IBM System/3 Magnetic Tape Program Planning Manual

GC21-5040-3
File No. S3-05

Page of GC21-5040-3
Issued 30 June 1978
By TNL: GN21-5604

Fourth Edition (March 1976)

This is a major revision of, and obsoletes, GC21-5040-2. This manual has been changed to include the Model 8 and the Model 12. Changes are indicated by a vertical line at the left of the change; new or extensively revised illustrations are denoted by a bullet ● at the left of the figure caption.

Changes are periodically made to the information herein; before using this publication in connection with the operation of IBM systems, refer to the latest *IBM System/3 Bibliography,* GC20-8080 for the editions that are applicable and current.

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 IBM 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. Comments become the property of IBM.

Preface

This manual introduces you to the 3410/3411 Magnetic Tape Subsystem for the IBM System/3. This manual will help installation managers decide if they want to use magnetic tape and aid in evaluating the effect that it will have on their current operations. Information contained in the manual will aid in planning for the use of magnetic tape.

The manual contains:

- A description of magnetic tape
- A description of the organizations of tape files
- A description of tape labels
- A description of control statement requirements
- A description of the Tape Initialization program
- A description of the functions supported by RPG II, COBOL, FORTRAN IV, Basic Assembler, Disk Sort, Magnetic Character Reader Utility, and Tape Sort program products

The reader should be familiar with the operation and programming concepts of the System/3.

A glossary is provided at the back of this manual to define important terms.

Related Publications

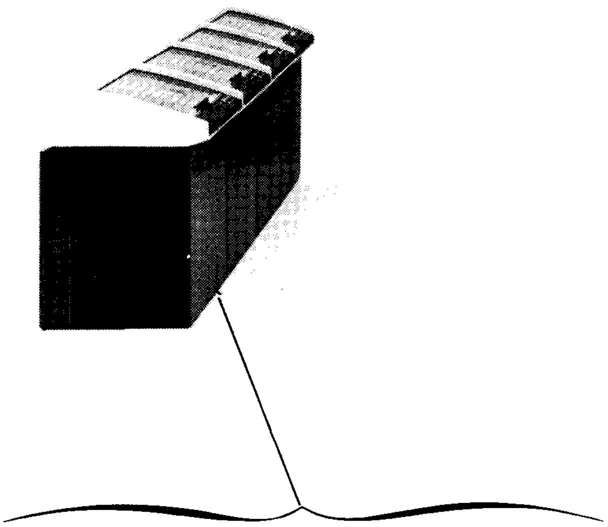
	Order Number of Publication for				
	System/3 Model 10 Card System	System/3 Model 8	System/3 Model 10 Disk System	System/3 Model 12	System/3 Model 15
Operator's Guide	GC21-7513	GC21-7634	▨		
RPG II Reference Manual	SC21-7500	SC21-7504			
System Control Programming Reference Manual	▨	GC21-7512		GC21-5130	GC21-5077[1] GC21-5162[2]
Halt Guide (System Messages)	▨	GC21-7540		GC21-5145	GC21-5076
System Control Programming Macros Reference Manual	▨	GC21-7562			GC21-7608
User's Guide	▨	▨	▨	GC21-5142	▨
Disk Concepts and Planning Guide	▨	GC21-7571			
Subset American National Standard COBOL Reference Manual	▨	GC28-6452			
FORTRAN IV Reference Manual	▨	SC28-6874			
Basic Assembler Reference Manual	▨	SC21-7509			
Disk Sort Reference Manual	▨	SC21-7522			
Tape Sort Reference Manual	SC21-7572				
Components Reference Manual	GA21-9236				

[1] Models 15A, B, C
[2] Model 15D

Contents

Chapter 1. Introduction

The IBM 3410/3411 Magnetic Tape Subsystem (Models 1, 2, and 3) reads and writes half-inch, 7- or 9-track magnetic tape. The subsystem provides an economical approach for increasing the storage capacity of the IBM System/3. (Seven-track magnetic tape is not available for the Model 10 Card System.)



3411 Magnetic Tape Unit and Control	3410 Magnetic Tape Unit	3410 Magnetic Tape Unit	3410 Magnetic Tape Unit

System/3 supports a maximum of four tape units. The 3411 is required. Up to three 3410s can be attached via the 3411 provided the 3410s are the same model as the 3411.

ADVANTAGES OF A TAPE SYSTEM

Compared with cards or diskettes, records on tape may be any size (within limits) depending on the storage capacity of the processing unit and the program products for the system (see *Magnetic Tape Support Summary* for allowable record lengths). This allows writing pertinent information in a continuous data record. Repeating the identification in multiple records is eliminated.

Magnetic tape records, like disk records, are permanent. However, during a tape write operation, the previously recorded information is erased. This means that tape may be used again and again with a significant cost saving over the use of cards. Diskettes are also reusable.

There are other advantages in using magnetic tape:

- Large amounts of data can be stored economically.
- With appropriate spacing between groups of records, approximately 112 boxes of System/3 96-column cards can be stored on a 2400-foot reel of tape.
- History files can be retained indefinitely.
- A duplicate of disk pack records can be kept.
- Information can be transferred easily to other data processing locations.
- System/3 magnetic tape is compatible with magnetic tape for other systems.

Chapter 2. Magnetic Tape Concepts

RECORDING DATA ON MAGNETIC TAPE

Data is recorded by moving tape across a read/write head. It is written while the tape is moving forward but may be read either forward or backward. Data is written as a series of magnetized bits in parallel tracks along the length of the tape.

The recording density is the number of bits in a single linear track measured per unit length of tape. For 9-track tape, the 3410/3411 Magnetic Tape Subsystem records in densities of 800 or 1600 bpi (bits or bytes per inch). A dual density feature is available that permits operation at either 800 or 1600 bpi. For 7-track tape, densities can be 200, 556, or 800 bpi.

FILE PROTECTION

The 3410/3411 uses a plastic write-enable ring mounted on the tape reel to permit writing. If a tape is mounted without the ring in position, writing cannot occur; therefore, the file is protected.

INTERSYSTEM TAPE EXCHANGE

Tapes produced on the 3410/3411 and all other IBM half-inch tape units operating in the same density are interchangeable; output data produced on one system, such as the System/370, can be used as direct input to another system, such as System/3.

TAPE FILE FORMATS

Three types of file formats can be used with the magnetic tape subsystem:

- Single file on a single volume (reel)
- Single file on several volumes (multivolume file)
- Several files on one or more volumes on the Model 15 (multifile volume)

The formats of the labeled files are shown in Figures 1 and 2; the formats of the unlabeled files are shown in Figures 3 and 4.

The labels of each reel of a multivolume file must be consistent. That is, all volumes of a labeled multivolume file must be labeled, and all volumes of an unlabeled multivolume file must be unlabeled.

Multivolume tape files cannot be specified on the Model 8, Model 10 Disk System, or Model 12 for programs that also use the BSCA or devices attached to the SIOC.

Similarly, files in a multifile volume must be consistent. That is, all files on the volume must be labeled or all must be unlabeled. See the system control programming reference manual for Model 15 listed in the *Preface* for more information on multifile tape volumes.

TAPE LABELS

System/3 follows either the American National Standards Institute (ANSI) for magnetic tape labels or IBM Magnetic Tape Label Standards. This provides compatibility with other systems that use either ANSI or the IBM Magnetic Tape Label Standards.

External Labels

You can attach external labels to the outside of tape reels for identification. These labels should include the reel (or volume) serial number, name of the file, file number, date the file was created, number of the volume (if a multi-volume file), and other information required for installation.

Internal Labels

The internal labels, volume and file, are written on tape to provide a programmed identification of the file. File labels provide a safeguard against mishandling files. Labels are checked by tape data management to determine whether the reel containing the desired file has been mounted before any processing occurs. Thus, file labels ensure that the correct reel is used for input, and that no current data is destroyed on output.

Volume labels for labeled tapes are written by the System/3 Tape Initialization program at the time a reel is prepared for use. The volume label identifies the reel and is always the first record in a reel. The information contained in the volume label is checked by tape data management and never altered during file processing. Any additional volume labels are ignored by tape data management.

Nonstandard labels contain whatever information the user desires to include. These labels may vary in length and format and are the complete responsibility of the user. Nonstandard labels are not checked by System/3 tape data management for input files, and they may not be written for output files. For EBCDIC input files with nonstandard labels, the first record on the volume cannot be an 80-byte record beginning with VOL1; for ASCII input files with nonstandard labels, records cannot be greater than 80 bytes beginning with VOL1.

