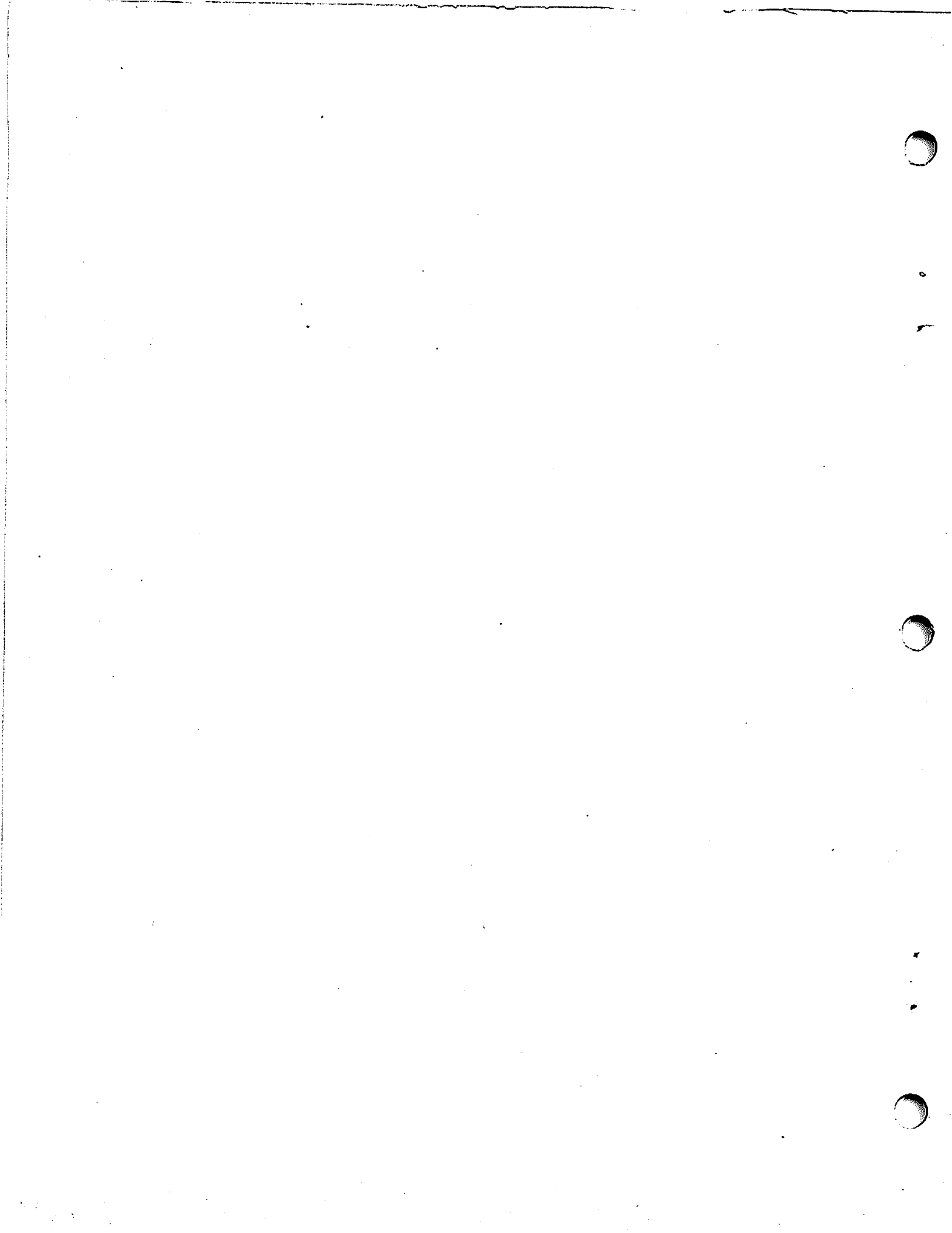


WANG

MODEL 2209A  
NINE-TRACK TAPE  
UTILITIES MANUAL

# SYSTEM 2200





# MODEL 2209A NINE-TRACK TAPE UTILITIES MANUAL

© Wang Laboratories, Inc., 1977



LABORATORIES, INC.

ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851. TEL. (617) 851-4111. TWX 710 343-6789. TELEX 94-7421

## **Disclaimer of Warranties and Limitation of Liabilities**

The staff of Wang Laboratories, Inc., has taken due care in preparing this manual; however, nothing contained herein modifies or alters in any way the standard terms and conditions of the Wang purchase, lease, or license agreement by which this software package was acquired, nor increases in any way Wang's liability to the customer. In no event shall Wang Laboratories, Inc., or its subsidiaries be liable for incidental or consequential damages in connection with or arising from the use of the software package, the accompanying manual, or any related materials.

### **NOTICE:**

All Wang Program Products are licensed to customers in accordance with the terms and conditions of the Wang Laboratories, Inc. Standard Program Products License; no ownership of Wang Software is transferred and any use beyond the terms of the aforesaid License, without the written authorization of Wang Laboratories, Inc., is prohibited.



**LABORATORIES, INC.**

ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851, TEL. (617) 851-4111, TWX 710 343-6769, TELEX 94-7421

## HOW TO USE THIS MANUAL

This manual contains information on the Model 2209A Nine-Track Utilities and instructions for their operation by the user. It assumes familiarity with the operation of the available Wang System 2200 and some programming experience in BASIC.

Users who are unfamiliar with their system should read the Wang BASIC Language Reference Manual and the Programming in BASIC Manual. For operation procedures and operator maintenance of the Model 2209A Nine-Track Tape Drive, see the Model 2209A Reference Manual. For more information on disk units and card readers see their respective manuals.

The utilities are provided on tape cassette (#701-0417), flexible disk (#701-5203) and diskette (#701-2203).

CHAPTER 3 CONTENTS

	Page
CHAPTER 1: INTRODUCTION . . . . .	1
CHAPTER 2: TAPE LABELS. . . . .	4
2.1 Volume Labels. . . . .	6
2.2 Header/Trailer Label 1 . . . . .	7
2.3 Header/Trailer Label 2 . . . . .	9
CHAPTER 3: PRIMARY ROUTINES . . . . .	11
3.1 The LOADER . . . . .	11
3.2 INITIALIZE/REWRITE VOLUME LABEL. . . . .	14
3.3 WRITE DATA ON TAPE . . . . .	20
3.4 READ DATA FROM TAPE. . . . .	27
3.5 TAPE TO DISK DATA TRANSFER . . . . .	30
3.6 DISK TO TAPE DATA TRANSFER . . . . .	35
3.7 CARD READER TO TAPE DATA TRANSFER. . . . .	38
3.8 TAPE DUMP. . . . .	42
CHAPTER 4: SUBROUTINES. . . . .	45
4.1 Logical I/O Control System . . . . .	45
4.2 Physical I/O Control System. . . . .	48
Appendix A Sample User Program for Read or Write. . . . .	50
Index. . . . .	51

## CHAPTER 1: INTRODUCTION

The Nine-Track Tape Utilities consist of a number of routines created for processing nine-track tapes to be read or written on a Model 2209A Nine-Track Tape Drive at 1600 bpi (bytes per inch). To run these routines, a System 2200 series CPU which has at least 12K of user RAM and supports the General I/O Statements must be available. To transfer data from card reader or disk to tape, or from tape to disk, the appropriate peripheral (card reader or diskette/disk) must be available; to dump data from tape, a printer must be available. The routines in this package can process single volume, single data file tapes. Tapes may have IBM/ANSI standard tape labels, or have no labels. Non-standard labels cannot be processed with the utility set; however, processing is available through the user programmable, General I/O Set. Physical records are limited to 4K (4096 bytes), the capacity of the I/O buffer.

The reader should note that the card reader to tape and disk to tape transfer utilities require special file formats. It may be necessary for the user to create restructured intermediate files in order to use the transfer utilities. To read from or write on nine track tape, the user must supply an object program which employs the utility subroutines.

The utility routines operate at four levels:

1. A LOADER to load all nine-track routines and subroutines into memory, and to provide a user 'menu' to simplify program selection.
2. A Physical I/O Control System (PIOCS) containing the subroutines which perform physical operations on tape such as rewinding, backspacing, writing a file mark (EOF), etc.
3. A Logical I/O Control System (LIOCS) containing the subroutines which perform logical I/O operations including opening and closing of files, translations from ASCII to EBCDIC (and vice versa), and creating or updating internal tape labels.

4. A set of primary routines to initialize or update volume labels, to read or write data and to transfer data from tape to disk, disk to tape and card to tape. To read or write tapes, the user must write a dimension (DIM) statement to specify block size, and must insert subroutine calls (GOSUB's) to the appropriate Logical and/or Physical IOCS subroutines.

The primary routines are:

"NAME"	Name	Operation
START	LOADER	Loads subroutines and menu.
TPUTO20A	INITIALIZE/REWRITE VOLUME LABEL	Initializes or rewrites tape volume labels.
TPUTO30A	WRITE DATA ON or READ DATA FROM TAPE	Writes on or reads from nine-track tape; the user must supply object program lines.
TPUTO40A	TAPE TO DISK DATA TRANSFER	Transfers data from tape to disk, with or without translation.
TPUTO50A	DISK TO TAPE DATA TRANSFER	Transfers data from disk to tape, with or without translation. Requires special disk file format.
TPUTO60A	CARD READER TO TAPE DATA TRANSFER	Transfers data from punched cards to tape. Requires special card file format.
TPUTO70A	DUMP THE TAPE CONTENTS	Reads variable length blocks and outputs block size, untranslated hex image, and translated alphanumeric image of the data on the available output device.
TPUTO80A	CONVERT 800 BPI TAPE TO 1600 BPI TAPE	This stand alone program to convert 800 BPI tapes to 1600 BPI tapes is not included in the user menu; it is loaded and run with LOAD DCF "TPUTO80A" (EXEC) RUN (EXEC). The utility addresses are 07B for the 800 BPI tape, and 17B for the 1600 BPI tape. Using this module requires the user system have both 800 and 1600 BPI drives drives on line since it is a tape to tape operation.



The Logical IOCS subroutines are:

Subroutine	Operation
DEFFN'42	Performs an 'OPEN WRITE'.
DEFFN'43	Performs an 'OPEN READ'.
DEFFN'44	Performs a 'CLOSE WRITE'.
DEFFN'45	Performs a 'CLOSE READ'.
DEFFN'204(y)	Performs translation (ASCII to EBCDIC or EBCDIC to ASCII), during read if y=0, during write if y=1.

The Physical IOCS subroutine is:

DEFFN'31(x), where x specifies the tape operation to be performed, as follows:

x	Operation
1	backspace and position to write
2	backspace a record
3	forward space a record
4	backspace a file
5	forward space a file
6	rewind the tape
7	"write" a gap
8	clean the tape
9	reread a block
10	write a Tape Mark (File Mark or EOF)
11	write a block
12	read a block
13	status request
14	master reset
21	buffer write
22	look-ahead-read
31	finish write
32	finish read

## CHAPTER 2: TAPE LABELS

Label processing provided in the Nine-Track Utilities generally conforms to the standard ANSI and IBM label usage. The following table compares the contents of Wang, IBM and ANSI tape labels.

