96768400

CONTROL DATA
CORPORATION

MAGNETIC TAPE UTILITY PROCESSOR REFERENCE MANUAL

CONTROL DATA®
1700 COMPUTER SYSTEMS
CYBER 18 COMPUTER SYSTEMS

REVISION RECORD

REVISION	DESCRIPTION
A	Manual released
5/76	
Publication No. 96768400	

REVISION LETTERS I, O, Q AND X ARE NOT USED

Address comments concerning this manual to:
Control Data Corporation
Publications and Graphics Division
4455 Eastgate Mall
La Jolla, California 92037
or use Comment Sheet in the back of this manual.

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

Page	Revision	SFC[†]
Cover	—	
Title Page	—	
ii	—	
iii	A	
v	A	
vii	A	
1-1	A	
1-2	A	
2-1	A	
2-2	A	
2-3	A	
2-4	A	
2-5	A	
2-6	A	
3-1	A	
3-2	A	
3-3	A	
3-4	A	
3-5	A	
3-6	A	
4-1	A	
4-2	A	
4-3	A	
4-4	A	
A-1	A	
B-1	A	
B-2	A	
C-1	A	
C-2	A	
C-3	A	
C-4	A	
D-1	A	
D-2	A	
Comment Sheet	—	
Envelope	—	
Back Cover	—	

Page	Revision	SFC[†]

[†] Software. Feature Change

PREFACE

This manual describes the magnetic tape utility processor used with CONTROL DATA® 1700 Computer Systems and CYBER 18 Computer Systems. It is assumed that users of this manual are familiar with 1700 Computer Systems and CYBER 18 Computer Systems.

The magnetic tape utility processor user should be familiar with the following manuals:

Publication	Publication Number
Real-Time Operating System Version 3 (CYBER 18-10/20) Reference Manual	96769500
Real-Time Operating System Version 3 (CYBER 18-17) Reference Manual	96769560
MSOS Version 4 Reference Manual	60361500
MSOS Version 4 Installation Handbook	39520900

This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or undefined parameters.

CONTENTS

APPENDIXES

TABLES

INTRODUCTION 1

The magnetic tape utility processor is a generalized tape utility for the 1700 Series computer systems or CYBER 18 Series computer systems under control of either MSOS (Mass Storage Operating System) or RTOS (Real-Time Operating System). The magnetic tape utility processor provides a set of functional operations to process magnetic tape files created on Control Data Corporation's or other manufacturers' equipment. The magnetic tape utility processor's capabilities encompass the more complex record formats and labeling structures usually found in data processing environments.

The magnetic tape utility processor provides a medium through which a 1700 Series system or a CYBER 18 Series system may be used to reduce input/output processing, such as off-line printing of listable tapes on other systems. The magnetic tape utility processor also provides data manipulation of 1700 Series or CYBER 18 Series system-created tapes to augment other 1700 Series or CYBER 18 Series features, such as readable tape dumps and improvement of input/output efficiency, through the blocking and deblocking of data files.

SYSTEM CONFIGURATION

The magnetic tape utility processor is designed to run under control of MSOS and RTOS with a minimum of 4K of available storage beyond the basic system.

Under MSOS, the magnetic tape utility processor is installed in the program library and executed as a background job. The background area may not be in the upper bank of core.

Under RTOS, the magnetic tape utility processor is installed on the system library tape and executed as a background job.

MINIMUM HARDWARE CONFIGURATION

The minimum hardware required by the magnetic tape utility processor includes the minimum hardware required by RTOS, with two magnetic tape transports and an optional line printer. The line printer is needed if the PRINT and DUMP functions are used extensively. When using MSOS, the minimum hardware required for the magnetic tape utility processor includes a minimum 3K available for background processing, a teletypewriter or conversational display terminal, tape transports, a disk pack, and a line printer.

FEATURES

The features of the magnetic tape utility processor are described below:

Copy	Copies data from tape to tape
Dump	Prints a dump from the tape to the line printer in either hexadecimal or character mode
Initialize	Writes volume I tape headers with volume serial numbers on any tape
Labels	Provides for the reading and writing of standard tape labels and trailer records
Print	Prints standard listings from a tape
Selection	Selects records for processing, based on specified criteria
Verify	Verifies data on two tape files and checks for equality
Blocking	Allows the processing of either blocked or unblocked records
Conversion	Allows the selection of the following data conversion options: ASCII to EBCDIC EBCDIC to ASCII ASCII to BCD BCD to ASCII EBCDIC to BCD BCD to EBCDIC
Positioning	Tapes may be positioned to given records, blocks, or records within blocks, using any of the processing functions.
Record formats	All functions of the magnetic tape utility processor can process the following record formats: Variable length unblocked records Variable length blocked records Fixed length records Fixed length blocked records Undefined records; for example, fixed or variable length records that follow no standard or are intermixed
Parameter input error correction	When an incorrect parameter is input to any of the functions, the magnetic tape utility processor allows for the correction of the errors by use of the following procedures: To delete a line of type, use a rubout, then a line feed followed by a carriage return. If there is a syntax error in the function statement, the magnetic tape utility processor allows a re-entry of only the parameters in error. The unrecognized parameters will be retyped by the magnetic tape utility processor for ease in discerning which parameter is in error.

MAGNETIC TAPE UTILITY PROCESSOR LOADING INSTRUCTIONS

The magnetic tape utility processor is loaded into memory from the MSOS or RTOS program library. The loading instructions and utility response for MSOS are as follows:

Loading Instruction	Utility Response
*MTUP	**TAPE UTILITY** *NEXT:

The loading instructions and utility response for RTOS are as follows:

Loading Instruction	Utility Response
MTUP	**TAPE UTILITY** *NEXT:

The magnetic tape utility processor executes through the use of a declarative and operational utility control language. Declarative control statements define file characteristics and internal processing options such as data conversion and blocking. The operational control statements define functional operations such as tape copy and transfer of data on tape to the line printer. Control statements consist of a mnemonic describing the function to be executed and a list of parameters. Parameters may be either optional or required in a control statement, depending on the function. If a parameter is required in a function, but omitted, a message is generated requesting that the parameter be stated. If a parameter is optional in a function, but required because of previous options, a message is generated requesting that the parameter be stated.

DECLARATIVE CONTROL STATEMENTS

The declarative control statements are used to define the characteristics of the files on which a function is to be applied. The declarative control statements also control the following functions:

Blocking

Data conversion

Labeling

Record formatting

File termination

Positioning of volumes

Data processing

Setting system and expiration dates

The parameters in a declarative control statement may be in any order, except that the UNIT parameter must be the first parameter in the list.

OPEN STATEMENT

The OPEN statement defines the characteristics of magnetic tape files used in a process. The statement is used to assign blocking factors, record lengths, record formats, label formats, conversion information on either input or output data files, and positioning to records or files.

The OPEN statement must be issued for all logical units used by a function before the function's use.

The control statement format is:

OPEN, unit lu, label, block, data, SFnn, BRnn, select,
LBnnnnn, LRnnnnn, LCnnnnn

Where: unit is the device type to which OPEN applies

I	Input device
O	Output device
VF	Verify file
PR	List device

lu is a one- to two-digit logical unit number of the device. If lu is omitted, the standard device for each type is opened.

label is the type of label processing.

SL	Process labels as standard tape labels (see appendix B).
BL	Bypass labels. If BL is specified, the tape is positioned to the first record of the specified file. No attempt is made to verify the volume, header, or trailer labels. BL must not be specified for output files.
NL or blank	No labels. The tape is not positioned.

block is the record format (see appendix C).

V	Variable length records. Within the first 32 bits of a record, 16 bits must contain a binary number specifying the record length. The last 16 bits must be zero filled.
VB	Variable length blocked records. Within the first 32 bits of a block, 16 bits must contain a binary number specifying the block length. The next 16 bits must be zero filled. The first variable length record of the block must follow the 32 bits.
F	Fixed length records. Each record must be exactly the same length.
FB	Fixed length blocked records. Each block must contain an integral number of fixed length records. There is no requirement that the block length remain fixed. Fixed blocking usually has a maximum length.
U or blank	Undefined record format. Each record is treated as an undefined variable length record. Blocking cannot be used in the undefined format. Print files may be specified as U only.

