPLETE.
```

```
THERE ARE NO DEFECTIVE CYLINDERS ON THIS DISKETTE.
```

```
PRESS EXECUTE TO CONTINUE
```

If defective cylinders were found, the following is displayed:

```
DISKETTE INITIALIZATION COMPLETE.  
THE FOLLOWING CYLINDERS ARE DEFECTIVE:
```

```
XX
```

```
PRESS EXECUTE TO CONTINUE.
```

The cylinder number, indicated by XX, of any defective cylinder is displayed. These numbers are recorded in the ERMAP label located at cylinder 0 sector 5. Up to two defective cylinders are allowed on each diskette. The diskette must be replaced if more than two cylinders are defective. The diskette must also be replaced if cylinder 0 is defective. Cylinder 0 is used by the system to record information about the diskette.

Hints

- Alphameric characters must be used for the volume and owner identifier fields. A maximum of 6 characters can be used for the volume identifier and 14 characters for the owner identifier.
- Error codes are displayed if any of the following conditions are found when the function is checking the header label:
 - Secure data set
 - Write-protected data set
 - Data set containing data
 - Nonblank expiration date field

Once one of these conditions is encountered, the error code is displayed and the option can be selected. If you elect to continue, no further checking of the labels is performed.

- Initialization can terminate unsuccessfully if an I/O error is encountered, if more than two defective cylinders were encountered, if cylinder 0 is defective, or if any extended label cylinder is defective.

DISKETTE FORMAT

Nine different diskette formats are available. The following chart shows the number of tracks per cylinder, sectors per track, bytes per sector, and the valid entries for the sequence-number field of the INIT command.

Diskette Type	Format	Tracks per Cylinder	Sectors per Track	Bytes per Sector	Capacity Tracks 1-74	Valid Sector Sequence Number
1	1*	1	26	128	246,272	01-13
	2*	1	15	256	284,160	01-08
	3**	1	8	512	303,104	01-04
2 or 2D	4*	2	26	128	492,544	01-13
	5*	2	15	256	568,320	01-08
	6**	2	8	512	606,208	01-04
	7*	2	26	256	985,088	01-13
	8	2	15	512	1,136,640	01-08
	9***	2	8	1024	1,212,416	01-04

* Exchange format (see *Diskette Exchange Files*)

** Maximum capacity

*** Can only be used for data files

Diskettes obtained from IBM have a colored label that shows the diskette type. The color of the label indicates the sector size used when IBM initialized the diskette.

Label Color	Sector Size
White with black print	128
Red with white print	256
Blue with white print	512
Yellow with black print	1024

Format 1, 2, or 3 can be used only with a Diskette 1 type of diskette. Formats 4 through 9 can be used with a Diskette 2 or Diskette 2D type of diskette. Diskette 1 diskettes are designed for single-side recording; Diskette 2 and Diskette 2D diskettes use both sides for recording. The higher the format number, the more data that can be recorded on the diskette.

The sector sequence number column contains the valid entries for the sequence-number field of the INIT command. This field modifies the normal sequence numbering of the sectors on the diskette. For example, assume format 1 and a sector sequence number of 02. The sectors would be numbered as follows: 1, 3, 5, 23, 25, 2, 4, 24, 26. For a format 1 and sector sequence number of 13, the sectors are alternately numbered as follows: 1, 14, 2, 15, 12, 25, 13, 26. This may enable you to read sector 1, process the data, and get ready to read sector 2 before the diskette rotates past sector 2. If the sectors were numbered sequentially, this type of processing would not be possible.

You can reduce the time required to search a key file by changing the sector numbering sequence. If you have a key larger than 20 characters, the internal processing to compare the key could take long enough to prevent the reading of the next sector. To provide sufficient time to compare the key and get ready to read the next sector, alternately number the sectors on the diskette. To alternately number the sectors, specify the largest possible sector sequence number. For format 1, 4, or 7 specify 13; for format 2, 5, or 8 specify 08; and for format 3, 6, or 9 specify 04. Other files on the diskette may process slower as a result of the sector numbering being changed.

Additional File Labels

A file label is required for each file you have on the diskette. Normally, cylinder 0 starting at sector 8 is used for labels. A Diskette 1 type of diskette (formats 1, 2, and 3) has room for 19 labels. A Diskette 2 type of diskette (formats 4, 5, and 6) has room for 45 labels. A Diskette 2D type of diskette (formats 7, 8, and 9) has room for 71 labels on cylinder 0.

Additional cylinders may be used for labels with format 7, 8, or 9. Each additional cylinder provides room for 104 additional labels. You should consider providing additional space for labels if you plan to use many small files on the diskette. A cylinder assigned for additional labels cannot be used for data. Therefore, assigning additional cylinders for labels reduces the number of cylinders that can be used for data and thus reduces the capacity of the diskette. Also, with an extended label area, some diskette operations such as mark can take longer.

Diskette Exchange Files

Diskette exchange files can be read and created with the 5110. Format 1, 4, or 7 diskettes must be used for exchange files. Formats 1 and 4 are used for basic exchange files, and normally format 7 for H-type interchange files. Basic exchange files use 128-byte sectors. H-type exchange files use 256-byte sectors. The exchange files:

- Must have a simple name
- Must have unblocked, unspanned record format (record length must not exceed the sector size)
- Must use a sequential access method
- For format 1 diskettes, must use sequential sector numbering and have the file between cylinders 1 and 73
- For formats 4 and 7, need not use sequential sector sequence numbering and can use cylinders 1 through 74
- For format 7, must have a file number from 1 to 71 (data set label must not be in the extended label area)

Chapter 4. Diskette-to-Diskette Copy and Image Copy

The diskette-to-diskette copy function copies an entire input diskette to an output diskette, copies a group of files from an input file to an output file, or copies a selected single input file to an output file. When the entire diskette is copied, the output diskette will be automatically compressed. When a single file is copied, you must mark the output file before the copy operation is started. When an entire diskette or a group of files from a diskette is copied, the format number of the output diskette must be the same as or larger than the format number of the input diskette, and the sector size of both diskettes must be the same. The format number represents the capacity (density) of the diskette. The higher the number the greater the capacity.

Your image copy function copies an entire input (copied from) diskette to an output (copied to) diskette without compressing space that may have been left between files. This function provides an exact copy of your input diskette for backup files and testing purposes. When more than 25 files are copied, image copy will be faster than diskette-to-diskette copy.

OPERATION OF DISKETTE-TO-DISKETTE COPY

The diskette-to-diskette copy function is loaded via the link command and is on file 3 on the IBM-supplied cartridge or diskette. The file name is DDCOPY. After the function is loaded, the following is displayed:

```
DISKETTE TO DISKETTE COPY FUNCTION
```

```
OPTIONS ARE:
```

1. COPY ONE FILE
2. COPY ALL FILES
3. COPY A GROUP OF FILES

```
SELECT OPTION, PRESS EXECUTE
```

Enter a 2 to copy the entire input diskette to the output diskette. Enter a 3 to copy a group of files from the input diskette to the output diskette. Enter a 1 to copy a single input file to an output file. To select the number of drives you are going to use, the following is displayed:

```
HOW MANY DISKETTE DRIVES WILL YOU BE USING
```

```
OPTIONS ARE:
```

1. ONE DISKETTE DRIVE
2. TWO DISKETTE DRIVES

```
SELECT OPTION, PRESS EXECUTE
```

Enter a 1 or 2 to select the number of the diskette drives.

One-Diskette-Drive Copy

If you select the one-diskette-drive option, the following is displayed:

```
INSERT DISKETTE BEING COPIED FROM  
INTO SELECTED DISKETTE DRIVE  
ENTER DRIVE NUMBER, PRESS EXECUTE
```

Enter a 1, 2, 3, or 4 to select the diskette drive you will be using. Then insert the diskette used for input into that drive and press the EXECUTE key.

The following is then displayed if you are copying a single file:

```
ENTER FILE NUMBER TO BE COPIED FROM  
  
ENTER FILE NUMBER TO BE COPIED TO
```

Enter the file number you want to use. The file number can have up to four numeric digits.

If you are copying a group of files, the following is displayed:

```
ENTER STARTING FILE NUMBER TO BE COPIED FROM  
  
ENTER ENDING FILE NUMBER TO BE COPIED FROM  
  
ENTER STARTING FILE NUMBER TO BE COPIED TO
```

Enter the file number you want to use. The file number can have up to four numeric digits. The output file will be marked by the copy function.

When the copy operation is completed, the following is displayed:

DISKETTE COPY COMPLETED

OPTIONS ARE:

1. RESTART COPY FUNCTION
2. RETURN TO THE SYSTEM

ENTER OPTION, PRESS EXECUTE

Hints

- When an entire diskette or a group of files is copied:
 - The sector size of the input and output diskettes must be the same.
 - The format number of the output diskette must be the same or larger than the input diskette.
 - The label area on the output diskette must be the same size or larger than the input diskette.
 - The ending file number to be copied from must be equal to or greater than the starting file number to be copied from.
- When copying a single file, make sure the output file:
 - Is marked
 - Does not contain valuable data
 - Is large enough to contain all the data on the input file
 - Is not write- or date-protected
- When entering a file number, you can start the number in any position. Up to four digits may be entered for the file number.
- When entering a file name for the output file, make sure the name conforms to the rules for simple or complex names.
- When an entire diskette is copied, the contents of the input diskette may not fit on the output diskette if the input diskette files were created on a different system. The 5110 allocates files in increments of 1K (1024 bytes); other systems may allocate files differently. For example, assume that the input diskette had a sector size of 128 bytes per sector, with a file containing only one sector. When this file is copied, the copy function will mark a 1K output file. This 1K file contains eight 128-byte sectors; the first sector contains data, and the last seven are empty. If the input diskette is full or nearly full, and has several files in which the preceding conditions exist, the output diskette may be filled before all files are copied from the input diskette.
- When an entire diskette is copied, all files on the output diskette are freed prior to the copy procedure. Make sure the output file does not contain important data.

OPERATION OF IMAGE COPY

The image copy function is loaded via the link command and is on file 12 on the IBM-supplied cartridge or diskette. The file name is IMAGCOPY. After the function is loaded, the following is displayed:

IMAGE COPY FUNCTION

HOW MANY DISKETTE DRIVES WILL YOU BE USING

OPTIONS ARE:

- 1.ONE DISKETTE DRIVE
- 2.TWO DISKETTE DRIVES

ENTER OPTION NUMBER AND PRESS EXECUTE

Enter a 1 or a 2 to select the number of diskette drives.

One-Diskette-Drive Copy

If you select the one-diskette-drive option, the following is displayed:

INSERT DISKETTE BEING COPIED FROM

INTO SELECTED DISKETTE DRIVE

ENTER DRIVE NUMBER AND PRESS EXECUTE

Enter a 1, 2, 3, or 4 to select the diskette drive you will be using. Then insert the diskette used for input into that drive and press the EXECUTE key.

When files containing data to be copied are found on the input diskette, data is read into the system's read/write storage. When the read/write storage is filled, the following is displayed:

INSERT DISKETTE BEING COPIED TO

PRESS EXECUTE

You should now remove the diskette used for input from the drive. Then insert the diskette to be used for output and press the EXECUTE key.

The data in read/write storage is then written to the output diskette, and the following is displayed:

```
INSERT DISKETTE BEING COPIED FROM
PRESS EXECUTE TO CONTINUE
```

You should now remove the output diskette from the diskette drive and insert the input diskette.

This procedure (reading, changing diskettes, writing, changing diskettes again) continues until all data is copied. When the copy operation is completed, the following is displayed:

```
IMAGE COPY COMPLETE
```

```
OPTIONS ARE:
```

- 1.RESTART FUNCTION
- 2.RETURN TO SYSTEM

```
ENTER OPTION NUMBER AND PRESS EXECUTE
```

If the diskette type, record length or label extension fields differ between the input diskette and the output diskette, the following is displayed:

	INPUT	OUTPUT
RECORD LENGTH	XXX	XXX
DISKETTE TYPE	XX	XX
EXTENDED LABELS	X	X

```
OPTIONS ARE:
```

- 1.RESTART FUNCTION
- 2.RETURN TO SYSTEM

```
ENTER OPTION NUMBER AND PRESS EXECUTE
```

```
COPY ERROR 415
```

You must now initialize the output diskette to the same format as the input diskette, or select another diskette for output that has the same format as the input diskette.

If there are marked files on the output diskette, the following is displayed:

DISKETTE BEING COPIED ONTO ALREADY MARKED

OPTIONS ARE:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

ENTER OPTION NUMBER AND PRESS EXECUTE

You must now delete all marked files on the output diskette, or select another diskette for output that has the same format as the input diskette and contains no marked files.

If an error occurs while data is being copied onto the diskette, the function attempts to mark the file in which the error occurred as unused. If the file cannot be marked as unused, the following is displayed:

COPY TERMINATED BECAUSE OF I/O ERROR

FILE XXXX IS INVALID ON OUTPUT DISKETTE

OPTIONS ARE:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

ENTER OPTION NUMBER AND PRESS EXECUTE

DISKETTE ERROR XX ON DRIVE X

During read/write header execution, the image copy will allow 10 files to be in error. If 10 files or less are in error, the file numbers in error are displayed within the COPY COMPLETE message:

IMAGE COPY COMPLETE

HEADERS FOR FILES LISTED ARE INVALID

XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX

OPTIONS ARE:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

ENTER OPTION NUMBER AND PRESS EXECUTE

If more than 10 files are in error when the headers are being copied onto the diskette, the following is displayed:

COPY TERMINATED BECAUSE OF I/O ERROR

THE HEADERS FOR THE LISTED FILES ARE INVALID
XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX XXXX
ON THE OUTPUT DISKETTE

OPTIONS ARE:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

ENTER OPTION NUMBER AND PRESS EXECUTE

DISKETTE ERROR XX ON DRIVE X

Two-Diskette-Drive Copy

If you selected the two-diskette-drive option, the following is displayed:

INSERT DISKETTE BEING COPIED FROM
INTO SELECTED DISKETTE DRIVE

ENTER DRIVE NUMBER AND PRESS EXECUTE

INSERT DISKETTE BEING COPIED TO
INTO SELECTED DISKETTE DRIVE

ENTER DRIVE NUMBER AND PRESS EXECUTE

Enter the number of the diskette drives you want to use for input and output.
Enter a 1, 2, 3, or 4 to select the diskette drive you will be using.

When the copy operation is completed, the following is displayed:

IMAGE COPY COMPLETE

OPTIONS ARE:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

ENTER OPTION NUMBER AND PRESS EXECUTE

Note: The format, marked file, and input/output error messages previously discussed for the one-diskette-drive system apply to the two-diskette-drive system.

Hints

- The format of the output diskette must be equal to the format of the input diskette.
- The output diskette must not contain any marked files.

Chapter 5. Tape-to-Diskette Copy

Tape-to-diskette copy and the diskette-to-tape copy are the same function. The diskette-to-tape copy is explained in Chapter 6. A chart at the end of this chapter shows the file types that can be copied. The file name of the tape input file is assigned to the diskette output file. However, if the file name of the tape input file does not conform to the rules for simple or complex names, a valid file name must be entered. The diskette output file must be marked before the copy operation is started.

OPERATION

Tape-to-diskette copy is loaded by the link command and is on file 4 on the IBM-supplied tape cartridge and diskette. The file name is TDTCOPY. After the function is loaded, the following is displayed:

```
TAPE TO DISKETTE, DISKETTE TO TAPE COPY FUNCTION
```

```
OPTIONS ARE:
```

1. TAPE TO DISKETTE COPY.
2. DISKETTE TO TAPE COPY.