Table-1. Tape Labels

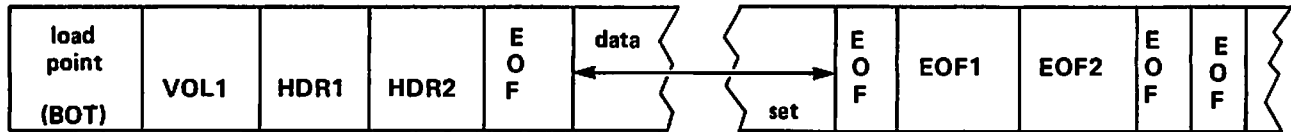
Item	Wang	IBM	ANSI
Volume Label	VOL1	VOL1	VOL1
Dataset Headers	HDR1 HDR2	HDR1 HDR2	HDR1 HDR2
Dataset Trailers	not used	EOV1*	EOV1*
	not used	EOV2*	EOV2*
Dataset Trailers	EOF1** EOF2**	EOF1** EOF2**	EOF1** EOF2**
User Headers	not used	UHL1***	UHL1***
	not used	UHL8***	UHLn***
User Trailers	not used	UTL****	UTL1****
	not used	UTL8****	UTLn****

\* = End-of-Volume  
\*\* = End-of-File  
\*\*\* = user-header-label  
\*\*\*\* = user-trailer-label

A labelled tape can be:

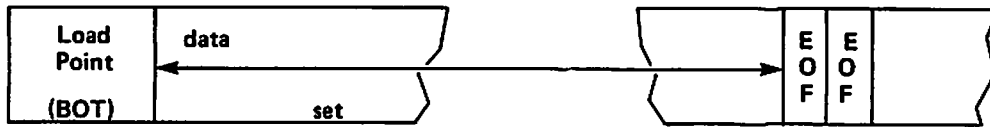
- a. IBM standard.
- b. ANSI standard.

The nine-track utilities can process either IBM or ANSI standard tapes containing single volumes and single data sets of character data. On such tapes, the physical label and file structure is as follows:



Labeled tapes can be processed, written and verified by the nine-track utilities; certain error conditions are automatically flagged.

An unlabelled tape has the following form:



Such an unlabelled tape can also be processed by the nine-track utilities.

## 2.1 VOLUME LABELS

The volume label identifies the volume and its owners; when a tape is read by the utilities, the volume label is checked to verify that the correct tape has been mounted. The volume label is 80 bytes long with its first four bytes = VOL1. A nine-track tape that has never been used can have such a volume label written by the INITIALIZE/REWRITE VOLUME LABEL routine. An existing volume label can also be updated with this same routine.

The format of the volume label is as follows:

Field No.	Starting and Ending Bytes	No. of Bytes	Description of Contents
1	1 to 4	4	Label identifier, always = VOL1.
2	5 to 10	6	Volume serial number can be less than 6 bytes but must be left-justified (any characters 0 to 9, A to Z and - (hyphen) are legal).
3	11	1	Reserved, not processed.
4	12 to 21	10	Not used, not processed.
5	22 to 31	10	Reserved, not processed.
6	32 to 41	10	Reserved, not processed.
7	42 to 51	10	Owner's name and address; can be any 10-byte code.
8	52 to 80	29	Reserved, not processed.

## 2.2 HEADER/TRAILER LABEL 1

Header labels contain data used to identify and describe the data set recorded on tape and can be used to protect data from unauthorized use. Header labels are written whenever the WRITE DATA ON TAPE or any data transfer routine (card or disk to tape) is executed. Except for the first field which contains either HDR1 (for a header) or EOF1 (for a trailer), header and trailer labels contain the same information.

The header and trailer labels each contain 80 bytes as follows:

Field	Starting and Ending Bytes	No. of Bytes	Contents
1	1 to 4	4	Label identifier: either HDR1 (if header) or EOF1 (if trailer).
2	5 to 21	17	Data set identifier: volume serial number, expiration date, protection code and password. An identifier of less than 17 bytes is padded (filled out) with blanks.
3	22 to 27	6	Data set serial number*: from volume serial number.)
4	28 to 31	4	Volume sequence number*: always = 0001.
5	32 to 35		Data set sequence number*: always = 0001.
6	36 to 39	4	Generation number*: entered by user during write operation. Should be a number between 0001 and 9999.
7	40 to 41	2	Version number of generation*: entered by user during write operation. Can be a number from 01 to 99.
8	42 to 47	6	Creation date*: entered by user during write operation; year and day-of-year in the form yyddd (yy=year, 00 to 99; ddd=day, 001 to 366).
9	48 to 53	6	Expiration date: checked by OPEN WRITE routine; year and day-of-year in yyddd format. If an attempt is made to write on an unexpired tape, an error message is issued.

\*not used or verified.

10	54	1	<p>Security flag:  0 = no protection.  1 = protected against read and write operations.  3 = protected against write operations.  When a tape is protected, the Data Set ID is used as a password to permit operations. A flag other than 0, 1 or 3 cannot be recorded (is rejected by routine).</p>
11	55 to 60	6	<p>Block count. 000000 in HDR1, contains count of blocks in the file in EOF1. (Value counted and stored by CLOSE WRITE routine.)</p>
12	61 to 80	20	<p>Reserved, not used.</p>

### 2.3 HEADER/TRAILER LABEL 2

Header Label 2 always follows Header Label 1; Trailer Label 2 always follows Trailer Label 1. Each contains the following 80 bytes of information:

Field	Starting and Ending Bytes	No. of Bytes	Contents
1	1 to 4	4	Label ID (HDR2 or EOF2).
2	5	1	Format Indicator: always = F (indicating record format of fixed length). Checked during OPEN READ operation. If not F, error message is issued and tape is rejected.
3	6 to 10	5	Block length in bytes: entered by user when writing tape. Can be up to 4096 bytes. It must be a multiple of the logical record length in Field 4.
4	11 to 15	5	Record length in bytes.
5	16	1	Tape density flag*: always = 3 (9-track, 1600 bpi).
6	17	1	Data Set Position*: always = 0 since only one volume can occur.
7	18 to 34	17	Job/Job Step ID*: identification of Job and Job Step which create data set. In the form nnnnnnnn/jjjjjjjj where n...n is the job name and j...j is the job step. The slash (/) must appear.
8	35 to 36	2	Tape Recording Technique*: always = blank indicates 9-track tape of odd parity and no translation.

\*not checked or verified.

9	37	1	Control character: a code indicating which control character set was used to create the data set and the type of control characters used. A = ASCII control characters. M = machine control characters. Blank = no control characters.
10	38	1	Reserved.*
11	39	1	Block attribute: a code indicating the type of blocking used in creating the data set. B = blocked records S = spanned records. R = both blocked and spanned records. If blank, neither blocked nor spanned records. (A spanned record crosses block boundaries.)
12	40 to 80	41	Reserved*.

\*not checked or verified.



## CHAPTER 3: PRIMARY ROUTINES

The LOADER and the primary nine-track programs are described in this chapter with operating instructions.

### 3.1 THE LOADER

The LOADER loads both a menu to simplify program selection, and the nine-track subroutines which are used for logical and physical tape operations.

Operating Instructions  
key.

EXEC = touch the RETURN(EXEC)

Display

Instruction

1. Mount program cassette or diskette in unit 1 (tape unit 10A or disk unit 310) and close the door.
2. Enter LOAD, (EXEC) (for tape) or LOAD DCF "START" (for disk), (EXEC). RUN, (EXEC).

#### NOTE:

On the Model 2209A, any utility routine which accesses the nine-track tape with the exception of read data from tape and write data on tape displays a tape status prompt:

STATUS OF TAPE Nonprotected or Protected  
STATUS OF TAPE DRIVE Ready or Not Ready  
Key "S" to stop, or "C" to continue. (EXEC.)

3. 2209A UTILITY LOADER ROUTINE  
ENTER THE NUMBER TO CHOOSE  
THE OPTION  
?\_

3. Enter the appropriate number  
on the numeric keyboard and  
touch (EXEC).

1. INITIALIZE/REWRITE VOLUME LABEL
2. WRITE DATA ON TAPE
3. READ DATA FROM TAPE
4. TAPE TO DISK DATA TRANSFER
5. DISK TO TAPE DATA TRANSFER
6. CARD READER TO TAPE DATA TRANSFER
7. DUMP THE TAPE CONTENTS
8. END OF PROCESS

Further operating instructions are found on the following pages:

Program	Page
INITIALIZE/REWRITE VOLUME LABELS	14
WRITE DATA ON TAPE	20
READ DATA FROM TAPE	27
TAPE TO DISK DATA TRANSFER	30
DISK TO TAPE DATA TRANSFER	35
CARD READER TO TAPE DATA TRANSFER	38
DUMP THE TAPE CONTENTS	42

NOTE:

A nine-track tape to be read or written must be mounted in the drive and positioned at the load point (BOT). To accomplish this, mount the tape (see instructions in the Model 2209A manual), press the POWER switch on tape drive control panel, and press the LOAD switch. The drive should briefly wind up tape in reverse and then advance tape (FORWARD lamp illuminated) until the load point is reached. When the tape is positioned at the load point, and only then, the LOAD lamp is illuminated. The ON LINE lamp should illuminate when the ON LINE selector is depressed. The WRITE ENABLE lamp is illuminated if the WRITE ENABLE ring has been inserted in the reel. (For more information on operation of the drive, see the Model 2209A Reference Manual.)

### Note on Error Messages:

Every routine in this package, whether a primary routine or a subroutine, contains at least one error message that is displayed to assist the user whenever a processing error occurs. Error messages for the primary routines are discussed with the Operating Instructions; those for the subroutines are described in Chapter 4, Subroutines. Since most primary routines call at least one of the subroutines, a subroutine error message may occur when running a primary routine. All error messages are referenced in the Index.

Four error messages are common to all routines and subroutines:

- 1) RE-ENTER indicates that a response to a prompt is invalid; either an inappropriate letter or number has been entered or the entry was in an incorrect form.
- 2) TAPE IS NOT READY indicates that the nine-track tape is not properly set up for operation. User should verify that:
  - a) POWER, LOAD, ON LINE (FILE PROTECT) lamps are illuminated.
  - b) Tape controller is on.
  - c) System is connected to tape unit.
- 3) END OF TAPE MARK IS DETECTED indicates that the EOT reflective marker at the end of the tape has been reached. Neither reading nor writing should be performed beyond this point.
- 4) SYSTEM ERROR indicates an important system (CPU) malfunction. If not cleared by Master Initialization, contact your Wang Service Representative.

### 3.2 INITIALIZE/REWRITE VOLUME LABEL

This routine creates a label or overwrites an existing label on a nine-track tape. Label information is entered by the user on the keyboard and recorded in the appropriate IBM or ANSI format on tape. To run this routine, the user must have at least one nine-track tape to be used for output.

The routine performs the following operations:

1. Requests the label type (IBM, ANSI or none).
2. Processes the volume label information which is input (Volume Serial Number, Owner Name, Tape File Status).
3. Performs an OPEN WRITE\* if tape status is 'old'; omits this, if new.
4. Rewrites the Volume Label and a dummy HDR1 (filled with zeroes); the label is left-justified.
5. Rewinds the tape.
6. Closes the file.

Multiple tapes can be handled; if label type entered is not the same as the label type on the tape, the tape is rejected. The label may then be reentered or another scratch tape mounted. If the label types match, the remaining OPEN WRITE checks are made. When all checks and verifications are complete, the volume label, a dummy header (HDR1 followed by 76 zeroes), and a file mark (EOF) are written. The volume serial number is read back from tape, displayed on the CRT and another 'mount tape' message is issued. This process is repeated until the requested number of tapes have been initialized.

The user must specify:

1. How many tapes are to be initialized (1 to 99).
2. Whether data are to be input in the tape label.
3. Label type (IBM, ANSI or none).
4. Volume serial number.
5. Owner's name and address code (optional).
6. Tape file status (new or old); if status is old, the date must be entered for comparison against the expiration date already on the tape.

\*For specific information on the operation of the OPEN WRITE, see the description of this subroutine (Chapter 4).

Once all information for a label has been entered, the user can verify it and change any byte or bytes as needed. When the routine completes execution, the nine-track tape rewinds and comes to rest at the load point (BOT).