Data is the type of data conversion. To specify conversion, both the input and output file data formats must be defined.

B	BCD data (seven-track only)
E	EBCDIC data
A or blank	ASCII or binary data

SF specifies skip nn files before processing.

nn	One- to two-digit decimal number

BR specifies bypass nnnnn records before processing.

nnnnn	One- to five-digit decimal number less than 32,767

select is the selected data to be processed. Selective processing occurs only on input files. Processing files that are not 1700 source data, listable, or relocatable records produces unreliable results.

PNAM = 'xxxxxx' Positioning of a data file to a name statement or name block containing the name specified by xxxxxx, where xxxxxx is a one- to six-character name. If xxxxxx is not specified, the first NAM block or statement is used. The PNAM parameter bypasses all records prior to the specification condition. Positioning of files occurs during an operational function. Therefore, the input file is not positioned during OPEN processing.

TNAM = 'xxxxxx' Termination of processing on the specified name statement or block. If xxxxxx is specified, termination occurs when the name specified by xxxxxx is located. If xxxxxx is not specified, the file is terminated on the first name statement or block. Where TNAM and PNAM are used jointly, termination tests are not executed until at least one record has been processed. This feature allows the processing of all records between specified name statements and blocks.

SNAM = 'xxxxxx' Selection of records following and including the specified name statement or block and termination of processing on the next name statement or block. SNAM is similar to using the PNAM and TNAM parameters, where PNAM specifies the name block and TNAM does not specify a particular name block. Thus, SNAM may be used to selectively process one file delimited by name statements.

NAM = 'xxxxxx' Selection of records that are only name blocks or statements. If xxxxxx is specified, the selection is limited to name blocks with the name specified by xxxxxx. Usually a name is not specified, since the NAM parameter generally is used to determine the names of all programs on a given file. For example, the NAM parameter used with the PR function results in a listing of all name statements on the associated input file.

LB is the maximum block length in characters. Default is 136 characters.

nnnnn	One- to five-digit decimal number (not greater than 32,767) specifying the number of characters in a block. For undefined, fixed, or variable length records, LB must equal LR. For blocked records LB must be greater than or equal to LR. For PR files, LB is used as the form's width. The value of nnnnn must be a multiple of two characters.

LR is the maximum record length in characters. Default is 136 characters.

nnnnn	Maximum record length where nnnnn is a one- to five-digit decimal number (not greater than 32,767) specifying the maximum number of characters in a record. If fixed block is specified, then LR is the exact length of each record. For PRINT files LR is used as the form's width. The value of nnnnn must be a multiple of two characters.

LC is the number of lines per page for the PRINT file. Default is 56 lines per page.

nnnnn	One- to five-digit decimal number. If nnnnn equals 0, line counting is inhibited.

CLOSE STATEMENT

The CLOSE statement terminates file processing and positions the file for further processing. All output files must be closed to ensure that the last block is written.

The control statement format is:

CLOSE, unit, motion

Where: unit is the device to which the CLOSE applies.

I	Input device
O	Output device
VF	Verify file
PR	List device

motion specifies the operations performed on the file before closing.

RW or blank — The tape is rewound and positioned at the load point. If the tape is an output file, a tape mark is written and all end-of-file labels are written before rewinding.

UN — The same procedure as in rewind (RW) except the tape is unloaded following rewind.

EOV — The function causes all trailer labels to be written including end-of-file labels (if the tape is to be labeled) followed by a double tape mark. The tape is rewound to load point and unloaded.

LEAVE — The tape is left-positioned at the next file. On input the tape is positioned at the next file. On output the last record is written followed by labels if they are specified. The tape is left-positioned following the last tape mark terminating the file.

BACKSPACE STATEMENT

The backspace function is available for tape control. It provides an easy method for positioning back to a name statement that was just used to complete an operation. The backspace instruction backspaces physical records and the instruction may cause unreliable results when working with blocked files.

The control statement format is:

BSPACE, unit, nnnnn

Where: unit is the device to which backspace applies.

I	Input unit
O	Output unit
VF	Verify file

nnnnn is the one- to five-digit decimal number indicating the number of physical records to backspace.

SET SYSTEM DATA STATEMENT

This command is used when the label processing option has been specified. It is used to set the creation date in the header labels of output volumes and to verify the expiration data in the header labels of output volumes.

The control statement format is:

SDATE = yyddd

Where: yy is the year.

ddd is the Julian calendar day.

SET EXPIRATION DATE STATEMENT

This command is used when the label processing option has been specified. It is used to set the expiration data in the header labels of output volumes. When the expiration data is reached, the tape may be used for other data.

The control statement format is:

EDATE = yyddd

Where: yy is the year.

ddd is the Julian calendar day.

NOTE

The SDATE and EDATE instructions may be inserted at any point in The program. If the two instructions are not used, both dates are assumed to be 99999. Verification of the date is not attempted.

OPERATIONAL CONTROL STATEMENTS

Operational control statements provide instructions to execute any of the following functional modules:

DUMP
PRINT
COPY
VERIFY
INITIALIZE VOLUME LABELS

Operational control statements usually follow OPEN control statements. Functions requiring input or output files are not executed unless the required files have been opened. In some cases files may be improperly defined, such as improper record type of length specifications. Recovery under these circumstances is not possible. The files must be closed and reopened before any function can be executed.

NEXT: OP, PR9
" OP, IC, LB6336, LR192
" OP, VFK, LB6336, LR192
" VERIFY, H

Through proper definition of input and output file characteristics, files may be blocked, deblocked, converted, and/or labeled. The parameters in an operational control statement may be entered in any order.

DUMP STATEMENT

The DUMP statement prints the input tape file on the standard list device or any list device assigned in a preceding OPEN command. The list file may be a teletypewriter or line printer, but it may not be a tape file since forms control records may be less than the minimum tape record size.

By specifying block or record sizes the user controls the length of a single output line. The number of lines per page are specified by setting the LC parameter of the OPEN statement to the required size. If LC is not specified, the number of lines per page is 56.

The control statement format is:

```
DUMP, FCnn, RCnnnn, mode, type, select
```

Where: FC indicates dumping of files. Default is one.

nn Number of files to be dumped

RC indicates dumping of records. Default is the number of files requested.

nnnn Number of records to be dumped

mode is the mode of the dump.

H Hexadecimal dump

C or blank Character dump

type is the format type of the dump (see appendix E).

FM or blank Formatted dump

UF Unformatted character dump. Hexadecimal dump must be formatted.

select is the selected data to be processed. Caution must be exercised when using the SELECT parameter because the conditions causing termination are determined from the record following the last record dumped. It is impossible to obtain the record which caused termination for subsequent processing, without first issuing the backspace function.

PNAM 'xxxxxx' NAM block in which all records are dumped until either an end-of-file or the conditions specified by TNAM are detected

TNAM 'xxxxxx' Dump is terminated upon detection of the specified NAM block or statement.

SNAM 'xxxxxx' Selection of the set of records following the specified NAM block or statement. SNAM then dumps the records and terminates the dump at the next NAM block or statement.

NAM 'xxxxxx' Only the specified NAM block or blocks are dumped until an end-of-file or the conditions specified by TNAM are detected.

PRINT STATEMENT

The PRINT statement prints an input tape on the standard list device or a device assigned in the OPEN statement. The print module provides for printer control characters that are in the first position of each record. The PRINT instruction is generally used for printing listable tapes.

By specifying block or record sizes, the user may control the length of a single line. Line counts are not used, since the input records are assumed to have standard print control characters.

The control statement format is:

```
PRINT, FCnn, RCnnnn, type, select
```

Where: FC indicates printing of files. Default is one.

nn Number of files to be printed

RC indicates printing of records. Default is the number of files requested.

nnnn Number of files to be printed

type is the type of control characters used in the records.

US USASCI standard printer control characters are in the first character position on each record. MSOS listable output has USASCI coded characters.

