

IDENTIFICATION

PRODUCT CODE: DEC-8E-UTDEA-A-D
PRODUCT NAME: TD8E DECTape Copy
User's Manual
DATE CREATED: May 1972
MAINTAINER: Development



DEC-8E-UTDEA-A-D

TD8E DECTape Copy
User's Manual

SOFTWARE SUPPORT CATEGORY

The software described in this document is supported by Digital Equipment Corporation under Category I, as defined on page iii of this document.

First Printing May, 1972

Copyright © 1972 by Digital Equipment Corporation

The material in this document is for information purposes and is subject to change without notice.

The following are trademarks of Digital Equipment Corporation, Maynard, Massachusetts:

| | | | |
|--------------|-----------|------------|-----------|
| CDP | Digital | LAB-8/e | RAD-8 |
| Computer Lab | DNC | OMNIBUS | RSTS |
| Comtex | Flip Chip | OS/8 | RSX |
| DEC | IDAC | PDP | RTM |
| DECTape | Indac | PHA | SABR |
| Dibol | KAL0 | PS/8 | Typeset 8 |
| | | Quickpoint | Unibus |

SOFTWARE SUPPORT CATEGORIES

Digital Equipment Corporation (DEC) makes available four categories of software. These categories reflect the types of support a customer may expect from DEC for a specified software product. DEC reserves the right to change the category of a software product at any time. The four categories are as follows:

CATEGORY I

Software Products Supported at no Charge

This classification includes current versions of monitors, programming languages, and support programs provided by DEC. DEC will provide installation (when applicable), advisory, and remedial support at no charge. These services are limited to original purchasers of DEC computer systems who have the requisite DEC equipment and software products.

At the option of DEC, a software product may be recategorized from Category I to Category II for a particular customer if the software product has been modified by the customer or a third party.

CATEGORY II

Software Products that Receive Support for a Fee

This category includes prior versions of Category I programs and all other programs available from DEC for which support is given. Programming assistance (additional support), as available, will be provided on these DEC programs and non-DEC programs when used in conjunction with these DEC programs and equipment supplied by DEC.

CATEGORY III

Pre-Release Software

DEC may elect to release certain software products to customers in order to facilitate final testing and/or customer familiarization. In this event, DEC will limit the use of such pre-release software to internal, non-competitive applications. Category III software is only supported by DEC where this support is consistent with evaluation of the software product. While DEC will be grateful for the reporting of any criticism and suggestions pertaining to a pre-release, there exists no commitment to respond to these reports.