To mount another tape, press RESET (tape drive), REWIND (tape drive), dismount the initialized tape and mount another. Press LOAD (tape drive) to place the newly mounted tape at the loadpoint, and touch (EXEC) to resume program execution.

This routine can display the following error message:

ERROR ON STARTING-TERMINATING BYTES indicates that the terminating byte input was smaller than the starting byte of a field; it must be equal or larger.

Operating Instructions

EXEC = touch the RETURN(EXEC) key

- Display
- 1.
  2. VOLUME LABEL INITIALIZATION PROGRAM  
ENTER THE NUMBER OF TAPES TO BE INITIALIZED  
?\_--/
  3. VOLUME...  
MOUNT TAPE #1, KEY (EXEC)  
TO RESUME  
?\_
  4. ENTER LABEL CONTENTS  
(1 = ENTER LABEL, 0 = SKIP)  
?\_-/
  5. ENTER LABEL TYPE  
(I = IBM, A = ANSI, N = NO)  
?\_-/
  6. ENTER THE VOLUME SERIAL NUMBER (REQUIRED)  
?\_-----  
  

0	1	.....
VOL1		
4	5	.....
  7. ENTER THE OWNER NAME & ADDRESS CODE  
?\_-----

Instruction

1. Touch 1, (EXEC) (to choose this routine).
2. Any number from 1 to 99 is valid. 1 (EXEC) (for example).
3. Mount the 9 track tape, (EXEC).  
  
If TAPE IS NOT READY message is displayed, check that POWER, LOAD and ON LINE lamps are illuminated.
4. To enter data in volume label, key 1, (EXEC).  
To skip, touch 0, (EXEC).
5. To create or update an IBM-type label, enter I, (EXEC); for an ANSI label, enter A, (EXEC) otherwise enter N, (EXEC).  
Any other reply is rejected with the message RE-ENTER. If Step 4 = 0, go to step 12.
6. The VSN must be entered. Any six characters are acceptable. The 80-bytes of the volume label are displayed on the CRT and receive all updated values.  
Enter VSN, (EXEC).
7. This field is used for optional identification; ten characters are allowed. They are stored in bytes 42 to 51 of the Volume label.  
Enter code, (EXEC).

- |   |   |
|---|---|
| <p>8. LABEL INFORMATION<br/>ROUTINE/(UPDATE LABEL)</p> <p>1. KEY '1' TO UPDATE THE LABEL.</p> <p>2. KEY '2' TO RE-ENTER LABEL</p> <p>3. KEY '3' TO CONTINUE<br/>PROCESSING.</p> | <p>8. To correct certain bytes in the Volume Label, touch 1; to re-enter the label from scratch, touch 2 (return to step 6); to continue, touch 3 (go to step 12).<br/>Do not use the (EXEC) key.</p>   |
| <p>9. ENTER THE STARTING BYTE<br/>LOCATION<br/>?--/</p>   | <p>9. Byte location (5 to 80) can be found by examining the CRT display of the Volume Label.<br/>Enter location, (EXEC).</p>  |
| <p>10. ENTER THE ENDING BYTE LOCATION<br/>?--/</p>  | <p>10. Enter location, (EXEC).</p> <p>(If only one byte is to be changed, starting byte = ending byte. A field to be updated cannot be larger than 64 bytes.)</p>   |
| <p>11. KEY IN THE NEW CHARACTERS<br/>?----</p>  | <p>11. The appropriate number of dashes is displayed. Preceding blanks and commas are not allowed unless the character string is enclosed in double quotes ("). (The quotes do not count as characters in the string.)</p> <p>Enter characters, (EXEC).</p> |
| <p>12. VOLUME LABEL INITIALIZATION<br/>PROGRAM<br/>ENTER TAPE FILE STATUS<br/>(0 = OLD, N = NEW)<br/>?_</p>   | <p>12. If file status is old, the old volume label and expiration date are checked before the new Volume Label is written; if new, this check is not made.</p> <p>Enter 0, (EXEC), if old<br/>or N (EXEC), if new.</p> <p>If N, skip steps 13 and 14.</p>   |
| <p>13. ENTER THE VOLUME SERIAL NUMBER<br/>?_-----</p>   | <p>13. Enter the serial number VSN. 6 characters are acceptable.</p>  |
| <p>14. ENTER TODAY'S DATE (YYDDD) FORMAT<br/>?_-----/</p>   | <p>14. Enter today's date in Julian format.<br/>YY=YEAR 19--(00 to 99)<br/>DDD=DAY OF THE YEAR (001 to 366)</p>   |

15.

15. After a label has been written, dismount the tape. If more than one tape is to be initialized, return to step 3 and repeat the sequence until all tapes are initialized. When all tapes requested have been initialized, the menu is reloaded for the user to select the next routine needed.

**Note:**

A number of error messages may be displayed at this point.

**THE VSN DO NOT MATCH**

(This indicates, for an old-status tape, that the new Volume Serial Number input and, the VSN on the tape are not the same.) The prompt appears "ENTER C TO CONTINUE, S TO STOP".

To proceed and write the label enter C; to stop, enter S. Do not touch (EXEC). If S, the prompt "ENTER 1 TO RE-ENTER LABEL 0 TO MOUNT TAPE" appears. Enter 1 or 0, (EXEC). Processing returns to step 4.

**LABEL TYPES ARE NOT IDENTICAL**

(This indicates a label type conflict.)

Enter 1 to reenter the label, or 0 to process another tape.

**ATTEMPT TO WRITE ON UNEXPIRED TAPE**

(For an old-status tape, this indicates that the date on the tape has not yet been reached.)



To override, enter C; to stop,  
enter S and return to step 4.

PERMANENT DATA ERROR ON  
WRITE

(This implies a permanently  
damaged tape; it must be re-  
moved i.e., the bad section cut  
from the tape.)

Dismount the tape and return  
to step 1.

END OF TAPE MARK IS  
DETECTED. WARNING  
ON WRITE

(Tape has either been in-  
correctly written or is not  
at load point.)

Dismount the tape and return  
to step 1.

THE DATA SET NAMES DO NOT  
MATCH

Enter 1 to reenter the label,  
or 0 to mount a new tape;  
processing returns to step 4.

(May indicate file is protected  
and user must supply the password  
to access tape.)

### 3.3 WRITE DATA ON TAPE

This routine is used to write data from memory on to the nine-track tape as specified by the user object program (lines 12 to 9120). Data may either be keyed in on the keyboard or loaded into memory from tape cassette or disk; it is then sent to the nine-track tape drive to be recorded.

When called from the menu, the routine automatically loads the Logical and Physical IOCS routines (Module "TPUTO30A") providing the user with all the logical and physical tape operations needed.

The user must have:

A pre-initialized nine-track tape on which the appropriate Volume Label has been written, if any. (For description of tape labels, see Chapter 2.)

The user must specify:

1. Volume Label Information (label type (IBM, ANSI or none), Volume Serial Number); data can be updated.
2. Header Label Information (Data Set Identifier, Generation Number, Version Number, Creation Date, Expiration Date, Security Flag, Block Length, Record Length, Job/Job Step ID, Control Character, Block Attribute).
3. Whether a program or data text is to be keyed in from the keyboard or loaded from tape cassette or disk.
4. A dimension (DIM) Statement in line 12 defining the number of bytes in the data transfer output buffer if more than 2000 bytes are needed for each record block. Block size must be less than or equal to the output buffer.

**NOTE:**

In the user object program, the buffer is defined by the alpha-array C\$(x)y, where x must be  $\geq 2$  and y must = 40.

5. Calls to the following logical IOCS subroutines in this order:

OPEN WRITE	(DEFFN'42)
TRANSLATE (before write)	(DEFFN'204 (1) )
WRITE A BLOCK	(DEFFN'31 (11))
CLOSE WRITE	(DEFFN'44)

\*In this CLOSE WRITE operation, if the tape is labelled, a file mark (EOF), two trailer labels, and two file marks (EOF) are written; if the tape is not labelled, only two file marks (EOF) are written (see the subroutine descriptions for details on their operations). At the termination of the CLOSE WRITE operation, the nine-track tape is rewound.

The dimension statement, if needed, and the four subroutine calls must be inserted by the user in the program; lines 12 through 9120 are left free for this purpose. Any user-defined processing can also be programmed within these lines. A sample routine is shown in Appendix A.

The OPEN WRITE routine verifies Volume Serial Number, expiration date, protection code and password, if needed. During the WRITE A BLOCK operation, the block count is incremented by one after each block is written (although the block count is not decremented after a backspace operation). During the write operation, the Physical IOCS routine provides error diagnostics for certain error conditions. When an unrecoverable error occurs, program execution terminates. File status is checked during the write operation; if the file is open (variable O\$ = "O" (letter O)), the file can be written on; if the file is closed (O\$ = "C"), it cannot be written on.

After all verifications are complete, the call to the OPEN WRITE routine positions a labelled tape at the end of the Volume Label, writes the new Header Labels (HDR1, HDR2) and a file mark (EOF). At this point the tape is correctly positioned for the addition of the new data set. If the tape is unlabelled, the tape is rewound to the load point after verifications are complete, and repositioned to add data on the tape.

OPERATING INSTRUCTIONS

(EXEC) = touch the RETURN (EXEC) key.

- | Display   | Instruction   |
|---|---|
| 1.  | 1. Touch 2, (EXEC).   |
| 2. 2209A UTILITY LOADER ROUTINE<br>ENTER THE VOLUME LABEL<br>TYPE (I/A/N)<br>?-<br>I = IBM LABEL<br>A = ANSI LABEL<br>N = NO LABEL  | 2. Enter I, A or N.<br>I, (EXEC) (for example).<br><br>LABEL TYPE=A, proceed to step 3.<br>LABEL TYPE=I, proceed to step 3.<br>LABEL TYPE=N, proceed to step 2.1. |
| 2.1 ENTER DATA TYPE (E=EBCDIC, A=ASCII)<br>?-/  | 2.1 Enter the data type. If E is selected, input data is translated into EBCDIC before being written onto the tape. Proceed to step 19.                           |
| 3. LABEL INFORMATION ROUTINE/<br>ENTER THE VOLUME SERIAL NUMBER<br>(REQUIRED)<br>?-----   | 3. Enter Volume Serial Number, (EXEC). (Six characters are allowed. This field must contain valid data.)  |
| 4. LABEL INFORMATION ROUTINE/<br>(UPDATE LABEL)<br><br>0                  1                  2...<br>- . - . - . - . - . - . - . - . - .<br>VOL<br>4                  1                  5                  6                  .<br>- . - . - . - . - . - . - . - . - . | 4. This permits changes to the displayed label (bytes 5 to 80).   |

Note:

No more than 64 bytes can be updated at a time.

- KEY '1' TO UPDATE THE LABEL
- KEY '2' TO REENTER ONE LABEL
- KEY '3' TO CONTINUE PROCESSING

To change certain bytes, key 1; to reenter the entire label, key 2 (return to step 3); to proceed with the write operation, key 3, and go to step 5. Do not key (EXEC).