TS A Tape SCOPE listable tape is to be printed, although Tape SCOPE listable tapes do not have printer control characters. This parameter duplicates the Tape SCOPE assembler output.

select is the selected data to be processed. See DUMP Statement for details.

Since the PRINT statement is used as a tape-to-print operation, a means to ensure proper forms alighment is included. After the PRINT instruction is issued, the forms in the line printer are ejected and the first record is printed, followed by a page eject. The following message is typed on the operator's console:

```
*FORMS ALIGNED?
```

open I6
open o16 , P29
verify

The operator's response may be any character or the carriage return. The carriage return directs the PRINT module to continue after reprinting the first line. Any character instructs the line printer that the forms are not aligned. The operator may realign the line printer forms before responding. The forms alignment statement and the printing are repeated until the carriage return is entered.

COPY STATEMENT

The COPY statement is designed to copy a tape file from the standard input device to the standard output device unless otherwise specified. Through the use of a declarative control statement blocking, deblocking, and/or code conversion may be implemented.

The COPY statement is used to block, deblock, convert, etc., any data tape or 1700 System tape. It also can be used by proper specification of selection parameters to replace, delete, or select relocatable program modules, listable program modules, and source program modules on 1700 System tape.

The control statement format is:

COPY, RCnnnn, FCnn, select

Where: RC indicates copying of records. Default is the number of files requested.

nnnn Number of records to be copied

FC indicates copying of files. Default is one.

nn Number of files to be copied

select is the selected data to be processed. Caution must be exercised when using the select parameter because the conditions causing termination are determined from the record following the last record copied. It is impossible to obtain the record that caused termination for subsequent processing without issuing the backspace function.

PNAM 'xxxxxx' a NAM block in which all records are copied until either an end-of-file or the conditions specified by TNAM are detected

TNAM 'xxxxxx' Copy terminated upon detection of the specified NAM or block statement

SNAM 'xxxxxx' Selection of the set of records following the specified NAM block or statement, which represents a specified NAM block and terminating copy when the next NAM block or statement is detected

NAM 'xxxxxx' Only the specified NAM block or blocks are copied until an end of file or the conditions specified by TNAM are detected

VERIFY STATEMENT

The VERIFY statement is designed to verify two tape files on a specified unit or tapes mounted on a standard input and output device. Tapes may be verified with or without the blocking or conversion options.

The VERIFY statement verifies data files on a record basis. Data files may be verified if the clocking factors are different, if the data is in different external codes (ASCII, EBCDIC, or BCD), or in some cases, if the data is in different record formats (see section 4). If records fail to match, the VERIFY instruction dumps both the input and output records to the standard list device. The number of characters not matching and the first character of the record in error are also printed. In the event that more than 10 consecutive reocrds are found in error, the system interrogates the operator whether to continue or terminate. A response of carriage return designates termination. Any other character followed by a carriage return designates continuation of the program.

The control statement format is:

VERIFY, RCnnnn, FCnn, mode, type, select

Where: RC indicates verifying of records. Default is the number of files requested.

nnnn Number of records to be verified.

FC indicates verifying of files. Default is one.

nn Number of files to be verified.

mode is the mode of the dump of error records.

H Hexadecimal dump (the dump module must be present)

C or blank Character dump

type is the format type of the dump of error records (see appendix E).

FM or blank Formatted dump

UF Unformatted dump. Each record is preceded by header lines specifying block size, record size, and the sequence of the records. (The dump module must be present.)

select is the selected data to be processed. See the COPY statement for details.

INITIALIZE STATEMENT

The INITIALIZE statement is used to write volume 1 header records following the load point on a specified unit. The INITIALIZE function differs from other functions in that once the INITIALIZE process is implemented it must be specially terminated. Once initialization is started, it requests a volume serial number and then writes the first header on the mounted tape. Upon completion, the tape is unloaded and a request for the next tape is made. The previous volume serial number is incremented (if numeric) and may be used as the next volume serial number.

The label type may be specified by using the OPEN conversion parameters. If the OPEN parameters are omitted, A is assumed. The INITIALIZE statement communicates with the operator through a series of messages. Operator responses are used to terminate the INITIALIZE statement, to determine if a tape is mounted and ready, and to assign the volume serial number. The mount message

```
*MOUNT, OUTPUT, SCRATCH:
```

instructs the operator to mount a scratch tape on the specified logical unit. Response of a carriage return designates that the tape is mounted. Any other character followed by a carriage return terminates the initialize function.

The volume serial number message

```
VOLSER = nnnnnn:
```

indicates the next volume serial number to be written on the mounted scratch tape. To change the serial number, the operator enters a one- to six-character volume serial identifier. The identifier may be alpha, alphanumeric, or numeric. Only characters from A through Z and 0 through 9 are permitted. A carriage return terminates the identifier. If the identifier is completely numeric, the next VOLSER message contains the last number plus one. Otherwise the next VOLSER message contains

```
VOLSER = 000001
```

To use the serial number contained in the message, the operator responds with a carriage return.

The control statement format is:

```
INIT, Onn, label
```

Where: O specifies the output unit. Default is the standard tape output.

nn One- to two-digit decimal number

label is the type of label to be generated.

E EBCDIC IBM standard volume labels. The volume label is compatible with 360/370 OS/DOS/VS operating systems.

B BCD code used in volume labels

A American National Standard volume labels

EXIT STATEMENT

The EXIT statement returns control to the operating system after terminating the job. All files not already closed by a CLOSE statement are closed with the rewind and unload function. If the magnetic tape utility processor is used to position a file for processing, the operator must ensure the file is closed without the rewind function before issuing an exit request.

The control statement format is:

```
EXIT
```

LABELING

Labels may be processed during the OPEN, READ, WRITE, and CLOSE operations. The three potential processing techniques available include: no labels, bypass labels, and standard labels. Each technique requires a specific file structure.

NO LABELS

When no labels are defined the volume may contain one or more files. The file structure for a single-volume unlabeled file is shown in the following illustration:

Single File	FILE A						*	*
Multifile	FILE A	*	FILE B	*	. . .	FILE N	*	*

The volume is terminated by two tape marks. Specific provisions to process files extending to more than one volume have been excluded. However, a double tape mark on input terminates all processing, which allows mounting of the next volume. Output processing terminates at the end of the tape marker. An error message is issued. A new tape is mounted and processing is continued by re-executing the function.

BYPASS LABELS

The bypass label processing ignores all labels; however, the magnetic tape utility processor expects the normal sequence of tape marks found in standard labels. The bypass label option is not available for output files.

STANDARD LABELS

Standard labeled volumes may contain one or more files and may extend to more than one volume. Each volume must contain a VOL1 header label as the initial record. Each file must be preceded by a HDR1 label and terminated by an end-of-file label. The magnetic tape utility processor processes VOL1, HDR1, HDR2, EOF1, EOF2, EOV1, and EOV2 labels.

REQUIREMENTS

GRAPHIC REPRESENTATION

In this section, n means any numeric digit, 0 through 9. An a means any of the characters occupying the center four columns of ASCII (except position 5/15) and those positions where there is provision for alternative graphic representation.

The limitation of a characters is intended as a guide to provide maximum interchangeability and consistent print especially when international interchange is a possibility. Checking for conformity to this limitation is not implied.

LABEL FORMAT

Each required label is an 80-character block recorded in even parity mode at the same density as the rest of the tape for seven-track tapes, or in convert mode ASCII for nine-track tapes.

TAPE MARKS

Use of tape marks is not permitted in positions not specified in this manual. A file containing nonstandard tape marks cannot be considered a legitimate interchange tape.

LABELED FILE STRUCTURE

Required labels and tape marks must be used to establish the file structure according to the formats displayed in table 3-1. The table shows that the beginning of the tape is

TABLE 3-1. STRUCTURE OF MAGNETIC TAPE FILES

Type of File	File Format	
Single Volume File	VOL1	HDR1* ----------- *EOF1**
Multivolume File	VOL1 VOL1	HDR1* ----------- *EOV1** HDR1* ----------- *EOF1**
Multifile Volume	VOL1	HDR1* ----------- *EOF1*HDR1* ----- *EOF1**
Multifile Multivolume	VOL1	HDR1* -- (A) -- *EOF1 *HDR1*---(B)----*EOV1**
	VOL1 VOL1	HDR1* -- (B) -------------- *EOV1** HDR1* -- (B) -- *EOF1 *HDR1*---(C)----*EOF1**

on the left and the end of the tape is on the right. Required labels are specified by their first four characters, and tape marks are specified by asterisks.