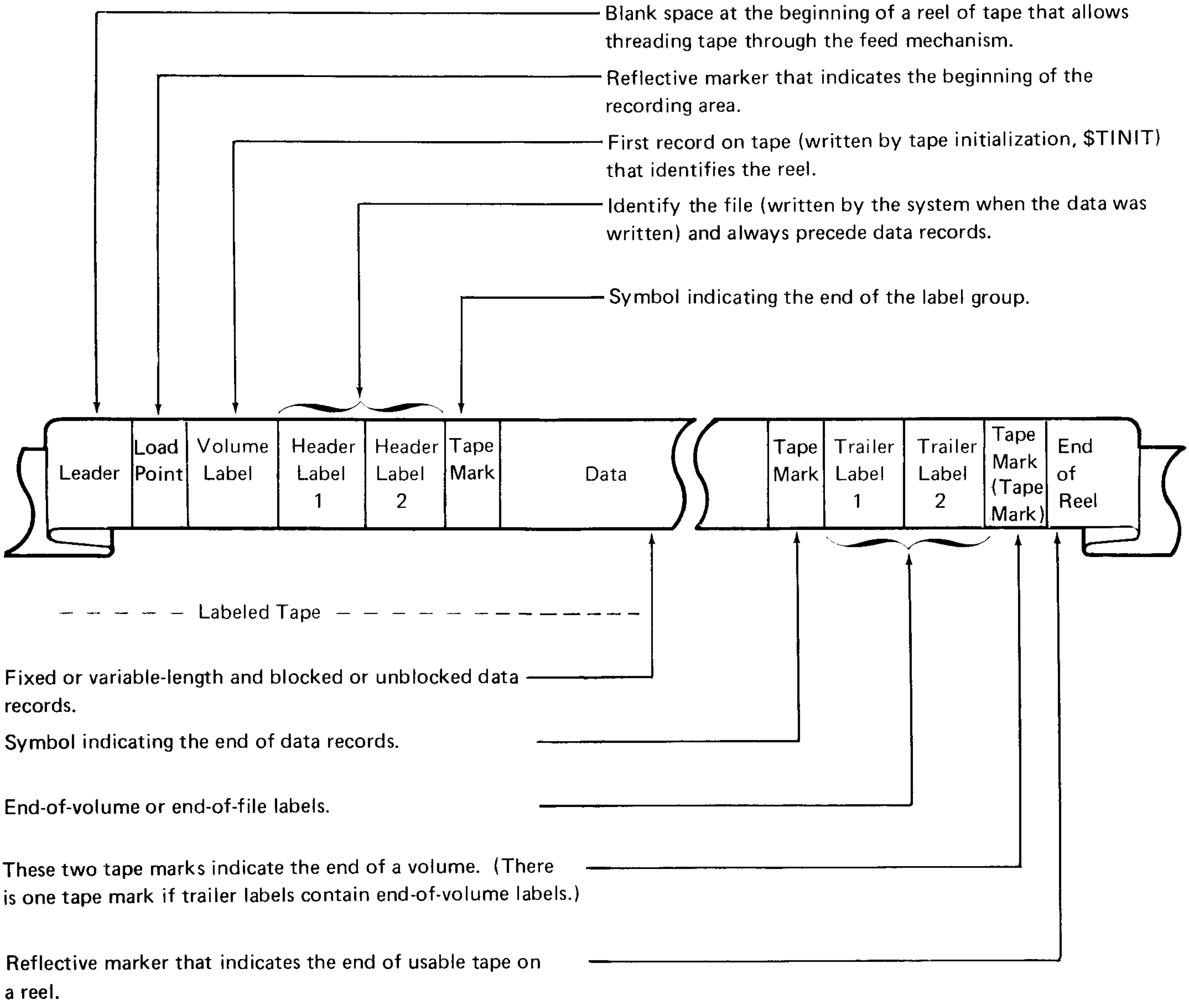
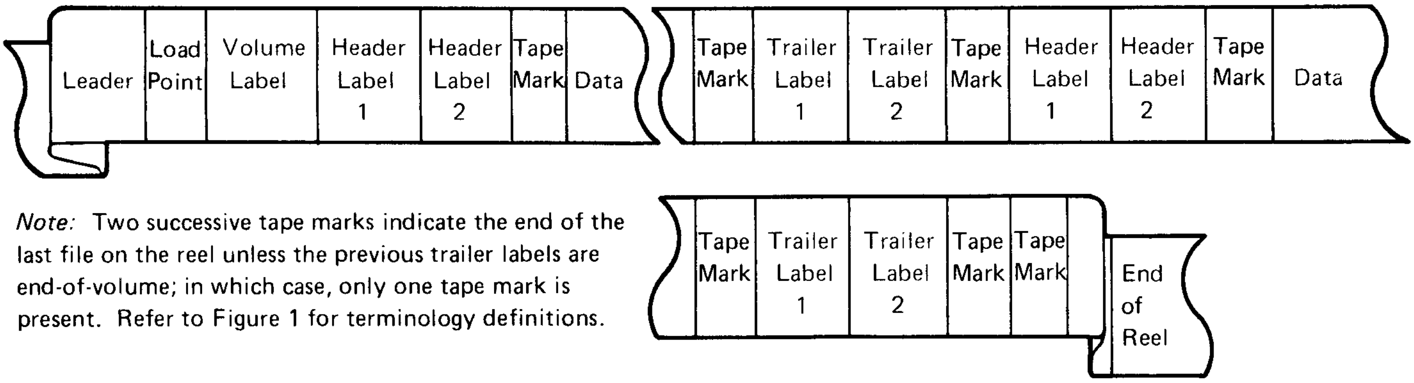
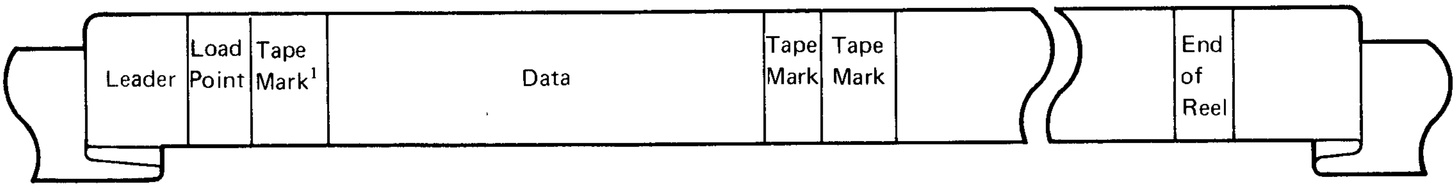

Blank space at the beginning of a reel of tape that allows threading tape through the feed mechanism.
Reflective marker that indicates the beginning of the recording area.
First record on tape (written by tape initialization, $TINIT) that identifies the reel.
Identify the file (written by the system when the data was written) and always precede data records.
Symbol indicating the end of the label group.
Leader
Load Point
Volume Label
Header Label 1
Header Label 2
Tape Mark
Data
Tape Mark
Trailer Label 1
Trailer Label 2
Tape Mark (Tape Mark)
End of Reel
Labeled Tape
Fixed or variable-length and blocked or unblocked data records.
Symbol indicating the end of data records.
End-of-volume or end-of-file labels.
These two tape marks indicate the end of a volume. (There is one tape mark if trailer labels contain end-of-volume labels.)
Reflective marker that indicates the end of usable tape on a reel.

Figure 1. Tape Formats: Labeled Single or Multivolume Files



Note: Two successive tape marks indicate the end of the last file on the reel unless the previous trailer labels are end-of-volume; in which case, only one tape mark is present. Refer to Figure 1 for terminology definitions.

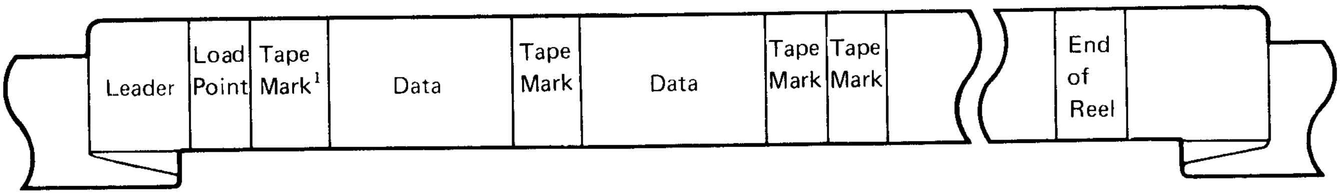
● **Figure 2. Tape Formats: Labeled Multifile Volume**



[1] Optional; if present, it is ignored.

Note: Two tape marks are written at the end of a single volume file and at the end of the last volume of a multivolume file. Other volumes of a multivolume file have one tape mark at the end. Refer to Figure 1 for terminology definitions.

● **Figure 3. Unlabeled Single or Multivolume Files**



[1] Optional; if present, it is ignored.

Note: Two tape marks indicate the end of the volume unless the last file on the volume is multivolume file, in which case, only one tape mark is present. Refer to Figure 1 for terminology definitions.

● **Figure 4. Tape Formats: Unlabeled Multifile Volume**

User-standard labels are file labels that follow standard header and trailer label conventions (ANSI or IBM). They are a variation of standard labels with a partially fixed format. These labels are sometimes provided by other systems. User-standard labels are not checked by System/3 tape data management and may not be written as part of the label group.

File labels are written before and after the logical file on a reel. These labels are referred to as file header labels or file trailer labels, depending on their position and use. Header labels precede the data file. Trailer labels are either end-of-file or end-of-volume. End-of-file trailer labels are written at the end of the logical file. End-of-volume trailer labels are written at the end of a reel indicating that the file is continued on another reel. The label formats are shown in Appendix A, *Tape Label Formats.*

TAPE RECORD ATTRIBUTES

Files consist of a collection of logical records. The logical record is the basic unit of information for a program. The logical records are either fixed or variable length.

Fixed Length Records

Fixed length records have a constant number of characters within a record. The records can be either blocked or unblocked.

Variable Length Records

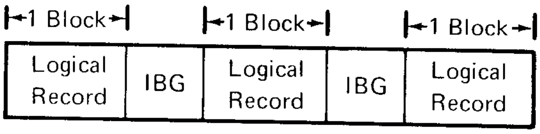
Variable length records have a different number of characters within each record. The records can also be either blocked or unblocked. The first four bytes of the logical record contain the length of that record. Variable length records cannot be used with the Model 10 Card System.

TAPE BLOCKING FACTORS

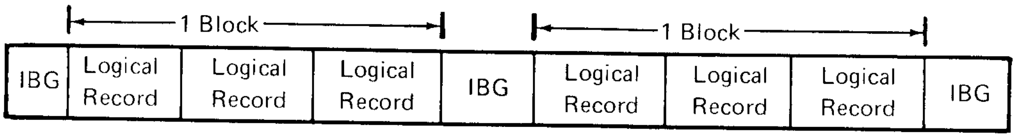
Records that are not grouped are unblocked; that is, each record makes up a physical block. Blocking the records is the process of combining two or more logical records before they are written on tape. Records are processed by the system in either the blocked or unblocked format. The system handles the blocking and deblocking of logical records so you need to determine only the most efficient block size for the file.

Records or physical blocks are separated on tape by an IBG (interblock gap). The system produces this gap during tape write operations.

Logical Unblocked Records



Logical Blocked Records



The number of logical records combined into one physical record or block is the blocking factor. If you specify an appropriate blocking factor for an application, many of the interblock gaps are eliminated, and the efficiency of processing is improved.

Blocking improves the effective processing speed by reducing the number of I/O operations required to process a file. For example, if 30 unblocked records were written on tape, 30 write operations would be required. By specifying a blocking factor of 10, only three write operations are required. The block length must be specified in multiples of the logical record length. Your selection of the block length is based on the amount of main storage available (greater tape efficiency is obtained with larger blocks).

Approximately 40,000 System/3 cards (20 boxes) punched in all 96 columns may be contained in 2400 feet of tape if the data is written in the 1600 bpi unblocked format. If a blocking factor of 10 were specified, approximately 224,000 cards (112 boxes) could be stored on the same amount of tape using the same density.

TAPE CHARACTER CODES

The system is capable of handling tape data files in one of two codes: EBCDIC (extended binary coded decimal interchange code) or ASCII (American National Standard Code for Information Interchange). The following list shows the tape formats supported by System/3 for the tape data files. (Variable length records are not supported by the Model 10 Card System.)