```
ENTER OPTION, PRESS EXECUTE.
```

Enter a 1 and press the EXECUTE key to select the tape-to-diskette copy. The following is then displayed:

```
INSERT TAPE INTO TAPE DRIVE YOU WILL BE USING
```

```
ENTER DRIVE NUMBER, PRESS EXECUTE
```

```
INSERT DISKETTE INTO DISKETTE DRIVE YOU WILL BE USING
```

```
ENTER DRIVE NUMBER, PRESS EXECUTE
```

Enter a 1 or 2 to select the tape drive you are using for the input file. Enter a 1, 2, 3, or 4 to select the diskette drive you are using for the output file. After the EXECUTE key is pressed, the following is displayed:

ENTER TAPE FILE NUMBER TO BE COPIED FROM

ENTER DISKETTE FILE NUMBER TO BE COPIED TO

Enter the number of the tape file you want to copy from. The file number can be up to four numeric digits. Enter the number of the diskette file you want to copy to. The file number can be up to four numeric digits. After the EXECUTE key is pressed, the copy operation begins. When the copy operation is completed, the following is displayed:

COPY COMPLETED

OPTIONS ARE:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

SELECT OPTION, PRESS EXECUTE

Hints

- Before trying to copy a tape file, check the file type and make sure that file type can be copied.
- When entering a file number, you can start the number in any position. Up to four digits may be entered for the file number.
- When entering a file name for the output file, make sure the name conforms to the rules for simple or complex names.
- Do not use a format 9 (1024-byte sector) diskette.
- Make sure the diskette output file:
 - Is marked
 - Is not write-protected
 - Does not contain valuable data
 - Is large enough to contain all the data on the input file

FILE TYPES COPIED

Not all file types can be copied with the tape-to-diskette copy function. Attempting to copy some types of files causes an error. The following chart summarizes the tape files that can be copied.

Tape File Type	System	File Name	Copied	Error	Diskette File Type
0	3	Marked and unused	No	357	
1	1	General exchange file	Yes		2
2	3	General exchange file or BASIC source	Yes		2
3	1	BASIC source file	Yes		2
4	1	BASIC work area file	No	358	
5	1	BASIC KEYS file	No	358	
6	1	APL continue file	No	359	
7	3	APL save file	Yes		7
8	3	APL internal data	Yes		8
9	2	Record I/O file	Yes		9
11	2	BASIC save file	Yes		11
12	2	BASIC KEYS file	Yes		12
16	1	Patch, tape recovery, and tape copy file	No	361	
17	1	Diagnostic file	No	361	
18	1	Communication file	No	361	
19	1	IMF file	No	361	
21	2	Function file	Yes		21
22	2	Feature file	Yes		22
23	2	IMF file	Yes		23
24	2	Diagnostic file	Yes		24
26	2	APL continued file	No	359	

System

1 = 5100

2 = 5110

3 = 5100 and 5110

Chapter 6. Diskette-to-Tape Copy

Diskette-to-tape copy and the tape-to-diskette copy are the same function. The tape-to-diskette copy is explained in Chapter 5. A chart at the end of this chapter shows the file types that can be copied. The file name of the diskette input file is assigned to the output tape file. The output media must be marked before the copy operation is started.

OPERATION

Diskette-to-tape copy is loaded via the link command and is on file 4 on the IBM-supplied tape cartridge or diskette. The file name is TDTCOPY. After the function is loaded, the following is displayed:

```
TAPE TO DISKETTE, DISKETTE TO TAPE COPY FUNCTION
```

```
OPTIONS ARE:
```

1. TAPE TO DISKETTE COPY.
2. DISKETTE TO TAPE COPY.

```
ENTER OPTION, PRESS EXECUTE.
```

Enter a 2 to select the diskette-to-tape copy. After the EXECUTE key is pressed, the following is displayed:

```
INSERT TAPE INTO TAPE DRIVE YOU WILL BE USING
```

```
ENTER DRIVE NUMBER, PRESS EXECUTE
```

```
INSERT DISKETTE INTO DISKETTE DRIVE YOU WILL BE USING
```

```
ENTER DRIVE NUMBER, PRESS EXECUTE
```

Enter a 1 or 2 to select the tape drive you are using for the output file. Enter a 1, 2, 3, or 4 to select the diskette drive you are using for the input file. After the EXECUTE key is pressed, the following is displayed:

ENTER DISKETTE FILE NUMBER TO BE COPIED FROM

ENTER TAPE FILE NUMBER TO BE COPIED TO

Enter the number of the diskette file you want to copy from. The file number can be up to four numeric digits. Enter the number of the tape file you want to copy to. The file number can be up to four numeric digits. After the EXECUTE key is pressed, the copy operation begins. When the copy operation is completed, the following is displayed:

COPY COMPLETED

OPTIONS ARE:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

SELECT OPTION, PRESS EXECUTE

Hints

- Before trying to copy a diskette file, check the file type and make sure that file type can be copied.
- When entering a file number, you can start the number in any position. Up to four digits may be entered for the file number.
- Make sure the tape output file:
 - Is marked
 - Does not contain valuable data
 - Is large enough to contain all the data on the input file

FILE TYPES COPIED

Not all file types can be copied with the diskette-to-tape copy. Attempting to copy some file types causes an error. The following chart summarizes diskette files that can be copied.

Diskette File Type	File	Copied	Error	Tape File Type
0	Marked and unused	No	366	
2	General exchange file	Yes		2
7	APL save file	Yes		7
8	APL internal data	Yes		8
9	Record I/O file	Yes		9
B9	Basic exchange	Yes		9
10	APL internal data	No	365	
11	BASIC save file	Yes		11
12	BASIC KEYS file	Yes		12
15	APL mixed record file	No	370	
21	Function file	Yes		21
22	Feature file	Yes		22
23	IMF file	Yes		23
24	Diagnostic file	Yes		24
26	APL continue file	No	364	

Chapter 7. Tape-to-Tape Copy

The tape-to-tape copy functions copies an entire input tape to an output tape or copies a selected single input file to an output file. When the entire tape is copied, the function marks the output tape exactly the same as the input tape was marked. When a single file is copied, you must mark the output file before the copy operation is started.

OPERATION

The tape-to-tape copy function is loaded via the link command and is on file 5 on the IBM-supplied tape cartridge or diskette. The file name is TTCOPY. After the function is loaded, the following is displayed:

```
TAPE TO TAPE COPY FUNCTION

SELECT DESIRED COPY OPTION

OPTIONS ARE:

    1. COPY ONE FILE
    2. COPY ALL FILES

SELECT OPTION, PRESS EXECUTE
```

Enter a 2 to copy the entire input tape to the output tape. Enter a 1 to copy a single selected input file to an output file. When you select a single file to be copied, the following is displayed:

```
HOW MANY TAPE DRIVES WILL YOU BE USING

OPTIONS ARE:

    1. ONE TAPE DRIVE
    2. TWO TAPE DRIVES

SELECT OPTION, PRESS EXECUTE
```

Enter a 1 or 2 to select the number of the tape drives.

One-Tape-Drive Copy

If you select the one-tape-drive option, the following is displayed:

```
INSERT TAPE BEING COPIED FROM

INTO SELECTED TAPE DRIVE

ENTER DRIVE NUMBER, PRESS EXECUTE
```

Enter a 1 to select tape drive 1 (built-in) or a 2 to select tape drive 2 (auxiliary). Then insert the tape cartridge used for input into that drive, and press the EXECUTE key.

If you are copying a single file, the following is displayed:

```
ENTER FILE NUMBER TO BE COPIED FROM
```

```
ENTER FILE NUMBER TO BE COPIED TO
```

Enter the file number you want to use. The file number can be up to three numeric digits.

On a one-drive copy, the function reads data from the input tape file into the system's read/write storage. When the read/write storage is filled or all the data is read from the input tape, the following is displayed:

```
INSERT TAPE BEING COPIED TO
```

```
PRESS EXECUTE TO CONTINUE
```

You should now remove the tape cartridge used for input from the drive. Then insert the tape cartridge to be used for output and press the EXECUTE key. If the input tape file and the output tape file are on the same tape cartridge, just press the EXECUTE key. The data in read/write storage is then written to the output tape file, and, if more data exists on the input tape file, the following is displayed:

```
INSERT TAPE BEING COPIED FROM
```

```
PRESS EXECUTE TO CONTINUE
```

You should now remove the tape cartridge used for output from the drive. Then insert the tape cartridge used for input and press the EXECUTE key. If the output tape file and the input tape file are on the same tape cartridge, just press the EXECUTE key. The function then reads the input file into read/write storage.

This procedure (reading, changing tape cartridge, writing, and changing tape cartridge again) continues until all data is copied. When the copy is completed, the following is displayed:

```
TAPE COPY COMPLETED
```

```
OPTIONS ARE:
```

1. RESTART COPY FUNCTION
2. RETURN TO THE SYSTEM

```
ENTER OPTION, PRESS EXECUTE
```


Two-Tape-Drive Copy

If you selected the two-tape-drive option, the following is displayed:

```
INSERT TAPE BEING COPIED FROM  
INTO SELECTED TAPE DRIVE
```

```
ENTER DRIVE NUMBER, PRESS EXECUTE
```

```
INSERT TAPE BEING COPIED TO  
INTO SELECTED TAPE DRIVE
```

```
ENTER DRIVE NUMBER, PRESS EXECUTE
```

Enter the number of the tape drives you want to use for input and output. Enter a 1 for tape drive 1 (built-in) or a 2 for tape drive 2 (auxiliary). If you enter the same tape drive number for the input tape drive and the output tape drive, the function defaults to a one-drive option.

If you are copying a single file, the following is displayed:

```
ENTER FILE NUMBER TO BE COPIED FROM
```

```
ENTER FILE NUMBER TO BE COPIED TO
```

Enter the file number you want to use. The file number can be up to three numeric digits.

When the copy operation is completed, the following is displayed:

```
TAPE COPY COMPLETED
```

```
OPTIONS ARE:
```

1. RESTART COPY FUNCTION
2. RETURN TO THE SYSTEM

```
ENTER OPTION, PRESS EXECUTE
```

Hints

- When entering a file number, you can start the number in any position. Up to four digits may be entered for the file number.
- On a file-to-file copy, the output file must be marked. If the output file contains data, an error message is displayed.
- On a file-to-file copy, make sure the output file is large enough to contain all the data from the input file.
- Make sure the file you are copying to does not contain valuable data.
- When a complete tape is copied, an error message is displayed if file number 1 is marked. If you continue when this error occurs, the output tape is re-marked and the original data on the tape is lost.
- Make sure the correct tape cartridge is in the drive you specify for input and output.

Chapter 8. Diskette Compress

As a result of the marking, dropping, freeing, and re-marking of files, unused sectors can exist between extents for files. The function of diskette compress is to place all unused sectors into one area. To do this, the function rewrites all files on the diskette as close to cylinder 0 (beginning of the diskette) as possible. The first file on the diskette that has unused space before it is rewritten starting at the end of the preceding file or at the end of the label area. The next file is then rewritten starting at the end of the first file that was rewritten. This continues until all files on the diskette are moved ahead as close to the preceding file as possible. Any unused space that existed between files is removed. This space now exists at the end of the diskette and can be used for additional files.

Note: Diskette compression is performed automatically when diskette-to-diskette copy is used to copy an entire diskette.

OPERATION

Diskette compress is loaded by the link command and is on file 6 on the IBM-supplied tape cartridge or diskette. The file name is COMPRESS. After the function is loaded, the following is displayed:

```
THIS IS THE COMPRESS FUNCTION
```

```
PRESS EXECUTE TO CONTINUE
```

After the EXECUTE key is pressed, the following is displayed:

```
SELECT DRIVE YOU WILL BE USING
```

```
1,2,3 OR 4
```

```
LOAD DISKETTE AND PRESS EXECUTE
```

Enter a 1, 2, 3, or 4 to select the diskette drive you will be using. Insert the diskette you want to compress in that drive, then press the EXECUTE key. The function now moves all the files. After all files have been successfully moved, the following is displayed:

COMPRESS COMPLETE

OPTIONS ARE:

1. RESTART FUNCTION
2. RETURN TO SYSTEM

ENTER OPTION NUMBER AND PRESS EXECUTE

Hints

- A diskette read or write error can prevent successful completion of the compress function and could make the file that had the error unusable. After the error occurs, a message is displayed indicating whether the data has been restored to its original address.
- The files on a format 9 (1024-byte sector) diskette cannot be compressed.
- The diskette-to-diskette copy function also compresses the output diskette.

When a header label error occurs, an attempt is made to restore the data back to the original location. A message is displayed indicating whether the data was restored. However, the data in the file is not usable because of the header. The diskette recovery function can be used to correct the header and thus recover the data in the file.

The label display function displays the diskette volume label, diskette ERMAP (error map) label, or diskette and tape file header labels. The contents of these labels are displayed and can also be printed.

A current record of the label information is helpful if a read error occurs on the diskette or tape label. The diskette and tape recovery functions request information to construct a new label. All the information requested by the recovery functions is available from the label display function output.

Labels on any diskette that is access-protected cannot be displayed.

OPERATION

The label display function is loaded by the link command and is on file 7 on the IBM-supplied tape cartridge or diskette. The file name is LDISPLAY.

After the function is loaded, the following is displayed:

```
SYSTEM LABEL DISPLAY OPTIONS ARE:
```

1. DISPLAY FROM TAPE
2. DISPLAY FROM DISKETTE
3. RETURN TO SYSTEM

```
ENTER OPTION NUMBER, PRESS EXECUTE
```

Enter a 1 to display tape labels, a 2 to display diskette labels, or a 3 to cancel the function and return to the language. The abbreviations used in the label displays and printouts are shown later in this chapter.

Display Tape Labels

The following is displayed when the display tape labels option is selected:

```
TAPE LABEL DISPLAY OPTIONS ARE:
```

1. DISPLAY TAPE HEADER LABEL(S)
2. DISPLAY FROM DISKETTE
3. RETURN TO SYSTEM

```
ENTER OPTION NUMBER, PRESS EXECUTE
```

Enter a 1 to display tape header labels, a 2 if you want to go back and display diskette labels, or a 3 to return to the system. When a 1 is entered, the following is displayed:

DISPLAY TAPE HEADER LABEL(S)

ENTER DISPLAY STATEMENT D,P,###,###

D---DRIVE NUMBER (1 OR 2),

P---PRINT LABEL? (0=NO, 1=YES, OR 2=AUTO),

#---FILE RANGE: FIRST FILE, LAST FILE

OR ENTER 0 TO STOP
AND THEN PRESS EXECUTE

The first parameter selects the tape drive. Enter a 1 to select tape drive 1 (built-in) or a 2 to select tape drive 2 (auxiliary).

The next parameter specifies whether you want the labels printed or just displayed. Enter a 0 to display the labels. Enter a 1 or 2 to print the labels on the printer. When a 1 is entered, the function stops after each label is printed; you must press the EXECUTE key to have the next label printed. When 2 is specified, the function does not stop until all labels are printed. Pressing the ATTN key causes the printing to stop.

The next two parameters specify the beginning and ending file numbers. The file numbers can be up to three digits. Only the beginning (FIRST FILE) file number need be entered if only one label is to be displayed.

A comma must be entered between the parameters. After all the parameters are entered and the EXECUTE key is pressed, the labels on the tape in the selected tape drive are displayed and printed if print is specified.

Display Diskette Labels

The following is displayed when the display diskette labels option is selected:

DISKETTE LABEL DISPLAY OPTIONS ARE:

1. DISPLAY ERMAP LABEL
2. DISPLAY VOLUME LABEL
3. DISPLAY DISKETTE HEADER LABEL(S)
4. DISPLAY FROM TAPE
5. RETURN TO SYSTEM

ENTER OPTION NUMBER, PRESS EXECUTE

Enter a 1, 2, or 3 to select the type of diskette labels you want displayed.
Enter 4 if you want to display a tape label, or a 5 to return to the system.

Display ERMAP Label

The following is displayed when the display ERMAP label option is selected:

```
DISPLAY ERMAP LABEL

      ENTER DISPLAY STATEMENT      D,P

      D---DRIVE NUMBER (1, 2, 3, OR 4),
      P---PRINT LABEL? (0=NO OR 1=YES)

      OR ENTER 0 TO STOP
      AND THEN PRESS EXECUTE
```

Enter the diskette drive number followed by a comma. Enter a 0 if the label is to be displayed, or a 1 if the label is to be printed. When the EXECUTE key is pressed, the ERMAP label on the diskette in the selected diskette drive is displayed and printed if print is specified.

Display Volume Label

The following is displayed when the display volume label option is selected:

```
DISPLAY VOLUME LABEL

      ENTER DISPLAY STATEMENT      D,P

      D---DRIVE NUMBER (1, 2, 3, OR 4),
      P---PRINT LABEL? (0=NO OR 1=YES)

      OR ENTER 0 TO STOP
      AND THEN PRESS EXECUTE
```

Enter the diskette drive number followed by a comma and a 0 if the label is just displayed, or a 1 if the label is to be printed. When the EXECUTE key is pressed, the volume label on the diskette in the selected diskette drive is displayed and printed.

Display Diskette Header Labels

The following is displayed when the display diskette header labels option is selected:

```
DISPLAY DISKETTE HEADER LABEL(S)

      ENTER DISPLAY STATEMENT      D,P,####,####

      D---DRIVE NUMBER (1, 2, 3, OR 4),
      P---PRINT LABEL? (0=NO, 1=YES, OR 2=AUTO),
      #---FILE RANGE: FIRST FILE, LAST FILE

      OR ENTER 0 TO STOP
      AND THEN PRESS EXECUTE
```

The first parameter selects the diskette drive. The next parameter specifies whether you want the labels printed or displayed. Enter a 0 to display the labels. Enter a 1 or 2 to print the labels on the printer. When a 1 is entered, the function stops after each label is printed; you must press the EXECUTE key to have the next label printed. When 2 is specified, the function does not stop until all labels are printed. Pressing the ATTN key causes the printing to stop.

The next two parameters specify the beginning and ending file numbers. The file numbers can have up to four digits. Only the beginning (FIRST FILE) file number need be entered if only one label is to be displayed.

A comma must be entered between the parameters. After all the parameters are entered and the EXECUTE key is pressed, the labels on the diskette in the selected drive are displayed and printed if print is specified.

DISPLAY FORMAT

The *IBM Diskette General Information Manual*, GA21-9182, provides detailed information on each of the fields in the diskette ERMAP label, volume label, and file header label. The following is a list of abbreviations used for the displays and printouts.

Abbreviation	Meaning
FLDNAME	Name of the field in the label
P	Position of the field in the label
LNTH	Length of the field in bytes
CONTENTS	Contents of the field

ERMAP Label

The ERMAP label contains information about the number and location of any defective cylinders on the diskette. The following is a list of abbreviations used for the ERMAP label.

Abbreviation	Meaning
LABLID	Label identifier (ERMAP)
DEFCYL	No defective cylinder indicator or first defective cylinder number
DEFCYL	No second defective cylinder indicator or second defective cylinder number
DEFECT	Diskette defect indicator
DRTIND	Error directory indicator
ERRDRT	Error directory

Volume Label

The volume label contains information about the format of the diskette. The following is a list of abbreviations used for the volume label.

Abbreviation	Meaning
LABLID	Label identifier and label number (VOL1)
VOLID	Volume identifier
ACCESS	Accessibility
SYSID	System identification
OWNRID	Owner identification
LBLXTN	Label extension indicator
VOLSRF	Volume surface indicator
XTNTAR	Extent arrangement indicator
SPLREQ	Special requirements indicator
PHYRLN	Physical record length of available space
PHYRSQ	Physical record sequence code
LBLSTD	Label standard version

Diskette File Header Label

The diskette file header label defines the contents and format of the associated diskette file. The following is a list of abbreviations used for the diskette file header label.

Abbreviation	Meaning
LABLID	Label identifier and label number (HDR1)
FILEID	Data set identifier
BLKLNG	Block length
RCDATT	Record attribute
BOE	Beginning of extent
PHYRLN	Physical record length
EOE	End of extent
RBLKFT	Record block format
BYPASS	Bypass indicator
DSSRTY	Data set security
WRTPRT	Write-protect indicator
XCHGTP	Exchange type indicator
MLTVOL	Multivolume data set indicator
VOLSQ#	Volume sequence number
CRDATE	Creation date
RCDLNG	Record length
OFFSET	Offset to next record space
EXDATE	Expiration date
VFYCPY	Verify/copy indicator
DSORG	Data set indicator
EOD	End-of-data address
SYSCOD	System code
FILTYP	File application type

Tape File Header Label

The tape file header label defines the contents and format of the tape file. The following is a list of abbreviations used for the tape file header label.

Abbreviation	Meaning
FILTYP	File type
FILE#	File number
FLSZKB	File size in K-bytes
SEQREL	Sequential relocate indicator
EOD	End of data
FILEID	File identification
BASTMT	First and last BASIC statement number
BASKEY	BASIC keys
APLMSP	APL read/write storage starting address
APLWS1	Length of the first half of the user's workspace
APLSVI	SVI starting address
APLWS2	Length of the second half of the user's workspace
APLMSZ	Machine size
APLMXL	Maximum logical record length
APLMX#	Maximum logical record number in the file

Chapter 10. Diskette Recovery

The diskette recovery function allows you to read a file when a read error occurs on the volume label (errors 015, 024, 025), header label (errors 015, 038), or data portion (error 017) of the file. The function saves as much of the data on the file as possible. The function can be used to recover data from the following file types:

- 02 General exchange
- 08 APL internal data
- 09 Record I/O

The diskette recovery function can be used on labels for all file types.

Note: To ensure a successful recovery from a read error on a label, you should have a current record of the output from the label display function (see Chapter 9). This record provides the information needed to reconstruct the label. You should also make sure you have room for one more header label on the diskette.

OPERATION

The diskette recovery function is loaded by the link command and is on file 8 on the IBM-supplied tape cartridge and diskette. The file name is DRECOVER. After the function is loaded, the following is displayed:

```
DISKETTE RECOVERY FUNCTION

OPTIONS ARE:

1. VOLUME LABEL RECOVERY
2. HEADER LABEL RECOVERY
3. DATA FILE RECOVERY

ENTER OPTION NUMBER AND PRESS EXECUTE
```

After the option is selected, the following is displayed:

```
ENTER DRIVE NUMBER

INSERT DISKETTE INTO SELECTED DRIVE

PRESS EXECUTE
```

After the diskette drive number is entered and if option 2 or 3 was selected, the following is displayed:

ENTER FILE NUMBER

PRESS EXECUTE

After the file number is entered, the function begins the recovery procedure.

Volume Label Recovery

The function attempts to read and check the volume label. If the label is found to be error free, an error message is displayed. If an error is found, the function requests the following information to create a new volume label:

- Label area extended indicator (valid entries are blank, 1-9) (LBLXTN)
- Volume surface indicator (valid entries are 1, 2, or 2D) (VOLSRF)
- Length of physical record (valid entries are blank for 128, 1 for 256, or 2 for 512) (PHYRLN)
- Physical record sequence (valid entries are blank, 1-13) (PHYRSQ)

The function uses default values for the other fields in the label. The new label is then written on the diskette. If the new label cannot be written without error, an error message is displayed. If the new label is written without error, the recovery is complete.

You can check if the volume label you created contains valid data. To do this, perform the volume label recovery option again. If error code 278 occurs, the label contains valid data. If the function requests information to create a new volume label, the information you supplied for the first volume label was not valid.

If recovery was successful, you should copy the data from the diskette that had the error to another diskette. Then initialize the diskette that had the error. If an error occurs on the initialize function, the diskette should not be used.

Header Label Recovery

The function attempts to read the header label. If the label can be read, an error message is displayed. If an error is found, the function requests the following information to create a new header label:

- Block length (BLKLNG)
- Record attribute (RCDATT)
- Beginning of extent (BOE)
- End of extent (EOE)
- Exchange type indicator (XCHGTP)
- Record length (RCDLNG)
- Offset to the next record (OFFSET)
- Data set organization indicator (DSORG)
- End of data (EOD)
- File type (FILTYP)

The function then tries to write the new label over the label causing the error. If the label can be written without error, recovery is complete. If an error occurs, the function requests a file number to write the new label. The function then writes the new label at the new location.

You can check if the header label you created contains valid data. To do this, perform the header recovery option again. If error code 284 occurs, the label contains valid data. If the function requests information to create a new header label, the information you supplied for the first header label was not valid.

When recovery is complete, you should copy the data from the diskette that had the error to another diskette. Then initialize the diskette that had the error. If an error occurs on the initialize function, the diskette should not be used.

Data File Recovery

This function reads the file until the sector that caused the error is found. The function then deletes the sector or writes blanks in the sector. Since the sector can contain data from more than one record, all records (in preceding and following sectors) with data in the sector with the error is lost.

If the function is successful in writing over the error, the recovery is complete. If an error occurs when the function is recovering data from a type 2 or 8 file, an error message is displayed. If an error occurs when the function is recovering data from a type 9 file, the following is displayed:

OPTIONS :

1. RESTART FUNCTION
2. RETURN TO SYSTEM
3. CONTINUE WITH FUNCTION

ENTER OPTION NUMBER AND PRESS EXECUTE

RECOVERY ERROR 293

When option 3 is selected, the file is copied to another file. You can select a file to which the file that caused the error will be copied. The two files may be on the same diskette or different diskettes. This copy is similar to the diskette-to-diskette copy described in this manual. The function issues prompting messages to assist you in doing the copy.

When recovery is complete, you should copy the data from the diskette with the error to another diskette. Then initialize the diskette that had the error. If an error occurs on the initialize function, the diskette should not be used.

Chapter 11. Tape Data Recovery

The tape data recovery function is used to read a file when a read error occurs. The function saves as much of the data in the file as possible. The function works on the following file types:

- 01 General exchange
- 02 General exchange or BASIC source
- 03 BASIC source
- 08 APL internal data

OPERATION

The tape data recovery function is loaded by the link command and is on file 9 on the IBM-supplied tape cartridge or diskette. The file name is TRECOVER.

After the function is loaded, the following is displayed:

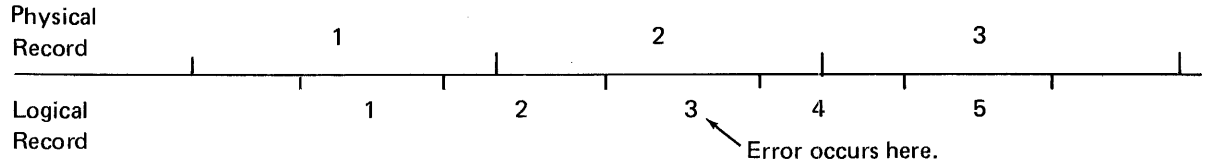
```
TAPE DATA RECOVERY FUNCTION  
ENTER DRIVE NUMBER
```

Enter a 1 to select tape drive 1 (built-in) or enter a 2 to select tape drive 2 (auxiliary). Press the EXECUTE key. The following is then displayed:

```
DATA IS RECOVERED FROM FILES HAVING TAPE READ ERRORS.  
FILE TYPES 01, 02, 03, AND 08 CAN BE PROCESSED.  
  
LOAD CARTRIDGE WITH READ ERRORS IN TAPE UNIT.  
ENTER FILE NUMBER (UP TO 3 DIGITS) AND PRESS EXECUTE.
```

Insert the tape with the read errors in the selected drive. Enter the file number of the file with the errors. Up to three digits can be entered for the file number.

The recovery function reads the file until the record that caused the read error is found. The function then writes blanks in the logical record with the error. If the record cannot be written back without error, all the data in the physical tape record (512 bytes) will be lost. Because the physical tape record may contain part of logical records from the preceding and following physical records, these logical records are lost also. For example:



Correcting the error in logical record 3 will cause logical records 2, 3, and 4 to be lost.

Chapter 12. Tape Header Recovery

The tape header recovery function is used to recover the data from a file when an error occurs while the header label is being read. The header label contains information that defines the file. The function works on headers for all files.

Note: To ensure a successful recovery from a read error on a header label, you should have a current record of the output from the label display (see Chapter 9). This record provides the information needed to reconstruct the tape header label.

OPERATION

The tape header recovery function is loaded by the link command and is on file 10 on the IBM-supplied tape cartridge or diskette. The file name is HRECOVER. After the function is loaded, the following is displayed:

```
TAPE HEADER RECOVERY FUNCTION
INSERT TAPE WITH HEADER CRC ERROR INTO SELECTED DRIVE
ENTER DRIVE NUMBER
PRESS EXECUTE
```

Insert the tape with the read errors in a tape unit. Enter a 1 to use tape drive 1 (built-in) or enter a 2 to use tape drive 2 (auxiliary).

The recovery function reads the tape until the header that caused the read error is found. The function then requests the following information to create a new header:

- File type (FILTYP)
- File size in K-bytes (FLSZKB)
- Sequential relocate indicator (SEQREL)
- End of data (EOD)
- First and last BASIC statement number (BASTMT)
- BASIC keys (BASKEY)
- APL read/write storage starting address (APLMSP)
- Length of the first half of the user's workspace (APLSV1)
- SVI starting address (APLSVI)
- Length of the second half of the user's workspace (APLSV2)
- Machine size (APLMSZ)
- Maximum logical record length (APLMXL)
- Maximum logical record number in the file (APLMX#)

The function then tries to write the new header over the header that caused the read error. If successful, the recovery is complete. If an error occurs, a message is displayed. This message gives you the option of copying the files from the tape with the read error to a different tape.

If you select the copy option, the function issues prompting messages to aid you in doing the copy. The file with the read error and all following files are copied to the other tape. The output tape is marked during the copy operation starting at the file copied to. For example, if a tape had 6 files and file 4 had a bad header, files 4, 5, and 6 would be copied. These three files could be added to another tape that already had 10 files. Files 11, 12, and 13 on the output tape would be marked as the data was being copied to the tape.

Chapter 13. Diskette Sort Feature

The Diskette Sort feature is optional on the IBM 5110. This feature allows you to change the order of the records in a file by doing a record sort or an address out sort. (The Diskette Sort feature is also referred to as *sort*.)

The record sort option creates a new file with the records in the sorted order. The address out option of the sort creates a file that contains the address of the records in the sorted order. The sort operates faster when an address out file is used because it does not have to write all of the records on the output file. Also, a smaller file can be used for output because the file contains only the addresses and not the data.

The Diskette Sort feature does not have to be loaded from tape or diskette before it is used. To execute sort use the)SORT (for APL) or UTIL SORT (for BASIC) command. You can also execute sort by using the procedure file from APL or BASIC. The use of the procedure file is discussed later in this chapter.

The remaining information on sort is divided into the following main topics:

Introduction to sort	Gives an overview of why you would want to sort
Files	Defines the characteristics of the files used by the sort program
Statements	Gives the syntax and a definition of each of the statements used by the sort program
Operation	Explains how to use the sort program and shows the displayed messages
Additional Information	Discusses performance considerations and programs to determine work file size

INTRODUCTION TO SORT

The following is an example of how you could use the sort program to prepare a report. The files and the report are described first. The actual sort statements are shown when the sort statements are discussed.

For this example, assume that you have a file, named MASTER, that contains a record for each of your customers. Each record has the customer number and name, the sales representative, the year-to-date (YTD) sales, and other information about the customer. The customer number has six digits. The first four are the customer identifier, and the last two are the sales territory associated with the customer. Assigning the customer numbers this way allows you to summarize your sales by territory easily. This file could have been created by an APL or BASIC program and would look like this if it were formatted and printed:

Customer Number	YTD Sales	Sales Rep	Customer Name
133101	900.00	CLAIR	SMITH LUMBER WEST
133102	830.00	MEL	SMITH LUMBER EAST
231101	900.00	CLAIR	R AND D PUMPS
287502	600.00	DON	DANS FLOWERS
303802	950.00	DON	SAMS BOOKS
369101	1700.00	MARTY	NORTHWOOD BOATS
407101	100.00	CLAIR	NORTHLAND CARS
457101	1000.00	CLAIR	K AND K WHOLESALE
508702	- 100.00	DON	SUNSHINE FLOORS
550101	2000.00	MARTY	DONS WORLD

Positions 1 through 4 are the customer identifier; positions 5 and 6 are the sales territory. Position 7 is the sign position for the YTD sales in positions 8 through 14. The sales representative is in positions 15 through 19. The customer name is in positions 20 through 36. Positions 37 through 128 are used for additional information. There are 1000 records in the file. The file is file 1 on a diskette in drive 1 and is named MASTER.

As you can see, the file is in ascending order by customer number. Now assume you want to print a report that shows by territory and sales representative the YTD sales for each customer. You want the customers within each group to appear in descending order of YTD sales on the report. The sort program allows you to use up to six different fields to determine the order of the records. Such a report could look like this:

Terr	Sales Rep	YTD Sales	Customer Number	Customer Name
01	CLAIR	1000.00	457101	K AND K WHOLESALE
		900.00	133101	SMITH LUMBER WEST
		900.00	231101	R AND D PUMPS
		100.00	407101	NORTHLAND CARS
01	MARTY	2000.00	550101	DONS WORLD
		1700.00	369101	NORTHWOOD BOATS
02	DON	950.00	303802	SAMS BOOKS
		600.00	287502	DANS FLOWERS
		- 100.00	508702	SUNSHINE FLOORS
02	MEL	830.00	133102	SMITH LUMBER EAST

The first column of the report shows the territory. The next column is the sales representative. The next three columns show the YTD sales, customer numbers, and customer name in order by the YTD sales. The highest YTD sales for each sales representative is shown first.

You must write an APL or BASIC program to print this report. But first you must sort the file to get the records in the order they are to appear on the report. The sorted file is named SALES. The examples used in this chapter deal with sorting the MASTER file to get the SALES file. The following shows the order of the records in the SALES file:

Customer Number	YTD Sales	Sales Rep	Customer Name
457101	1000.00	CLAIR	K AND K WHOLESALE
133101	900.00	CLAIR	SMITH LUMBER WEST
231101	900.00	CLAIR	R AND D PUMPS
407101	100.00	CLAIR	NORTHLAND CARS
550101	2000.00	MARTY	DONS WORLD
369101	1700.00	MARTY	NORTHWOOD BOATS
303802	950.00	DON	SAMS BOOKS
287502	600.00	DON	DANS FLOWERS
508702	- 100.00	DON	SUNSHINE FLOORS
133102	830.00	MEL	SMITH LUMBER EAST

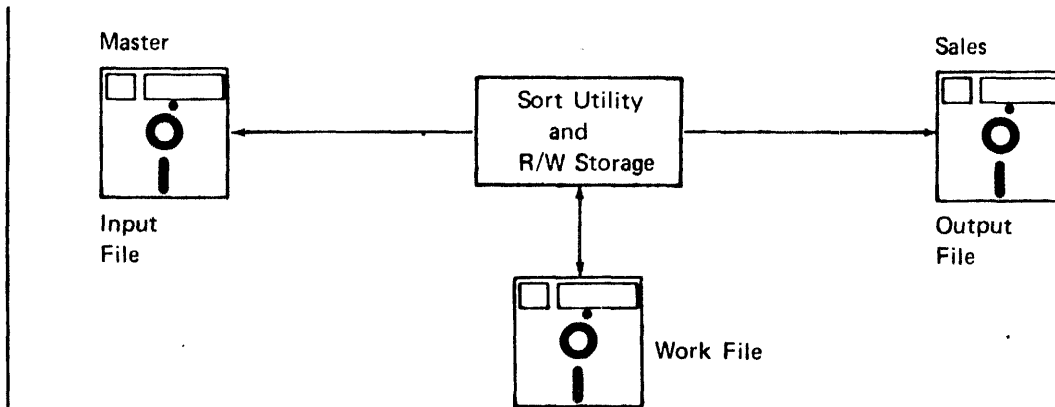
FILES

The sort program uses an input, output, and, for some sort jobs, a work file. All files must be diskette files. The program reads the records from the input file. The sort fields (fields of the record that are used to determine the order of the records) and the location of the record in the input file (addresses) are called keys. The keys are stored in the system. The sort program uses the same area of the system that APL uses for its workspace or that BASIC uses for its work area to store the keys. This part of the system is called the read and write storage (R/W storage).

When a large input file is sorted, more space may be required to store the keys than is available in R/W storage. The work file is used to provide the additional space.

The sort program fills R/W storage with keys, and the keys are sorted. The sorted keys are then written on the work file. The sort program reads the complete input file in this manner. After all input records are read, the program does a final sort on all the keys.

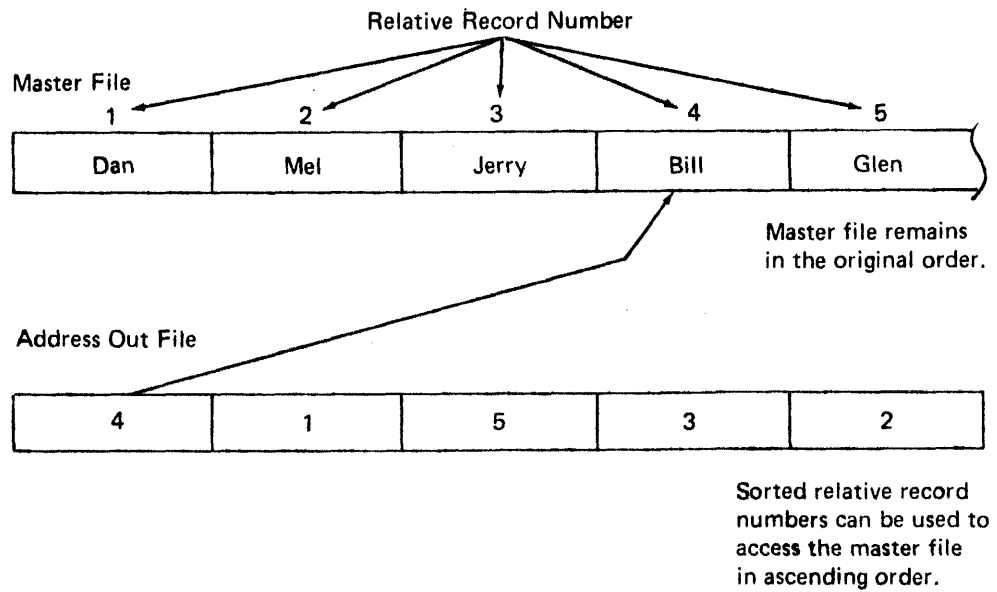
In the standard sort, the program uses the sorted keys to read the input file again. This time the records are read from the input file and are written to the output file. Thus the input file remains in the original order, and the output file is in the sorted order.



If an address out sort is specified, the program uses the sorted keys to write the location of the input record to the output file. Thus the output file (address out file) contains a 4-byte binary logical record number (relative record number) for each of the records in the input file.

You can then read the relative record numbers from the address out file and use them to directly access records from the input file.

For example:



The following statements might be used in a program to access a file in order using the record numbers in the address out file:

```
◊  
◊  
◊  
0030 OPEN FILE FL1, 'D80', 1, 'NAMES', ALL  
0040 OPEN FILE FL2, 'D80', 2, 'ADDSORT', IN  
0050 READ FILE USING 060, FL2, R  
0060 FORM B4  
0070 READ FILE FL1, REC=R, A#  
◊  
◊  
◊
```

This page is intentionally left blank.

Input File

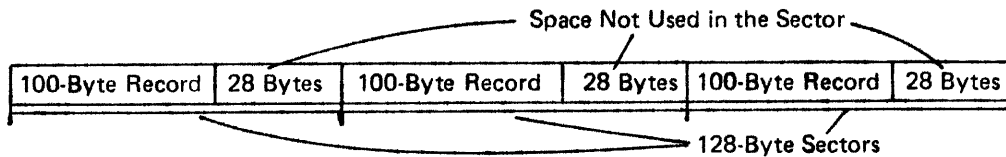
The input file must be a record I/O (type 9) data file created on the IBM 5110. Record I/O data files created on other devices may be used if the characteristics are the same as those of the 5110 files. If the characteristics are not correct, one of the following error codes will result:

Error	Meaning
-------	---------

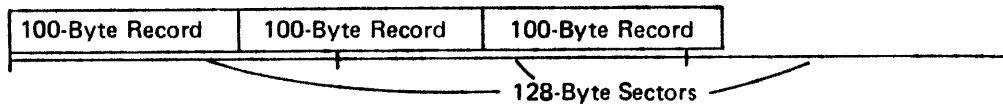
- | | |
|-----|---|
| 336 | The block length for a blocked-spanned file does not equal the sector size. |
| 342 | A multivolume file cannot be used. |
| 344 | The input file exchange type is not supported. |
| 345 | A variable-length logical record cannot be used. |
| 355 | The diskette file expiration date is not blank. |

All diskette formats except format 9 (1024-byte sectors) can be used for this file. The file can have either sequential or random organization. The records can be any valid length and can be blocked-spanned or unblocked-unspanned.

In an unblocked-unspanned file, one record is written on each sector. The records cannot be longer than the size of the sectors. If they are shorter, the additional space in the sector is unused.



For a blocked-spanned file, the records span the limits of the sectors. For example, if you have a sector size of 128 bytes and a record size of 100 bytes, the first record and 28 bytes of the second record are written on the first sector of the file.



Output File

The file used as an output file by the sort program must be a marked, unused file. When you are using blocked-spanned input files:

- After a standard sort is completed, the output file has the same organization, logical record length, and record format as the input file.
- After an address out sort is completed, the output file has a blocked-spanned format with a record length of four.

When you are using unblocked-unspanned input files, you can only do an address out sort. In this case, the output file has a blocked-spanned format with a record length of four.

Output File Size for Blocked-Spanned File

The marked size of the output file in a standard sort for blocked-spanned records must be equal to or larger than the input file. This allows all the records from the input file to be written to the output file after they are sorted.

Output File Size for Address Out File

The marked size of the output file in an address out sort for blocked-spanned or unblocked-unspanned input file is computed as follows:

$$K\text{-bytes} = \frac{\text{(number of records in the input file)}}{256}$$

Use the next K larger file size.

Work File

Some sort jobs do not need a work file. However, if the storage needed to do the sort is greater than your system's R/W storage size, your sort job needs a **work file**. You can determine approximately how large the work file must be by using the charts in Appendix A. Because these charts do not show all possible combinations, a formula is shown later in this chapter for that purpose. The charts assume you have no IMFs loaded and you are not using a procedure file. IMFs and a procedure file use part of the system's R/W storage.

How To Use Work File Size Charts

Charts are provided in Appendix A for all system R/W storage sizes and all work file diskette sector sizes. Use the chart that matches your system and diskette.

Across the top of each chart are the sort control field sizes. The size of the control field is the combined length of all the fields used to determine the sort order. Down the left side are the number of records in the file. The numbers in the chart show the approximate minimum size for the work file in K-bytes. If your exact sort field size or number of records does not appear in the chart, use the next higher value to make sure that you have a large enough work file. A zero entry in the chart means that a work file is not needed. Also, a work file is not needed if fewer than 100 records are sorted.

The numbers that appear on the extreme right side of the chart are diskette format numbers. The lines in the chart show the maximum capacities at which that format diskette can be used. Your work file size must appear above the line for the diskette format you are using.

For example, assume you have a 16K system with a 128-byte diskette sector size and format 1. You would use the following chart from Appendix A.

SYSTEM SIZE= 16K
 WORK FILE SECTOR SIZE= 128

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	9	12	12	17	17	17	17	17	17	33
500	0	0	9	12	14	17	34	44	44	66	66	66	66	66	66	128
750	7	11	13	34	40	50	50	66	66	96	96	96	98	98	98	190 ¹
1000	9	14	34	46	54	66	66	86	86	128	128	128	128	128	128	252
1500	26	40	50	66	78	98	96	128	130	190	190	190	192	192	192	378 ⁴
2000	34	54	66	88	104	128	128	170	172	252	252	252	254	254	254	
2500	42	66	82	108	130	160	160	212	214	316	316	316	318	318	318	
3000	50	80	98	130	154	192	190	252	256	378	378	378	380	380	380	
3500	58	92	112	150	180	222	222	294	298	440	440	440	444	444	444	
5000	82	130	160	214	256	316	316	420	424							
7500	122	194	238	320	382	474	472									
10000	160	258	316	426												
15000	240	384	474													
20000	318															
30000	476															

For our example, the records have a 15-position control field with 1000 records in the file. We must use a 16-position control field because the chart does not show a 15-position control field. From the chart you can see that a work file of at least 46K would be required.

The work file must be a marked, unused file. When the sort is finished, the program displays a message to indicate work file usage. Then the program marks the work file as an unused file.

Control File

The control file contains sort statements. The sort statements are read from the control file each time the sort program is executed, eliminating the need to rekey the sort statements. The sort statements for a given sort are identified by the sort group name parameter in the sort group name statement.

You can write an APL or BASIC program to create the control file. Two simple methods of creating a control file are shown later. Any diskette format except format 9 can be used for the control file. The control file must:

- Be a record I/O data file
- Have blocked-spanned records
- Have a logical record length of 64

The sort statements for each sort group must be written on the control file in the same order as they are specified for the sort program. The order of the statements is:

1. Sort group name
2. Comment (optional)
3. SORT
4. MASK
5. ALTS (optional)
6. WORK (optional)

Each group of sort statements must have a unique name specified for the sort-group-name parameter of the sort group name statement. Up to 150 sort statements can be in a single group on a 16K system; more statements can be used with a larger system. The amount of R/W storage available determines the number of statements that can be in a sort group. The sort-group-name is used to retrieve the correct group of sort statements from the control file. The sort statements are read from the control file by a FILE statement.

The following is the sort group for our example.