VOLUME HEADER LABEL

Every volume has a volume header label (VOL1) as the first block in the volume. No volume header label is used at any other place in the volume.

END-OF-VOLUME LABEL

An end-of-volume label (EOV) should follow the last block of a file when a volume ends within the file. One tape mark immediately precedes and two tape marks immediately follow every end-of-volume label group. No file sets should be terminated by an end-of-volume label.

FILE HEADER LABEL

Every file must be preceded by a file header label (HDR1). Whenever a volume ends within a file, the continuation of the file in the next volume must also be preceded by a file header label. Every file header label group must immediately be followed by a tape mark.

END-OF-FILE LABEL

The last block of every file must be followed by an end-of-file label (EOF1). A tape mark must immediately precede, and another tape mark immediately follow, every end-of-file label group. The end-of-file label that appears at the end of the last file in a set must be followed by two tape marks.

If end-of-file and end-of-volume coincide on a multifile multivolume tape, the following configurations are used. Letters in parentheses indicate the file the labels belong to. A dash indicates that the label belongs to neither file.

```
VOL1  HDR1*(FILE A)------*EOF1*HDR1**EOV1**
(-)                       (A)    (B)   (-)

VOL1  HDR1*-----------(FILE B)------
(-)    (B)
```

OPTIONAL LABEL FORMAT

Optional operating system labels and user labels must fit into the file structure according to the following rules so the relationship between required labels and file is not affected.

OPTIONAL FIELDS

When optional is used in defining a field, it means the field may contain information already described. If an optional field does not contain the designated information, the field contains blanks.

REQUIRED FIELDS

Fields not defined as optional are required and are written or read as specified in this manual.

VOLUME HEADER LABEL (VOL1)

The fields in the volume header label are arranged as shown in table 3-2. Anyone using a magnetic tape that is not his must return the entire volume header label unless otherwise authorized by the user. However, a volume header label may be written if authorized.

TABLE 3-2. VOLUME HEADER LABEL (VOL1)

Character Position	Field	Name	Length	Description
1–3	1	Label Identifier	3	Must be VOL
4	2	Label Number	1	Must be 1
5–10	3	Volume Serial Number	6	Six a characters permanently assigned by the user to identify this physical volume (reel of tape)
11	4	Accessibility	1	An a character indicating any restrictions on who may have access to the information in the volume. A blank means unlimited access. Any other character means special handling,

TABLE 3-2. VOLUME HEADER LABEL (VOL1) (Continued)

Character Position	Field	Name	Length	Description
12–31	5	Reserved for future use	20	Must be blanks
32–37	6	Reserved for future use	6	Must be blanks
38–51	7	Owner Identification	14	Any a characters, identifying the owner of the physical volume. May be blank.
52–79	8	Reserved for future use	28	Must be blanks
80	9	Label Standard Level 8	1	1 means the labels and data format on this volume conform to the requirements of this manual. Blanks mean the labels and data formats on this volume require the agreement of the interchange parties.

FILE HEADER LABEL

The fields in the file header label are arranged as shown in table 3-3.

END-OF-FILE LABEL (EOF1)

The fields of the end-of-file label are arranged as shown in table 3-4.

TABLE 3-3. FILE HEADER LABEL 1

Character Position	Field	Name	Length	Description
1–3	1	Label Identifier	3	Must be HDR
4	2	Label Number	1	Must be 1
5–21	3	File Identifier	17	Any a characters agreed on between originator and recipient
22–27	4	Set Identification	6	Any a characters to identify the set of files of which this is one. This identification must be the same for all files of a multifile set.
28–31	5	File Section Number	4	The file section number of the first header label of each file is 0001. This applies to the first or only file on a volume and to subsequent files on a multifile volume. This field is incremented by one on each subsequent volume of the file.
32–35	6	File Sequence Number	4	Four n characters denoting the sequence (0001, 0002, etc.) of files within the volume or set of volumes. In all the labels for a given file, this field contains the same number.

TABLE 3-3. FILE HEADER LABEL 1 (Continued)

Character Position	Field	Name	Length	Description
36–39	7	Generation Number (Optional)	4	Four n characters denoting the current stage in the succession of one file generation by the next. When a file is first created, its generation number is 0001.
40–41	8	Generation Version	2	Two n characters distinguishing successive repetitions of the same generation. The generation version number of the first attempt to produce a file is 00.
42–47	9	Creation Date	6	A blank followed by two characters for the year, followed by three n characters for the day (001 to 366) within the year
48–53	10	Expiration Date	6	Same format as field 9. This file is regarded as expired when today's date is equal to or later than the date given in this field. When this condition is satisfied, the remainder of this volume may be overwritten. To be effective on multifile volumes, the expiration date of a file must be less than or equal to the expiration date of all previous files on the volume.
54	11	Accessibility	1	An a character indicating restrictions on who may have access to the information in this file. A blank means unlimited access; any other character means special handling in a manner agreed on by the interchange parties.
55-60	12	Block Count	6	Must be zeros.
61–73	13	System Code (Optional)	13	Thirteen a characters identifying the operation system that recorded this file
74–80	14	Reserved for Future Use	7	Must be blanks

TABLE 3-4. END-OF-FILE LABEL (EOF1)

Character Position	Field	Name	Length	Description
1–3	1	Label Identifier	3	Must be end-of-file
4	2	Label Number	1	Must be 1
5–54	3–11	Same as corresponding fields in the first file header label (optional)	Total 50	Same as corresponding fields in the first file header label
55–60	12	Block Count	6	Six n characters denoting the number of data blocks (exclusive of labels and tape marks) since the preceding HDR label group
61–80	13,14	Same as corresponding fields in the first file header label (optional)	Total 20	Same as corresponding fields in the first file header label

SECOND END-OF-FILE LABEL (OPTIONAL)

The option constitutes a second end-of-file label specified by EOF2 as the label identifier and label number. The label contains the same information in fields 3 through 8 (all optional) as does HDR2.

END-OF-VOLUME LABEL (EOV1)

The fields of the end-of-volume label are arranged as shown in table 3-5.

SECOND END-OF-VOLUME LABEL (OPTIONAL)

This option constitutes a second end-of-volume label specified by EOV2 as the label identifier and label number. The label contains the same information in fields 3 through 8 (all optional) as does HDR2. The fields of the second file header label are arranged as shown in table 3-6.

TABLE 3-5. END-OF-VOLUME LABEL (EOV1)

Character Position	Field	Name	Length	Description
1–3	1	Label Identifier	3	Must be end-of-volume
4	2	Label Number	1	Must be 1
5–54	3–11	Same as corresponding fields in the first file header label (optional)	Total 50	Same as corresponding fields in the first file header label
55–60	12	Block Count	6	Number of data blocks (exclusive labels and tape marks) since the preceding HDR label group
61–80	13,14	Same as corresponding fields in the first file header label (optional)	Total 20	Same as corresponding fields in the first file header label

TABLE 3-6. SECOND FILE HEADER LABEL (OPTIONAL)

Character Position	Field	Name	Length	Description
1–3	1	Label Identifier	3	Must be HDR
4	2	Label Number	1	Must be 2

5	3	Record Format	1	F = Fixed length
				V = Variable with the number of characters in the record specified in binary
				U = Undefined

6–10	4	Block Length	5	Maximum number of characters per block
11–15	5	Record Length	5	Maximum record length including any count fields for F and V record formats. Undefined for U record format
16–50	6	Reserved for Operating Systems	35	Reserved for operating systems use. Any a characters
51–52	7	Buffer Offset (Optional)	2	Input is ignored. Output is zeros.
53–80	8	Reserved for Future Use	28	Must be blanks

I/O PROCESSING WITH STANDARD LABELS

INPUT FILE PROCESSING

The OPEN statement initiates label processing by reading the VOL1 header record.