Fixed length records:
  Blocked or unblocked
  EBCDIC or ASCII code

Variable length records:
  Blocked or unblocked
  EBCDIC code (Format V)
  ASCII code (Format D)

ASCII Files

ASCII is a 7-bit code containing 128 characters. ASCII files are specified on a FILE control statement (see *Control Statement Requirements*). When translating ASCII files, an entire record is translated without field differentiation. The input files are translated to EBCDIC from ASCII by tape data management, and the output files are translated from EBCDIC to ASCII before being written. Labels on tape files which contain ASCII data must conform to the ANSI format.

EBCDIC Files

EBCDIC is an 8-bit code containing 256 characters. If there is no installation requirement for ASCII data interchange, magnetic tape files should be recorded in EBCDIC; EBCDIC is the natural code representation for the System/3 processing unit. The use of EBCDIC eliminates the object program overhead which results from the translation of ASCII to EBCDIC.

Chapter 3. Magnetic Tape System Support

System/3 program support provides input and output capability for data storage files. The FILE statement, which defines the tape file, and the tape utility programs are described in this chapter.

CONTROL STATEMENT REQUIREMENTS

Model 10 Card System

// LOG $\left\{\begin{array}{l}\underline{\text{ON}}\\ \text{OFF}\end{array}\right\}$

The LOG control card indicates whether you want the FILE cards and diagnostic messages printed. The format of the card is illustrated using the conventions shown in Figure 5.

If the LOG card is not used, the FILE cards, diagnostic messages, and RUN cards will be printed as if LOG ON was specified. The LOG control card precedes the FILE cards.

```
// FILE list of parameters
```

A FILE control card contains information concerning the tape files. You must supply one FILE card for each new tape file created by your program and for each of the existing tape file that your programs use.

The parameters that are allowed on the control card are shown in Figure 6. The keyword parameters may be listed on the control card in any order. General coding rules for the FILE card are contained in the *IBM System/3 Card System Operator's Guide*, GC21-7513.

```
// RUN
```

A RUN control card must follow the last FILE control card. This card serves to delimit the end of the control card specifications and requests that the tape program be executed. Model 10 Card System control cards are upward compatible with those of the other System/3 models.

Model 8, Model 10 Disk System, Model 12, and Model 15

A set of OCL statements must be supplied for each program run. Within this set of statements is the FILE statement which provides the system with information about the tape files. The formats of the statements are illustrated using the conventions shown in Figure 5.

```
// FILE list of parameters
```

You must supply one FILE statement for each new tape file created by your program and for each of the existing tape files your program uses. The parameters allowed on the FILE statement are shown in Figure 6. The keyword parameters may be listed on the FILE statement in any order.

General coding rules for the FILE statement are contained in the SCP publications listed in the Preface.

BRACES { } indicate a choice of entries.

BRACKETS [ ] indicate an optional entry.

UNDERSCORE in an optional entry means that the underscored value is assumed if the entry is not present in the statement.

HYPHENS separate keywords and parameters.

COMMAS separate all entries.

SPECIAL CHARACTERS, such as //, must appear as they are shown in the statement format.

WORDS OR ABBREVIATIONS IN CAPITAL LETTERS, such as BLKL and NL, must appear as shown in the statement format.

SMALL LETTERS, such as yyy and mn, represent numbers or other information you must supply. The number of letters indicates the number of characters in the information or the number of digits in the numbers. Three dots, as in x...x, indicate that the number of characters or digits may vary.

Figure 5. Coding Conventions for Control Statements

Keyword Parameters for the FILE Statement

```
NAME-filename
```

NAME is a required keyword parameter in the FILE statement. It specifies the name your program uses to refer to the tape file. The name can be up to eight characters long. The first character must be alphabetic, and the remaining characters can be alphameric.

Parameters	Model 10 Card System		Other System/3 Models	
	Required	Optional	Required	Optional
NAME	X		X	
UNIT	X		X	
REEL				
Input files	X		X	
Output files		X		X
LABEL		X		X
DATE		X		X
RETAIN		X		X
BLKL		X		X
RECL		X		X
RECFM				X
END		X		X
DENSITY		X		X
ASCII		X		X
DEFER		X		X
CONVERT				X
TRANSLATE				X
PARITY				X
SEQNUM[1]				X

[1] Model 15 only

Figure 6. Keyword Parameters for the FILE Statement

```
         ⎧ T1             ⎫
         ⎪ T2             ⎪
UNIT-    ⎨ T3             ⎬
         ⎪ T4             ⎪
         ⎩ 'Tn,......,Tn' ⎭
```

UNIT is a required keyword parameter in the FILE statement. It tells the system which tape units contain the file. For a multivolume file, the units must be coded in the order the files are to be processed, and the parameter must be enclosed in quotes. Quotes are not required if only one tape unit is used for a multivolume file.

If your file resides on more reels than you have tape units, the tape units are used alternately. For example, if the system configuration includes two tape units (specified by T1 and T2 on the UNIT parameter) and your multivolume file is contained on three volumes, the system processes the first volume on T1, the second volume on T2, and the third volume on T1

```
       ( nnnnnn 1                                )
       ( 'nnnnnn, . . . , . . . , nnnnnn' 2      )
       ( NL <-------- 3                          )
REEL-  ( 'NL,n' <---- 4                          )
       ( NS <-------- 5                          )
       ( 'NS,n' <---- 6                          )
       ( BLP <------- 7                          )
       ( 'BLP,n' <--- 8                          )
```

REEL is a required keyword parameter in the FILE statement for input files and is optional for output files. Its function is to identify the volume label of the tape on which the file resides. The following describes the numbered parameter entries:

1 This entry is used for labeled volumes. A single volume file is identified by coding a maximum of six characters not including the quote, comma, and blank.

2 This entry is used for labeled volumes. Each reel of a multivolume file must be identified separately. The reels are processed in the order they appear. Six characters, not including the quote, comma, and blank, identify a reel. The entries specified must be enclosed in quotes.

3 The specification NL (not labeled) is used if the file resides on an unlabeled tape. The first record of an unlabeled tape must not be an 80-byte record beginning with VOL1.

4 If a multivolume file resides on unlabeled tapes, the number of unlabeled tapes that contain the file is specified in this entry. The default value for n is 1.

5 The specification NS (nonstandard) indicates an input file with labels that do not adhere to the IBM Tape Label Standard. The first record of a nonstandard labeled tape must not be an 80-byte record beginning with VOL1.

6 If a multivolume input file resides on tapes that do not adhere to the IBM Tape Label Standard, the number of tapes that contain the file is specified in this entry. The default value for n is 1.

7 The specification BLP (bypass label processing) indicates that the input file resides on a standard labeled volume and label processing is to be bypassed. REEL-BLP is not supported on the Model 10 Card System.

8 If a multivolume input file resides on standard labeled tapes and label processing is bypassed, the number of labeled tapes that contain the file is specified in this entry. The default value for n is 1.

If the REEL keyword is not used for an output file, then the system assumes a standard labeled tape is mounted.

Multivolume tape files cannot be specified on the Model 8, Model 10 Disk System, or Model 12 for programs that also use the BSCA or devices attached to the SIOC.

```
        ( name               )
LABEL-  (                    )
        ( 'character string' )
```

LABEL is an optional keyword parameter in the FILE statement. It is used to specify the name (label) of the tape file as it exists in the header label. If LABEL is not specified, the system defaults to the NAME parameter unless REEL-NS or REEL-NL is also specified. LABEL cannot be used when REEL-BLP is specified.

The name entry must begin with one of the 29 alphabetic characters; the remaining characters must not be a quote, comma, or blank. If the name of the file in the header label exceeds eight characters, only the first eight characters are used.

A label may also be specified using special characters. The character string must be enclosed in quotes, cannot contain commas, and is restricted to eight characters in length.

```
       ( mmddyy )
DATE-  (        )
       ( ddmmyy )
```

DATE is an optional keyword parameter in the FILE statement. It ensures that the proper input or output file is referenced by verifying the date in the file label (standard labels). DATE cannot be used when REEL-BLP is specified.

The date may be coded in one of two forms, month-day-year (mmddyy) or day-month-year (ddmmyy). The format must match the system date format chosen at system generation time (except Model 10 Card System). The system date is used as the creation date for output files. It may be changed for a program being run on the Model 8, Model 10 Disk System, Model 12, or Model 15 by placing the OCL DATE statement after the LOAD statement.

RETAIN-nnn

RETAIN is an optional keyword parameter in the FILE statement; it indicates the number of days a file should be retained before it is scratched. The number may be from 0 to 999. Leading zeros are not required. After the number of days has elapsed, the file expires, and the system allows the file to be written over. The system assumes a value of zero when the parameter is omitted. The value of 999 is specified to create a nonexpiring permanent tape file.

If an attempt is made to write over an unexpired file, the system halts and allows the operator to cancel the job or continue.

RETAIN is used for standard labeled files only. RETAIN cannot be used if REEL-BLP is specified.

BLKL-block length

BLKL is an optional keyword parameter in the FILE statement; it indicates the number of bytes in a physical block of data on tape. The minimum size, fixed length block (FB) that can be specified is 18 bytes. Variable length (VB or DB) blocks are padded with X'00' (EBCDIC) or X'5E' (ASCII), when necessary, to meet the 18-byte minimum block length requirement. The maximum size block that can be specified, regardless of record format, is 32,767.

When fixed blocked EBCDIC records are used, the block length must be an integral multiple of the record length. When fixed blocked ASCII records are used, the block length must include the buffer offset length. When the buffer offset length is subtracted from the block length, the remainder must be an integral multiple of the record length. For a file containing blocked EBCDIC variable length (VB) records, the block length must include the 4-byte block descriptor and the 4-byte record descriptor(s). For blocked ASCII variable length (DB) records, the buffer offset length and the 4-byte record descriptor(s) must be included in the block length.