```
**SALES
*SORT CUSTOMER FILE BY TERRITORY, SALES REP AND YTD SALES
SORT 1,MASTER,1,1,SALES,2
MASK 5,2,A,15,5,A,7,8,0
WORK 2,1
```

How to Create A Control File

The following are two methods of creating the control file. One method uses **APL**, the other **BASIC**. Both methods create a file named **CONTROL** on file 1 on diskette drive 1.

APL: Use the following APL statements to create the control file for our example.

```
1. 1 DSV0 'OUT'  
2. OUT←'OUTF 11001 ID=(CONTROL) TYPE=I'  
3. OUT←64↑'**SALES'  
3. OUT←64↑'*SORT CUSTOMER FILE BY TERRITORY, SALES REP AND YTD SALES'  
3. OUT←64↑'SORT 1,MASTER,1,1,SALES,2'  
3. OUT←64↑'MASK 5,2,A,15,5,A,7,8,0'  
3. OUT←64↑'WORK 2,1'  
4. OUT←\0
```

Statement 1 offers a shared variable called **OUT**. Statement 2 assigns the open parameter string to the shared variable. Statement 3 is repeated for each sort statement you want to write on the file. Statement 4 closes the file.

BASIC: Use the following steps in BASIC to create the control file for our example.

```
1. LOAD 0,DATA  
2. 0010:**SALES  
2. 0020:*SORT CUSTOMER FILE BY TERRITORY, SALES REP AND YTD SALES  
2. 0030:|SORT 1,MASTER,1,1,SALES,2  
2. 0040:|MASK 5,2,A,15,5,A,7,8,0  
2. 0050:|WORK 2,1  
3. SAVE 1, 'CONTROL', RECL=64, D80
```

Statement 1 allows you to enter data from the keyboard into the work area. Statement 2 is the data for the sort statements. Enter each line after the line number and colon are displayed. Statement 3 writes the data from the work area on the file.

STATEMENTS

The sort statements may be entered from the keyboard or read from a control file. The sort statement length must not exceed 64 characters. Spaces are not allowed within or between the parameters of sort statements. If a space is found, error 302 is displayed. Also blanks (spaces) must follow the last parameter; if they do not, error 304 results. An exception is that a comma may follow the last parameter of an ALTS statement.

The six types of sort statements and their order of entry are:

1. Sort group name statement (required only with control file)
2. Comment statements (optional only with control file)
3. SORT statement
4. MASK statement
5. ALTS statement (optional)
6. WORK statement (optional)

FILE Statement

A FILE statement is entered from the keyboard or read from a procedure file. The FILE statement specifies where the control file is located and which sort group to read. The command syntax is:

```
FILE      sort-group-name, control-file-number, [control-filename],  
          drive-number
```

where:

sort-group-name, is the name of the sort group. This name must be the same as the name given in the sort group name statement you want.

control-file-number, is the file number of the control file. A maximum of four digits can be used.

control-file-name, (optional) is the name assigned to the control file. A maximum of 17 characters can be used. If the name is not used, the comma must still be entered.

drive-number is the control file diskette drive number. The valid entries are 1, 2, 3, or 4.

For our example, assume you created a control file on file 5 and that the file is called EXAMPLE. The file statement would be:

```
FILE SALES, 5, EXAMPLE, 1
```

Sort Group Name
File Number of the Control File
Name of the Control File
Drive Number of the Control File

Sort Group Name Statement

The sort group name statement is used only when the sort statements are read from a control file. The control file can contain several sets of sort statements. Each set of sort statements is identified and starts with a sort group name statement. All sort statements that follow the sort group name statement belong to that sort group.

The sort group name statement has *(asterisks) in positions 1 and 2. The sort group name starts in position 3. The name can have up to 8 alphameric characters. The first character of the name must be alphabetic (A through Z).

For our example, if we were to write the sort statements on the control file, the sort group name statements and the other sort statements would look like this:

```
**COST
*SORT THE COST FILE
SORT * * * * *
MASK * * * * *
**SALES
*SORT CUSTOMER FILE BY TERRITORY, SALES REP AND YTD SALES
SORT 1,MASTER,1,1,SALES,2
MASK 5,2,A,15,5,A,7,8,0
WORK 2,1
**PAYROLL
*PAYROLL FILE IS SORTED FOR CHECK RUN
SORT * * * * *
MASK * * * * *
WORK * * * * *
```

The **SALES is the sort group name statement.

Comment Statement

The comment statement can be used only in a control file. Any number of comment statements may be used as long as the maximum number of statements for the sort group is not exceeded. The comment statements along with the other statements in the sort control group are displayed after being read from the control file. The comment statement can be used to identify the type of sort and the input and output files. It also can provide special information to the operator about the sort. The comment statement is not required. If used, it follows the sort group name statement and comes before the sort statement.

The comment statement has an *(asterisk) in position 1. The remaining 63 positions can be used for comments as in our example:

```
*SORT CUSTOMER FILE BY TERRITORY, SALES REP AND YTD SALES
```


SORT Statement

The SORT statement is required. The SORT statement describes the data files used and the type of sort to be performed by the sort program. The command syntax is:

```
SORT    input-number, [input-name], input-drive, output-number,  
        [output-name], output-drive [,sort-type]
```

where:

input-number,	is the file number of the input file. A maximum of four digits can be used (0001-9999).
input-name,	(optional) is the name assigned to the input file. A maximum of 17 characters can be used. If the name is not used, a comma must still be entered.
input-drive,	is the input file diskette drive number. The valid entries are 1, 2, 3, or 4.
output-number,	is the file number of the output file. A maximum of four digits can be used (0001-9999).
output-name,	(optional) is the name to be assigned to the output file. A maximum of 17 characters can be used. If the name is not used, a comma must still be entered and the name that exists on the diskette will be used.
output-drive,	is the output file diskette drive number. The valid entries are 1, 2, 3, or 4.
sort-type	(optional) leave blank for a standard sort (the complete record is sorted), enter an A for address out sort.

In our example, the SORT statement looks like this:

```
SORT 1, MASTER, 1, 1, SALES, 2
```

Input File Number
Input File Name
Input File Drive Number
Output File Number
Output File Name
Output File Drive Number

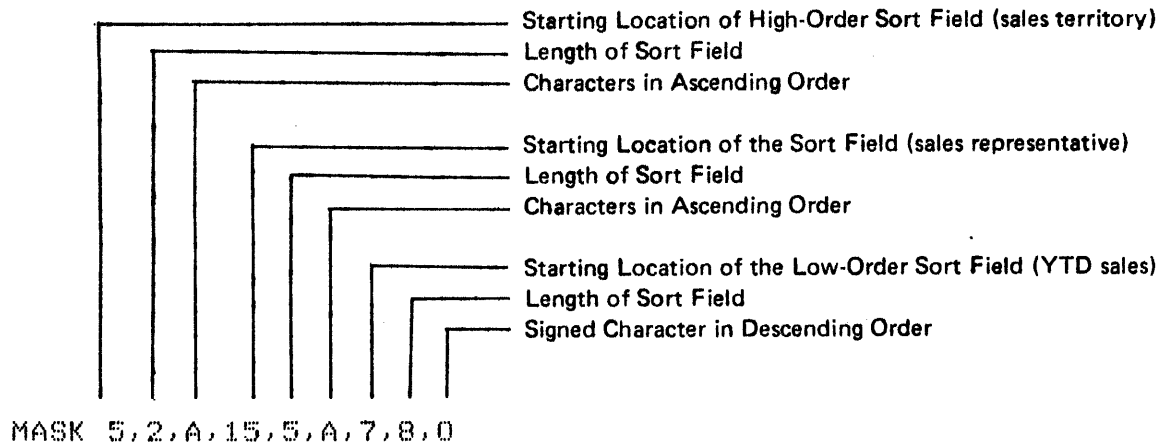
Character data is written by an APL program and by a BASIC program when a character variable or character constant is specified for output. The FORM statement in BASIC can also be used to change numeric data into character data. BASIC internal numeric (floating point) data is written when a numeric variable in BASIC is specified for output. Sign character data is written when PIC is used in the FORM statement in BASIC.

When signed character (entry N or O) data is sorted, the following conditions must be met:

- A sign must be in the leftmost position of the field.
- A blank in the leftmost position of the field is the same as a plus sign.
- The digits must be right-justified in the field.
- The complete field must be used as a sort field.
- Leading blanks are the same as zeros.
- Characters such as \$ and . can appear in the field as long as they always appear in the same position.

For more information on the types of data, see the APL or BASIC reference manual listed in the preface.

In our example, the MASK statement looked like this:



ALTS Statement

The ALTS statement is used when you want to alter the collating sequence used by the sort program. The collating sequence is restored to the original sequence when the next SORT statement is read. *Note:* The ALTS statement should only be used with the character sorts.

The collating sequence used by the sort is shown in the chart. The command syntax is:

ALTS new-location original-location[,...]

where:

new-location is the hexadecimal number from the entry column of the chart for the new location of the character you want to change.

original -location is the hexadecimal number from the entry column of the chart of the character you want to move.

Note that a comma can be used after the last entry but that it is not required.

For example, if you want the numbers to sort before the alphabetic characters, the ALTS statement would be:

ALTS 30F0,31F1,32F2,.....,39F9

When a character is moved into a sequence position normally assigned to another, both the new and the original character will occupy the same position in the sequence and will be considered equal. If the two characters are not to be equal, the character that normally occupies that position must also be moved.

Entry	Graphic	Entry	Graphic	Entry	Graphic	Entry	Graphic	Entry	Graphic	Entry	Graphic
00		34		68	Y	9C		D0	⌘		
01		35		69	Z	9D	o	D1	J		
02		36		6A	[9E		D2	K		
03		37		6B	,	9F	+	D3	L		
04		38		6C	%	A0	-	D4	M		
05		39		6D	_	A1	~	D5	N		
06		3A		6E	>	A2	s	D6	O		
07		3B		6F	?	A3	t	D7	P		
08		3C		70	&	A4	u	D8	Q		
09		3D		71	^	A5	v	D9	R		
0A		3E		72	:	A6	w	DA	⌘		
0B		3F		73	Δ	A7	x	DB	!		
0C		40		74	"	A8	y	DC	⌘		
0D		41	A	75	∅	A9	z	DD	⌘		
0E		42	B	76		AA	0	DE	⌘		
0F		43	C	77		AB	U	DF	⌘		
10		44	D	78	v	AC	L	E0	\		
11		45	E	79	~	AD	E	E1			
12		46	F	7A	:	AE	2	E2	S		
13		47	G	7B	#	AF	o	E3	T		
14		48	H	7C	@	B0	α	E4	U		
15		49	I	7D	.	B1	ε	E5	V		
16		4A	⌘	7E	=	B2	ι	E6	W		
17		4B		7F	~	B3	ρ	E7	X		
18		4C	<	80	~	B4	ω	E8	Y		
19		4D	(81	a	B5		E9	Z		
1A		4E	+	82	b	B6	x	EA	/		
1B		4F	⌘	83	c	B7	\	EB	⌘		
1C		50	&	84	d	B8	-	EC	⌘		
1D		51	J	85	e	B9		ED	⌘		
1E		52	K	86	f	BA	v	EE	⌘		
1F		53	L	87	g	BB	Δ	EF	⌘		
20		54	M	88	h	BC	T	F0	0		
21		55	N	89	i	BD	I	F1	1		
22		56	O	8A	↑	BE	⌘	F2	2		
23		57	P	8B	↓	BF	I	F3	3		
24		58	Q	8C	⌘	C0	⌘	F4	4		
25		59	R	8D	⌘	C1	A	F5	5		
26		5A	!	8E	⌘	C2	B	F6	6		
27		5B	\$	8F	→	C3	C	F7	7		
28		5C	*	90	⌘	C4	D	F8	8		
29		5D)	91	j	C5	E	F9	9		
2A		5E	:	92	k	C6	F	FA			
2B		5F	-	93	l	C7	G	FB	⌘		
2C		60	_	94	m	C8	H	FC	⌘		
2D		61	/	95	n	C9	I	FD	⌘		
2E		62	S	96	o	CA	X	FE	⌘		
2F		63	T	97	p	CB	⌘	FF	■		
30		64	U	98	q	CC	⌘				
31		65	V	99	r	CD	⌘				
32		66	W	9A	o	CE	⌘				
33		67	X	9B	e	CF	⌘				

Note: The EBCDIC graphics are shown in this chart. See the appropriate language reference manual for the graphics displayed for other national character sets. Graphics are assigned to all blank positions (except hex 40) for maintenance purposes. These graphics are incompatible with other systems and cannot be used for exchange purposes. These graphics may be removed or changed as a result of maintenance or new versions of this product.

WORK Statement

The WORK statement follows the ALTS statement. The WORK statement identifies the work file if one is required. The command syntax is:

WORK file-number, drive-number

where:

file-number, is the file number of the work file. A maximum of four digits can be used.

drive-number is the work file diskette drive number. The valid entries are 1, 2, 3, or 4.

In our example, the WORK statement looked like this:

```
WORK 2,1
```

In our example, we have a 16K system with a 128-byte sector work file diskette. There are 1000 records in the file. Each record has 15 positions in the control field. Now, based on the table in Appendix A, the work file should be at least a 46K file. When using the chart for this example you must use a 16-position control field because the chart does not have an entry for a 15-position field.

OPERATION

The)SORT (for APL) or UTIL SORT (for BASIC) command is used to activate the sort program. The sort program then determines the source of the sort statements by displaying the following:

CONTROL GROUP STATEMENT INPUT

OPTIONS ARE:

1. CONTROL FILE ON DISKETTE
2. KEYBOARD ENTRY
3. STOP

ENTER OPTION NUMBER, PRESS EXECUTE

When sort statements are entered from the keyboard, the sort program displays the sort statement keyword on the input line. The cursor is then positioned to enter the first parameter. Anytime the sort program is requesting input from the keyboard you can enter STOP. STOP must appear in the first four positions of the input line. When STOP is entered, you can restart the sort or return to the language.

Control File

Entering a 1 in response to the first display means that the sort statements are located on the control file. The following is then displayed:

ENTER FILE STATEMENT

SORT GROUP NAME,
CONTROL FILE NUMBER,
CONTROL FILE NAME,
DRIVE NUMBER

OR ENTER STOP
AND THEN PRESS EXECUTE

FILE ...

After the information for the FILE statement is entered, the control group is read. A message to mount the sort files is then displayed:

MOUNT SORT FILES.

AND THEN PRESS EXECUTE

After the sort files are mounted and the EXECUTE key is pressed, the sort statements are read and displayed.

The sort program also checks the sort statements for errors. If any errors are found, you will be allowed to correct them. However, these corrections are not saved on the control group file. Only 12 statements can be displayed at a time. When the sort control group contains more than 12 statements, press the EXECUTE key to display the additional statements. After all sort statements have been displayed and corrections have been made, press the EXECUTE key to begin the sort or enter STOP to end the sort.

Keyboard Entry

Entering a 2 in response to the control group statement input display means that the sort statements are entered from the keyboard. A message to remind you to mount the sort file is displayed. After the sort files have been mounted and the EXECUTE key is pressed, the following is displayed:

ENTER SORT STATEMENT

INPUT NUMBER,
INPUT NAME,
INPUT DRIVE,
OUTPUT NUMBER,
OUTPUT NAME,
OUTPUT DRIVE,
SORT TYPE

OR ENTER STOP
AND THEN PRESS EXECUTE

SORT _

After the parameters for the SORT statement have been entered and the EXECUTE key is pressed, the following is displayed:

ENTER MASK STATEMENT

FIELD 1 STARTING POSITION NUMBER,
FIELD 1 LENGTH,
TYPE 1 (A, D, N, O, F, G, Z, Y, P, Q, B, OR C)
FIELD 2 STARTING POSITION NUMBER,
FIELD 2 LENGTH,
TYPE 2 (A, D, N, O, F, G, Z, Y, P, Q, B, OR C)

UP TO 6 FIELDS

OR ENTER STOP
AND THEN PRESS EXECUTE

MASK _

After the parameters for the MASK statement have been entered and the EXECUTE key is pressed, the following is displayed:

```
PRESS EXECUTE TO USE THE STANDARD COLLATING SEQUENCE,  
OR ENTER ALTERNATE COLLATING SEQUENCE STATEMENT(S)
```

```
NEW-LOCATION 1 ORIGINAL-LOCATION 1,  
NEW-LOCATION 2 ORIGINAL-LOCATION 2,
```

```
ENTER AS MANY PAIRS AS NEEDED
```

```
OR ENTER STOP  
AND THEN PRESS EXECUTE
```

ALTS ...

If you do not want to change the collating sequence, just press the EXECUTE key.

To change the collating sequence, enter hexadecimal numbers that represent the changes, then press the EXECUTE key. From one to 12 changes can be entered on a single line. The same message will be displayed again to allow you to make additional changes. After you have made all your collating sequence changes, press the EXECUTE key.

The following is then displayed:

```
PRESS EXECUTE TO DO THE SORT,  
OR ENTER A WORK STATEMENT
```

```
FILE NUMBER,  
DRIVE NUMBER
```

```
OR ENTER STOP  
AND THEN PRESS EXECUTE
```

WORK ...

If a work file is not needed, just press the EXECUTE key. If you need a work file, enter the file number and diskette drive number before you press the EXECUTE key. The sort begins after the EXECUTE key is pressed.

If the sort program determines that a work file is needed and one was not specified, error 337 occurs. The previous message requesting a work statement is displayed along with the 337 error code. You can then enter a work statement and continue the sort.

Stop

Entering a 3 in response to the control group statement input display ends the sort program. The sort program is also ended when STOP is entered in positions 1 through 4 anytime the program is expecting input from the keyboard. When the sort program ends, the following is displayed:

SORT COMPLETED.

OPTIONS ARE:

1. EXECUTE SORT AGAIN
2. RETURN TO APL OR BASIC

ENTER OPTION NUMBER, PRESS EXECUTE

This message can be displayed when error 325, 326, 338, or 340 occurs or when any diskette error occurs during the sort. Therefore, check to make sure an error code is not displayed with this message.

If a work file was used during the sort, the amount of the work file storage used is shown in this display also.

Procedure File

The sort program can be executed by statements in the procedure file. The)PROC (APL) or PROC (BASIC) command is used to activate the procedure file. The procedure file must contain the)SORT (APL) or UTIL SORT (BASIC) statement and the FILE statement for each file you want to sort. The control, input, output, and work (if one is required) files must all be online before the procedure file is executed. All operator responses are bypassed unless an error is found. Except for error 337, errors cause the sort to return to the manual method of operation, and the procedure file operation is terminated. If error 337 is the only error encountered, the procedure file will continue when SORT completes. The BASIC system device is not changed by the sort program.

Following is a sample SORT program executed in an APL procedure file:

```
>LOAD 11002
PGM1
>SORT
FILE SALES,5,EXAMPLE,1
>LOAD 11003
PGM2
.
.
.
.
```

The equivalent procedure file for BASIC is as follows:

```
UTIL SYS D80
LOAD 2
RUN
UTIL SORT
FILE SALES,5,EXAMPLE,1
LOAD 3
RUN
.
.
.
.
```

The procedure file is accessed when you enter a PROC command. A PROC command instructs the system to begin using the procedure file in the indicated file, as shown below:

```
PROC 4,D80
```

The system then loads file 4 and begins executing the lines in the file, one record at a time.

Hints

- File names may be either simple or complex. The names SYSAREA and ERRORSET cannot be used.
- The work and output files must be marked and unused.
- The control file must be a record I/O file with blocked-spanded records. The record length must be 64.
- When the input file contains records with equal values in the control field, these records in the output file may not be in the same order as they were in the input file.

This page is intentionally left blank.

ADDITIONAL INFORMATION

Performance Considerations

The following conditions can lessen the time required to perform sort:

- Having fewer records to sort
- Having a smaller control field
- Having a smaller logical record size
- Having a blocked-spanned input file
- Having the input, output, and work files on different diskettes
- Not using an alternate collating sequence
- Using the address out sort

Work File Storage Requirements

The following APL and BASIC programs will determine the size of your work file if one is needed.

Diskette Format	Maximum File Size
1	240K
2	277K
3	296K
4	481K
5	555K
6	592K
7	962K
8	1,110K

For information about format, refer to Chapter 3, *Diskette Initialization*.

APL Program

```

      VWORKFILEC[]V
V WORKFILE;K;W;M;C;S;R;A;B;K1;T;G;I;I1;L;S1;S2;Z
[1]  'ENTER MACHINE SIZE IN # OF K'
[2]  K←[]
[3]  'ENTER WORK FILE SECTOR SIZE IN # OF BYTES'
[4]  W←[]
[5]  'ENTER MASK LENGTH IN # OF BYTES'
[6]  M←[]
[7]  'ENTER # OF RECORDS IN INPUT FILE'
[8]  R←[]
[9]  C←1↑((K≤ 16 32 48 64)/ 550 1650 2595 3855)
[10] Z←0
[11] L0:
[12] S←C
[13] L1:
[14] C←C+1
[15] T←L2⊙C
[16] →(T≤16)/L2
[17] T←16
[18] L2:
[19] A←2×(1+T+2×2×T)
[20] K1←7+(A+C×(M+7))÷1024
[21] →(K1>K)/L3
[22] →(C≥R)/L4
[23] →L1
[24] L3:
[25] B←C-1
[26] →(B=S)/L5
[27] T←⌊W÷M+4
[28] G←⌈B÷T
[29] I←⌊I1÷R÷B
[30] L←1+(W×G×I)÷1024
[31] S1←R-B×I
[32] S2←⌈S1÷T
[33] Z←⌈L+(S2×W)÷1024
[34] Z←(1+~2>I1)×Z
[35] L4:
[36] '      MACHINE SIZE: ';K
[37] 'WORK FILE SECTOR SIZE: ';W
[38] '      MASK LENGTH: ';M
[39] '      # OF RECORDS: ';R
[40] '      WORK FILE SIZE: ';Z
[41] →0
[42] L5:
[43] C←⌊C×0.9
[44] →L0
      V

```

BASIC Program