CATEGORY IV

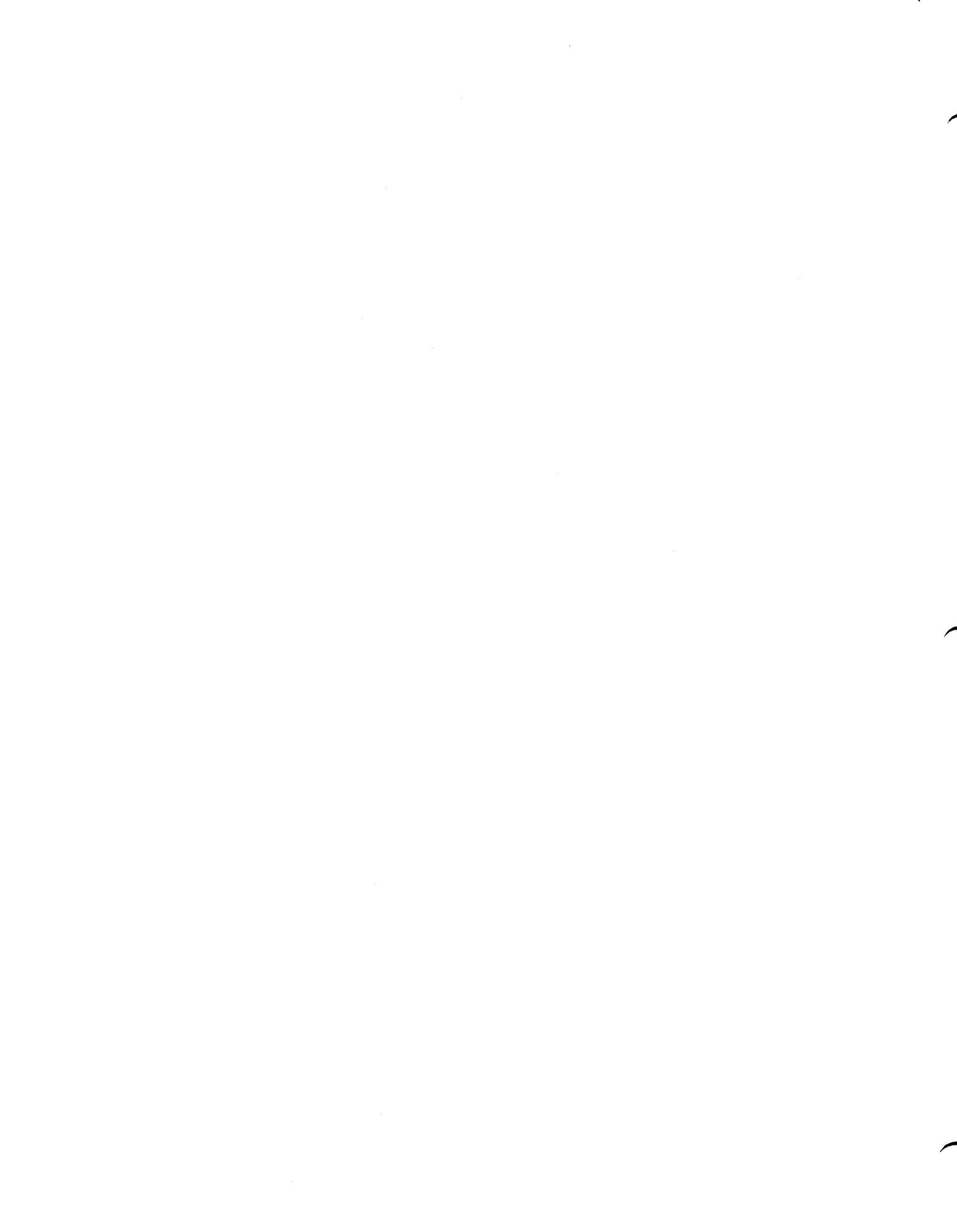
Non-Supported Software

This category includes all programs for which no support is given.