RECL-record length

RECL is an optional keyword parameter in the FILE statement; it indicates the number of bytes in a logical tape record. The minimum record length permitted for fixed (F) or fixed blocked records is 18 bytes. Unblocked variable length (V or D) records are padded with X'00' (EBCDIC) or X'5E' (ASCII), when necessary, to meet the 18-byte minimum, block length requirements. For files containing variable length (V or D) records, the record length must include the 4-byte record descriptor.

$$\text{RECFM-}\left\{\begin{array}{c}\text{F}\\ \text{V}\\ \text{D}\\ \text{FB}\\ \text{VB}\\ \text{DB}\end{array}\right\}$$

RECFM is an optional keyword parameter in the FILE statement; it identifies the format of the input or output records. It is not used with the Model 10 Card System. The parameter entries are:

F – Fixed length, unblocked records. Logical and physical records are the same length.

V – Variable length, unblocked records. Each physical record contains one logical record; the logical record can vary in length.

D – Variable length, unblocked records in the D-type ASCII format.

FB – Fixed length, blocked records. All records are of equal length and all blocks are of equal length. Each physical record can contain more than one logical record.

VB – Variable length, blocked records. Each physical record contains logical records of various lengths.

DB – Variable length, blocked records in the D-type ASCII format.

Note: U-type (unspecified) records are not supported.

$$\text{END-}\begin{Bmatrix}\text{LEAVE}\\ \text{UNLOAD}\\ \underline{\text{REWIND}}\end{Bmatrix}$$

END is an optional keyword parameter in the FILE statement; its function is to control the position of the tape after the file has been processed. The tape remains in its current position if LEAVE is specified. If UNLOAD is specified, the system rewinds the tape and positions it for removal from the tape unit at job termination time. The tape is rewound to the load point if the REWIND option is specified. The system assumes REWIND if the END parameter is not specified.

$$\text{DENSITY-}\begin{Bmatrix}1600\\ 800\\ 556\\ 200\end{Bmatrix}$$

DENSITY is an optional parameter in the FILE statement that is used to specify the number of bpi at which tape files are to be written or read. For 9-track tape, this parameter affects only the density of unlabeled output tapes and is ignored for input tapes. When standard labeled or nonstandard labeled tapes are used, the 9-track tape hardware automatically determines the density at which the tape has been initialized. When a file is initialized to 1600 bpi with standard labels, any file that is written on that tape is 1600 bpi regardless of the DENSITY parameter. No halts occur if the wrong 9-track density is specified.

The parameter entries are 800 or 1600 bpi for 9-track tape, and 200, 556, or 800 bpi for 7-track tape. If the DENSITY parameter is omitted, 1600 bpi is assumed for 9-track tape and 800 bpi for 7-track tape.

$$\text{ASCII-}\begin{Bmatrix}\text{YES}\\ \underline{\text{NO}}\end{Bmatrix}$$

ASCII is an optional keyword parameter in the FILE statement; it informs the system when an ASCII file is being used. If ASCII files are being processed, ASCII-YES must be specified. ASCII-YES is invalid for 7-track tape; if this parameter is omitted or coded ASCII-NO, an EBCDIC file is assumed.

$$\text{DEFER-}\begin{Bmatrix}\text{YES}\\ \underline{\text{NO}}\end{Bmatrix}$$

DEFER is an optional keyword parameter in the FILE statement. It is required if the tape volume is not online at the beginning of the job. If this parameter is omitted, the system assumes that the tape volume is online.

DEFER-YES cannot be specified for programs on the Model 8, Model 10 Disk System, or Model 12 that also use BSCA or devices attached to the SIOC.

$$\text{CONVERT-}\begin{Bmatrix}\text{ON}\\ \underline{\text{OFF}}\end{Bmatrix}$$

CONVERT is an optional keyword parameter in the FILE statement. This parameter is valid only for 7-track tape files. It informs the system that the data converter is on or off. CONVERT-ON causes 7-track data to be processed in 8-bit binary form. The converter writes three main storage characters as four tape characters and converts the opposite way when reading. If this parameter is omitted, CONVERT-OFF is assumed. When processing variable length records on 7-track tape files, CONVERT-ON is required.

$$\text{TRANSLATE-}\begin{Bmatrix}\text{ON}\\ \underline{\text{OFF}}\end{Bmatrix}$$

TRANSLATE is an optional keyword parameter in the FILE statement. This parameter is valid only for 7-track tape files. It informs the system that the data translator is on or off. TRANSLATE-ON causes 7-track data to be processed in six-bit BCD form. The translator writes 8-bit EBCDIC main storage characters as 6-bit BCD tape characters and translates the opposite way when reading. If this parameter is omitted, TRANSLATE-OFF is assumed.

Note: If CONVERT-OFF and TRANSLATE-OFF are specified, only the six low-order bits of the main storage character are written on the tape. When reading with CONVERT-OFF and TRANSLATE-OFF, the two high-order bits of the main storage characters are set to zeros.

$\text{PARITY-}\begin{Bmatrix}\underline{\text{ODD}}\\ \text{EVEN}\end{Bmatrix}$

PARITY is an optional keyword parameter in the FILE statement. This parameter is valid only for 7-track tape files. It is used to specify the parity at which tape characters will be processed. If this parameter is omitted, PARITY-ODD is assumed.

Note: Certain combinations of CONVERT, TRANSLATE, and PARITY are invalid.

	CONVERT	TRANSLATE	PARITY
Valid	OFF	OFF	ODD
	OFF	OFF	EVEN
	OFF	ON	ODD
	OFF	ON	EVEN
	ON	OFF	ODD
Invalid	ON	OFF	EVEN
	ON	ON	ODD
	ON	ON	EVEN

$\text{SEQNUM-}\begin{Bmatrix}\text{number}\\ \text{X}\end{Bmatrix}$

SEQNUM is an optional keyword parameter in the FILE statement. It is used only on the Model 15 to position a file in a multifile volume for processing. It can also be used to indicate that tape positioning is not required.

If SEQNUM-number is used with REEL-BLP, the system searches the tape for a standard label HDR1 record containing the file number. When the record is found, further processing of the label group is terminated. The system then positions the tape to the file data. See the system control programming reference manual for Model 15 listed in the *Preface* for additional information on multifile tape volumes.

TAPE DATA MANAGEMENT

IBM System/3 tape management routines regulate the use of the magnetic tape units. The tape data management routines are included in the object program when your source programs are compiled.

Tape data management performs initial setup and label checking before the file is opened. It determines if the file is online. Each file, described by the FILE statement, which does not contain a DEFER parameter, is expected to be online at the beginning of the execution of the program. When a new file is being written, tape data management checks for an existing file (if it has standard labels) to ensure that an unexpired file is not destroyed.

Tape data management provides support for sequential processing of tape files. The two logical access methods are consecutive input and consecutive output. Consecutive input retrieves records from a sequential file with records obtained from contiguous locations beginning with the first record. Consecutive output writes the records sequentially on tape building a tape file beginning with the first record.

End of Volume Processing

The end-of-reel reflective marker indicates the end of tape when a file is written. When the marker is detected, tape data management writes a tape mark. Trailer labels (EOV, end-of-volume) are written if the file is a labeled file. Tape data management then issues a rewind/unload command to the tape unit. The next tape volume, as specified on the FILE statement, is then opened, header labels are written for a labeled file, and data management continues to write the file. When the last volume of the output file has been written, data management writes trailer labels (end-of-file) for a labeled file and issues the optional rewind/unload command (END parameter on the FILE statement) to the last tape unit.

When reading an input file, recognition of a tape mark indicates an end condition. Trailer labels are read and checked if the tape file is labeled. For a multivolume tape file, the next volume is opened as specified on the FILE statement.

Dual Programming (Models 8, 10, and 12) or Multiprogramming (Model 15)

You can address separate tape units from any program level (partition) on these systems; however, a tape unit cannot be in use by two program levels (partitions) simultaneously. Required tape units are allocated to the program by the system to ensure that two program levels (partitions) are not using the same unit.

For the Model 8 or the Model 10 Disk System, a processing unit with a minimum of 16K main storage is required to support a tape environment in a dual programming system.

MAGNETIC TAPE UTILITY PROGRAMS

Four card-resident support programs are available for tape on the Model 10 Card System:

- The Magnetic Tape Error Logout program punches the contents of counters accumulated during the execution of any Model 10 Card System program that updates the magnetic tape error counters.
- The Magnetic Tape Error Summary program prints a report of the error statistics using the cards punched by the Magnetic Tape Error Logout program as input.
- The Tape Dump program lists blocks of tape data on the printer in both hexadecimal and character representation until a tape mark is read. This program also spaces a tape block backward or forward and rewinds a tape to the load point.
- The Tape Initialization program creates IBM Standard or ANSI tape volume labels, checks for unexpired files, and displays existing volume and file labels. The Tape Initialization program is described in the following section. (This program is also available for the other System/3 models.

In addition to the Tape Initialization program (previously mentioned and described in the following section), the Tape Volume Error Summary program is available for tape on the Model 8, Model 10 Disk System, Model 12, and Model 15. The Tape Volume Error Summary program prints a report of statistics that have been accumulated during processing and stored on F1.

TAPE INITIALIZATION PROGRAM

The Tape Initialization program prepares tapes for use. It writes IBM standard volume labels on tape allowing tape data management to perform IBM standard label processing.

The Tape Initialization program can perform these functions at your request:

- CHECK labeled tapes for a volume label and an unexpired file before writing a new volume label.
- CLEAR labeled or unlabeled tapes by bypassing CHECK and unconditionally initializing the tape.
- DISPLAY the volume and header labels.

Note: DISPLAY causes tape runaway if it is performed on a tape that has never been initialized or if it is performed on a tape that has not been initialized since its last cleaning.

All tapes must be initialized before use. Tapes that have been initialized need not be reinitialized unless you want a new volume label written. This program can either initialize (CLEAR or CHECK) or DISPLAY one tape per unit during the same program run.

Control Statements

The control statements you supply for the Tape Initialization program depend on the type of initialization and the number of tapes initialized. Figure 7 shows the statements required by this utility program to initialize a reel of tape or display volume and header labels.

Control statements for the Model 10 Card System are read from the primary hopper of the MFCU. If the system finds an error in a control statement, you may correct the error and continue the job or the job can be canceled.

The system requires a set of OCL statements to run the disk resident Tape Initialization program. The two statements that must be supplied for this utility program are LOAD and RUN. These statements are part of the standard System/3 OCL statements as explained in the system control programming reference manuals listed in the *Preface.*

The LOAD statement identifies the program to be run. It is the first of the required statements. The program name for the Tape Initialization program is $TINIT. The unit parameter contains a code that indicates the location of the disk pack containing the Tape Initialization program. The codes are F1, R1, F2, and R2.