Header 1 overlays the VOL1 header in a 40-word buffer label, and header 1 is retained. If header 2 is available it is added to the 40-word buffer and, if necessary, the contents of the block length and record length fields are used to complete open processing. All other label records are placed in a chained buffer until a tape mark is encountered. The saved labels may be used by the magnetic tape utility processor for output labels with copy function.

OUTPUT FILE PROCESSING

All output files must contain a VOL1 header record; however, output files may contain other header records. If the output file contains a HDR1 header, the expiration date is checked against the system date. The output volume may not be used if the expiration date is greater than the system date unless the operator responds to the following message:

```
VOL NOT EXPIRED:     U      (cr)
                     X      (cr)
                            (cr)
```

Where: U implies use of the volume.
X implies the volume is not available.
(cr) implies carriage return.

Once the initial verification is complete, new headers must be written. Since COPY is the only instruction that uses output files, the new headers may originate from either of two sources: the input volume or the operator. If the new headers are to be copied from the input volume, the input volume must contain standard labels. All labels except the VOL1 label are transferred from the input to the output file. To specify new output labels, the input file must be opened as BL (bypass labels) or NL (no labels) and the output must be opened as SL (standard labels). In this instance the operator enters the following label information:

*DATA SET NAME: DSN = 'a. . .a'

where a is from 1 to 17 alphanumeric characters.

All other header fields are generated by the magnetic tape utility processor in accordance with the preceding standards.

END-OF-VOLUME, END-OF-PROCESSING

The magnetic tape utility processor recognizes and writes end-of-volume (EOV) or end-of-file (EOF) labels. Automatic volume sequencing occurs on input and when the EOV label is sensed. On output, the sensing of the end-of-tape (EOT) marker causes an EOV label to be written and automatic sequencing to a new volume.

System messages are designed to inform the operator of the system's functions, cue the operator to take a certain action, output statistical data, and indicate errors that have occurred during execution. The messages are written either on the system console or the system list device depending on the purpose of the message.

INFORMATION MESSAGES

Statistical data and information messages such as those indicating errors during the verify process should appear on the standard list device if a standard device has been activated by an OPEN statement. If not, the messages appear on the console.

The format for the statistical data messages is:

```
*NEXT: COPY,RC100
00100 INPUT BLOCKS
00100 INPUT RECORDS
00020 OUTPUT BLOCKS
00100 OUTPUT RECORDS
```

NOTE

Output block and record totals may not reflect the same information as input totals. The input/output discrepancy is a result of blocking, deblocking, multiple files, and other functions.

All totals reflect the total number of records that have been read and written. For INPUT files the total number of records are cumulative over the entire volume. For OUTPUT files the totals reflect only the last file written minus the last applicable unwritten block. Labeled files may generate zero totals in the block field because the counts are cleared during end-of-file processing.

ACTION MESSAGES

Whenever action messages occur, an operator response is required. The action message indicates that a decision on whether to proceed is possible, an operation has been completed, or more information is necessary before processing can continue. Table 4-1 lists all action messages.

TABLE 4-1. ACTION MESSAGES

Message	Response	Description
*NEXT:	Any utility operational declarative statement	System has completed all prior requests and can accept next request.
*INVALID PARM= 'XXX. . .' *RETYPE PARM:_	Enter corrected parameter.	The characters within quotes are invalid and may be corrected (refer to functional actions).
10 ERRORS *CONTINUE:	Type carriage return to terminate, or type one character followed by carriage return to continue.	Verify function has located 10 consecutive records that contain errors.
*MOUNT,OUTPUT, SCRATCH:	Type carriage return, which implies tape is ready, or type any other character followed by a carriage return to terminate the initialize function.	See Initialize Statement in section 2.
*VOLSER=nnnnnn:	See Initialize Statement in section 2.	See Initialize Statement in section 2.
VOL NOT EXPIRED USE:	Carriage return implies do not use. U implies use, ignoring expiration date.	Label processing – Output volume header records are checked against the system date.

TABLE 4-1. ACTION MESSAGES (Continued)

Message	Response	Description
*DATA SET NAME:	DSN='xxxxx'	Label processing – Output volumes require a data set name if not available from input (see section 3).
VOLSER=nnnnnn	None	Informative tape file just opened with the specified volume serial number.

SYSTEM ERROR MESSAGES

System error messages are always issued to a comment device. There are four types of system errors: descriptive, critical, serious, and warning errors.

DESCRIPTIVE ERROR MESSAGES

The descriptive error messages indicate the error and direct that the previous instruction could not be executed. The descriptive error messages are listed in table 4-2.

TABLE 4-2. DESCRIPTIVE ERROR MESSAGES

Message	Description	Action
FILE(S) NOT OPEN	A required file is not open and the specified function cannot be executed.	Open file and re-enter function.
*INVALID OPEN OR CLOSE	The file being opened or closed is already in that state.	Open or close the proper file.
*FUNCTION NOT AVAILABLE	An attempted function is not available in the system. The function is not invalid; rather the system was configured without the requested module.	Use another function, if possible.
*PARM NOT AVAILABLE	A parameter is not available in the system. The parameter is not invalid; rather the system was configured without the requested module.	Use another function, if possible.
*INCORRECT VOL MOUNT:	The volume mounted does not contain a volume label or the header label sequence is incorrect; i.e., the wrong volume of a multiple volume file is mounted.	Mount correct volume and type a carriage return.

CRITICAL, SERIOUS, AND WARNING ERROR MESSAGES

Critical error messages imply that an error has occurred that prevents further processing by the magnetic tape utility processor. If the critical error occurs, control is returned to the operating system.

Serious errors are a result of mistakes such as improper specification of valid parameters and tape errors. If a serious error occurs, control returns to the magnetic tape utility processor and the user may continue programming.

Warning messages indicate that parameters may be inconsistent or a possible processing error has occurred. Execution continues following a warning. Warning messages are used to allow processing flexibility, since the inconsistent parameters may be required to execute a function. For example, when converting fixed length records to variable length records, the copy function checks for length mismatches. In this case, the warning may be ignored.

Tables 4-3, 4-4, and 4-5 describe corrective action for all coded error messages.

TABLE 4-3. CRITICAL ERRORS

Error Code	Description	Action
****C000****	Data buffer linkage has been destroyed. This is caused by input/output or central processing unit malfunction.	Reload utility.

TABLE 4-4. SERIOUS ERRORS

Error Code	Description	Action
****S000****	Available memory has been filled.	Free memory by closing a file.
****S001****	Attempt to close file already closed.	Close proper file.
****S002****	1. Read end-of-file 2. Attempt to write on file not opened for write 3. Input/output error; i.e., parity, read or write error, lost data, or alarm	Retry the function.
****S003****	Variable length block does not match actual length read or variable read length is greater than specified block size.	Close all files. Open input as undefined and dump records to locate the erroneous record. File cannot be processed as variable length.
****S004****	Blocking has been requested and specified block size is smaller than specified record size.	Reopen file with proper parameters.
****S005****	Variable size error was detected prior to write.	Attempt to re-execute function after closing and reopening all files. Possible hardware malfunction.
****S006****	Fixed block error was detected prior to write. Record length is not specified.	Close file and reopen with proper record size or dump file to locate erroneous records.
****S007****	Labeled file sequence number is in error. (File is not opened.)	Mount proper volume and reopen.
****S008****	Labeled file EOF1 trailer label contains invalid information that does not correspond to header label one.	This file cannot be processed with standard labels.

TABLE 4-4. SERIOUS ERRORS (Continued)

Error Code	Description	Action
****S009****	Labeled file is missing end-of-file trailer labels.	File cannot be processed as labeled.
****S010****	End of tape is sensed on output file (unlabeled).	Close the file with end of volume and reopen after mounting new tape. Re-enter function to complete processing.
****S011****	A double file mark has been sensed on an input file. Processing is terminated.	Close input file and mount next volume. Re-enter function to complete processing.
****S012****	Invalid date	Re-enter date function with proper date.
****S013****	Labeled volume sequence number is incorrect. (Occurs after OPEN file is not opened.)	Mount proper volume and reopen file.
****S014****	ZERO LENGTH block specified in OPEN FILE is not opened.	Reopen, specifying proper block length.
****S015****	Block or record length specified is not a multiple of two. FILE, is not opened.	Reopen, specifying even block and record length. If either block or record length is odd, the data cannot be processed by the system.