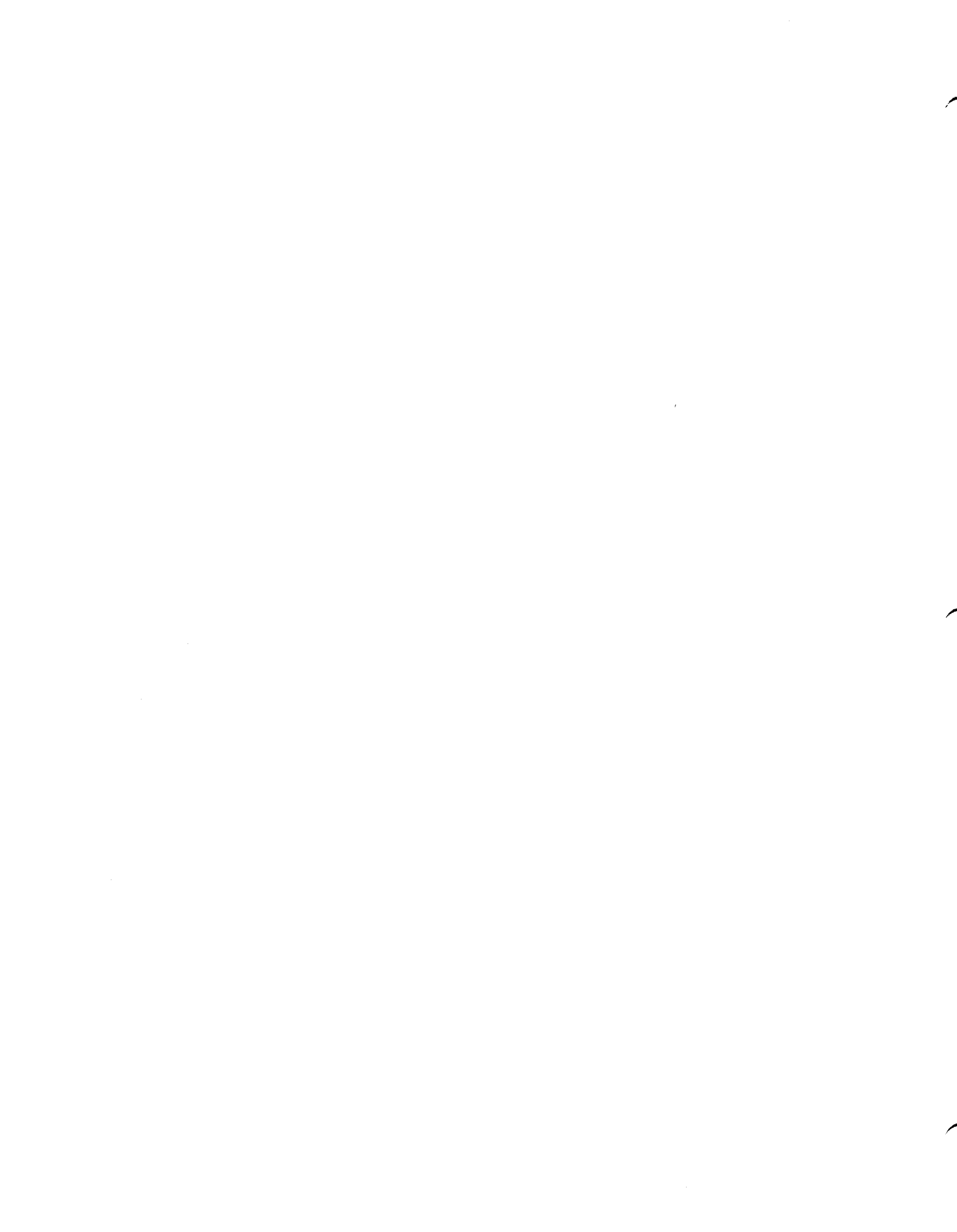
CONTENTS

| | | |
|----------|------------------------|-----|
| ABSTRACT | | vii |
| 1.0 | EQUIPMENT | 1 |
| 1.1 | Memory Requirements | 1 |
| 2.0 | OPERATING INSTRUCTIONS | 1 |
| 2.1 | Loading | 1 |
| 2.2 | Dialog | 2 |
| 3.0 | ERROR MESSAGES | 4 |
| 4.0 | DETAILS OF OPERATION | 5 |



ABSTRACT

The TD8E DEctape Copy program is used to copy any properly formatted DEctape, using a PDP-8/E or PDP-8/M with TD8E control and dual TU56 tape drives. An integral number of blocks may be copied from any position on the input tape to any position on the output tape, or the entire tape may be copied.



TD8E DECTAPE COPY

1.0 EQUIPMENT

TD8E Copy requires a PDP-8/E or PDP-8/M with 4K of read/write memory, a terminal, a TD8E control, and at least one dual TU56 tape drive. The program will also support up to 32K of memory and up to 4 dual TU56 drives. The DECTapes to be used should have been formatted by any of Digital Equipment Corporation's standard DECTape formatters.

1.1 Memory Requirements

TD8E Copy occupies locations 0-2577 of field 0. The remainder of field 0 is used for buffer space. Additional fields may also be used for buffer space, increasing the speed at which a tape may be copied. Locations 7600-7777 of each field are reserved. The starting address is 00200.

2.0 OPERATING INSTRUCTIONS

2.1 Loading

TD8E Copy may be loaded with the Binary Loader (DEC-08-LBAA) or with the Self-starting Binary Loader (DEC-8E-XBINA). The loading procedures for these loaders are in Introduction to Programming 1972, chapter 5. The binary tape is supplied with a starting address at the end, so that it will be started automatically if it is loaded with the Self-starting Binary Loader. The terminal control must be set to LINE before loading.

If the tape is loaded with the Binary Loader, it must be started with the programmer's console according to the following procedure:

1. Set the terminal control to LINE.
2. Set the switch register to 0200.
3. Press EXT D ADDR LOAD and ADDR LOAD.
4. Press CLEAR and then CONT.