The program is loaded after the system reads the RUN statement. If the control statements are read from the system input device and an error is found, you may correct the error and continue the job, or you may cancel the job.

When the control statements are read from a procedure on disk and an error is found, the job must be canceled. The END statement is required for every program and follows the last of the VOL statements. The program begins initialization or display of the tape after the system reads the END statement.

VOL Statement

The Tape Initialization program requires one or more utility control statements (// VOL). These statements supply information needed to initialize or display a reel of tape. The format of the VOL statement is shown in Figure 7. A maximum of four of these statements may appear in the job stream.

UNIT parameter informs the system which tape unit contains the tape to be initialized or displayed. A separate VOL statement is required for each tape unit that contains a tape to be initialized or displayed.

REEL parameter specifies either initialization of an unlabeled tape (NL) or the volume serial number which the program writes on the tape (xxxxxx). This optional parameter is not needed if TYPE-DISPLAY is specified. NS cannot be specified.

$$\left[\begin{array}{l} \text{// LOAD \$TINIT,}\left\{\begin{matrix}\text{F1}\\\text{R1}\\\text{R2}\\\text{F2}\end{matrix}\right\} \\ \\ \text{// RUN}\end{array}\right]$$

Note: These statements are not used with the Model 10 Card System.

$$\text{// VOL UNIT-}\left\{\begin{matrix}\text{T1}\\\text{T2}\\\text{T3}\\\text{T4}\end{matrix}\right\}\left[\text{,REEL-}\left\{\begin{matrix}\text{NL}\\\text{xxxxxx}\end{matrix}\right\}\text{,TYPE-}\left\{\begin{matrix}\text{DISPLAY}\\\text{CLEAR}\\\underline{\text{CHECK}}\end{matrix}\right\}\text{,ASCII-}\left\{\begin{matrix}\text{YES}\\\underline{\text{NO}}\end{matrix}\right\}\text{,DENSITY-}\left\{\begin{matrix}\underline{\text{1600}}\\\text{800}\\\text{556}\\\text{200}\end{matrix}\right\}\text{,ID-YY . . . . YY}\right]$$

$$\left[\text{,FILES-}\left\{\begin{matrix}\underline{\text{001}}\\\text{ALL}\\\text{nnn}\end{matrix}\right\}\right]$$

Note: A separate VOL statement is required for each tape drive unit containing a tape to be initialized or displayed.

// END

Figure 7. Specifications for the Tape Initialization Program

TYPE parameter indicates if the program should check for a volume label and an unexpired file or print the volume and header labels. This is an optional parameter.

If CLEAR is specified, the checking procedures are not performed, and a new volume label is written on tape. This option should always be specified for new or unlabeled tapes, because the CHECK option would attempt to read a blank tape.

The CHECK option is assumed if the parameter is omitted. It checks for an unexpired file and a volume label before writing a volume label on the tape.

The DISPLAY parameter specifies that the volume and header labels from the specified unit be printed.

ASCII parameter specifies the initialization of tapes in either EBCDIC or ASCII code. The YES option tells the system to write ANSI standard labels in the ASCII format. The NO option tells the system to write IBM standard EBCDIC labels. The NO option is assumed if the parameter is omitted.

DENSITY parameter indicates whether the label to be written is 200, 556, 800, or 1600 bpi. If the parameter is omitted, the system assumes 1600 bpi for 9-track tape and 800 bpi for 7-track tape.

ID parameter provides an additional identification field in the volume label. A maximum of 10 characters can be used in this field if ASCII-NO is specified; 14 characters can be used if ASCII-YES is specified. This is an optional parameter.

FILES parameter indicates the number of files for which header labels are to be displayed. From 1 to 999 can be specified, or all header labels can be displayed (FILES-ALL). If the FILES parameter is not used, the system assumes that only the first file's labels are to be displayed. This parameter is supported for the Model 15 only.

Chapter 4. Magnetic Tape Program Product Support

The following program products and features for the System/3 support the 3410/3411 Magnetic Tape Subsystem:

	Model 10 Card System	Model 10 Disk System	Model 8	Model 12	Model 15
RPG II				X	X
RPG II Magnetic Tape Feature	X	X	X		
Subset ANS COBOL		X	X	X	X
Disk FORTRAN		X	X	X	X
Basic Assembler		X	X	X	X
Disk Sort		X	X	X	X
Magnetic Tape Sort	X	X	X	X	X
1255 Magnetic Character Reader Utility		X	X	X	

CARD RPG II FOR MODEL 10

The RPG II magnetic tape feature supports sequential processing of input and output data files for the IBM 3410/3411 Magnetic Tape Subsystem. (The Model 10 Card System supports only 9-track tape with 800 or 1600 bpi.) To use the RPG II magnetic tape feature, the programmer codes only the necessary entries on the RPG II specification sheets. Normal input and output specifications are supported. A magnetic tape unit is not required in the minimum system configuration for compiling a program using the RPG II magnetic tape feature. The following functions are supported:

- *EBCDIC and ASCII record formats.* ASCII files can be specified in the File Description Specification to include tape data management capable of handling either ASCII or EBCDIC code (if ASCII files are not specified, EBCDIC-only data management will be included). The ASCII parameter in the FILE card determines the mode in which data management functions.

- *Fixed length, blocked or unblocked records.* The block prefix must be included in the block length for ASCII tapes. Record or block size can be from 18 bytes to 4096 bytes.

- *Tape rewind.* The entries can be specified on the File Description Specification sheet or the control card. Entries on the control card override entries on the File Description Specification sheet.

- *Dual I/O areas.* An entry on the File Description Specification sheet provides two I/O areas that are used alternately to improve performance.

- *Object time tables or arrays.* These can be loaded or dumped from tape files. Tables or arrays are defined in the Extension Specifications and become part of the object program.

- *FORCE, READ, DEBUG, and EXCPT operation codes.* These codes alter the normal RPG II program cycle, allowing input and output operations during calculations. These operation codes are specified on the Calculations Specifications sheet.

- *Multivolume tape files.* A maximum of 31 volumes can be supported for each multivolume tape file.

- *Overlay tape OPEN.* This permits tape-open routines to be overlaid, which saves main storage after files have been opened.

- *Tape resident compiler.* The compiler can be placed on magnetic tape and loaded from tape into main storage for program compilation.

Note: Tape label information is specified on the control card.

DISK RPG II FOR MODELS 8, 10, 12, AND 15

Disk RPG II support for magnetic tape is available as a feature for the Model 8 and the Model 10 Disk System; this support is standard for Model 12 and Model 15. RPG II for these systems supports consecutive input and output files on 9-track tape with 800 or 1600 bpi, or 7-track tape with 200, 556, or 800 bpi. Magnetic tape storage can be used for data or record address files; however, record address files can only be used to process tape files within limits. To use RPG II, the programmer codes only the necessary entries on the RPG II specification sheets. Normal input and output specifications are supported.

A magnetic tape unit is not required in the minimum system configuration for compiling a program using RPG II. The File Description Specification sheet defines the control information and device entry for tape to the RPG II compiler. Columns 40-46 of this form identify the input/output device for the file. Tape data files may be referenced by specifying TAPE in those columns. For Model 15, device-independent data management can be used. If device-independent data management is used, the device name is left blank. At execution time, assign the file to tape.

The following functions are supported:

- *EBCDIC and ASCII record formats.* ASCII files can be specified in the File Description Specification to include tape data management capable of handling either ASCII or EBCDIC code (if ASCII files are not specified, EBCDIC only data management will be included). The ASCII parameter in the FILE statement determines the mode in which data management functions.

- *Fixed length, blocked or unblocked records.* The block prefix must be included in the block length for ASCII tapes. Record or block size can be from 18 bytes to 9999 bytes.

- Variable length, blocked or unblocked records.

- *Label-type control information.* This information is accepted but not checked by tape data management. Label-type information must be specified on the OCL FILE statement.

- *Dual I/O areas.* An entry on the File Description Specification sheet provides two I/O areas that are used alternately to improve performance.

- *Tape rewind.* A tape can be rewound to the load point positioning it at the beginning of the data file. This specification can be overridden by the appropriate OCL FILE statement specifications.

- *Object time tables or arrays.* These can be loaded or dumped from tape files. Tables or arrays are defined in the Extension Specifications and become part of the object program.

- *FORCE, READ, DEBUG, and EXCPT operation codes.* These codes alter the normal RPG II program cycle allowing input and output operations during calculations. These operation codes are specified in the calculations specifications.

- *Multivolume tape files.* A maximum of 99 volumes can be supported for each unlabeled or nonstandard labeled input file. The maximum number of volumes supported for labeled input is 40 (192 on Model 15D). If the REEL parameter is specified on the FILE statement for labeled output, a maximum of 40 volumes can be specified (255 volumes are supported); when the REEL parameter is not specified for labeled output a maximum of 255 volumes is supported. An unlimited number of unlabeled output volumes is supported. Multivolume tape files are not supported on the Model 8, Model 10 Disk System, or Model 12 if BSCA or devices attached to the SIOC are used in the program (not applicable to Model 15).

SUBSET ANS COBOL

Magnetic tape files can be created and read using the IBM System/3 Subset American National Standard (ANS) COBOL. To use the magnetic tape subsystem, the programmer uses normal COBOL statements in conjunction with OCL statements.

The compiler and library require a minimum main storage size of 12K bytes exclusive of system control program requirements. For the Model 10 Disk System, the minimum system required for COBOL compilation is 16K. However, on a 16K system with 5445 Disk Storage Drive and/or 3410/3411 Magnetic Tape Subsystem, the supervisor cannot include support for the dual program feature.

ANS COBOL supports:

- Sequential input and output files
- ASCII or EBCDIC record formats
- Fixed or variable length records
- Blocked or unblocked records
- Label type information specified on the OCL FILE statement

FORTRAN IV

Magnetic tape files can be created and read using the IBM System/3 FORTRAN IV. You can specify a device (unit) number and the block size for each tape file at compile time. At execution time, the FILE statements are used to allocate up to four tape units for the FORTRAN program.

The compiler and library require a minimum main storage size of 9K bytes exclusive of system control program requirements. The FORTRAN IV compiler can be executed on a minimum configuration system. For object programs supporting magnetic tape on the Model 10 Disk System, a processing unit with at least 16K bytes of main storage is required.

Logical records can span two or more single record blocks. The logical record size of unformatted records depends solely on machine capacity allowing tape work files to be used advantageously when handling large arrays.