```
0010 PRINT 'ENTER MACHINE SIZE IN # OF K'
0020 INPUT K
0030 PRINT 'ENTER WORK FILE SECTOR SIZE IN # OF BYTES'
0040 INPUT W
0050 PRINT 'ENTER MASK LENGTH IN # OF BYTES'
0060 INPUT M
0070 PRINT 'ENTER # OF RECORDS IN INPUT FILE'
0080 INPUT R
0090 IF K<=16 GOTO 500
0100 IF K<=32 GOTO 520
0110 IF K<=48 GOTO 540
0120 Z=0
0130 C=3855
0140 S=C
0150 C=C+1
0160 T=2↑INT(LTW(C)+1E-10)
0170 IF T<=16 GOTO 190
0180 T=16
0190 A=2*(2*T-1+T)
0200 K1=7+((A+(C*(M+7)))/1024)
0210 IF K1>K GOTO 240
0220 IF C>R GOTO 420
0230 GOTO 150
0240 B=C-1
0250 IF B=S GOTO 480
0260 T=INT(W/(M+4))
0270 G=B/T
0280 IF G=INT(G) GOTO 300
0290 G=INT(G+1)
0300 I1=R/B
0310 I=INT(I1)
0320 L=((W*G*I)/1024)+1
0330 S1=R-(B*I)
0340 S2=S1/T
0350 IF S2=INT(S2) GOTO 370
0360 S2=INT(S2+1)
0370 Z=(S2*W)/1024+L
0380 IF Z=INT(Z) GOTO 400
0390 Z=INT(Z+1)
0400 IF I1<=2 GOTO 420
0410 Z=Z*2
0420 PRINT '          MACHINE SIZE: ',K
0430 PRINT 'WORK FILE SECTOR SIZE: ',W
0440 PRINT '          MASK LENGTH: ',M
0450 PRINT '          # OF RECORDS: ',R
0460 PRINT '          WORK FILE SIZE: ',Z
0470 STOP
0480 C=INT(C*.9)
0490 GOTO 140
0500 C=550
0510 GOTO 140
0520 C=1650
0530 GOTO 140
0540 C=2595
0550 GOTO 140
```


Appendix A: Sort Work File Size Charts

The number within the charts are the K-byte size of the work file. The numbers on the extreme right side of the chart are diskette format numbers. The lines in the chart show the maximum capacities at which that format diskette can be used. A zero entry in the chart means that a work file is not needed.

SYSTEM SIZE= 16K
 WORK FILE SECTOR SIZE= 128

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	9	12	12	17	17	17	17	17	17	33
500	0	0	9	12	14	17	34	44	44	66	66	66	66	66	66	128
750	7	11	13	34	40	50	50	66	66	96	96	96	98	98	98	190 ¹
1000	9	14	34	46	54	66	66	86	86	128	128	128	128	128	128	252
1500	26	40	50	66	78	98	96	128	130	190	190	190	192	192	192	378 ⁴
2000	34	54	66	88	104	128	128	170	172	252	252	252	254	254	254	
2500	42	66	82	108	130	160	160	212	214	316	316	316	318	318	318	
3000	50	80	98	130	154	192	190	252	256	378	378	378	380	380	380	
3500	58	92	112	150	180	222	222	294	298	440	440	440	444	444	444	
5000	82	130	160	214	256	316	316	420	424							
7500	122	194	238	320	382	474	472									
10000	160	258	316	426												
15000	240	384	474													
20000	318															
30000	476															

SYSTEM SIZE= 16K
 WORK FILE SECTOR SIZE= 256

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	9	10	12	14	14	17	17	17	17	22
500	0	0	9	12	14	16	34	38	44	54	52	66	66	66	66	86
750	8	11	13	34	42	46	50	56	66	78	78	98	98	98	98	128
1000	9	14	34	46	54	60	66	74	86	104	102	130	128	130	130	170
1500	28	40	50	66	80	88	96	110	130	154	152	192	192	194	194	252
2000	36	52	66	88	106	116	128	148	172	204	202	256	254	258	258	336
2500	44	64	82	108	132	146	160	184	214	254	252	320	318	322	322	420
3000	52	76	98	130	156	174	190	220	256	304	302	382	380	386	386	502
3500	60	88	112	150	182	202	222	256	298	354	352	446	444	450	450	586
5000	84	126	160	214	260	288	316	364	424	506	502	636	632	642	642	836
7500	124	186	238	320	388	430	472	544	634	756	752	952	946	960	962	
10000	164	248	316	426	516	574	628	724	844							
15000	246	370	474	636	772	858	940	1084								
20000	326	492	630													
30000	488	736	944													
40000	648	982														
50000	810															

SYSTEM SIZE= 16K
 WORK FILE SECTOR SIZE= 512

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	10	10	12	13	14	16	16	18	18	19
500	0	0	10	12	14	16	36	38	44	48	52	60	60	68	68	74
750	8	11	13	34	40	46	52	56	66	72	78	88	90	102	102	110
1000	10	14	36	44	52	60	68	74	86	96	102	116	118	134	134	146
1500	28	40	52	66	76	88	100	110	130	142	152	174	176	200	200	218
2000	36	54	68	86	102	116	132	148	172	188	202	230	234	266	266	288
2500	44	66	84	108	126	146	164	184	214	234	252	288	292	332	332	360
3000	54	78	100	130	152	174	196	220	256	280	302	344	350	396	398	432
3500	62	92	116	150	176	202	228	256	298	326	352	402	408	462	464	502
5000	88	130	166	214	250	288	326	364	424	464	502	572	582	660	662	718
7500	130	194	246	320	374	430	486	544	634	696	752	858	870	988	992	1074
10000	172	256	328	426	500	574	648	724	844	926	1002					
15000	256	384	490	636	748	858	970	1084								
20000	340	512	654	848	996											
30000	510	766	980													
40000	680	1020														
50000	848															
60000	1018															

SYSTEM SIZE= 32K
 WORK FILE SECTOR SIZE= 128

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	33	33	33	33	33	64
750	0	0	0	0	0	0	25	33	33	48	48	48	48	48	96	190
1000	0	0	0	0	27	33	33	43	43	64	128	128	128	128	128	252
1500	0	20	25	33	39	48	96	128	128	190	190	190	190	190	190	378
2000	17	27	33	43	104	128	128	170	170	254	254	252	252	254	252	
2500	21	33	82	108	128	160	160	212	212	316	316	316	316	316	316	
3000	25	78	98	128	154	190	190	252	254	378	378	378	378	378	378	
3500	29	90	112	150	178	222	222	294	296	442	442	440	440	442	440	
5000	82	128	160	212	254	316	316	420	422							
7500	120	192	238	316	380	472										
10000	160	254	316	420												
15000	240	380	474													
20000	318															
30000	476															

SYSTEM SIZE= 32K
 WORK FILE SECTOR SIZE= 256

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	27	33	33	33	33	43
750	0	0	0	0	0	0	25	28	33	39	39	48	48	49	98	128
1000	0	0	0	0	27	29	33	37	43	52	104	128	128	128	128	170
1500	0	20	25	33	39	43	96	110	128	154	154	192	190	192	192	252
2000	17	26	33	43	104	114	128	148	170	204	204	254	252	254	254	336
2500	21	31	82	108	128	142	160	184	212	254	254	316	316	318	316	420
3000	25	76	98	128	154	170	190	218	256	304	306	380	378	380	380	502
3500	29	88	112	150	180	198	222	256	298	356	356	442	440	444	442	586
5000	82	124	160	212	256	282	316	364	424	506	506	630	628	632	632	836
7500	122	184	238	318	382	422	472	544	634	758	758	944	940	948	946	
10000	162	244	316	424	508	562	628	724	844							
15000	240	364	474	634	760	842	940									
20000	320	484	632	844												
30000	480	726	946													
40000	638															
50000	796															
60000	956															

SYSTEM SIZE= 32K
 WORK FILE SECTOR SIZE= 512

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	27	29	30	33	33	38
750	0	0	0	0	0	0	25	29	33	36	39	43	44	49	98	112
1000	0	0	0	0	26	30	33	38	43	47	104	114	116	128	128	148
1500	0	20	25	32	37	44	96	112	128	140	156	170	172	192	192	220
2000	17	26	33	42	100	116	128	148	170	186	206	226	228	254	254	292
2500	21	32	82	104	124	144	160	186	212	232	258	282	286	318	316	366
3000	25	76	98	124	148	172	190	220	256	278	308	338	342	380	380	438
3500	29	88	114	144	172	200	222	258	298	324	360	394	398	444	442	510
5000	82	124	162	204	244	286	316	368	424	462	512	564	568	632	632	728
7500	122	186	242	304	364	428	472	550	634	692	766	844	850	948	946	1090
10000	162	246	320	404	484	570	628	732	844	922	1020					
15000	240	370	480	604	726	852	940	1096								
20000	320	490	640	804	966											
30000	480	736	958													
40000	638	980														
50000	796															
60000	956															

SYSTEM SIZE= 48K
 WORK FILE SECTOR SIZE= 128

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0	0	48	48	48	48	95
1000	0	0	0	0	0	0	0	0	43	64	64	64	64	64	64	126
1500	0	0	0	0	39	48	48	64	64	95	95	190	190	190	190	378
2000	0	0	33	43	52	64	64	85	170	254	252	254	252	254	254	
2500	0	33	41	54	64	160	160	212	212	316	316	316	316	316	316	
3000	0	39	48	64	154	190	190	252	254	378	378	378	378	378	378	
3500	29	45	56	150	178	222	222	294	296	440	440	440	440	442	442	
5000	41	128	160	212	254	316	316	420	420							
7500	120	192	238	316	378	472	472									
10000	160	254	316	420												
15000	238	380	474													
20000	316															
30000	472															

SYSTEM SIZE= 48K
 WORK FILE SECTOR SIZE= 256

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0	0	49	48	49	48	64
1000	0	0	0	0	0	0	0	0	43	52	52	64	64	64	64	85
1500	0	0	0	0	39	43	49	55	64	77	77	192	190	192	190	254
2000	0	0	33	43	52	57	64	73	170	204	204	254	252	254	254	338
2500	0	31	41	54	64	142	160	182	212	254	254	316	316	316	316	420
3000	0	37	48	64	154	170	192	218	254	304	304	380	378	380	378	504
3500	29	43	56	150	178	198	222	254	296	354	354	442	440	442	442	588
5000	41	122	160	212	254	282	316	362	420	504	504	630	628	630	628	838
7500	120	182	238	318	380	420	474	542	630	754	756	944	940	944	942	
10000	160	242	316	422	506	560	632	722	838							
15000	238	362	474	632	756	838	946									
20000	316	482	630	842												
30000	472	722	944													
40000	628	962														
50000	784															
60000	942															

SYSTEM SIZE= 48K
 WORK FILE SECTOR SIZE= 512

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0	0	43	44	49	49	55
1000	0	0	0	0	0	0	0	0	43	47	52	57	57	64	64	73
1500	0	0	0	0	37	43	49	56	64	70	77	170	172	192	192	218
2000	0	0	33	42	49	57	64	73	170	186	204	226	226	256	254	290
2500	0	31	41	52	61	142	160	184	212	232	256	282	284	318	318	362
3000	0	37	49	62	146	170	192	220	254	278	306	338	340	382	382	436
3500	29	43	57	144	170	198	222	256	296	324	356	394	396	444	444	508
5000	41	122	160	204	242	282	316	364	422	462	508	562	564	634	632	722
7500	120	182	240	304	362	420	474	546	634	692	760	840	846	950	948	1084
10000	160	242	320	404	480	560	632	726	842	922	1012					
15000	238	362	476	604	720	838	946	1088								
20000	316	482	636	804	958											
30000	472	722	952													
40000	628	962														
50000	784															
60000	942															

SYSTEM SIZE= 64K
 WORK FILE SECTOR SIZE= 128

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1000	0	0	0	0	0	0	0	0	0	0	0	0	64	64	64	126
1500	0	0	0	0	0	0	0	64	64	95	95	95	95	95	95	189
2000	0	0	0	0	0	64	64	85	85	126	127	127	252	254	252	0
2500	0	0	0	54	64	80	80	106	106	316	316	316	316	316	316	0
3000	0	0	48	64	77	95	95	254	254	378	378	378	378	378	378	0
3500	0	45	56	74	89	222	222	294	294	440	440	440	440	440	440	0
5000	41	64	80	212	254	316	316	420	420							
7500	60	190	238	316	378	472	472									
10000	160	254	316	420												
15000	238	378	472													
20000	316															
30000	472															

SYSTEM SIZE= 64K
 WORK FILE SECTOR SIZE= 256

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1000	0	0	0	0	0	0	0	0	0	0	0	0	64	64	64	85
1500	0	0	0	0	0	0	0	55	64	77	77	95	95	95	95	127
2000	0	0	0	0	0	57	64	73	85	102	102	127	254	254	252	336
2500	0	0	0	54	64	71	80	91	106	254	254	316	316	316	316	420
3000	0	0	48	64	77	85	95	218	254	304	304	378	378	380	378	504
3500	0	43	56	74	89	198	222	254	296	354	354	440	442	442	440	588
5000	41	61	80	212	254	282	316	360	420	504	504	628	630	630	628	838
7500	60	182	238	316	378	420	472	540	630	754	754	942	942	944	940	
10000	160	242	316	420	504	560	628	718	840							
15000	238	362	472	628	754	838	942									
20000	316	482	628	838												
30000	472	720	940													
40000	628	960														
50000	784															
60000	940															

SYSTEM SIZE= 64K
 WORK FILE SECTOR SIZE= 512

RECORDS	MASK LENGTH															
	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
250	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
500	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1000	0	0	0	0	0	0	0	0	0	0	0	0	57	64	64	73
1500	0	0	0	0	0	0	0	55	64	70	77	85	85	95	95	109
2000	0	0	0	0	0	57	64	73	85	92	102	113	226	254	252	290 ³
2500	0	0	0	52	61	71	80	91	106	230	254	282	280	316	316	362
3000	0	0	49	62	73	85	95	218	254	276	304	336	336	380	378	434
3500	0	43	56	72	85	198	222	254	296	322	356	392	392	442	440	506 ⁶
5000	41	61	80	204	242	282	316	362	422	458	506	560	558	630	628	722
7500	60	182	238	306	362	422	472	542	632	686	758	838	836	944	940	1080 ⁸
10000	160	242	318	406	480	562	628	722	842	912	1010					
15000	238	362	474	608	720	842	942	1080								
20000	316	482	632	810	958											
30000	472	720	946													
40000	628	960														
50000	784															
60000	940															

Appendix B. Glossary

address out: A type of sort in which the relative record numbers of the sorted records are provided as output.

basic exchange: A diskette exchange format that uses 128-byte sectors.

blocked: A term used to indicate that one or more logical records are written in each sector (physical record).

bytes: Eight bits used as a processing unit and a unit of measure for the 5110.

complex file name: A file name made up of two or more simple file names with a period between names. Up to 17 characters, including the periods, are allowed in a complex file name.

cylinder: The area of a diskette that can be accessed without moving the access mechanism.

cylinder 0: The outermost cylinder on the diskette that is used to record information about the data on the diskette.

data set: Another name for a file.

data set label: A record that defines a file. The name, size, location, and organization of a file are defined in the data set label.

ERMAP: A sector in cylinder 0 that is used to record the defective cylinders on the diskette.

exchange: A diskette format or a data organization that allows for the exchange of data between different systems.

extended label area: Additional area of a diskette that is used for data set labels.

file: An organized collection of related records treated as a unit (data set).

format: The size of the sectors and the number of sectors in a cylinder of a diskette.

H-type exchange: A diskette exchange format that uses 256-byte sectors.

IMF: IMFs (internal machine fixes) are supplied by IBM to correct machine problems.

logical record: All the data that relates to a common identifier.

physical record: The amount of data read at one time governed by the hardware device and media.

relative record number: The number of the record within a file relative to the beginning of the file.

sector: Part of a track on a diskette that contains one physical record.

simple file name: A file name of up to 8 characters. The first character must be uppercase alphabetic (A-Z). Other characters may be alphanumeric (A-Z, 0-9).

spanned: A logical record spans the sector (physical record) boundaries.

track: Part of a cylinder on a diskette.

unblocked: One logical record is written in each sector (physical record).

unspanned: A logical record does not span the sector (physical record) boundaries.

Appendix C. Error Codes

The error codes in this appendix are arranged in ascending number order. Each customer support function is assigned a group of numbers. The group of numbers assigned to each function are:

Numbers Function

000-249	See your language reference manual
250-264	Tape data recovery
265-274	Tape header recovery
275-299	Diskette recovery
300-349	Sort
350-374	Tape-to-diskette and diskette-to-tape copy
375-399	Tape-to-tape copy
400-424	Diskette-to-diskette copy
425-429	Compress
430-449	Label display
450-474	Initialization
475-499	Loader
500-999	See your language reference manual

TAPE DATA RECOVERY

Error	Meaning	User Response
250	An invalid file number was entered.	Press the EXECUTE key to get the option to stop or specify another file.
251	Invalid file type for data recovery. Only file types 01, 02, 03, or 08 may be used.	Press the EXECUTE key to get the option to stop or specify another file.
252	Invalid entry mode.	Press the EXECUTE key to get the option menu.
253	Read error on a header label.	Press the EXECUTE key to get the option menu. Then use the tape header recovery function to correct the header.

TAPE HEADER RECOVERY

Error	Meaning	User Response
265	No header that caused a read error was found.	Select another option.
266	The header causing the read error cannot be recovered.	Select another option.
267	An incorrect length parameter was entered.	Reenter the parameter, making sure the length is correct. The input line following the parameter must be blank.
268	The data that was entered is not valid for the field requested.	Enter valid data.
269	Invalid entry was made for the selection of an option.	Enter the correct option entry.
270	Invalid tape file number.	Enter a correct tape file number.
271	R/W storage is not large enough to copy the file.	Try the recovery function again without features or IMFs loaded.
272	A two-tape-drive copy was specified and the input and output drive numbers are the same.	Specify different drive numbers for the input and output file.

DISKETTE RECOVERY

275	Invalid entry was made for the selection of an option.	Enter a correct option entry.
276	Invalid diskette drive number.	Enter 1, 2, 3, or 4 to select the diskette drive.
277	Invalid diskette file number.	Enter a correct file number.
278	No read errors were found in the volume label.	Select another option.
279	Invalid entry for the label extension indicator.	Enter a valid label extension indicator.
280	Invalid entry for the volume surface indicator.	Enter a valid volume surface indicator.

DISKETTE RECOVERY (continued)

Error	Meaning	User Response
281	Invalid entry for the physical record length.	Enter a valid physical record length.
282	Invalid entry for the physical record sequence.	Enter a valid physical record sequence.
283	Volume label error.	Retry recovery using option 1.
284	No read errors were found in the header you specified.	Select another option.
285	Invalid entry for the header label.	Make a valid header label entry.
288	Sector size is not supported.	Use a different diskette that has a valid format.
289	Header label error.	Retry recovery using option 2.
290	File specified for data recovery is not file type 2, 8, or 9, or the file specified contains no data.	Data cannot be recovered for other than type 2, 8, or 9 files.
291	No read errors were found for the file you specified.	Select another option.
292	The file type is 9, but sequential relocate is not permitted.	Data recovery cannot be performed on this file.
293	Unable to recover data on this file. Data can be copied to another file on the same or different diskette.	Continue with the function; specify the file number you want the data copied to.
294	R/W storage is not large enough to copy the file.	Try the recovery function again without features or IMFs loaded.
295	The diskette selected to copy to has a different format than the input diskette.	Use a diskette for output that has the same format as the input diskette.

SORT ERROR CODES

300	Response to option was: – Not numeric – Not one of the valid responses	Enter a valid option number.
301	Invalid name in a sort statement.	Enter, starting in position 1, one of the valid names: FILE, SORT, MASK, ALTS, WORK, or STOP.

SORT ERROR CODES (continued)

Error	Meaning	User Response
302	A space appeared in a sort statement parameter.	Delete the space from the statement, or enter STOP to end the sort.
303	The sort-group-name in the FILE statement is too long.	Shorten the name to 8 characters, or enter STOP to end the sort.
304	The last parameter of a statement was not followed by blanks or the SORT statement contains an invalid SORT type.	Make sure blanks follow the sort statement, or enter STOP to end the sort.
	The logical not character (¬) appears on the screen in a sort control statement.	Use the following procedure and add the calculated number of comment statements to the control group: <ol style="list-style-type: none"> 1. Subtract 123 from the input file record length. 2. Divide the result from Step 1 by 64. 3. Round the result from Step 2 up to the next whole number.
305	Alphameric characters were expected. The logical not character (¬) appears on the screen in a sort control statement.	Change the entry to alphameric characters, or enter STOP to end the sort. See three-step procedure under Error 304.
306	Alphabetic characters were expected. The logical not character (¬) appears on the screen in a sort control statement.	Change the entry to alphabetic characters, or enter STOP to end the sort. See three-step procedure under Error 304.
307	Numeric characters were expected. The logical not character (¬) appears on the screen in a sort control statement.	Change the entry to numeric characters, or enter STOP to end the sort. See three-step procedure under Error 304.
308	Missing comma between input file and output file parameters.	Enter the comma, or enter STOP to end the sort.
309	File number contains more than four digits.	Enter file number with four digits or less, or enter STOP to end the sort.
310	File name contains more than 17 characters.	Enter file name with 17 characters or less, or enter STOP to end the sort.

SORT ERROR CODES (continued)

Error	Meaning	User Response
311	Comma was entered in place of the diskette drive number.	Enter the diskette drive number, or enter STOP to end the sort.