The following procedure will load and save TD8E Copy using OS/8. The procedure assumes that the TD8E Copy binary is on paper tape and will be saved on the system device.

```
.R ABSLDR
*PTR:$↑
($ means ALTMODE)
(type any character to continue
transfer)

.SAVE SYS TDCOPY 0-2577;00200=3001
```

The program may then be called and used with:

```
.R TDCOPY
```

2.2 Dialog

TD8E Copy is controlled through a dialog on the terminal. The responses to the questions are in the form of octal numbers followed by a carriage return. Where more than one answer is required to a question, the answers are separated by semicolons. Alphabetic or other illegal characters will cause an error message to be generated and the question to be repeated. If too many digits are typed for the response expected, only the last ones typed will be used. If the response was to be either 0 or 1 (YES or NO), a non-zero final digit will be assumed to be 1.

Before answering the dialog's questions, the user must ensure that all the DECTapes involved are mounted on their respective drives. All the drives must be set to REMOTE. The input drive may be set to WRITE LOCK or WRITE ENABLE; all output drives must be set to WRITE ENABLE. No two drives may have the same unit number.

When the routine is first loaded and started, the dialog begins:

```
TD8E COPY
HIGHEST FIELD AVAILABLE:
```

The user responds with the number of the highest field he wishes used for buffer space. This response may allow data to be preserved in any higher field or may make full use of the memory available. This question is asked only once, immediately after the program has been loaded. To change the response, the program must be reloaded. If 4K

of memory is to be used, the response is 0; if 8K, the response is 1, and so forth.

For each set of copies, the dialog is as follows (the user's response is underlined; (CR) means carriage return):

| <u>Dialog</u> | <u>Comments</u> |
|---|---|
| TD8E COPY FROM UNIT: <u>0</u> (CR) | User may specify one unit number. |
| TO UNITS: <u>1;2;3;4;5;6;7</u> (CR) | User may specify up to 7 unit numbers, separated by semicolons. |
| FIRST INPUT BLOCK: <u>100</u> (CR) | User may supply any legal DECTape block number. |
| FIRST OUTPUT BLOCK: <u>200</u> (CR) | User may supply any legal DECTape block number. |
| NUMBER OF BLOCKS TO COPY: <u>50</u> (CR) | User may supply appropriate number of blocks. |
| VERIFY OUTPUT (YES=1,NO=0): <u>1</u> (CR) | |
| <u>0201</u> 12-BIT WORDS PER BLOCK | Determined by program from tape on input unit. |

The block length of all the specified tapes is checked. If any are found to be different from the input tape, the ILLEGAL FORMAT UNIT n error message is generated.

When all specified copies have been finished, the tapes are rewound and the dialog continues:

DONE

REPEAT (YES=1,NO=0):

If there are more tapes to be copied with the same set of specifications, they should be placed on the drives before typing 1 to repeat the previous operation. If a different set of specifications is desired, 0 should be typed to restart the dialog.

Occasionally a TD8E drive will not stop fast enough after the tapes have been rewound and the end of the tape will spin off the reel. If this should happen, the drive may be stopped manually by setting it to OFF and stopping the reel by hand. This will not affect the validity

of the copy. If the dialog does not continue properly after one or more tapes have spun off, the program may be restarted as if it had been loaded with the Binary Loader (see Loading, Section 2.1).

In response to any question in the dialog, the user may type either CTRL/S to restart the dialog at REPEAT (YES=1,NO=0) or CTRL/C to exit to 7600 of field 0. Either CTRL/S or CTRL/C may also be typed during DEctape motion; it will be acknowledged when possible, usually after a small amount of further motion. If CTRL/S is typed during the dialog the response to the REPEAT question should be NO; this option is mainly for cases where a complete set of specifications is already available.

(CTRL/ characters are typed by holding the CONTROL key down while typing the character. The procedure is similar to that used with the SHIFT key on a typewriter. CTRL/ characters are not echoed (printed).)

A special case of the dialog allows the entire input tape to be copied onto the output tape with a minimum of effort. This case eliminates the need to specify the starting block numbers and number of blocks to copy. In this case, the answer to FIRST INPUT BLOCK: is only a carriage return. The shortened dialog will be as follows:

```
TD8E COPY
FROM UNIT:0 (CR)
TO UNITS:1;2;3;4;5;6;7 (CR)
FIRST INPUT BLOCK: (CR)
VERIFY OUTPUT (YES=1,NO=0): 1 (CR)
0201 12-BIT WORDS PER BLOCK
```

The preceding sample dialog will cause the entire tape on unit 0 to be copied onto the other 7 tapes and verified.