System/3 FORTRAN IV supports:

- Formatted, list-directed, and unformatted tape input and output
- Fixed length, unblocked records with record lengths of from 18 to 32,767 characters (bytes) read or written under control of a FORMAT statement
- Variable length, unformatted records with a block size of from 18 to 32,767 bytes
- READ (with optional end-file, END, and error processing, ERR, parameters), WRITE, END FILE, BACKSPACE, and REWIND statements

MAGNETIC CHARACTER READER UTILITY PROGRAM

The 1255 Magnetic Character Reader Utility supports the tape unit as an output device for creating single volume files. This utility program can be executed on a minimum configuration system with the exception of a Model 10 Disk System for which a minimum of 16K bytes of main storage is required. No change in coding is required in the 1255 utility program specifications to support tape output rather than disk output.

TAPE SORT PROGRAM

The Magnetic Tape Sort programs use the tape drives as work files to sort a tape file into either ascending or descending sequence. RPG II-type statements are used to specify record selection, sort parameters, and data fields. The sort specifications statements are similar in function and arrangement to the tag-along sort function of the disk sort program. The summary sort function is not supported.

System Requirements

The magnetic tape sort programs operate with the minimum System/3 configurations.

Tape Sort Files

The Tape Sort program requires three types of files: input, output, and work.

Input and Output Files

Tape input and output files can be recorded in either EBCDIC or ASCII codes. Records in these files must be fixed length but can be either blocked or unblocked. EBCDIC files are sorted directly; no translation is necessary.

ASCII files can be labeled or unlabeled. ANSI labels are used for labeled ASCII files. Labeled files must be translated to EBCDIC before processing. After all processing is finished, the file can be translated back to ASCII and written on the output file.

Unlabeled ASCII files can be sorted directly, using the ASCII collating sequence. Only character comparisons are valid; specified sort constants are treated as EBCDIC data. The sorted records are written in ASCII code on the output file. You must not specify translation of output files for untranslated ASCII files.

Work Files

The tape sort program requires at least three work files; a fourth work file is optional. Work files can share tape units with the input and output files.

Note: When using SEQNUM-x as a work FILE statement parameter for multifile volumes, you must be sure that the tape is positioned past the last file on the volume that you want to preserve (Model 15 only).

Sort Functions

The Tape Sort program creates a sorted file of all records selected. Only those fields you request from the input records are included in the sorted records. Output records are sequenced by the designated control fields from the corresponding input records. Fields identified as data fields are included in the output records, but have no effect on the sequence of the records. After records are sorted, the sort control fields can be dropped from the output records.

Record Selection

Records to be sorted are selected from the input file by the tape sort program according to your specifications. You can specify that records be selected on the basis of a comparison of two fields within the record, or that selection be based on a comparison of a field within the record and a constant. You can also specify multiple tests such as combinations of field-to-field and field-to-constant comparisons. These comparisons direct the sort program to accept all or only particular records from the input file.

Field Selection

You must specify the fields on the input record that are control fields and/or data fields. Figure 8 shows an example of field selection. Fields specified as control fields are used by the sort program to determine the order of the records in the output file. Multiple control fields up to a combined length of 256 bytes can be specified. Ascending or descending sequence can be specified for each field. You must specify which data fields from the input records are to be included in the output records; all other fields are dropped when the output record is created.

In the output record, all control fields from the input record are placed ahead of all data fields (if the key is not dropped), but you can choose the order of the control and data fields in the output record. Fields specified as control fields can also be duplicated as data fields within the data portion of the output record.

Multiple record types with different formats can be sorted in each sort run. For example, you could sort debit and credit records at the same time.

Additional Features

The tape sort program provides a checkpoint option that allows you to restart processing from a point other than the beginning if a system failure occurs. When a checkpoint is taken, the program records enough information for a restart to be made from that checkpoint. SEQNUM-x may not be used as a work FILE statement parameter when checkpoint restart has been specified on the header statement.

The tape sort program also provides diagnostic messages. These messages describe errors and unusual conditions in the sort control statements.

Audit controls in the tape sort program keep record counts. The number of input records, records selected for sorting, and output records is recorded by the audit control feature to ensure that no records are lost in the sort process. Sequence checking is also performed on the sort output file to ensure that the output records are in the proper order.

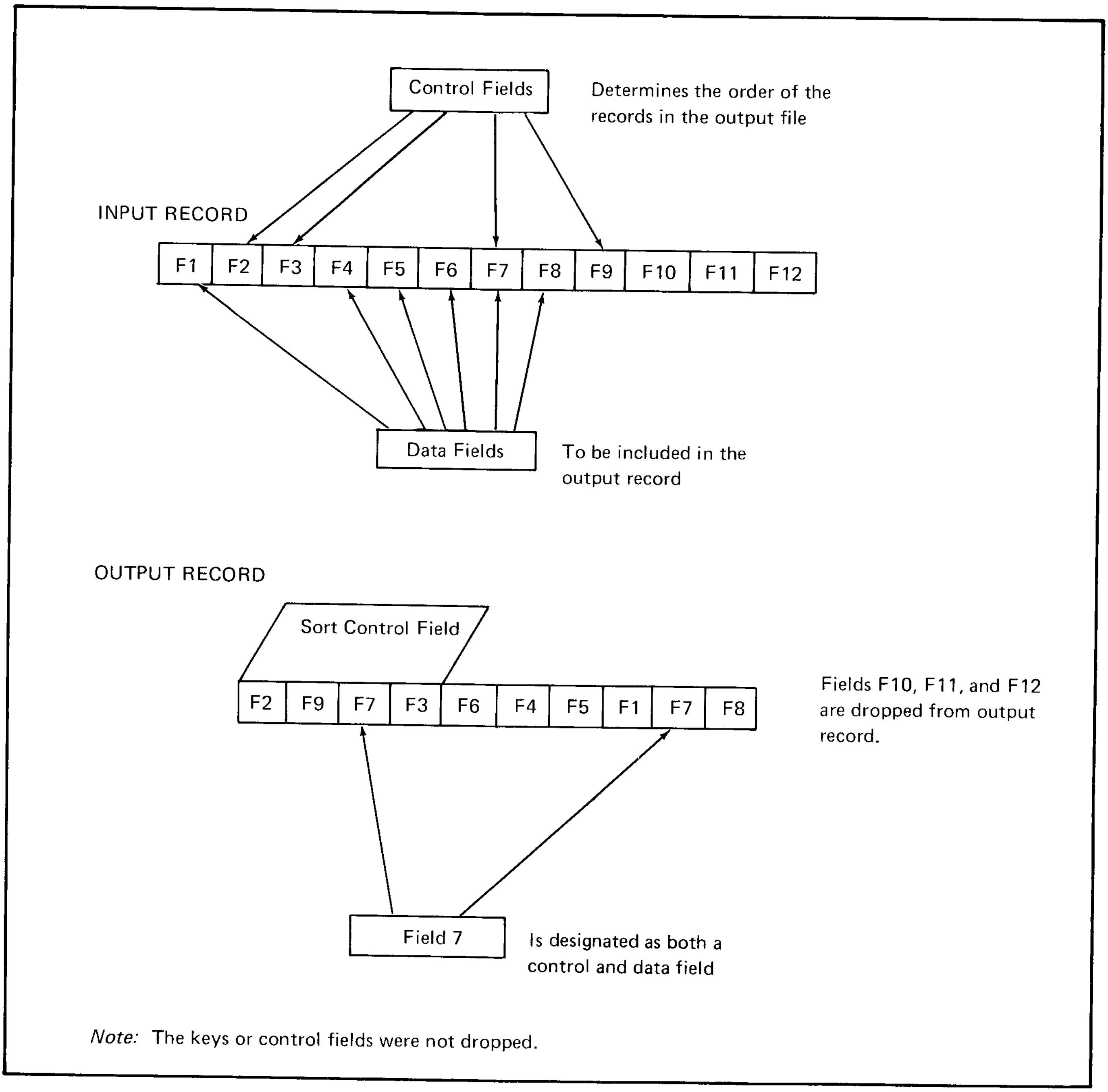

Control Fields
Determines the order of the records in the output file
INPUT RECORD
F1
F2
F3
F4
F5
F6
F7
F8
F9
F10
F11
F12
Data Fields
To be included in the output record
OUTPUT RECORD
Sort Control Field
F2
F9
F7
F3
F6
F4
F5
F1
F7
F8
Fields F10, F11, and F12 are dropped from output record.
Field 7
Is designated as both a control and data field
Note: The keys or control fields were not dropped.

Figure 8. Comparison of Input and Output Sort Records

BASIC ASSEMBLER

By using the Basic Assembler, you can write assembler language programs that support magnetic tape. Magnetic tape is not used for program assembly.

System macro instructions are available with the SCP to simplify coding required by the assembler. Macros perform the following functions for tape:

- Build a preopen DTF for tape and assign its offsets.
- Build the interfaces required to read input records from a tape device via a GET or READ.
- Build the interfaces required to write output records to a tape device via a PUT or WRITE.
- Build the interface required to issue tape control commands.
- Wait for completion of read, write, or tape control operations.

See the macros reference manuals listed in the *Preface* for more information concerning macro instructions.

DISK SORT PROGRAM

The Disk Sort programs are used to sort a tape file into either ascending or descending sequence using disk drives for work files. The output can be on either a tape file or a disk file. Disk sort functions are similar to those for the Tape Sort program except that the summary sort function is available in disk sort, and disk sort does not provide a checkpoint option.

Special Considerations for Variable Length Records

When running a tape job with variable length records, there are conventions that must be followed concerning the BLKL (block length) and the RECL (record length) parameters in the OCL FILE statement.

There are two rules:

- The specified record length (in the RECL parameter) must be at least 4 bytes larger than the largest record.
- The specified block length (in the BLKL parameter) must be at least 4 bytes larger than the specified record length.

A halt is issued if either of the above is violated.

For maximum efficiency, however, the following algorithm is recommended:

$$BLKL = N (R + 4) + 4$$
$$RECL = R + 4$$

where R is the size of the largest record and N is the minimum number of records desired in a block.

For example, to create a variable record length tape with the largest record 60 bytes and at least two records per block, specify BLKL-132 and RECL-64.

Note: Every block would contain at least two records with the possible exception of the last block.