- |   |   |
|---|---|
| <p>10. ENTER THE DATA SECURITY<br/>(REQUIRED)<br/>?=<br/>_</p>  | <p>10. Enter security flag. (EXEC)</p> <p>0 = no protection.<br/>1 = protected against read and write operations.<br/>3 = protected against write operations.</p> <p>When a tape is protected, the DATA SET ID (step 5) is used as a password.</p>    |
| <p>11. LABEL INFORMATION ROUTINE/<br/>(UPDATE LABEL)<br/>KEY '1' TO UPDATE THE LABEL.<br/>KEY '2' TO REENTER THE LABEL.<br/>KEY '3' TO CONTINUE PROCESSING.</p> | <p>11. To change certain bytes, key 1; to reenter the entire label, key 2; to proceed, key 3. If 1, processing returns and displays step 4 and then returns to step 11. If 2, processing returns to step 5. If 3, processing proceeds to step 12.</p> |
| <p>12. LABEL INFORMATION ROUTINE/<br/>(ENTER HEADER LABEL 2)<br/>ENTER THE BLOCK LENGTH<br/>(REQUIRED)<br/>?-----/<br/>_</p>                                    | <p>12. Enter block length (in bytes), (EXEC).</p> <p style="text-align: center;">Note:</p> <p>Must be a multiple of the logical record length.</p>  |
| <p>13. ENTER THE RECORD LENGTH<br/>(REQUIRED)<br/>?-----/<br/>_</p>   | <p>13. Enter record length, (EXEC).</p>   |
| <p>14. ENTER THE JOB/<br/>JOB STEP I.D.<br/>?-----/<br/>_</p>   | <p>14. Enter ID, (EXEC).</p> <p style="text-align: center;">Note:</p> <p>Must be in the form nnnnnnnn/<br/>jjjjjjjj where n...n is the name and j...j is the job step. The slash (/) must appear.</p>   |
| <p>15. ENTER THE CONTROL CHARACTER<br/>(REQUIRED)<br/>?=<br/>_</p>  | <p>15. Enter <u>code</u>, (EXEC).</p> <p>A = ASCII control characters.<br/>M = machine control characters.<br/>Blank = no control characters.</p>   |

16. ENTER THE BLOCK ATTRIBUTE  
?\_

16. Enter block attribute, (EXEC).

B = blocked records.  
S = spanned records.  
R = both blocked and spanned records.  
Blank = neither blocked nor spanned.

17. LABEL INFORMATION ROUTINE/  
(UPDATE LABEL)

17. Enter 1, 2 or 3.  
If 2 processing returns to step 12.

KEY '1' TO UPDATE THE LABEL.  
KEY '2' TO REENTER THE LABEL.  
KEY '3' TO CONTINUE PROCESSING.

If 1 processing returns to step 4 (4.1, 4.2, 4.3) and then returns to step 17.  
If 3 processing proceeds to step 18.

This permits final verification and correction of the label before writing it on tape.

18. SEARCH & LOAD THE PROGRAM  
2209A READ/WRITE DATA UTILITY

18. To load a user-designed program from tape cassette or disk, touch 1.  
to enter program lines from the keyboard, touch 0; if 0, go to step 25.

KEY 1 TO LOAD PROGRAM,  
KEY 0 TO STOP  
?-/

19. KEY D FOR DISK, KEY C FOR  
CASSETTE

19. This permits a choice of program input medium-disk or tape cassette. Enter D or C, (EXEC). If D, go to step 23.

?\_  
This prompt does not appear if entering lines on the keyboard, or if operating the 2200VP.

20. MOUNT THE CASSETTE IN THE TAPE  
DRIVE. KEY (EXEC) TO RESUME  
?\_

20. Mount tape cassette or disk, (EXEC).

21. SELECT DISK DEVICE ADDRESS  
?\_

21. Select the device of the unit in which the program disk is mounted. Enter 1,2,3,4,5 or 6, (EXEC).

<u>REPLY</u>	<u>DEVICE</u>
1	310
2	B10
3	320
4	B20
5	350
6	B50

22. ENTER PROGRAM NAME  
?\_-----

23. ENTER TODAY'S DATE IN  
(YYDDD) FORMAT  
?\_-----

ENTER DATA SET NAME  
?\_-----

24. ENTER THE BLOCK SIZE  
?\_-----/

25. STATEMENT NO.  
FOR PROGRAM ARE

12 - 9120

26. END OF WRITE

22. Enter name, (EXEC).  
The user's program is loaded and  
takes control.

23. If the tape has a label, the  
user's call to 'OPEN WRITE'  
requests the date for  
expiration date checking. If  
there is a protection code (a  
1 or 3 in byte 54 of header 1),  
the user is also asked to 'ENTER  
THE DATA SET NAME'. The answer  
must agree with the 17 bytes of  
step 5.

24. If a tape has no label, the  
user's call to 'OPEN WRITE'  
asks for the block size.  
Enter BLOCKSIZE, (EXEC).

25. This prompt permits the user  
to insert any required pro-  
gram lines in the program.  
Line 12 must contain the DIM  
statement that defines buffer  
size. (A sample routine is pro-  
vided in Appendix A.)

26. After the user routine has been  
inserted and run, the file can be  
closed and the tape rewound to  
the load point through the user's  
call to CLOSEWRITE. To dismount  
the tape, press REWIND (tape  
drive) and remove the tape from  
the unit.



### 3.4 READ DATA FROM TAPE

This routine reads data from tape as specified by the user object program (lines 12 to 9120). When called from the menu, the routine automatically loads the Logical and Physical IOCS routines (Module "TPUTO30A"), providing the user with all the necessary subroutines.

To run this routine, the user must have:

A nine-track tape containing data.

The user must specify:

1. The label type (IBM, ANSI or none).
2. A dimension statement (in line 12 of the user program) defining the number of bytes in the transfer buffer if more than 2000 bytes are needed. Block size must not be larger than the input buffer, 4K bytes. The variable C\$(x)40 in line 12 represents the buffer; x must be  $2 \leq x \leq 255$ ; element length must = 40.
3. Calls to the following IOCS subroutines in this order:

REWIND TAPE	DEFFN'31(6)
OPEN READ	DEFFN'43
READ a Block	DEFFN'31(12)
TRANSLATE after Read	DEFFN'204(0)
CLOSE READ	DEFFN'45 .

During the read process, block count is incremented and stored in the variable U1; block count is not decremented during backspace operations. The CLOSE READ operation closes the file and rewinds the tape.

As in writing on tape, the appropriate dimension statement and subroutine calls can be inserted in lines 12 through 9120 which are left free for this purpose. Any user-defined processing can also be programmed within these lines. A sample routine is shown in Appendix A.

The OPEN READ process opens the file, verifies the label and checks the security code to prevent unauthorized use. When verification is complete, the tape is positioned at the beginning of the data set. If the tape is unlabelled, the OPEN READ opens the file and positions the tape at the load point.

The READ a block and TRANSLATE routines operate together to read in a block and translate it appropriately. CLOSE READ closes the file, terminates the read process and rewinds the tape to the load point.

OPERATING INSTRUCTIONS

(EXEC) = touch the RETURN (EXEC) key

Display

Instruction

- 1.
2. 2209A UTILITY LOADER ROUTINE

ENTER THE VOLUME LABEL  
TYPE (I/A/N)  
?\_

I = IBM LABEL  
A = ANSI LABEL  
N = NO LABEL

- 2.1 ENTER THE DATA TYPE  
(E=EBCDIC, A=ASCII)  
?\_/\_

3. 2209A UTILITY LOADER ROUTINE  
SEARCH & LOAD THE PROGRAM

4. 2209A READ/WRITE DATA UTILITY  
KEY 1 TO LOAD PROGRAM,  
KEY 0 TO STOP  
?\_/\_

5. KEY D FOR DISK, KEY C FOR CASSETTE  
?\_/\_

PROMPT DOES NOT APPEAR ON THE 2200VP OR WHEN ENTERING PROGRAMS FROM THE KEYBORAD.

- 5.1 SELECT DISK DEVICE ADDRESS  
?\_

<u>REPLY</u>	<u>DEVICE</u>
1	310
2	B10
3	320
4	B20
5	350
6	B50

1. Touch 3, (EXEC).
2. Enter A, I, or N, I, (EXEC) (for example) LABEL TYPE=A, processing proceeds to step 3. LABEL TYPE=1, processing proceeds to step 3. LABEL TYPE=N, processing proceeds to step 2.1.
- 2.1 Enter the data type, (EXEC). If E is chosen, the data read from the tape is translated from EBCDIC to ASCII as it is read.
3. This display appears as the utility and subroutines are loaded into memory.
4. Key 1 to load a user-routine from tape cassette or disk; key 0 to enter user statements from the keyboard. If 0, go to step 8. Enter 1 or 0, (EXEC).
5. Enter C or D, (EXEC). If C, proceed to step 6.

- 5.1 Mount the disk.  
Select the device address of the unit in which the program disk is mounted. Enter 1, 2, 3, 4, 5 or 6, (EXEC).

5.2 ENTER PROGRAM NAME

?\_-----  
\_

6. MOUNT THE CASSETTE TAPE  
KEY (EXEC) TO RESUME

?\_

7. ENTER THE VSN (volume serial  
number labelled tapes only)

?\_-----  
\_

ENTER THE DATASET NAME

?\_-----  
\_

8. STATEMENT NOS. FOR PROGRAM  
ARE 12 TO 9120

STOP

5.2 Enter program name, (EXEC).  
The user's program is loaded  
and takes control.

6. Mount program cassette, (EXEC).  
The user's program is loaded  
and takes control.

7. The user's call to 'OPEN READ'  
requests the VSN.

If the tape is read protected  
(security flag = 1), the user's  
call to 'OPEN READ' requests  
the data set password.

8. The user can insert his own code  
in BASIC within these lines.  
Line 12 must contain the dimen-  
sion statement defining the I/O  
buffer. When user statements  
have been inserted, touching  
RUN (EXEC) starts processing  
which continues until End of file  
is reached.

### 3.5 TAPE TO DISK DATA TRANSFER

This routine transfers data from nine-track tape to disk with or without translation. Any Wang disk can be used for output. A tape to be read on the nine-track tape drive must conform to the usual utility requirements (either have an IBM or ANSI label or be unlabelled). A disk data file name up to 5 characters in length with F as the fifth character, must be input by the user. The utility automatically inserts three additional volume number and sequence flag characters. A choice of the output code (EBCDIC or ASCII) is made by the user. As blocks are written to disk a count of the sectors is displayed.

File structure on the disk is as follows:

1. For a labelled tape, the first disk sector contains 256 bytes as follows:

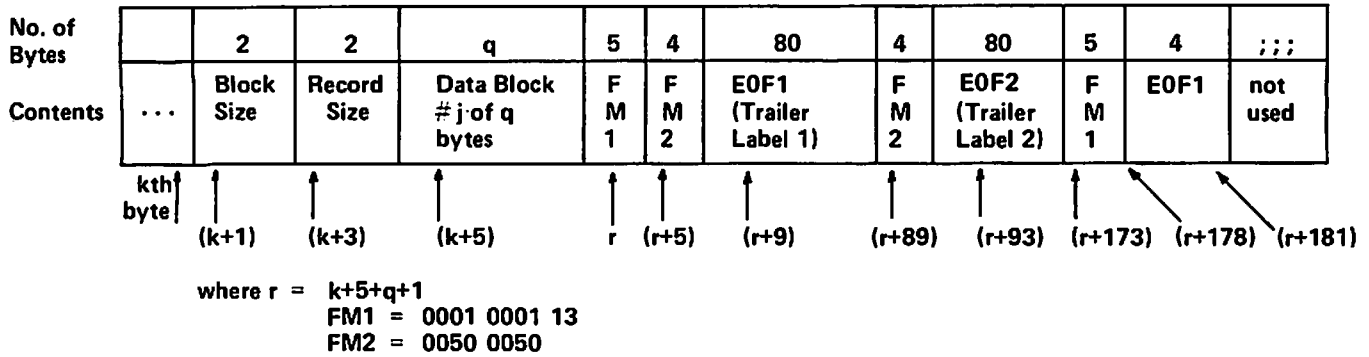
No. of Bytes	80	80	80	16	
Contents	Volume Label	HDR1 (Header Label 1)	HDR2 (Header Label 2)	Not Used	
Bytes	1	80 81	160 161	240 241	256

The next disk sector contains:

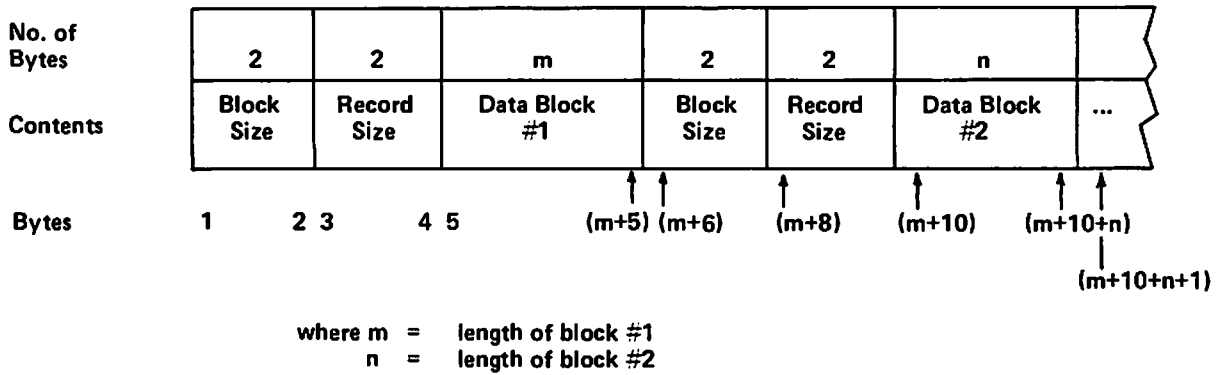
No. of Bytes	5	2	2	m	2	2	n	2	
Contents	FM1	Block Size (HEX)	Record Size (HEX)	Data Block #1	Block Size	Record Size	Data Block #2	Block Size	...
Bytes	1	5 6	7 8	9 10	(m+10)	(m+11)	(m+12)	(m+15)	(m+15+n+1)

where m = No. of bytes in block #1  
n = No. of bytes in block #2, etc.

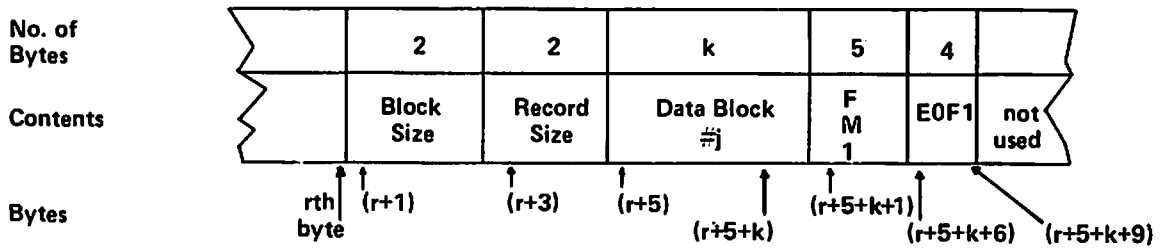
The final disk sector contains:



2. For an unlabelled tape, the first disk sector contains:



The final disk sector contains:



A nine-track tape containing data (in one file) can store up to 35 megabytes (35 million bytes) of information. Since all Wang disks contain fewer bytes than this (0.25 to 10 megabytes), each disk file written with the TAPE TO DISK routine contains a Sequence Number and Code as the last three characters of the File Name, as follows:

Character	Contents	Example
1 to 4	User assigned name	PAY1
5	F	F
6 and 7	Volume Number 00 to 99	00
8	Sequence Flag, 0 = final disk, 1 = intermediate disk	1

For example, in the above list, the file name is PAY1F001. If the disk were the final disk for the volume, the file name would be PAY1F000. The volume and sequence flags are automatically written by the nine-track utility.

To run this routine, the user must have:

- a) A nine-track tape containing data.
- b) A disk (flexible or hard) to receive the data.

The user must specify:

- a) Volume label type (IBM, ANSI or none).
- b) Disk device address.
- c) Disk file status (old or new).
- d) Five-character name for disk.
- e) Sector number of the End Catalog Area.
- f) Volume Serial No. of tape.
- g) Translation object code (EBCDIC or ASCII).

This routine can display three error messages:

- 1) FILE ALREADY CATALOGED indicates that the file name input is already on the working disk.
- 2) REACH THE "END OF CATALOG BOUNDARY" indicates that an attempt was made to write beyond the available catalog area on the disk.
- 3) LABEL TYPES ARE NOT IDENTICAL indicates that the volume label type specified in step 2 does not match the type on the tape.

OPERATING INSTRUCTIONS

(EXEC) = touch the RETURN (EXEC) key.

- | Display   | Instruction   |
|---|---|
| 1.  | 1. Touch 4, (EXEC).   |
| 2. 2209A UTILITY LOADER ROUTINE<br>ENTER THE VOLUME LABEL TYPE<br>?_<br>I = IBM LABEL<br>A = ANSI LABEL<br>N = NO LABEL   | 2. Enter type of label, (EXEC).<br>If I, proceed to step 3.<br>If A, proceed to step 3.<br>If N, proceed to step 2.1.                           |
| 2.1 ENTER THE DATA TYPE (E=EBCDIC,<br>A=ASCII)<br>?_  | 2.1 Enter the type of data on the<br>tape, (EXEC).  |
| 3. SEARCH & LOAD THE PROGRAM  | 3. The program is loaded into<br>memory.  |
| 4. 2209A - TAPE TO DISK<br>DATA TRANSFER UTILITY<br>SELECT DISK DEVICE ADDRESS<br>?_<br><br>DEVICE        ADDRESS<br><br>1. 310        4. B20<br>2. B10        5. 350<br>3. 320        6. B50 | 4. Enter a number from 1 to 6<br>which represents the disk<br>device address currently on<br>your system. (For example)<br><br>1, (EXEC).       |
| 5. MOUNT THE PLATTER,<br>KEY (EXEC) TO RESUME<br>?_   | 5. Mount the disk platter for out-<br>put on the appropriate unit.<br>Be sure the unit lamp is illu-<br>minated, (EXEC)                         |
| 6. ENTER DISK FILE STATUS<br><br>(O = OLD, N = NEW)<br>?_   | 6. Enter status, (EXEC).<br><br>(If "old", the file name etc.<br>is verified. If new, <u>the disk<br/>is scratched.</u> )                       |
| 7. ENTER THE DATAFILE NAME<br>(FOR DISK) IN 5 CHARS<br>?_-----  | 7. Enter file name, (EXEC).<br><br>Only 5 characters are allowed;<br>the fifth must = F. If step 6<br>= "old" processing proceeds to<br>step 9. |

- |   |  |
|---|--|
| <p>8. ENTER THE SECTOR # OF<br/>'END CAT. AREA'<br/>?_----/<br/>_</p> <p>9. MOUNT THE TAPE, KEY (EXEC)<br/>TO RESUME<br/>?_<br/>_</p>   | <p>8. Enter number, (EXEC).</p> <p>9. Mount the nine-track tape to<br/>be read, (EXEC).<br/>If non-labelled tape, proceed<br/>to step 11.</p>  |
| <p>Note:</p> <p>Be sure POWER, ONLINE, LOAD<br/>and FILE PROTECT lamps are illu-<br/>minated.</p>   |  |
| <p>10. ENTER THE VOLUME SERIAL NO.<br/>?_-----<br/>_</p> <p>11. ENTER THE OBJECT<br/>CODE (E=EBCDIC/A=ASCII)<br/>?_<br/>_</p> <p>11.1 ENTER THE BLOCK LENGTH<br/>?_----/<br/>_</p> <p>12. DATASAVE ON DISK SECTOR --</p> <p>13. END OF PROGRAM<br/>STOP</p> | <p>10. Enter the volume serial number<br/>as recorded on tape, (EXEC).</p> <p>11. Data are recorded on disk in<br/>the code specified. Enter<br/>code, (EXEC).</p> <p>11.1 For an unlabelled tape, enter<br/>the block length, (EXEC).</p> <p>12. This display changes as sectors<br/>are recorded.</p> <p>13. When this display occurs, pro-<br/>cessing is complete, the disk<br/>file has been written and the<br/>input tape is rewound to the<br/>load point.</p> |

To dismount the tape, press  
REWIND (tape drive) and re-  
move the tape from the unit.  
Remove the disk from its unit  
and label it.



### 3.6 DISK TO TAPE DATA TRANSFER

This routine reads an appropriately formatted disk file and writes it on a nine-track tape, with or without translation, performing an OPEN WRITE. The disk file must be catalogued and must be recorded in the format described in the TAPE TO DISK DATA TRANSFER UTILITY. If the block size recorded in the control bytes on the tape is not equal to the block size input, an error message appears and processing is terminated. If block sizes are equal, data is transferred into the output buffer and written on the tape. As each disk is read to its End-Catalog-Area, a request prompt to mount the next disk platter is issued.

The data file must contain valid data. The first disk sector read is assumed to contain valid label information. If the actual data block size is greater than that recorded on tape or specified by the user, an error message is issued and processing terminates; if the block size is less than that recorded on tape or specified by the user, the remaining unused bytes in the block are filled with blanks.

To run this routine, the user must have:

- a) A pre-initialized nine-track tape.
- b) A disk containing appropriately formatted data; the first disk sector must contain a valid tape label which must match the output tape.

The user must specify:

- a) Disk device address.
- b) Disk data file name.
- c) Tape label type (IBM, ANSI or none).
- d) Type of data on disk (EBCDIC or ASCII).
- e) Date in YYDDD format.

These error messages can be displayed by this routine:

- 1) DATA FILE IS NOT ON THE WORKING DISK indicates that the file name requested has not been recorded on the disk being read.
- 2) MISSING "EOF1" MARK ON DISK indicates that the disk being read does not have the format required (see format specifications in Section 5 TAPE TO DISK DATA TRANSFER).
- 3) THE VSN DO NOT MATCH indicates that the internal and external VSN displayed are not the same. The tape label must be made to correspond to the disk label.
- 4) LABEL TYPES ARE NOT IDENTICAL indicates that the volume label specified in step 6 does not match that on the tape.

OPERATING INSTRUCTIONS

(EXEC) = touch the RETURN (EXEC) key.

- | Display  | Instruction   |
|--|---|
| 1.   | 1. Touch 5, (EXEC).   |
| 2. 2209A UTILITY LOADER<br>ROUTINE<br>SEARCH & LOAD THE PROGRAM                      | 2. The routine is loaded into memory.   |
| 3. 2209A - DISK TO TAPE DATA<br>TRANSFER UTILITY<br>SELECT DISK DEVICE ADDRESS<br>?_ | 3. Must be a number from 1 to 6.<br><br>Enter disk address number,<br>(EXEC).                   |
| DEVICE            ADDRESS  |   |
| 1. 310            4. B20   |   |
| 2. B10            5. 350   |   |
| 3. 320            6. B50   |   |
| 4. MOUNT THE PLATTER, KEY (EXEC)<br>TO RESUME<br>?_                                  | 4. Mount platter containing the file to be transferred, (EXEC).                                 |
| 5. ENTER THE DISK DATA FILE NAME<br>(1 to 5 CHARACTERS)<br>?-----                    | 5. Enter file name, (EXEC).   |
|  | Note:   |
|  | Must end with 'F'.  |
| 6. ENTER THE TAPE LABEL TYPE<br>(I/A/N)<br>?_  | 6. Enter type, (EXEC).<br>If I or A, proceed to step 7.<br>If N, proceed to 6.1.                |
| 6.1 ENTER THE TAPE DATA TYPE<br>(E=EBCDIC, A=ASCII)<br>?_                            | 6.1 Enter the data type on the tape,<br>(EXEC).   |
| 7. ENTER THE DISK DATA TYPE<br>(E=EBCDIC/A=ASCII)<br>?_                              | 7. Enter data type, (EXEC).   |
| 8. MOUNT THE TAPE ON 2209A,<br>KEY (EXEC) TO RESUME<br>?_                            | 8. Mount nine-track tape, (EXEC).<br><br>Be sure POWER, LOAD, ON LINE<br>lamps are illuminated. |

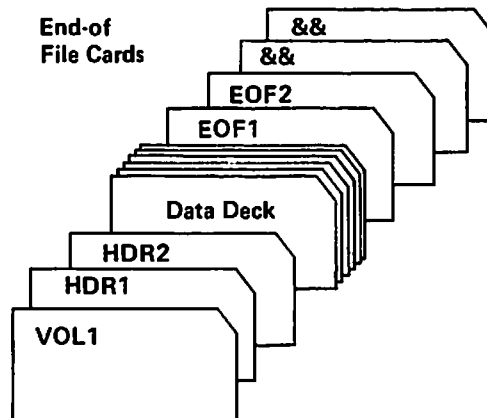
9. ENTER TODAY'S DATE IN (YYDD) FORMAT  
?\_----/
  - 9.1 ENTER THE BLOCK SIZE  
?\_----/
  10. WRITE BLOCK n ON TAPE
  11. END OF PROGRAM
9. If the tape is labelled enter date, (EXEC). Proceed to step 10.
  - 9.1 If the tape is unlabelled, enter the block size, (EXEC).
  10. This prompt increments by one with each block written.
  11. This display appears when EOF is written; tape returns to load point (BOT) and processing terminates. To unload tape, press REWIND (tape drive).