3.0 ERROR MESSAGES

ILLEGAL RESPONSE The user's response to the dialog was not correct; for example, an alphabetic character was typed or carriage return was typed before an octal number was given where one was required. The question will be restated and any previous answer ignored. Nothing should be typed until the terminal has stopped printing.

SELECT ERROR UNIT n During attempted data transfer, unit n was not found. The program waits for the user to correct the cause of the error. The user should check to see that:

1. if unit n is an output drive, it is set to WRITE ENABLE.
2. unit n is set to REMOTE.
3. there is only one unit n.
4. all units are set to numbers appropriate to their TD8E internal wiring.

When the cause of the error has been corrected, the user may type CTRL/R to resume transfer or he may type CTRL/S to restart the dialog.

TAPE ERROR BLOCK x UNIT n

During attempted transfer, a parity error or timing error was detected, or too great a block number was requested near block x on the tape on unit n. The tapes are rewound and the dialog is automatically restarted at DONE, REPEAT (YES=1,NO=Ø).

VERIFY ERROR BLOCK x UNIT n

The data on the input tape does not match the data which was written on block x of the output tape on unit n. The user may type CTRL/R to ignore the error and continue with the transfer, CTRL/T to try the last transfer again and continue if the error does not recur, or CTRL/S to restart the dialog.

ILLEGAL FORMAT UNIT n

Either the number of words per block on unit n does not agree with the number of words per block on the input unit or, when the number of blocks on the tape was calculated from the block length of the input tape, the length was found to be illegal. The number of blocks is only calculated if the whole tape copy option is requested. In either case, when the error has been corrected, the user may type CTRL/R to check the formats of all tapes again and continue, or CTRL/S to restart the dialog.

4.0 DETAILS OF OPERATION

After the answers to the dialog have been stored, the following procedure is used:

1. The number of words per block is determined from the input tape. All output tapes are checked to see if they have the same format as the input tape. If the shortened dialog option was used, the number of blocks on the tape is determined using the formula:

of blocks = $(636,160 / (\text{words per block} + 17)) + 2$ (octal)

or

of blocks = $(212,080 / (\text{words per block} + 15)) + 2$ (decimal)

2. The response to the VERIFY question is checked. The copying loop is set up to verify or not, as was requested.

3. The loop is entered which copies the input tape, using the same set of specifications for each output tape.
 - a. The buffers are filled from the input tape.
 - b. All output tapes are written with the contents of the buffers.
 - c. If verification was requested, a separate set of buffers is filled from the output tape and the two sets of buffers are compared. If there are any discrepancies a VERIFY ERROR has occurred.
 - d. If more blocks remain to be copied, the loop is entered again.
4. When all the specified blocks have been copied onto the output tapes, all the tapes are rewound.
5. The REPEAT option is offered.

The number of fields to be used for buffer space is determined immediately after loading. As soon as the question has been answered, it is removed from the program.

If the output tape is to be verified, each available field, including that part of field 0 not occupied by the program, is divided in half. The lower half is used as the input and output buffer; the upper half is used for verification. The output tape is read back into the upper half and the two halves are compared. If they are not identical, a VERIFY ERROR has occurred.

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 1

/TD8E DECTAPE COPY, V4

/

/

/

/

/

/

//

/

/

/

/

/COPYRIGHT (C) 1972, 1975

/DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

/

/

/

/THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON A
/SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH THE INCLU-
/SION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE, OR ANT OTHER
/COPIES THEREOF, MAY NOT BR PROVIDED OR OTHERWISE MADE AVAILABLE
/TO ANY OTHER PERSON EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE WHO
/AGREES TO THESE LICENSE TERMS. TITLE TO AND OWNERSHIP OF THE
/SOFTWARE SHALL AT ALL TIMES REMAIN IN DEC.

/

/

/THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT
/NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
/EQUIPMRNT COROPATION.

/

/DEC ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
/SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DEC.