There is an additional consideration when running a variable record length tape job with RPG II. First, determine BLKL and RECL for the FILE statement using the rules stated above. The block length in the RPG II file specifications (columns 20-23) must match BLKL, but the record length in RPG II file specification (columns 24-27) must be RECL-4. Failure to match either of these causes a halt.

For example, if the FILE statement has BLKL-132 and RECL-64, then 132 must appear as the block length and 60 as the record length in the RPG II File Description Specification.

When tape containing variable length records is read, the block length and record length specified on the FILE statement need not match the ones on the FILE statement when the tape was created as long as:

- They obey the rules above.
- BLKL is at least as large as the largest block and RECL at least as large as the largest record on the tape.

MAGNETIC TAPE SUPPORT SUMMARY

	SCP	Disk RPG II	COBOL	FORTRAN	Disk Sort	Tape Sort	ASM/ macros	1255	Card RPG II	Card Sort
Minimum number of tape drives supported	1	1	1	1	1	3	1	1	1	3
Maximum number of tape drives supported	4	4	4	4	4	4	4	4	4	4
Fixed unblocked	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Fixed blocked	yes	yes	yes	no	yes	yes	yes	yes	yes	yes
Variable unblocked	yes	yes	yes	no	no	no	yes	no	no	no
Variable spanned unblocked	no	no	no	yes	no	no	no	no	no	no
Variable blocked	yes	yes	yes	no	no	no	yes	no	no	no
Maximum record size supported	32767	9999	32767	32767	9999 (input) 4096 (output)	9999	32767	55	4096	9999
Maximum block size supported	32767	9999	32767	32767	32767	9999	32767	32767	4096	9999
Minimum record size supported	18	18	18	18	18	18	18	55	18	18
Minimum block size supported	18	18	18	18	18	18	18	55	18	18
ANSI labels	yes	yes	yes	no	yes	yes	yes	yes	yes	yes
IBM standard labels	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Unlabeled tapes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Nonstandard labels (input only)[1]	yes	yes	yes	yes	yes	yes	yes	NA	yes	yes
Bypass label processing	yes	NA	NA	NA	NA	NA	NA	NA	NA	NA
Read backwards	yes	no	no	no	no	yes	yes	no	no	yes
Write backwards	no	no	no	no	no	no	no	no	no	no
Multiple buffers	NL	2	2	NA	NA	NA	NL	2	2	NA
9 track 800/1600 bpi	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
7 track 200/556/800 bpi	yes	yes	yes	yes	yes	yes	yes	no	no	no
Multivolume files (unlabeled or labeled; input or output)	yes	yes	yes	no	yes	yes	yes	no	yes	yes
Multifile volumes (Model 15 only)	yes	yes	yes	yes	yes	yes	yes	yes	no	no
Maximum number of volumes[2] (input/output)		SCP	SCP	1/1	SCP	SCP	SCP	0/1	31/31	31/31
Labeled[3]	40/255									
Unlabeled	99/NL									
ASCII data (9 track only)	yes	yes	yes	no	yes	yes	yes	yes	yes	yes
EBCDIC data (7 or 9 track)	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
Program options:										
Rewind	yes	yes	OCL	yes	OCL	OCL	yes	OCL	yes	OCL
Rewind unload	yes	yes	OCL	OCL	OCL	OCL	yes	OCL	yes	OCL
Forward space file/block	yes	no	no	no	no	no	yes	no	no	no
Backward space file/block	yes	no	no	yes	no	no	yes	no	no	no
Write tape mark (end-file)	yes	no	no	yes	no	no	yes	no	no	no

[1] Nonstandard labels are supported only in that they are bypassed on input.

[2] Multivolume tape files are not supported on the Model 8, Model 10 Disk System, or Model 12 if the program also uses BSCA or devices attached to the SIOC.

[3] The maximum number of format 1 labels that are allowed (not applicable to Model 10 Card System) is 40 on Models 8, 10, 12, 15A, B, C and 192 on Model 15D. This number must accommodate all labels, including disk. Two hundred fifty-five is the limit for the output if there is no specification of the particular output volumes; otherwise, the limit is 40 volumes specified in the REEL parameter. If more standard labeled volumes are required than specified for the output file, a halt occurs for each additional volume as it is needed. By mounting another standard labeled tape and by selecting the continue option to the halt, the system continues writing the file. A maximum of 255 volumes are supported for output (including those specified in the REEL parameter).

Note: NA = not applicable, NL = no limit, OCL = program options through OCL only, SCP = limited by SCP.

The following chart describes the column headings and the applicable program numbers:

Column Heading	Description	Model 10 Card System	Model 10 Disk System and Model 8	Model 12	Model 15	Model 15D
SCP	System control programming with magnetic tape feature	NA NA	5702-SC1 6024/6025	5705-SC1 NA	5704-SC1 NA	5704-SC2 NA
Disk RPG II	Disk RPG II with magnetic tape feature	NA NA	5702-RG1 6016/6018	5705-RG1 NA	5704-RG1 NA	5704-RG2 NA
COBOL	Subset ANS COBOL compiler	NA	5702-CB1	5705-CB1	5705-CB1	5704-CB2
FORTRAN	FORTRAN IV	NA	5702-FO1	5705-FO1	5704-FO1	5704-FO2
Disk Sort	Disk Sort program	NA	5702-SM1	5705-SM1	5704-SM1	5704-SM9
Tape Sort	Disk Resident Magnetic Tape Sort program	NA	5702-SM2	5705-SM2	5704-SM2	5704-SM8
Asm/Macros	Basic Assember with SCP macros feature	NA NA	5702-AS1 6020/6021	5705-AS1 NA	5704-AS1 NA	5704-AS2 NA
1255	Utility program for 1255 Magnetic Character Reader	NA	5702-UT2	5705-UT2	NA	NA
Card RPG II	Card RPG II with magnetic tape feature	5701-RG1 6004	NA NA	NA NA	NA NA	NA NA
Card Sort	Card Resident Magnetic Tape Sort program	5701-SM1	NA	NA	NA	NA

Note: NA = not applicable.

Chapter 5. 3410/3411 Magnetic Tape Subsystem

The following chart shows the three models of the tape subsystem with the characteristics of each that are available:

	Model 1	Model 2	Model 3
Tape speed (in/s)	12.5	25	50
Interblock gap (IBG)[1]			
Length/time (9-track)	0.6 in/48 ms	0.6 in/24 ms	0.6 in/12 ms
Length/time (7-track)	0.75 in/60 ms	0.75 in/30 ms	0.75 in/15 ms
Write access time[2]	15 ms	12 ms	6 ms
Read access time[2]	15 ms	12 ms	6 ms
Data rate 1600 bpi 800 bpi 556 bpi 200 bpi	 20K bytes/s 10K bytes/s 6.95K bytes/s 2.5K bytes/s	 40K bytes/s 20K bytes/s 13.9K bytes/s 5.0 bytes/s	 80K bytes/s 49K bytes/s 27.8K bytes/s 10.0K bytes/s
Time per byte 1600 bpi 800 bpi 556 bpi 200 bpi	 50 μs 100 μs 144 μs 400 μs	 25 μs 50 μs 72 μs 200 μs	 12.5 μs 25 μs 36 μs 100 μs
Rewind time (±10%)	3 min (2400 ft)	3 min (2400 ft)	2 min (2400 ft)
Reel sizes (in)	10.5, 8.5, 7, 6	10.5, 8.5, 7, 6	10.5, 8.5, 7, 6
Tape threading	Manual	Manual	Manual

(chart continued on next column)

Tape motion	Tape is driven by a single capstan which is directly coupled to a low-inertia, high-torque, dc motor.
Read/write head	The chrome-plated, two-gap head is located in the left vacuum column.

[1] An interblock gap is erased tape which separates blocks of data.

[2] Time given is for 0.6 in interblock gap.

Metric Equivalents:

1600 bpi	=	63 bytes/mm	0.6 in	=	15.2 mm
800 bpi	=	31.5 bytes/mm	0.75 in	=	19 mm
556 bpi	=	21.9 bytes/mm	6 in	=	152.4 mm
200 bpi	=	7.9 bytes/mm	7 in	=	177.8 mm
50 in/s	=	1270 mm/s	8.5 in	=	216 mm
25 in/s	=	635 mm/s	10.5 in	=	266.7 mm
12.5 in/s	=	317.5 mm/s	2400 ft	=	732 m

SPECIAL FEATURES

Each 3410 and 3411 tape unit must be equipped with a special feature that specifies the read/write format desired:

- Single density
- Dual density
- Seven-track

Any tape unit in the subsystem (a 3411 or an attached 3410) can be equipped with either the dual density feature or the 7-track feature, but not both. A subsystem can have the following combination of features:

- Each tape unit has the single density feature.
- Each tape unit has the dual density feature.
- Each tape unit has the 7-track feature.
- Some tape units have the single density feature; some have the dual density feature.
- Some tape units have the single density feature; some have the seven track feature.

Single Density Tape Unit Feature

This feature is installed on tape units to enable 9-track, PE (phase-encoded) operations. Single density control is standard on the 3411.

Dual Density Tape Unit Feature

This feature is installed on tape units to enable 9-track operations in both 1600 bpi PE mode and 800 bpi NRZI (non-return-to-zero) mode. If any tape unit is equipped with the dual density feature, the 3411 must also be equipped with the dual density control feature.

Seven-Track Tape Unit Feature

This feature can be installed on the tape unit portion of the 3411 or on any 3410. It enables the tape unit to read and write data in NRZI mode on 7-track magnetic tape. Reading and writing are done at densities of 200, 556, or 800 bpi. Odd or even parity is provided.

If the 7-track tape unit feature is installed in the 3411 or any attached 3410, the 3411 must aso be equipped with a 7-track tape control feature. A 3410 or 3411 tape unit equipped with the 7-track tape unit feature cannot be equipped with either a single density tape unit feature or a dual density tape unit feature.

Dual Density Control Feature

This feature, available for the 3411 control unit, enables the tape units to read and write 9-track tape in either the 800 bpi NRZI or 1600 bpi PE mode.

Seven-Track Control Feature

This feature, available for the 3411 control unit, enables the 3411 to control any tape unit (including the tape unit portion of the 3411) that is equipped with the 7-track tape unit feature.

A translator and data converter are included with the 7-track control feature. The translator, when set on, translates 8-bit bytes from main storage to 6-bit BCD tape characters and vice versa. Each main storage byte becomes a tape character; each tape character becomes one byte in main storage. The data rate is not changed by the translator.