### 3.7 CARD READER TO TAPE DATA TRANSFER

This routine accepts data from standard 80-column Hollerith punched cards and creates either an unlabelled or a labelled tape depending on the structure of the card deck.

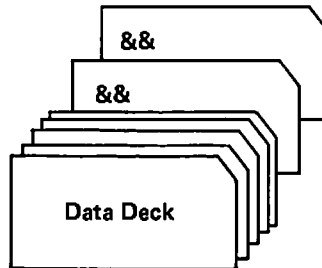
To create a labelled tape, the first three cards in the deck must contain Volume Label, Header Label 1 and Header Label 2 information, respectively, and the last four cards must contain Trailer Label 1, Trailer Label 2, &&, &&, respectively. The && cards represent End-of-File marks; the deck must end with two of them. The volume serial number in the card deck must be the same as recorded on the tape label.

The card deck structure for a labelled tape is thus:



To create an unlabelled tape, the first four bytes of any data card must not contain any of the following:

VOL1  
HDR1  
HDR2  
EOF1  
EOF2



The card deck should be terminated with a pair of && cards (End-of-File). (&) is the Hollerith punch in the top row of the card; columns 1 and 2 must both contain this punch. No punches are allowed in any other column of the End-of-File card. From a card deck used to create an unlabelled tape, a tape is created containing one record per block, with a block length of 80 bytes. Actual block size must equal block size specified on HDR2 or by the user.

The routine performs an OPEN WRITE on the tape; a labelled tape is checked to verify that label types are identical. If they are not, an error message is issued. A READ ERROR indicates a bad card in the card reader. The user can reread the card by pressing RESET on the card reader and (EXEC). The setup of the card reader switches (rear panel of reader) should be as follows:

Switch	Set To
Index Marks	Non-Clock
Data Mode	Man/Punch
Control Mode	Remote

For more information on the card reader, see the Card Reader reference manual.

To run this routine, the user must have:

- a) A pre-initialized nine-track tape.
- b) A deck of punched cards.

The user must specify:

- a) Label type (IBM, ANSI or none).
- b) Date.

This routine can display two error messages:

READ ERROR indicates that a card has been improperly read by the card reader; it can be reread or repunched to be reread as needed.

LABEL TYPES ARE NOT IDENTICAL - Indicates that the tape label type specified does not match that of the tape.

OPERATING INSTRUCTIONS

(EXEC) = touch the RETURN (EXEC) key.

Display	Instruction
1.	1. Touch 6, (EXEC).
2. 2209A UTILITY LOADER ROUTINE SEARCH & LOAD THE PROGRAM	2. The routine is loaded into memory.
3. 2209A - CARD READER TO TAPE DATA TRANSFER UTILITY LOAD CARDS INTO CARD READER, KEY (EXEC) TO RESUME ?_	3. Load cards into hopper, (EXEC).  Be sure card reader is on with its power lamp illuminated.
4. MOUNT THE TAPE, KEY (EXEC) TO RESUME ?_	4. Mount nine-track tape, (EXEC).  Be sure POWER, LOAD and ON LINE lamps are illuminated.
5. ENTER THE LABEL TYPE OF CARD DATAFILE (I/A/N) ?_	5. Enter type, (EXEC).
6. ENTER TODAY'S DATE IN (YYDD) FORMAT ?-----/	6. For a labelled tape enter date, (EXEC). Processing proceeds to step 7.
6.1 ENTER THE BLOCK SIZE ?-----/	6.1 For an unlabelled tape, enter the block size, (EXEC).
6.2 ENTER THE OBJECT FILE DATA TYPE (E=EBCDIC, A=ASCII)	6.2 For unlabelled tape, enter the input data type, (EXEC).
7. WRITE BLOCK x ON TAPE	7. This display increments as each block is written.
8. END OF PROGRAM STOP :_	8. When the entire deck has been read to the End-of-File in the card deck, the tape rewinds to the load point and processing terminates. To unload the tape, press (tape drive) REWIND and dismount the tape.

### 3.8 TAPE DUMP

This routine dumps data from tape in EBCDIC hexadecimal form and in ASCII alphanumeric form. Blocks up to 4096 bytes long can be read and dumped. If the tape is a labelled tape, the label is also output. An OPEN READ operation is performed when reading the tape.

To run this program the user must have:

- a) A nine-track tape containing data.
- b) A printer containing full-width (132 column) paper.

The user must specify:

- a) Volume label type (IBM, ASCII or none).
- b) Volume Serial No. of the tape to be read.
- c) The output data set name (printed on every page of the dump); seven characters are allowed.
- d) The date (mm/dd/yy format; slashes must appear).

The dataset name and data are printed with original tape hex codes and Wang ASCII characters. When End-of-File is reached, tape rewinds and returns to the load point (BOT). See Figure 1 for a sample of blocks dumped from a tape. A full page can contain the dump for 800 bytes of information.

The user must know and specify the VSN of a labelled tape; it cannot be dumped in the same manner as a non-labelled tape. A VSN for a tape must conform to the contents described on page 6. The user must specify whether his tape has been recorded in EBCDIC or ASCII or the tape dump will be meaningless.



2200 CORRESPONDING CHARF

BLOCK	1	SIZE	80	BYTES	TAPE CONTENTS <EBCDIC HEXIDECIMAL>
					96L1FH080870
BLOCK	1	SIZE	80	80	E50202F1D7 C8F0F8F0F7 F040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040
BLOCK	2	SIZE	80	80	C8C4D9F1E3 C1D7C5C6C9 D3C5404040 4040404040 400768F0F8 F0F7F0F0F0 F1F0F0F0F1 4040404040 4040404040 F2F440F0F0 F0F0F0F0F0 F0F0F0F0F0 0000000000 0000000000 0000000000 4040404040
BLOCK	3	SIZE	80	80	C8C4D9F2C5 F0F0F0F0F0 F0F0F0F0F0 F2F0E5C17C C2E7E2C6C1 61E5E4D5D7 D9C5E2E240 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040
BLOCK	4	SIZE	1	13	
BLOCK	5	SIZE	800	800	6161656F1F0 4001D6C240 D9E2C76807 D6D6C4C503 D266D4E2C7 D3C5E5C5D3 7EF16E07D9 E3E87EF540 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 615C0906E4 E3C5404007 D9C2C5E240 D9C5E406E3 C5F3F74040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 615C0906E2 E2C1C7C540 40C4C503C9 E5C5C940E3 C640D9C1D3 D7C840C7D6 D6C4C5C7D3 40C1E3310E6 C1D5C74003 C1C2E24040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 616140C5E7 C5C340C1E2 D1C6C24040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 4040404040 6161C1E2C4 4BE2E8E2C9 D540C4C440 5C40404040 4040404040

HCPLTAPEFILE PH08  
0700010001 74324 00  
0000000000

HCPL2F00000000020WR08XSFH  
/MURPRESS 6

//H10 JOB P99,GOODCELL,MSG  
LEVEL=L,PRINT=S

//FOUTE PRINT PRINT

MSG LABS // EXEC AS  
MFC

SYSDIN DD \*

TITLE 'WRAPG 830  
MOD 1 SIMULATOR 13MDV74

MACRO

NAME FORNAT

Figure 1. Sample of Output from TAPE DUMP

OPERATING INSTRUCTIONS

(EXEC) = touch the RETURN (EXEC) key

- | Display   | Instructions  |
|---|---|
| 1.  | 1. Touch 7, (EXEC).   |
| 2. 2209A UTILITY LOADER ROUTINE<br>ENTER THE VOLUME LABEL<br>TYPE (I/A/N)<br>?_<br>_                      | 2. Be sure printer is connected to your CPU, with its POWER and SELECT lamps illuminated, and it is filled with full-width (132 column) paper.<br><br>Enter label type, (EXEC).                   |
| 3. 2209A UTILITY LOADER ROUTINE<br>SEARCH & LOAD THE PROGRAM  | 3. The program is loaded into memory.   |
| 4. 2209A - TAPE (CONTENTS)<br>DUMP UTILITY<br>MOUNT THE TAPE ON 2209A,<br>KEY (EXEC) TO RESUME<br>?_<br>_ | 4. Mount tape to be dumped on nine-track drive. Be sure that POWER, ON LINE, LOAD and lamps are illuminated, (EXEC).  |
| 5. ENTER THE VOLUME SERIAL NO.<br>?-----<br>_   | 5. This must be the volume serial number recorded on the tape. Enter volume serial no., (EXEC).   |
| 6. ENTER THE OUTPUT DATASET NAME<br>?-----<br>_   | 6. This must be the name you want printed on the tape dump.<br><br>Enter name, (EXEC).  |
| 7. ENTER TODAY'S DATE IN<br>(MM/DD/YY) FORMAT<br>?-----<br>_  | 7. The date must be in the indicated format; leading zeroes and slashes must appear.<br><br>Enter date, (EXEC).   |
| 8. TURN ON THE PRINTER,<br>KEY (EXEC) TO RESUME<br>?_<br>_  | 8. Be sure printer is on and selected, (EXEC).  |
| 9. DUMPING THE TAPE   | 9. This message is displayed as the dump proceeds. When finished, tape returns to the load point. To dismount it push REWIND (tape drive). To end the run, turn off printer and nine-track drive. |

## CHAPTER 4: SUBROUTINES

### 4.1 LOGICAL I/O CONTROL SYSTEM

These subroutines perform the logical control operations and the translation from EBCDIC to ASCII or ASCII to EBCDIC.

DEFFN'42 performs the OPEN WRITE operation which contains the following steps for a labelled tape.

- a) Checks the current file status.
- b) Rewinds nine-track tape.
- c) Opens the file.
- d) Reads the first block and checks the label type.
- e) Checks the Volume Serial No.
- f) Checks the expiration date.
- g) If the security code is on, checks the data set name (the password).
- h) Rewrites the Header Labels (HDR1, HDR2) and the file mark.

For an unlabelled tape, only checks (a), (b), (c) and (d) are made; the user must enter block size (less than or equal to 4096 bytes). The tape is then rewound and user control restored.