/

/

/

/

/

/

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 2

/DEFINITIONS FOR PAL8 AND PAL10

7002 BSW=7002
7421 MQL=7421
7501 MQA=7501
7621 CAM=7621
7521 SWP=7521
7701 ACL=7701
6007 CAF=6007
6203 CDI=6203
6030 KCF=6030
6771 SDSS=6771
6772 SDST=6772
6773 SDSQ=6773
6774 SDLC=6774
6775 SDLD=6775
6776 SDRC=6776
6777 SDRD=6777
FIXTAB

7402 HALT=HLT

/UNIT NUMBER DEFINITIONS FOR TD8E IOT'S

0770 UNIT01=0770
0760 UNIT23=0760
0750 UNIT45=0750
0740 UNIT67=0740

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 3

7600 LIMIT=7600

0011 *11

00011 0000 X11, 0
 00012 0000 X12, 0

/PAGE 0 CONSTANTS AND VARIABLES

0020 *20
 00020 0000 INPUT, 0 /INPUT UNIT CONSTANT
 00021 0000 OUTPUT, 0 /OUTPUT UNIT CONSTANTS
 00022 0000 0
 00023 0000 0
 00024 0000 0
 00025 0000 0
 00026 0000 0
 00027 0000 0
 00030 0000 OCOUNT, 0 /NUMBER OF OUTPUT UNITS SPECIFIED
 00031 0000 OPOINT, 0
 00032 0020 LIST, OUTPUT-1
 00033 0000 OUTNUM, 0
 00034 0000 IBLOCK, 0 /STARTING INPUT BLOCK
 00035 0000 OBLOCK, 0 /STARTING OUTPUT BLOCK
 00036 0000 NUMBER, 0 /NUMBER OF BLOCKS TO TRANSFER
 00037 0000 FIELDS, 0 /-(HIGHEST FIELD AVAILABLE)
 00040 0000 COUNT, 0 /TEMPORARY COUNTERS
 00041 0000 COUNT1, 0 / "
 00042 0000 COUNT2, 0 / "
 00043 0000 COUNT3, 0 / "
 00044 0000 COUNT4, 0 / "
 00045 0000 UNIT, 0 /UNIT CONSTANT--THIS TRANSFER
 00046 0000 VERF, 0 /VERIFY SWITCH (1=YES,0=NO)
 00047 0000 WDCNT, 0 /-(NUMBER OF WORDS PER BLOCK)
 00050 0000 RW, 0 /READ/WRITE BIT--THIS TRANSFER
 00051 0000 FLD0, 0 /# OF BLOCKS IN FIELD 0 BUFFER
 00052 0000 FLDN, 0 /# OF BLOCKS IN FIELD N BUFFER
 00053 0000 BUF0, 0 /START OF FIELD 0 BUFFER
 00054 0000 BUFN, 0 /START OF FIELD N BUFFER
 00055 0000 XNUMB, 0 /# OF BLOCKS LEFT TO TRANSFER
 00056 0000 BLOCKN, 0 /STARTING BLOCK NUMBER--THIS TRANSFER
 00057 0000 NUMB1, 0
 00060 0000 NUMB2, 0
 00061 0000 VB, 0
 00062 0000 END0, 0 /BEGINNING OF FIELD 0 VERIFY BUFFER
 00063 0000 ENTRY, 0 /ENTRY TO TD8E HANDLER
 00064 0000 INB, 0
 00065 0000 OUTB, 0
 00066 0000 OHOLD, 0

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 4

| | | |
|-------|------|--|
| 00067 | 2404 | MESSG1, TEXT @TD8E COPY V4A@ |
| 00070 | 7005 | |
| 00071 | 4003 | |
| 00072 | 1720 | |
| 00073 | 3140 | |
| 00074 | 2664 | |
| 00075 | 0100 | |
| 00076 | 4061 | MESSG3, TEXT @ 12-BIT WORDS PER BLOCK@ |
| 00077 | 6255 | |
| 00100 | 0211 | |
| 00101 | 2440 | |
| 00102 | 2717 | |
| 00103 | 2204 | |
| 00104 | 2340 | |
| 00105 | 2005 | |
| 00106 | 2240 | |
| 00107 | 0214 | |
| 00110 | 1703 | |
| 00111 | 1300 | |