312	File number and name of the input or control file do not match.	Enter the correct file number and name, or mount the correct diskette, or enter STOP to end the sort.
313	Input file and output file are the same file.	Specify different files for input or output, or enter STOP to end the sort.
314	Work file is the same as the input file or output file.	Specify a work file that is different from the input or output file, or enter STOP to end the sort.
315	Output file name contains more than 8 characters or has more than 8 characters between periods.	Specify a valid name for the output file, or do not use an output file name, or enter STOP to end the sort.
316	Invalid use of periods in a complex output file name.	Specify a valid name for the output file, or delete the output file name, or enter STOP to end the sort.
317	One of the names reserved for system use, SYSAREA or ERRORSET, was specified for the output file name.	Change the output file name, or delete the output file name, or enter STOP to end the sort.
318	ALTS statement contains an invalid character. The logical not character (¬) appears on the screen in a sort control statement.	Enter only hexadecimal characters (0-9, A-F), or enter STOP to end the sort. See three-step procedure under Error 304.
319	Invalid number of characters between commas in the ALTS statement.	Enter 4 hexadecimal characters between commas, or enter STOP to end the sort.
320	R/W storage is not large enough to contain a control file physical record, or an ** statement (sort group name) was not found.	Specify another file or enter STOP to end the sort.

SORT ERROR CODES (continued)

Error	Meaning	User Response
321	Control group was not found.	Specify a correct name or file, or enter STOP to end the sort.
322	Too many records were encountered in a control group before the specified group was found.	Specify another group name or enter STOP.
323	Logical record length of control file not 64.	Specify another file or enter STOP.
324	Control file is not a record I/O file with blocked-spanned records.	Specify another file or enter STOP.
325	Too many records in input file or the 5110 user storage is not large enough to do the sort.	Respond to the options.
326	R/W storage is not large enough to do the sort.	Respond to the options.
327	R/W storage is not large enough to contain the control group that was found.	Specify another group or enter STOP.
328	Invalid length field for BASIC internal numeric short or long data type.	Enter 4 for short precision, or 8 for long precision, or enter STOP to end the sort.
329	Mask location is not within logical record length.	Correct the position or length of the mask field, or enter STOP to end the sort.
330	Mask position or length entry contains too many digits.	Correct mask position (maximum of four digits) or length (maximum of two digits) entry, or enter STOP to end the sort.
331	No mask position or length entry.	Specify a mask position and length entry, or enter STOP to end the sort.
332	Mask position or length entry must be greater than zero.	Correct the error, or enter STOP to end the sort.
333	Mask type is not A, D, F, G, N, or O.	Correct the error or enter STOP to end the sort.
334	Total MASK length is greater than 64.	Reduce the total MASK length, or enter STOP to end the sort.

SORT ERROR CODES (continued)

Error	Meaning	User Response
335	The block length of an unblocked-unspanned file is greater than the sector size.	Specify a different file, or enter STOP to end the sort.
336	The block length for a blocked-spanned file does not equal the sector size.	Specify a different file, or enter STOP to end the sort.
337	A work file is needed.	Enter a WORK statement, or enter STOP to end the sort.
338	Work file too small.	Enter option. Return to the calling language to increase size of work file, or enter STOP to end the sort.
339	Name of the output file already exists on the diskette.	Change name of output file, or change diskette, or enter STOP to end the sort.
340	Output file too small.	If detected when the SORT statement was entered, specify another output file, or enter STOP to end the sort. If detected during the sort, the sort completed display appears; respond to the options.
341	Output file or work file is a used file.	Specify unused files, or enter STOP to end the sort.
342	Multivolume file was specified but cannot be used by the sort program.	Specify a different input file, or enter STOP to end the sort.
343	The referenced file application type not supported.	Specify a different file, or enter STOP to end the sort.
344	The referenced file exchange type not supported.	Specify a different file, or enter STOP to end the sort.
345	Variable-length logical records were specified but cannot be used by the sort program.	Specify a different file, or enter STOP to end the sort.

SORT ERROR CODES (continued)

Error	Meaning	User Response
346	The input file must be a blocked-spanned or unblocked-unspanned file, or the control file must be a blocked-spanned file.	Specify a different file, or enter STOP to end the sort.
347	Input file does not contain any records.	Specify a different input file, or enter STOP to end the sort.
349	The sector size of the control, input, output, or work file is not supported.	Specify a file with a sector size of 128, 256, or 512 bytes.

TAPE-TO-DISKETTE AND DISKETTE-TO-TAPE ERROR CODES

350	Response to option was: – Not numeric – Not one of the valid responses	Enter correct response.
351	Invalid file number was entered.	Enter the file number, up to four digits.
352	Tape file name is invalid and will not be copied to the diskette file.	Enter a valid file name. This name will be used as the file name for the copied diskette file. (See Chapter 1).
353	File name that was entered is invalid.	Enter a valid file name. (See Chapter 1).
354	Diskette file is write-protected.	Restart the function and specify a non-write-protected diskette file for output.
355	Diskette file expiration date is not blank.	Restart the function and specify an unused diskette file for output.
356	File name of the tape file or the file name entered from the keyboard already exists on the diskette file (see Chapter 1).	Enter a valid file name that does not already exist on the diskette file.
357	Tape file to be copied from is unused. An unused file cannot be copied.	Make sure you have not specified an unused file.

TAPE-TO-DISKETTE AND DISKETTE-TO-TAPE ERROR CODES
(continued)

Error	Meaning	User Response
358	Tape file to be copied from is type 4 or 5. These file types cannot be copied.	Make sure you have not specified a type 4 or 5 file.
359	Tape file to be copied from is type 6 or 26. These file types cannot be copied.	Make sure you have not specified a type 6 or 26 file.
361	Tape file to be copied from is type 16, 17, 18, or 19. These file types cannot be copied.	Make sure you have not specified a type 16, 17, 18, or 19 file.
362	Tape file has an invalid file type.	Make sure the tape file has a valid file type.
363	Diskette file has an invalid file type.	Make sure the diskette file has a valid file type.
364	Diskette file to be copied from is type 26. These file types cannot be copied.	Make sure you have not specified a file type 26 file.
365	Diskette file to be copied from is type 10. A type 10 file cannot be copied.	Make sure you have not specified a file type 10 file.
366	Diskette file to be copied from is unused (EOD = BOE). An unused file cannot be copied.	Make sure you have not specified an unused file.
367	Diskette sector size is 1024 bytes. This size is not supported.	Use a different diskette that has a valid format.
368	The header label on the diskette is not correct.	Use diskette recovery function to create a valid header.
369	The logical record length in the tape header is zero.	Use diskette recovery function to create a valid header.
370	Diskette file to be copied from is type 15. This file type cannot be copied.	Make sure you have not specified a type 15 file.

TAPE-TO-TAPE ERROR CODES

375	Invalid entry was made to select an option.	Reenter the option entry.
376	Invalid file number was entered.	Reenter file number.
377	Invalid tape drive number was entered.	Reenter tape drive number.
378	Input file contains no data (file type 00).	Reenter input file number.
379	Output tape file is not marked.	Mark the output file and restart the copy operation.

DISKETTE-TO-DISKETTE ERROR CODES

Error	Meaning	User Response
400	Invalid response to an option.	Enter the correct response.
401	Invalid file number was entered.	Reenter the file number, up to four digits.
402	Invalid diskette drive number was entered.	Reenter correct diskette drive number: 1, 2, 3, or 4.
403	The output diskette has a different sector size, a smaller format number (less capacity), or fewer cylinders set aside for the extended label area than the input diskette. When the label area causes the error, the number of cylinders for each diskette is displayed.	Use an output diskette that has the same sector size, the same or larger format number, and the same or larger label area than the input diskette.
404	Input diskette sector size is not equal to that of the output diskette.	Make sure both diskettes have the same sector size.
405	Output file is not marked.	Mark the output file or use a different file.
406	Input diskette sector size is 1024 bytes. This size is not supported.	Use a different diskette that has a valid format.
407	Output diskette sector size is 1024 bytes. This size is not supported.	Use a different diskette that has a valid format.
408	Input diskette file contains no data.	Reenter the input file number.
409	Invalid file name was entered.	Enter a valid file name (see Chapter 1).
410	The output file has the file-protect indicator on.	Use a file that is not file-protected for output.
411	The expiration date on the output file is not blank.	Use a different file with a blank expiration date for output.
412	No files exist on the input diskette.	Use a diskette that contains files.
413	The diskette cover was not opened.	Press the EXECUTE key if the input files are on the same diskette, or insert the correct diskette.

DISKETTE-TO-DISKETTE ERROR CODES (continued)

Error	Meaning	User Response
414	During image copy on a one drive system, the original diskette was not used when prompting for the <i>from</i> diskette.	Restart image copy or return to system.
415	During image copy, the output diskette has different sector size, diskette type, or extended labels than the input diskette.	Initialize the output diskette to the same attributes as the input diskette.
416	During image copy, the response indicated two diskette drives were to be used and the same drive number was used for the <i>from</i> and the <i>to</i> drive.	If two drives are to be used, a different number must be specified for the <i>from</i> and the <i>to</i> drives.

This page intentionally left blank.

COMPRESS ERROR CODES

Error	Meaning	User Response
425	Invalid entry was made to select an option.	Reenter the option entry.
426	Input diskette sector size is 1024 bytes. This size is not supported.	Use a different diskette that has a valid format.

LABEL DISPLAY

430	Invalid selection of options or the input line is not correct format.	Reenter, starting in position 1, the options given in the prompting message.
431	A file number of zero was specified or the first file number of a range was greater than the second.	Enter a valid file number or range.
432	A file number larger than what can exist on the media was specified.	Select another option.
433	The volume is access-protected. Labels cannot be displayed when volume is access-protected.	Select another option.
434	Diskette drive door was opened before last file header was read.	Select another option.
436	The end of the marked tape was found.	Respond to the message by selecting an option.
437	The tape is not marked for the tape header that was specified.	Respond to the message by selecting an option.

INITIALIZATION ERROR CODES

450	INIT is not in correct position.	Make sure INIT is in positions 1 to 4 of the initialization command.
452	Characters used for volume identification are not alphameric.	Make sure all characters are alphameric.
453	VOL-ID field of initialization command contains more than 6 characters.	Make sure not more than 6 characters are entered for the volume identification field.

INITIALIZATION ERROR CODES (continued)

Error	Meaning	User Response
454	Characters used for owner identification are not alphameric.	Make sure all characters are alphameric.
455	OWNER-ID field of initialization command contains more than 14 characters.	Make sure not more than 14 characters are entered for the owner identification field.
457	Invalid format specified.	The valid entries are the digits 1 through 9.
458	Sector sequence number is not a two-digit number.	Enter sector sequence number as a two-digit number.
459	Invalid number specified for sector sequence.	Enter correct number; see chart in <i>Diskette Format</i> for valid entries.
460	Invalid entry made for increasing the number of labels on format 7, 8, or 9 diskettes.	Valid entries are blank and digits 1 through 9. The digits must be entered in position 1 with position 2 blank. Reenter.
461	Diskette drive number was not found in position 1.	Enter diskette drive number in position 1; valid entries are 1, 2, 3, or 4.
462	A nonnumeric character was found for the diskette drive number in position 1.	Enter 1, 2, 3, or 4 to select the diskette drive in position 1.
464	Response to option was: – Not in position 1 – Not numeric – Not one of the valid responses	Retry. If error occurs again, diskette could be defective.
465	More than two bad cylinders found during initialization.	Retry. If error occurs again, diskette could be defective.
466	A comma was not found where one was expected in the initialization command.	Make sure all commas are present; a comma must separate each parameter.
467	The position after the two-digit sector sequence number is not blank.	Make sure a blank follows the sector sequence number.

INITIALIZATION ERROR CODES (continued)

Error	Meaning	User Response
468	A secure data set was found.	Respond to the message by selecting an option.
469	A write-protected data set was found.	Respond to the message by selecting an option.
470	A used data set (EOD is greater than BOE) was found.	Respond to the message by selecting an option.
471	A data set with a nonblank expiration field was found.	Respond to the message by selecting an option.
472	A Diskette 1 type diskette is in the drive, and a format of 4 through 9 was specified.	Format 1, 2, or 3 must be specified for Diskette 1 type diskettes.
473	A Diskette 2 type diskette is in the drive, and a format of 1, 2, or 3 was specified.	Formats 4 through 9 must be specified for Diskette 2 type diskettes.
474	Cylinder zero is defective. Cylinder 0 must be defect free.	Retry; if error occurs again, diskette could be defective.

LOADER ERROR CODE

475	Invalid number entered for an option.	Enter a valid number.
476	The file to be loaded from is not a type 22 file.	Feature programs must be on a type 22 file. Press the EXECUTE key to get the option to stop or continue.
477	The program you are trying to load does not have a valid feature code.	Press the EXECUTE key to get the option to stop or continue.
478	You are trying to load a feature program that is already loaded.	Press the EXECUTE key to get the option to stop or continue.
479	Not enough space remains in the work area to load the feature program.	Press the EXECUTE key to get the option to stop or continue.
480	You are trying to load the translated sort messages, and they are already loaded.	Press the EXECUTE key to get the option to stop or continue.

LOADER ERROR CODE (continued)

Error	Meaning	User Response
481	An invalid or incorrect hexadecimal character was entered.	Correct and reenter the character.
482	There is no IMF for the problem number you entered.	Enter a different problem number.
483	The IMF is not for your system.	Enter a different problem number.
484	The IMF problem number you entered already exists.	Enter a different problem number for your IMF.
485	There is no preformatted record in the IMF file for your key-entered IMF.	You cannot enter the IMF. Get a different IMF file from IBM.
486	An invalid or incorrect file number was entered.	Enter the file number, up to four digits.
487	Blank in the file name.	Enter a valid file name (see Chapter 1).
488	Too many characters entered for a response to a prompt.	Enter the correct number of characters for the response to the prompt.

Index

)LINK 3
)SORT 3, 57

ERMAP label 47
exchange diskette files 18

additional file labels 17
ALTS statement 72
ATTN key 3

collating sequence 73
color label 16
command INIT 12
comment statement 68
compress diskette 41
control file 64
copy
 diskette to diskette 19
 diskette to tape 33
 tape to diskette 29
 tape to tape 37

data file recovery 52
device numbers 4
diskette
 compress 41
 data file recovery 52
 exchange files 18
 file header label 48
 files copied 35
 format 16
 header label recovery 51
 image copy 24
 initialization 9
 label display 44
 recovery 49
 sort 57
 to diskette copy 19
 to tape copy 33
 volume label recovery 50
display label 43

FILE statement 67
file
 control 64
 input 61
 labels 14
 labels additional 17
 name rules 5
 names 1, 5
 number 1
 output 62
 procedure 78
 types copied 31, 35
 work 59
files exchange 18
FORMAT 12
format diskette 16

header label recovery
 diskette 51
 tape 55

image copy 24
 one-diskette-drive copy 24
 two-diskette drive copy 27
IMF 8
INIT command 12
input file 61

key ATTN 3

label color 16
label display 43
labels file 14
labels file additional 17
LINK 3
loader 7

MASK statement 70
messages 3

names file 1, 5
number file 1
numbers device 4

operation 3
output file 62
output file size 62
OWNER-ID 12

parameters 4
performance considerations sort 79
procedure file 78

recovery
diskette data file 52
diskette header label 51
diskette volume label 50
tape data 53
tape header 55
rules for file names 5

sector sequence 17
sequence collating 73
sequence sector 17
SEQUENCE-NUMBER 12
size output file 62
size work file 63
SORT statement 69
SORT GROUP NAME statement 68
sort diskette 57
sort performance considerations 79
statement
ALTS 72
comment 69
FILE 67
MASK 70
SORT 69
SORT GROUP NAME 68
WORK 74

tape
data recovery 53
file header label 46
files copied 31
header recovery 55
label display 43
to diskette copy 29
to tape copy 37

UTIL SORT 3, 57

volume label 47
volume label recovery 50
VOL-ID 12

WORK statement 74
work
file 63
file size 63
file storage requirements 63

READER'S COMMENT FORM

Please use this form only to identify publication errors or request changes to publications. Technical questions about IBM systems, changes in IBM programming support, requests for additional publications, etc. should be directed to your IBM representative or to the IBM branch office nearest your location.

Error in publication (typographical, illustration, and so on). No reply.

Page Number Error

Inaccurate or misleading information in this publication. Please tell us about it by using this postage-paid form. We will correct or clarify the publication, or tell you why a change is not being made, provided you include your name and address.

Page Number Comment

Note: All comments and suggestions become the property of IBM.

Name _____

Address _____

● No postage necessary if mailed in the U.S.A.

Cut Along Line

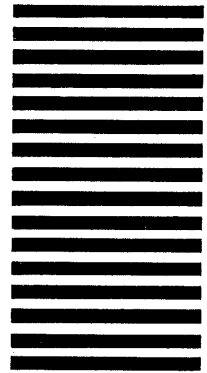
Fold

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
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)**

**General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)**

IBM 5110 Customer Support Functions Reference Manual Printed in USA SA21-9311-1



International Business Machines Corporation

**General Systems Division
4111 Northside Parkway N.W.
P.O. Box 2150
Atlanta, Georgia 30301
(U.S.A. only)**

**General Business Group/International
44 South Broadway
White Plains, New York 10601
U.S.A.
(International)**

IBM 5110 Customer Support Functions Reference Manual Printed in USA SA21-9311-1

**IBM 5110
Customer Support
Functions Manual**

SA21-9311-1