Five error messages can be displayed by this routine:

- 1) ATTEMPT TO OPEN A NON-CLOSED FILE (a non-closed file is either improperly positioned on the tape drive or has not been written correctly).
- 2) THE DATASET NAMES DO NOT MATCH (the data set name is requested for any tape with a security code; it must match the name recorded on the tape).
- 3) THE VSN DO NOT MATCH (Volume Serial No. input on the keyboard and recorded on the tape must be identical; both are displayed with this error message if different).

- 4) ATTEMPT TO WRITE ON UNEXPIRED TAPE (if tape is protected with an expiration date, it cannot be written on until the expiration date exceeds today's date).
- 5) LABEL TYPES ARE NOT IDENTICAL (indicates that the specified volume label does not match that on the tape).

DEFFN'43 performs the OPEN READ operation which contains the following steps for a labelled tape.

- a) Checks the current file status.
- b) Opens the file.
- c) Rewinds the tape to the load point.
- d) Reads the first block.
- e) Checks the label type.
- f) Checks the Volume Serial No.
- g) If the security code is on, checks the data set name (password).
- h) Checks the record format.
- i) Positions the tape (after the header and tape marker) to read the first block of data.

For an unlabelled tape, only operations (a), (b), (c), (d) and (e) are performed. The tape is rewound and control returns to the user program.

Three error messages can be displayed by this routine:

- 1) ATTEMPT TO OPEN A NON-CLOSED FILE (a non-closed file is either improperly positioned on the tape drive or has not been written correctly).
- 2) THE DATASET NAMES DO NOT MATCH (dataset name is requested for any tape with security code on; it must match the name recorded on the tape).
- 3) THE VSN DO NOT MATCH (Volume Serial No. input on the keyboard and recorded on the tape must be identical; both are displayed with this error message if different).
- 4) ILLEGAL RECORD FORMAT (the record format must be reentered. The field containing the record format on header label 2 is not 'F').
- 5) LABEL TYPES ARE NOT IDENTICAL (indicates that the specified volume label does not match that on the tape).

DEFFN'44 performs the CLOSE WRITE operation which contains the following steps for a labelled tape:

- a) Writes a File Mark (EOF).
- b) Writes EOF1 (Trailer Label 1).
- c) Writes EOF2 (Trailer Label 2).
- d) Writes two File Marks (EOF).
- e) Rewinds the tape.
- f) Closes the file.

For an unlabelled tape, only steps (d), (e) and (f) are performed.

One error message can be displayed by this routine:

ATTEMPT TO CLOSE A CLOSED FILE

A closed file has the variable O\$ set to 'C' (closed); for a file to be closed, the variable must initially contain 'O' (open).

DEFFN'45 performs the CLOSE READ operation which contains the following steps for both labelled and unlabelled tapes:

- a) Rewinds tape.
- b) Closes the file.

The error message that can be displayed by this routine is:

ATTEMPT TO CLOSE A CLOSED FILE  
(see comment for DEFFN'44).

DEFFN'204 (x) performs TRANSLATION either before writing or after reading data into the transfer buffer.

For reading, x must equal 0.  
For writing, x must equal 1.

A translate flag T is automatically set at T = 0 if no translation is to be performed, and T = 1 if a translation is to be performed. Before a write operation, this routine should be called. If the tape to be written is labelled, block size is equal to the block size specified on the Header Label; for an unlabelled tape, block size must be specified by the user in the variable LO (Immediate Mode operation can be used).

For example, for a write operation, the following sequence will work:

100 GOSUB'204 (1)	(translate prior to write)
110 LO = 2000	(define block size)
120 GOSUB'31 (11)	(write a block)

## 4.2 PHYSICAL I/O CONTROL SYSTEM

Physical IOCS contains a single subroutine which can perform eighteen I/O control operations. The subroutine name is DEFFN'31(x), where x can be a value from 1 to 32 as follows:

x	Operation
1	backspace to write
2	backspace a record
3	forward-space a record
4	backspace a file
5	forward-space a file
6	rewind
7	"write" a gap
8	clean tape
9	reread a block
10	write a File Mark (EOF)
11	write a block
12	read a block
13	status request
14	master reset
21	buffered write
22	look-ahead-read
31	finish write
32	finish read

Any write or read sequence must conform to the following rule.

- 1) When writing a block, any translation must be performed first, the value of x must be (11).

The error messages that can be displayed are:

- 1) ATTEMPT TO WRITE ON PROTECTED TAPE
  - during a write operation, no FILE PROTECT ring on tape.
- 2) BLOCK SIZE NOT EQUAL
  - during a write operation, the block size input by the user is not equal to the block size stored on the header label.
- 3) BUFFER OVERFLOW ON READ
  - during a read operation, the input buffer is too small to hold one complete data block from the tape. (This message will not appear on reading records greater than 4096 bytes.)

- 4) BUFFER OVERFLOW ON WRITE
  - during write operations, the user is trying to write more than 4096 bytes of data.
- 5) FILE NOT OPEN
  - during a read or write operation, the file status is closed, that is, 0\$="C".
- 6) PERMANENT DATA ERROR ON WRITE
  - during a write operation, an unrecoverable error was detected when writing to the tape.
- 7) \*STATUS OF TAPE DRIVE = READY \*  
\*STATUS OF TAPE DRIVE = NOT READY\*
  - issued after a status request, one of these appears depending on the status bits.
- 8) \*STATUS OF TAPE DRIVE = PROTECTED \*  
\*STATUS OF TAPE DRIVE = NON-PROTECTED\*
  - issued after a status request, one of these appears depending on status bits.
- 9) END OF TAPE MARK IS DETECTED, WARNING ON WRITE
  - issued during a write operation as a warning when EOT is sensed.
- 10) KEY "C" TO CONTINUE, "S" TO STOP
  - issued after the status is displayed on a status request. If the user keys in an "S", the program stops. If he keys in a "C", or hits C/R, the program continues.

APPENDIX A: Sample user program to be inserted in lines 12 to 9120 of utility for reading or writing on nine-track tape.

```
9 REM SAMPLE PROGRAM TO READ/WRITE ON 9-TRACK TAPE
10 REM UTILITY IN DRIVE 310 DATA IN DRIVE B10
11 REM DEFINE THE I/O BUFFERS
12 DIM C$(5)40,D$(5)40
22 INIT(20)C$(),D$()
23 SELECT #3 B10
30 PRINT HEX(03)
32 PRINT "S.F.O=WRITE DATA ON TAPE S.F.15=READ DATA FROM TAPE"
52 STOP
62 DEFFN'O
65 DATA LOAD DC OPEN T#3, "DATAF"
72 GOSUB '42: REM OPEN WRITE
75 SELECT PRINT 215(80)
76 PRINT "DISK TO TAPE TRANSFER"
82 DATA LOAD DC #3 ,D$(): REM READ DATA FROM DISK
85 IF END THEN 152
92 FOR I=1 TO 5
102 C$(I) = D$(I): REM COPY DATA TO TAPE BUFFER
112 NEXT I
122 GOSUB '204(1): REM TRANSLATE BEFORE WRITE
132 GOSUB '31(11): REM WRITE A BLOCK
142 PRINT " BLOCK #",U1, "LENGTH OF",LO
145 GOTO 82
152 SELECT PRINT 005
155 GOSUB '44: REM CLOSE WRITE
162 RETURN
172 DEFFN'15
182 GOSUB '43: REM OPEN READ
185 SELECT PRINT 215(80): INIT (20) C$()
186 PRINT "CONTENTS OF TAPE"
192 GOSUB '31(12): REM READ A BLOCK
202 GOSUB '204(0): REM TRANSLATE AFTER READ
203 W$ = HEX(04): REM CHECK FOR EOF
204 AND (W$,STR(B$,7,1))
205 IF W$ = HEX(04) THEN 282
206 PRINT "BLOCK #",U1: REM PRINT CONTENTS OF THE TAPE BUFFER
212 FOR I = 1 TO 5
222 PRINT C$(I)
232 NEXT I
272 GOTO 192
282 SELECT PRINT 005
284 GOSUB '45: REM CLOSE READ
292 RETURN
```



INDEX

	PAGE
ATTEMPT TO CLOSE A CLOSED FILE (ERROR MESSAGE) . . . . .	47
ATTEMPT TO OPEN A NON-CLOSED FILE (ERROR MESSAGE). . . . .	45,46
ATTEMPT TO WRITE ON PROTECTED TAPE (ERROR MESSAGE) . . . . .	48
ATTEMPT TO WRITE ON UNEXPIRED TAPE (ERROR MESSAGE) . . . . .	18
BLOCK ATTRIBUTE. . . . .	10,24
BLOCK COUNT. . . . .	8
BLOCK LENGTH . . . . .	9,24
BLOCK SIZE NOT EQUAL (ERROR MESSAGE) . . . . .	48
BLOCKED RECORD . . . . .	10,24
BUFFER OVERFLOW ON READ (ERROR MESSAGE). . . . .	48
BUFFER OVERFLOW ON WRITE (ERROR MESSAGE) . . . . .	49
CARD DECK STRUCTURE FOR LABELLED TAPES . . . . .	38
CARD DECK STRUCTURE FOR UNLABELLED TAPES . . . . .	39
CARD READER TO TAPE DATA TRANSFER . . . . .	38
CLOSE READ . . . . .	47
CLOSE WRITE. . . . .	46
CLOSE WRITE OPERATION. . . . .	21
CONTROL CHARACTER. . . . .	10,24
CONTROL CODE . . . . .	24
CREATION DATE. . . . .	7,23
DATA FILE NOT ON THE WORKING DISK (ERROR MESSAGE). . . . .	35
DATA SECURITY FLAG . . . . .	23
DATA SET IDENTIFIER. . . . .	7
DATA SET NAME. . . . .	25
DATA SET POSITION. . . . .	9
DATA SET SEQUENCE NUMBER . . . . .	7
DATA SET SERIAL NUMBER . . . . .	7
DEFFN'204 (TRANSLATION). . . . .	47
DEFFN'31 (PHYSICAL IOCS) . . . . .	48
DEFFN'42 (OPEN WRITE). . . . .	45
DEFFN'43 (OPEN READ) . . . . .	46
DEFFN'44 (CLOSE WRITE) . . . . .	46
DEFFN'45 (CLOSE READ). . . . .	47
DISK FILE NAME . . . . .	32
DISK FILE STRUCTURE. . . . .	30
DISK TO TAPE DATA TRANSFER . . . . .	35
END OF TAPE (ERROR MESSAGE). . . . .	13
END OF TAPE MARK IS DETECTED WARNING MESSAGE . . . . .	19
ERROR MESSAGES . . . . .	13
ERROR ON STARTING-TERMINATING BYTES (ERROR MESSAGE). . . . .	15
EXPIRATION DATE. . . . .	7,23
FILE ALREADY CATALOGED (ERROR MESSAGE) . . . . .	32
FILE NOT OPEN (ERROR MESSAGE). . . . .	48
FORMAT INDICATOR . . . . .	9
GENERATION NUMBER. . . . .	7,23