The data converter allows writing and reading of binary data on 7-track tape units. Writing a tape when the data converter is on causes four tape characters (24 bits) to be written for every three storage bytes (24 bits). Reading such a tape reverses the process by converting four tape characters into three storage bytes. Data conversion reduces the data transfer rate by 25 percent of that for 9-track NRZI operations. An odd/even count is made during read/write data converter operations to ensure correct transfer of data.

Appendix A. Tape Label Formats

This appendix describes the tape volume label, header label 1, trailer label 1, header label 2, and trailer label 2.

VOLUME LABEL FORMAT

The volume label is written in EBCDIC format with a block length and record length equal to 80 bytes. Figure 9 shows the format of the volume label.

Displ	Lng in bytes	Name	Description
00	3	Label identifier	This is the first record on tape that identifies the reel. This position always contains VOL.
03	1	Volume label number	This indicates the sequence of a volume label; this position contains a 1 for System/3 tapes.
04	6	Volume identifier	This position uniquely identifies the volume. This field contains from one to six alphabetic and/or numeric characters. If the volume identifier is less than six characters, it is left-justified and the remainder of the field is padded with blanks. In the case of the first or only volume of a file, the contents of this field are also used as the file identifier in the header and trailer labels.
A	1	Volume security	This character is for installation use as a security indicator for the entire volume. It is installation defined, and may be any alphabetic or numeric character or a space. The System/3 tape initialization program initially sets this position to zero.
B	30	Reserved	This field is reserved and initially set to blanks (X'40').
29	10	Owner identification	This field is not used by System/3 label processing. However, it indicates the name and address of the installation or user to whom the volume belongs. The 10 most significant (leftmost) user supplied characters are used to set this field. If the user supplied name and address code is less than 10 characters in length, it is left-justified and the remainder of the field is padded with blanks.
33	29	Reserved	This field is reserved and initially set to blanks (X'40').

Figure 9 (Part 1 of 2). Tape Volume Label

Figure 9 (Part 2 of 2). Tape Volume Label

HEADER LABEL 1 AND TRAILER LABEL 1 FORMAT

The first header and trailer labels identify the data contained on tape. These labels are written for the exclusive use of the system. They are written in EBCDIC format with a block length and record length equal to 80 bytes. Figure 10 shows the format of the first IBM standard header and trailer labels.

Displ	Lng in bytes	Name	Description
00	3	Label identifier	This field identifies the type of label. HDR – Header label EOF – End-of-file trailer label EOV – End-of-volume trailer label (for all but the final volume of a multi-volume file)
03	1	Label number	Indicates the sequence of the label. This position contains a 1 for System/3 tapes.
04	17	File identifier	These characters are assigned by the user to identify the file. System/3 is concerned only with the leftmost eight characters. These characters are left-justified by System/3 and the remainder of the field padded with blanks (X'40').
15	6	Aggregate identifier	These characters identify the aggregate on which the file is included. The characters are identical to the volume identifier field VOL (04). In the case of an aggregate on two or more volumes, this field contains the identifier of the first volume.

Figure 10 (Part 1 of 3). IBM Standard for Tape Header Label 1 and Trailer Label 1

Displ	Lng in bytes	Name	Description
1B	4	Volume sequence number	This is a number field (0001-9999) indicating the order of a volume within a multivolume file. The volume sequence number is the same for every header and trailer label on the volume. It is incremented by one for each additional volume.
1F	4	File sequence number	This is a numeric field (0001-9999) indicating the order of a file within a multifile volume. The sequence number is the same for every label associated with portions of the same file.
23	4	Generation number	This is a numberic field (0001-9999) indicating a single stage in the succession in which one file replaces another. This field is not verified on input by System/3.
27	2	Version number of generation	This is a numeric field (00-99) indicating an iteration of a generation. This field is not verified on input by System/3.
29	6	Creation date	This field indicates the year and day of the year the data set was created. It is of the form b̸YYDDD; a space, followed by the year (YY=00-99), followed by the day of the year, (DDD-001-366). e.g., February 1, 1970 would be recorded as "b̸70032".

Figure 10 (Part 2 of 3). IBM Standard for Tape Header Label 1 and Trailer Label 1

Displ	Lng in bytes	Name	Description
2F	6	Expiration	This field indicates the year and day of the year when the file may be purged. The format of this field is identical to the creation date field.
35	1	Security	This field indicates whether or not additional qualifications must be supplied in order to access the file. This field is set to zero on System/3 tapes and is not verified by System/3.
36	6	Block count	This is a numeric field (000000-999999) indicating the number of blocks between the last label of the header label group and the first trailer label. The count is recorded in the trailer label only (EOF1 or EOV1). These characters are recorded as zeros in the header label.
3C	13	System code	This field contains 'IBMƀSYSTEM/3ƀ'.
49	7	Reserved	This field is not used by System/3.

Figure 10 (Part 3 of 3). IBM Standard for Tape Header Label 1 and Trailer Label 1

HEADER LABEL 2 AND TRAILER LABEL 2 FORMAT

The second header and trailer labels provide additional information about the file. These labels are checked by System/3 if they are present for input files and are created for output files. However, System/3 will accept a file without the second header or trailer label. The labels are written in EBCDIC format with a block length and record length equal to 80 bytes. Figure 11 shows the format of the second IBM standard header and trailer labels.

Displ	Lng in bytes	Name	Description
00	3	Label identifier	This field identifies the type of label: HDR – Header label. EOF – End-of-file trailer label. EOV – End-of-volume trailer label (for all but the final volume of a multivolume file).
03	1	Label number	Indicates the sequence of the label. This position contains a 2 for System/3 tapes.
04	1	Record/ block format	Indicates the record/block format of the file: V – Variable length records in variable length blocks. F – Fixed-length records in fixed length blocks. D – Variable length ASCII records.
05	5	Block length	This field indicates the following length attributes: V – Maximum block length (00009-32767). F – Constant block length (00018-32767). *Note:* The minimum physical record length for magnetic tapes is 18; therefore, if the block length is to equal the physical record length, 18 should be used as the minimum in place of those given above.

Figure 11 (Part 1 of 2). IBM Standard for Tape Header Label 2 and Trailer Label 2

Displ	Lng in bytes	Name	Description
A	5	Record length	This field indicates the following length attributes: V – Maximum record length (00005-32767). F – Constant record length (00018-32767).
F	19	Reserved	This field is not used by System/3.
22	2	Tape recording technique	This field indicates the tape recording technique used to create the file: Cƀ– Data conversion required. Eƀ– Even parity. Tƀ– BCD to EBCDIC translation required. ET– Even parity and BCD to EBCDIC translation required. ƀƀ – Odd parity–no translation required (9-track).
24	2	Reserved	This field is not used by System/3.
26	1	Record attributes	This field indicates one of the following record attributes: B – Records are blocked and not spanned. S – Records are spanned and not blocked. R – Records are blocked and spanned. ƀ – Records are not blocked and not spanned.
27	41	Reserved	This field is not used by System/3.

Figure 11 (Part 2 of 2). IBM Standard for Tape Header Label 2 and Trailer Label 2

ANSI. American National Standards Institute.

ASCII. American National Standard Code for Information Interchange.

bpi. Bits per inch per row (track) or the bytes per inch on tape. This term is used when referring to the recording density of the tape unit.

block. A collection of contiguous records recorded as a unit. Blocks are separated by interrecord gaps, and each block may contain one or more records.

control program. A program for allocation and management of system resources.

control statement. A statement that contains information to tailor a program to the requirements of the user.

EBCDIC. Extended binary coded decimal interchange code.

field. A contiguous set of digits or characters that form a meaningful entity.

fixed length record. A record having the same length as other records with which it is logically or physically associated.

format. The general make-up of data or of a control card, record, or a file.

initialize. To set counters, switches, and addresses to zero or other starting values at the beginning of, or at prescribed points in, a computer routine.

input. Data to be processed by a computer program.

interblock gap. A blank space on magnetic tape that separates physical records.

loadpoint. The beginning of the usable portion of a reel of tape, indicated by a load point marker, where reading or writing is to begin.

multifile tape volume. More than one file is stored on a reel of tape.

multivolume tape file. A file stored on more than one tape reel.

NRZI (non-return-to-zero IBM). A method of recording on tape where only the one bits are written as magnetized spots on tape.

operand. The representation of a value that must be supplied to define a selective function to the program.

output. The resultant data produced by a computer program.

parameter. A value that must be supplied to define a selective function to the program.

PE (phase encoding). A method of recording on tape where both zero and one bits are written as magnetized spots. The zero and one bits are opposite in polarity. This method allows distinction between zero bits and no recording.

record. A collection of related data fields.

statement. A meaningful expression or generalized instruction in a source language.

tape labels. Special records at the beginning and ending of tape files. There are volume, header, and trailer labels. They are used to identify the reel of tape and the recorded data file. They also contain certain housekeeping information.

tape mark. A special symbol that can be read from or written on magnetic tape. It is used to indicate the end of a file or file segment, and to segregate the labels from data. Tape marks must be read in the same density in which they were written.

variable length records. Logical records in a file in which the number of bytes in each record is not a fixed value, but may vary within prescribed limits.

Index

READER'S COMMENT FORM

IBM System/3
Magnetic Tape
Program Planning Manual

GC21-5040-3

YOUR COMMENTS, PLEASE . . .

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

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

Corrections or clarifications needed:

Page *Comment*

Please include your name and address in the space below if you wish a reply.

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

GC21-5040-3

Cut Along Line

Fold

Fold

FIRST CLASS
PERMIT NO. 387
ROCHESTER, MINN.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

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

Fold

Fold

IBM

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

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

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 System/3 Magnetic Tape Planning (File No. S3-05) Printed in U.S.A. GC21-5040-3

Technical Newsletter

This Newsletter No.	GN21-5604
Date	30 June 1978
Base Publication No.	GC21-5040-3
File No.	S3-34
Previous Newsletters	None

IBM System/3
Magnetic Tape
Program Planning Manual

This technical newsletter applies to the current versions and modifications of the applicable System/3 programs listed in the edition notice and provides replacement pages for the subject publication. These replacement pages remain in effect for subsequent versions and modifications unless specifically altered. Pages to be inserted and/or removed are:

Cover, Edition Notice
iii, iv
1 through 4
7 through 14
17,18
21 through 24
Back Cover

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

Summary of Amendments

Miscellaneous technical changes

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

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