TABLE 4-5. WARNING MESSAGES

Error Code	Description	Action
W000	Blocking is not specified but block size and record size have been specified differently in OPEN.	Open file with proper parameters or continue statement.
W001	File count is specified as zero.	Re-enter function with proper parameters or continue statement.
W002	Record count is specified as zero.	Re-enter function with proper parameters or continue statement.
W003	Input and output record lengths have been specified differently for COPY.	Re-enter function with proper parameters or continue statement.

GLOSSARY A

FIELD A specified area in a record used for a particular category of data

FILE A collection of data consisting of records pertaining to a general subject. The delineation of a file may be arbitrary. The absence of information may result in the creation of a file, delineated by adjacent tape marks, without information.

FILE SET A collection of one or more related files, recorded on one or more volumes. A file set consists of:

One file recorded on a single volume

More than one file recorded on a single volume

One file recorded on more than one volume

More than one file recorded on more than one volume

LABEL A block at the beginning or end of a volume or a file that identifies and/or delineates that volume or file.

LABEL GROUPS A collection of contiguous labels related to a file. The label group either precedes or follows the related file on a single volume. The volume header labels and the following file header labels are the first label group on a volume.

TAPE MARK A special configuration record on magnetic tape, synonymous with an end-of-file mark, indicating the boundary between files and labels and between certain label groups. The tape mark configuration for nine-track tape is defined in CDC-STD 1.10.005 and CDC-STD 1.10.006. The tape mark configuration for seven-track tape is an octal 17 written in even parity.

VOLUME A physical unit of storage media. In this manual, volume is synonymous with a reel of magnetic tape.

RECORD PROCESSING

The following sections describe record formats and processing requirements for the various record types.

RECORD FORMATS

FIXED LENGTH RECORDS

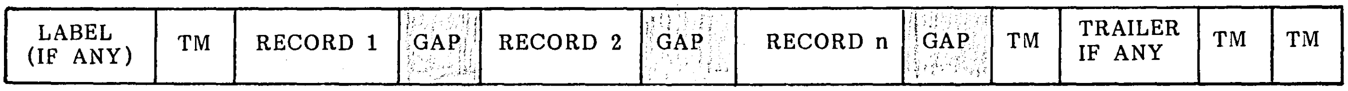
Each record must be exactly the same length (see figure B-1).

FIXED LENGTH BLOCKED RECORDS

Each block must contain an integral number of fixed-length records (see figure B-2). However, there is no requirement that the block length remain fixed. Only the last block may be shorter than any preceding blocks. A maximum length is associated with fixed blocking.

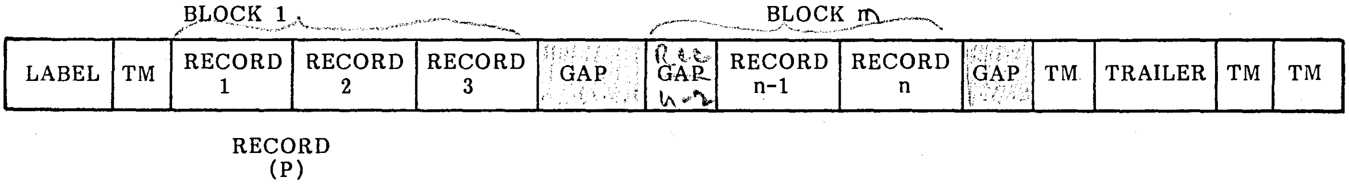
VARIABLE LENGTH RECORDS

A standard variable-length record consists of a record descriptor word and data characters (see figure B-3). The number of characters is contained in the record descriptor word.



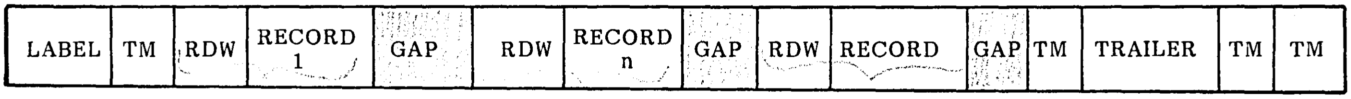
NOTE: TM = TAPE MARK

Figure B-1. Fixed Length Records



NOTE: TM = TAPE MARK

Figure B-2. Fixed Length Blocked Records



NOTES: TM = TAPE MARK
RDW = RECORD DESCRIPTOR WORD (32 BITS)
UPPER 16 BITS = RECORD SIZE
LOWER 16 BITS = 0 (RESERVED)

Figure B-3. Variable Length Records

VARIABLE LENGTH BLOCKED RECORDS

A variable-length blocked record consists of a block descriptor word and any number of variable length records (see figure B-4). The block descriptor word holds the block length; i.e., the sum of the record descriptor words plus four.

UNDEFINED RECORDS

An undefined record has either variable or fixed length. The user must provide record length using the LR parameter. The record format is uncontrolled and blocking is not possible. Print files may be specified only as undefined.

RECORD PROCESSING

Record processing of the acceptable format types is related to the OPEN statement.

Formats may be converted only as follows.

Fixed length records may be converted to:

Fixed blocked
Variable
Variable blocked
Undefined

Fixed length blocked records may be converted to:

Fixed
Variable
Variable blocked
Undefined

Variable length records may be converted to:

Variable blocked
Undefined

Variable length blocked records may be converted to:

Variable
Undefined

Undefined records may be converted to:

Fixed (if undefined records are fixed length)
Fixed blocked (if undefined records are fixed length)
Variable
Variable blocked

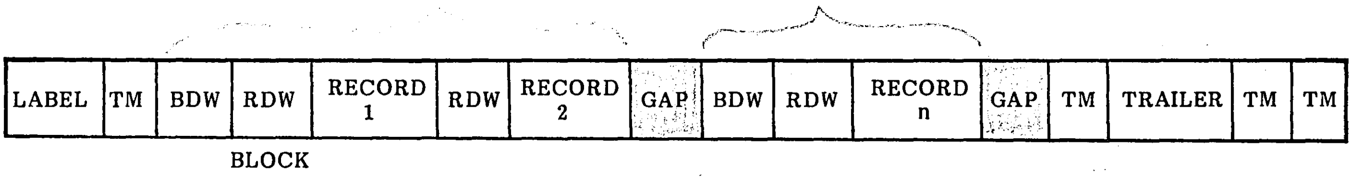
The open parameters LB and LR must conform to the following rules if blocking is used:

Fixed block records:

LB must be equal to or greater than the longest record to be read or written
LR must be less than or equal to LB and must be the exact record size

Variable and variable blocked records:

LB must be equal to or greater than the largest record
LR must be equal to or greater than the largest record
LR must be less than or equal to LB



NOTES: TM = TAPE MARK

BDW = BLOCK DESCRIPTOR WORD (32 BITS)
UPPER 16 BITS = BLOCK SIZE
LOWER 16 BITS = 0

RDW = RECORD DESCRIPTOR WORD
UPPER 16 BITS = RECORD SIZE
LOWER 16 BITS = 0 (RESERVED)

Figure B-4. Variable Length Blocked Records

CONVERSION CODES C

Table C-1 shows the relationships among codes in an ascending collating sequence. The chart explains the equivalence between characters in the various machine and teletype codes.