HEADER/TRAILER LABEL 1 . . . . .	7
HEADER/TRAILER LABEL 2 . . . . .	9
HOLLERITH PUNCHED CARDS. . . . .	38
ILLEGAL RECORD FORMAT (ERROR MESSAGE). . . . .	46
INITIALIZE/REWRITE VOLUME LABEL. . . . .	14
JOB/JOB STEP ID. . . . .	9,24
LABELLED FORMAT . . . . .	5
LABEL TYPES ARE NOT IDENTICAL (ERROR MESSAGE). . . . .	18,32,35 40,46
LOGICAL I/O CONTROL SYSTEM . . . . .	3,45
MISSING "EOF1" MARK ON DISK (ERROR MESSAGE). . . . .	35
MULTIPLE TAPES . . . . .	14
OPEN READ. . . . .	27,46
OPEN WRITE . . . . .	21,45
PERMANENT DATA ERROR ON READ (ERROR MESSAGE) . . . . .	48
PERMANENT DATA ERROR ON WRITE (ERROR MESSAGE). . . . .	19,49
PHYSICAL I/O CONTROL SYSTEM. . . . .	3,48
PRIMARY ROUTINES . . . . .	11
PRIMARY ROUTINES (LIST). . . . .	2
PUNCHED CARDS. . . . .	38
RE-ENTER (ERROR MESSAGE) . . . . .	13
REACH THE END OF CATALOG BOUNDARY (ERROR MESSAGE). . . . .	32
READ DATA FROM TAPE. . . . .	27
READ ERROR (ERROR MESSAGE) . . . . .	40
RECORD LENGTH. . . . .	9
SAMPLE USER OBJECT PROGRAM . . . . .	50
SECURITY FLAG. . . . .	8
SPANNED RECORD . . . . .	10,24
STANDARD LABEL FORMAT. . . . .	5
SUBROUTINES. . . . .	45
SYSTEM ERROR (ERROR MESSAGE) . . . . .	13
TAPE CAPACITY. . . . .	32
TAPE DENSITY FLAG. . . . .	9
TAPE DUMP. . . . .	42
TAPE NOT READY (ERROR MESSAGE) . . . . .	13
TAPE RECORDING TECHNIQUE . . . . .	9
THE DATA SET NAMES DO NOT MATCH (ERROR MESSAGE). . . . .	19,45,46
THE VSN DO NOT MATCH (ERROR MESSAGE) . . . . .	18,35,45, 46
TRANSLATION. . . . .	47
UNLABELLED TAPE FORMAT . . . . .	5

VERSION NUMBER . . . . .	7
VOLUME LABELS. . . . .	6
VOLUME SEQUENCE NUMBER . . . . .	7
VOLUME SERIAL NUMBER . . . . .	6,22
WRITE BUFFER . . . . .	20
WRITE DATA ON TAPE . . . . .	20
WRITE ENABLE RING. . . . .	12



To help us to provide you with the best manuals possible, please make your comments and suggestions concerning this publication on the form below. Then detach, fold, tape closed and mail to us. All comments and suggestions become the property of Wang Laboratories, Inc. For a reply, be sure to include your name and address. Your cooperation is appreciated.

700-4325

TITLE OF MANUAL: 2209A NINE-TRACK TAPE UTILITIES MANUAL

COMMENTS:

Fold

Fold



Fold

FIRST CLASS  
PERMIT NO. 16  
Tewksbury, Mass.

**BUSINESS REPLY MAIL**  
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

- POSTAGE WILL BE PAID BY -

WANG LABORATORIES, INC.  
ONE INDUSTRIAL AVENUE  
LOWELL, MASSACHUSETTS 01851

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

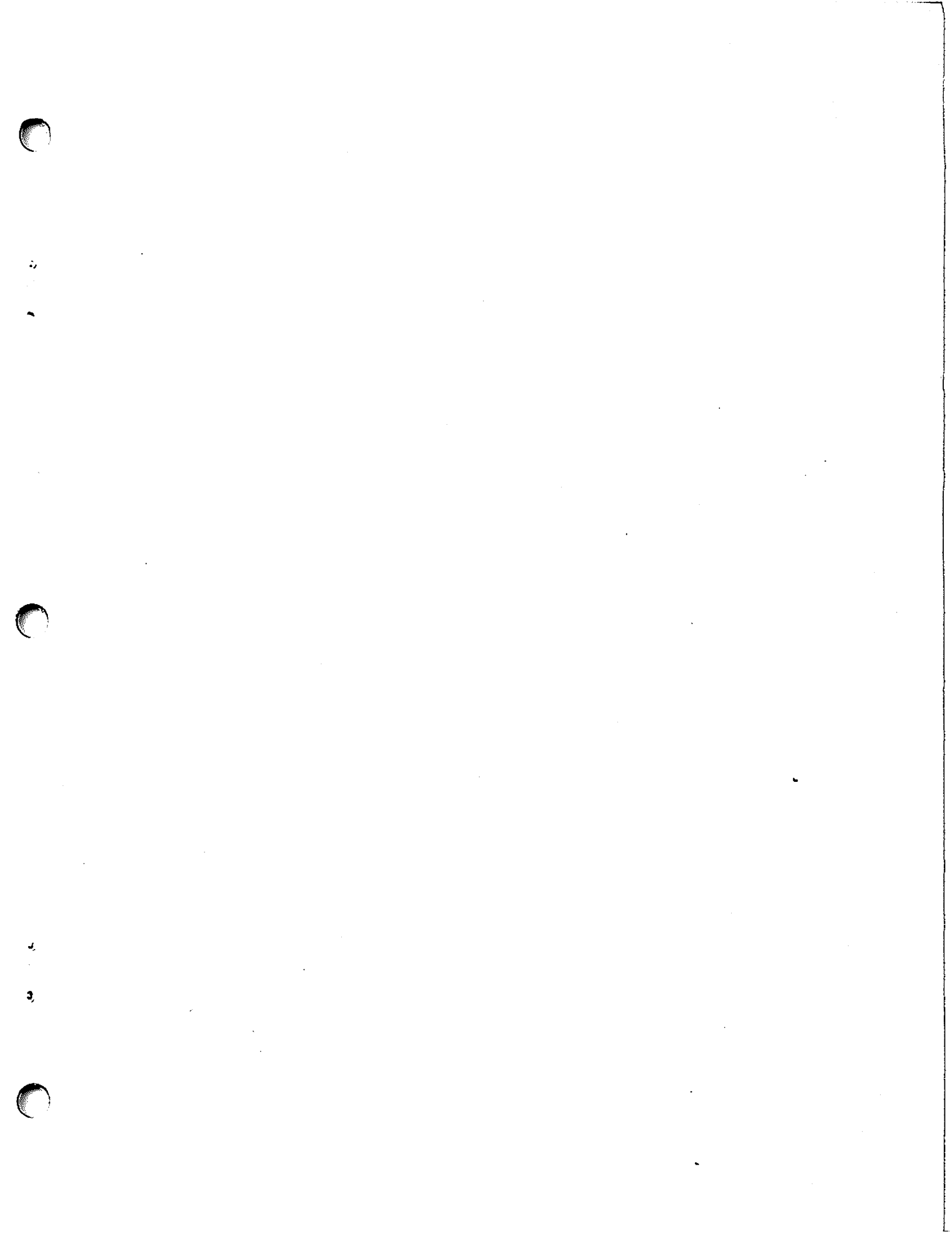
\_\_\_\_\_

\_\_\_\_\_

Attention: Technical Writing Department

Fold

Cut along dotted line.



**WANG LABORATORIES  
(CANADA) LTD.**  
49 Valleybrook Drive  
Don Mills, Ontario M3B 2S6  
TELEPHONE (416) 449-2175  
Telex: 069-66546

**WANG EUROPE S.A./N.V.**  
(European Headquarters)  
250, Avenue Louise  
1050 Brussels, Belgium  
TELEPHONE 02/640.37.80  
Telex: 12430-12398

**WANG EUROPE S.A./N.V.**  
(Belgian Sales)  
350, Avenue Louise  
1050 Brussels, Belgium  
TELEPHONE 02/648.91.00  
Telex: 62691

**WANG DO BRASIL  
COMPUTADORES LTDA.**  
Praca Olavo Bilac No. 28  
SL1801/1803  
Rio de Janeiro, Centro, RJ, Brasil  
TELEPHONE 232-7503, 232-7026

**WANG COMPUTERS  
(SO. AFRICA) PTY. LTD.**  
Corner of Allen Rd. & Garden St.  
Bordeaux, Transvaal  
Republic of South Africa  
TELEPHONE (011) 48-6123  
Telex: 960-83297

**WANG INTERNATIONAL  
TRADE, INC.**  
One Industrial Avenue  
Lowell, Massachusetts 01851  
TELEPHONE (617) 851-4111  
Telex: 94-7421

**WANG SKANDINAVISKA AB**  
Pyramidvaegen 9A  
S-171 36 Solna, Sweden  
TELEPHONE 08/27 27 98  
Telex: 11498

**WANG COMPUTER LTD.**  
Shindaiso Building No. 5  
2-10-7 Dogenzaka Shibuya-Ku  
Tokyo, Japan  
TELEPHONE (03) 464-0644  
Telex: 2424909 WCL TKO J

**WANG NEDERLAND B.V.**  
Produktieweg 1  
Ijsselstein, Netherlands  
TELEPHONE (03408) 41.84  
Telex: 47579

**WANG PACIFIC LTD.**  
9th Floor, Lap Heng House  
47-50, Gloucester Road  
Hong Kong  
TELEPHONE 5-274641  
Telex: 74879 Wang HX

**WANG INDUSTRIAL CO., LTD.**  
7, Tun Hwa South Road  
Sun Start Tun Hwa Bldg.  
Taipei, Taiwan, China  
TELEPHONE 7522068, 7814181-3  
Telex: 21713

**WANG GESELLSCHAFT MBH**  
Murlingengasse 7  
A-1120 Vienna, Austria  
TELEPHONE 85.85.33  
Telex: 74640 Wang a

**WANG GESELLSCHAFT MBH**  
Wiedner Hauptstrasse 68  
A-1040 Vienna, Austria  
TELEPHONE 57.94.20  
Telex: 76424 Wang a

**WANG S.A./A.G.**  
Markusstrasse 20  
Postfach 423  
CH 8042 Zurich 6, Switzerland  
TELEPHONE 41-1-60 50 20  
Telex: 59151

**WANG COMPUTER PTY. LTD.**  
55 Herbert Street  
St. Leonards, 2065, Australia  
TELEPHONE 439-3511  
Telex: 24569

**WANG ELECTRONICS LTD.**  
Argyle House, 3rd Floor  
Joel Street  
Northwood Hills  
Middlesex, HA6 1NS, England  
TELEPHONE (09274) 28211  
Telex: 923498

**WANG FRANCE S.A.R.L.**  
Tour Gallieni, 1  
78/80 Ave. Gallieni  
93170 Bagnolet, France  
TELEPHONE 33.1.3602211  
Telex: 680958F

**WANG LABORATORIES GmbH**  
Moselstrasse 4  
6000 Frankfurt AM Main  
Postfach 16826  
West Germany  
TELEPHONE (0611) 252061  
Telex: 04-16246

**WANG DE PANAMA (CPEC) S.A.**  
Apartado 6425  
Calle 45E, No. 9N. Bella Vista  
Panama 5, Panama  
TELEPHONE 69-0855, 69-0857  
Telex: 3282243

**WANG COMPUTER LTD.**  
302 Great North Road  
Grey Lynn, Auckland  
New Zealand  
TELEPHONE Auckland 762-219  
Telex: CAPENG 2826

**WANG COMPUTER PTE., LTD.**  
Suite 1801-1808, 18th Floor  
Tunas Building, 114 Anson Road  
Singapore 2, Republic of Singapore  
TELEPHONE 2218044, 45, 46  
Telex: RS 24160 WANGSIN

**WANG COMPUTER SERVICES**  
One Industrial Avenue  
Lowell, Massachusetts 01851  
TELEPHONE (617) 851-4111  
TWX 710-343-6769  
Telex: 94-7421

**DATA CENTER DIVISION**  
20 South Avenue  
Burlington, Massachusetts 01803  
TELEPHONE (617) 272-8550

**WANG**

LABORATORIES, INC.

ONE INDUSTRIAL AVENUE, LOWELL, MASSACHUSETTS 01851. TEL. (617) 851-4111. TWX 710 343-6769. TELEX 94-7421

Printed in U.S.A.  
700-4325  
8-77-1.2M

Price: see current list