TABLE C-1. COLLATING SEQUENCE

Collating Sequence	EBCDIC Character	ANSI Character	Internal and ASCII Tape Code (Hexadecimal)	Tape Code BCD (Octal)	Tape Code EBCDIC (Hexadecimal)
00	Others	Del	7F		3E
01	Rej.	@	40	56	3F
02	Space	Space	20	20	40
03					
04					
05					
06					
07					
08					
09					
10					
11					
12	\	[	5B		4A
13	.	.	2E	73	4B
14	<				4C
15	(	(	28	34	4D
16	+	+	2B	60	4E
17	|				4F
18	&	&	26	15	50
19					
20					
21					
22					
23					
24					
25					
26					
27					
28	!	!	21	52	5A
29	$	$	24	53	5B
30	*	*	2A	54	5C
31	)	)	29	74	5D
32	;	;	3B	77	5E
33	¬	—	5F	36	5F
34	-	-	2D	40	60
35	/	/	2F	21	61
36					
37					
38					
39					
40					
41					
42					
43					
44			7C		6A
45	,	,	2C	33	6B

TABLE C-1. COLLATING SEQUENCE (Continued)

Collating Sequence	EBCDIC Character	ANSI Character	Internal and ASCII Tape Code (Hexadecimal)	Tape Code BCD (Octal)	Tape Code EBCDIC (Hexadecimal)
46	%	%	25	16	6C
47	--	^	5E	37	6D
48	>				6E
49	?	?	3F	75	6F
50					
51					
52					
53					
54					
55					
56					
57					
58					
59	\	\	60		79
60	:	:	3A	00	7A
61	#	£	23	55	7B
62	@	£			7C
63	┌┘	┌┘	27	14	7D
64	=	=	3D	13	7E
65	"	"	22	76	7F
66					
67	a	a	61		81
68	b	b	62		82
69	c	c	63		83
70	d	d	64		84
71	e	e	65		85
72	f	f	66		86
73	g	g	67		87
74	h	h	68		88
75	i	i	69		89
76					
77					
78					
79					
80					
81					
82					
83	j	j	6A		91
84	k	k	6B		92
85	l	l	6C		93
86	m	m	6D		94
87	n	n	6E		95
88	o	o	6F		96
89	p	p	70		97
90	q	q	71		98
91	r	r	72		99
92					
93					
94					
95					
96					
97					
98					
99	~	~	7E		A1
100	s	s	73		A2
101	t	t	74		A3
102	u	u	75		A4
103	v	v	76		A5
104	w	w	77		A6

TABLE C-1. COLLATING SEQUENCE (Continued)

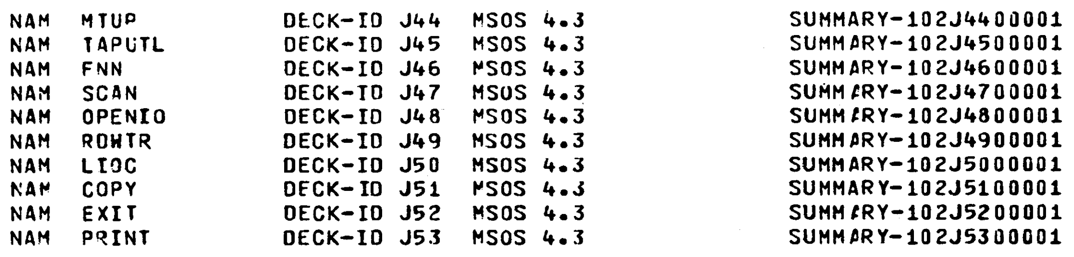
Collating Sequence	EBCDIC Character	ANSI Character	Internal and ASCII Tape Code (Hexadecimal)	Tape Code BCD (Octal)	Tape Code EBCDIC (Hexadecimal)
105	x	x	78		A7
106	y	y	79		A8
107	z	z	7A		A9
108					
109					
110					
111					
112					
113					
114					
115					
116					
117					
118					
119					
120					
121					
122					
123					
124					
125					
126					
127					
128					
129					
130	{	{	7B		C0
131	A	A	41	61	C1
132	B	B	42	62	C2
133	C	C	43	63	C3
134	D	D	44	64	C4
135	E	E	45	65	C5
136	F	F	46	66	C6
137	G	G	47	67	C7
138	H	H	48	70	C8
139	I	I	49	71	C9
140					
141					
142	⑁	]	5D	32	CC
143					
144	⑂	<	3C	72	CE
145					
146	}	}	7D		D0
147	J	J	4A	41	D1
148	K	K	4B	42	D2
149	L	L	4C	43	D3
150	M	M	4D	44	D4
151	N	N	4E	45	D5
152	O	O	4F	46	D6
153	P	P	50	47	D7
154	Q	Q	51	50	D8
155	R	R	52	51	D9
156					
157					
158					
159					

TABLE C-1. COLLATING SEQUENCE (Continued)

Collating Sequence	EBCDIC Character	ANSI Character	Internal and ASCII Tape Code (Hexadecimal)	Tape Code BCD (Octal)	Tape Code EBCDIC (Hexadecimal)
160					
161					
162					
163					
164	S	S	53	22	E2
165	T	T	54	23	E3
166	U	U	55	24	E4
167	V	V	56	25	E5
168	W	W	57	26	E6
169	X	X	58	27	E7
170	Y	Y	59	30	E8
171	Z	Z	5A	31	E9
172					
173					
174	⑂	>	3E	57	EC
175					
176					
177					
178	0	0	30	12	F0
179	1	1	31	01	F1
180	2	2	32	02	F2
181	3	3	33	03	F3
182	4	4	34	04	F4
183	5	5	35	05	F5
184	6	6	36	06	F6
185	7	7	37	07	F7
186	8	8	38	10	F8
187	9	9	39	11	F9
188	|	\	5C	35	FA

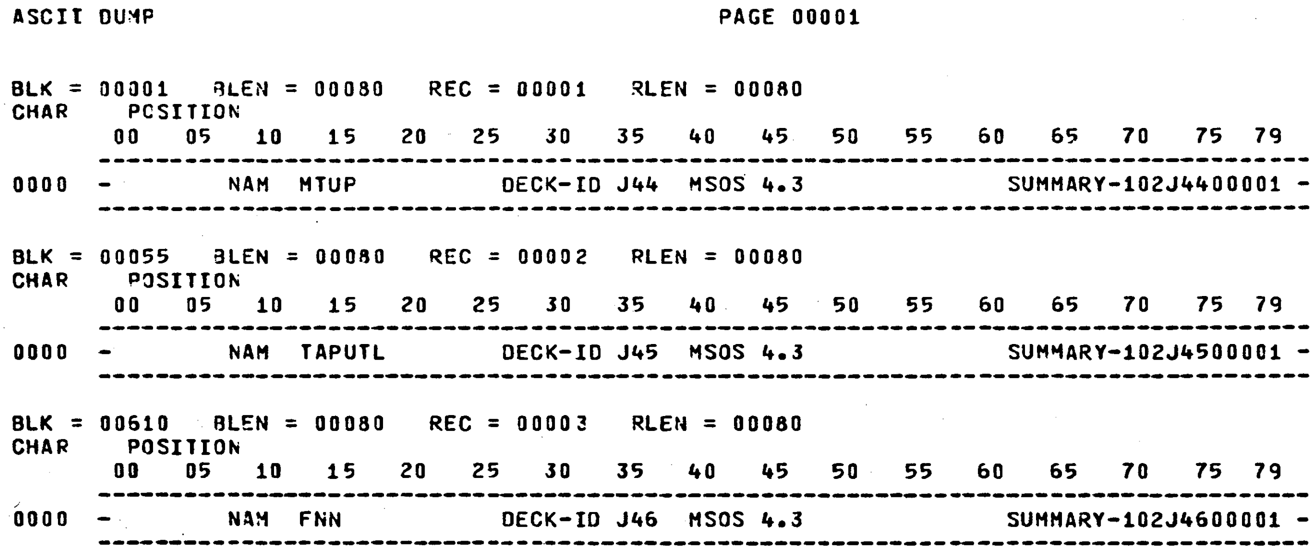
SAMPLE DUMPS D

Figure D-1, figure D-2, and figure D-3 are lists of dumps that include an unformatted character dump, a formatted character dump, and a formatted hexadecimal dump.



```
NAM   MTUP        DECK-ID J44  MSOS 4.3           SUMMARY-102J4400001
NAM   TAPUTL      DECK-ID J45  MSOS 4.3           SUMMARY-102J4500001
NAM   FNN         DECK-ID J46  MSOS 4.3           SUMMARY-102J4600001
NAM   SCAN        DECK-ID J47  MSOS 4.3           SUMMARY-102J4700001
NAM   OPENIO      DECK-ID J48  MSOS 4.3           SUMMARY-102J4800001
NAM   ROWTR       DECK-ID J49  MSOS 4.3           SUMMARY-102J4900001
NAM   LIOC        DECK-ID J50  MSOS 4.3           SUMMARY-102J5000001
NAM   COPY        DECK-ID J51  MSOS 4.3           SUMMARY-102J5100001
NAM   EXIT        DECK-ID J52  MSOS 4.3           SUMMARY-102J5200001
NAM   PRINT       DECK-ID J53  MSOS 4.3           SUMMARY-102J5300001
```

Figure D-1. Unformatted Character Dump



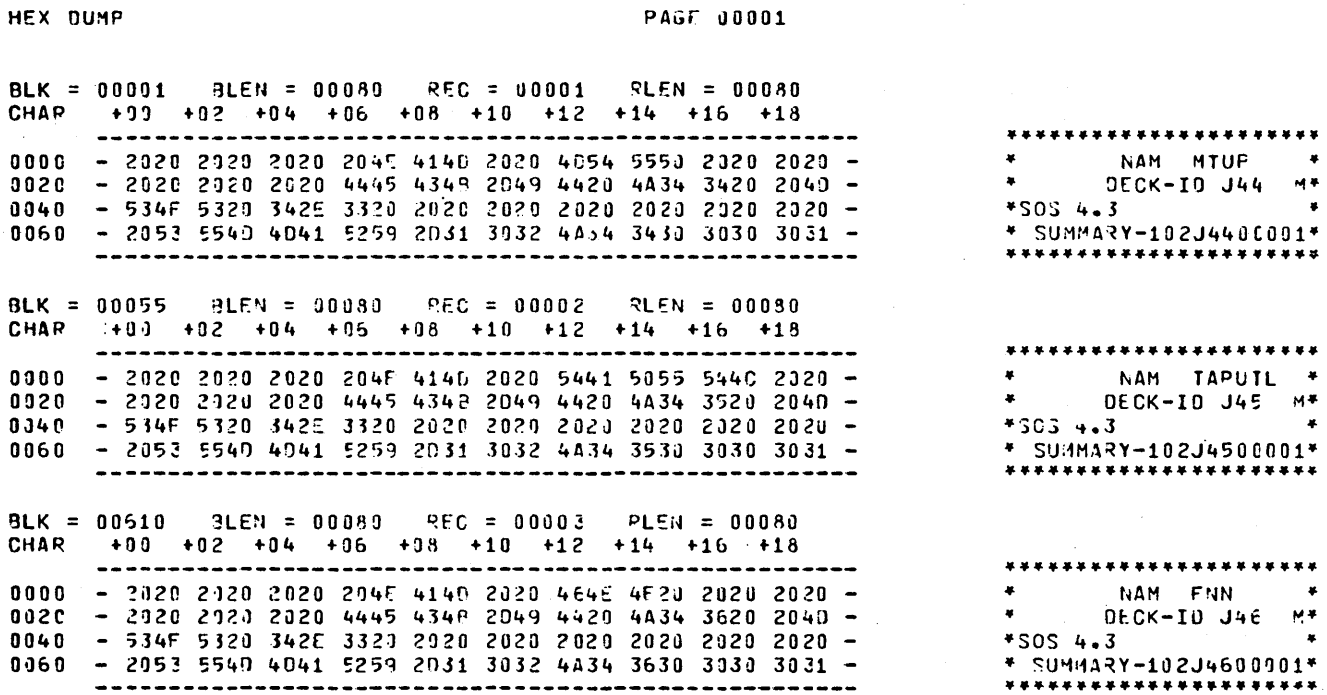
```
ASCII DUMP                                       PAGE 00001

BLK = 00001   BLEN = 00080   REC = 00001   RLEN = 00080
CHAR    POSITION
        00   05   10   15   20   25   30   35   40   45   50   55   60   65   70   75  79
     ------------------------------------------------------------------------------------
0000 -        NAM  MTUP          DECK-ID J44  MSOS 4.3                SUMMARY-102J4400001 -
     ------------------------------------------------------------------------------------

BLK = 00055   BLEN = 00080   REC = 00002   RLEN = 00080
CHAR    POSITION
        00   05   10   15   20   25   30   35   40   45   50   55   60   65   70   75  79
     ------------------------------------------------------------------------------------
0000 -        NAM  TAPUTL        DECK-ID J45  MSOS 4.3                SUMMARY-102J4500001 -
     ------------------------------------------------------------------------------------

BLK = 00610   BLEN = 00080   REC = 00003   RLEN = 00080
CHAR    POSITION
        00   05   10   15   20   25   30   35   40   45   50   55   60   65   70   75  79
     ------------------------------------------------------------------------------------
0000 -        NAM  FNN           DECK-ID J46  MSOS 4.3                SUMMARY-102J4600001 -
     ------------------------------------------------------------------------------------
```

Figure D-2. Formatted Character Dump



```
HEX DUMP                                         PAGE 00001


BLK = 00001   BLEN = 00080   REC = 00001   RLEN = 00080
CHAR   +00  +02  +04  +06  +08  +10  +12  +14  +16  +18
       ---------------------------------------------------          **********************
0000   - 2020 2020 2020 204E 414D 2020 4D54 5550 2020 2020 -          *         NAM  MTUP    *
0020   - 2020 2020 2020 4445 434B 2049 4420 4A34 3420 204D -          *       DECK-ID J44   M*
0040   - 534F 5320 342E 3320 2020 2020 2020 2020 2020 2020 -          *SOS 4.3                 *
0060   - 2053 554D 4D41 5259 2D31 3032 4A34 3430 3030 3031 -          * SUMMARY-102J4400001*
       ---------------------------------------------------          **********************

BLK = 00055   BLEN = 00080   REC = 00002   RLEN = 00080
CHAR   +00  +02  +04  +05  +08  +10  +12  +14  +16  +18
       ---------------------------------------------------          **********************
0000   - 2020 2020 2020 204F 414D 2020 5441 5055 544C 2020 -          *         NAM  TAPUTL  *
0020   - 2020 2020 2020 4445 434B 2049 4420 4A34 3520 204D -          *       DECK-ID J45   M*
0040   - 534F 5320 342E 3320 2020 2020 2020 2020 2020 2020 -          *SOS 4.3                 *
0060   - 2053 554D 4D41 5259 2D31 3032 4A34 3530 3030 3031 -          * SUMMARY-102J4500001*
       ---------------------------------------------------          **********************

BLK = 00610   BLEN = 00080   REC = 00003   PLEN = 00080
CHAR   +00  +02  +04  +06  +08  +10  +12  +14  +16  +18
       ---------------------------------------------------          **********************
0000   - 2020 2020 2020 204E 414D 2020 464E 4E20 2020 2020 -          *         NAM  FNN     *
0020   - 2020 2020 2020 4445 434B 2049 4420 4A34 3620 204D -          *       DECK-ID J46   M*
0040   - 534F 5320 342E 3320 2020 2020 2020 2020 2020 2020 -          *SOS 4.3                 *
0060   - 2053 554D 4D41 5259 2D31 3032 4A34 3630 3030 3031 -          * SUMMARY-102J4600001*
       ---------------------------------------------------          **********************
```

Figure D-3. Formatted Hexadecimal Dump

COMMENT SHEET

MANUAL TITLE CONTROL DATA® Magnetic Tape Utility Processor Reference Manual

PUBLICATION NO. 96768400 REVISION A

FROM NAME:

BUSINESS
ADDRESS:

COMMENTS: This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number to which your comment applies.

CUT ALONG LINE

STAPLE STAPLE

FOLD

FIRST CLASS
PERMIT NO. 333

LA JOLLA, CA.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
PUBLICATIONS AND GRAPHICS DIVISION
4455 EASTGATE MALL
LA JOLLA, CALIFORNIA 92037

CUT ALONG LINE

FOLD

STAPLE STAPLE

CORPORATE HEADQUARTERS, P.O. BOX 0, MINNEAPOLIS, MINNESOTA 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A.

CONTROL DATA CORPORATION