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 5

```

0200 *200

00200 6046 START, TLS
00201 4317 JMS CRLF
00202 4577 JMS I [MESSGE
00203 0067 MESSG1 /@TD8E COPY@
00204 4317 JMS CRLF
00205 3040 DCA COUNT
00206 5576 JMP I [END /ONCE ONLY CODE FOR MULTIPLE FIELD TEST
00207 4336 START1, JMS QUEST
00210 2457 MESSG4 /@FROM UNIT:@
00211 7521 SWP
00212 4352 JMS UNITNO /MAKE UNIT NUMBER CONSTANT
00213 3020 DCA INPUT
00214 1032 TAD LIST
00215 3031 DCA OPOINT
00216 7410 SKP
00217 4325 AGAIN, JMS ERR4 /*ILLEGAL RESPONSE*
00220 3030 DCA OCOUNT
00221 3040 DCA COUNT
00222 4577 JMS I [MESSGE
00223 2465 MESSG5 /@TO UNITS:@
00224 4575 MORE2, JMS I [ANSWER
00225 5217 JMP AGAIN
00226 7701 ACL
00227 4352 JMS UNITNO /MAKE UNIT NUMBER CONSTANT
00230 7421 MQL /STORE IN MQ
00231 7501 MQA /RESTORE TO AC
00232 7041 CIA
00233 1020 TAD INPUT
00234 7650 SNA CLA /IS OUTPUT UNIT = INPUT UNIT ?
00235 5225 JMP MORE2+1 /YES--ERROR
00236 2031 ISZ OPOINT
00237 2030 ISZ OCOUNT /COUNT ONE MORE OUTPUT UNIT
00240 1030 TAD OCOUNT
00241 1174 TAD [-10
00242 7710 SPA CLA /WERE MORE THAN 7 UNITS SPECIFIED?
00243 5246 JMP .+3
00244 4317 JMS CRLF /YES--CARRIAGE RETURN
00245 5255 JMP OALL /IGNORE EXTRA ONE
00246 7521 SWP /NO--
00247 3431 DCA I OPOINT /STORE UNIT CONSTANT IN LIST
00250 1040 TAD COUNT
00251 7041 CIA
00252 1030 TAD OCOUNT
00253 7750 SPA SNA CLA /ALL UNITS IN?
00254 5224 JMP MORE2 /YES

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 6

```

00255 3040 OALL, DCA COUNT
00256 4577 JMS I [MESSGE
00257 1040 MESSG6 /@FIRST INPUT BLOCK:@
00260 4575 JMS I [ANSWER
00261 5306 JMP WHOLE /COPY WHOLE TAPE
00262 1040 TAD COUNT
00263 7650 SNA CLA /WERE TOO MANY SPECIFIED?
00264 5271 JMP .+5
00265 4577 JMS I [MESSGE /YES
00266 2511 ERROR4 /@ILLEGAL RESPONSE@
00267 4317 JMS CRLF
00270 5255 JMP OALL /REPEAT THE QUESTION
00271 7701 ACL /NO
00272 3034 DCA IBLOCK /STORE
00273 4336 JMS QUEST
00274 1052 MESSG7 /@FIRST OUTPUT BLOCK:@
00275 7701 ACL
00276 3035 DCA OBLOCK
00277 4336 JMS QUEST
00300 1064 MESSG8 /@NUMBER OF BLOCKS TO COPY:@
00301 7701 ACL
00302 7450 SNA /WERE 0 BLOCKS SPECIFIED?
00303 5345 JMP QUEST1 /YES--REPEAT QUESTION
00304 3036 DCA NUMBER
00305 5311 JMP .+4
00306 3034 WHOLE, DCA IBLOCK
00307 3035 DCA OBLOCK
00310 3036 DCA NUMBER /0 MEANS WHOLE TAPE
00311 4336 JMS QUEST
00312 1101 MESSG9 /@VERIFY OUTPUT (YES=1,NO=0):@
00313 7701 ACL
00314 0173 AND [7
00315 3046 DCA VERF
00316 5777 JMP I (SETUP

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 7

/OUTPUT CARRIAGE RETURN/LINE FEED

```
00317 0000 CRLF, 0
00320 1376 TAD (215
00321 4572 JMS I [TYPE
00322 1171 TAD [212
00323 4572 JMS I [TYPE
00324 5717 JMP I CRLF /--RETURN--

00325 0000 ERR4, 0
00326 4577 JMS I [MESSGE
00327 2511 ERROR4 /@ILLEGAL RESPONSE@
00330 4317 JMS CRLF /OUTPUT CARRIAGE RETURN/LINE FEED
00331 1170 TAD [-4
00332 1325 TAD ERR4
00333 3325 DCA ERR4
00334 3040 DCA COUNT
00335 5725 JMP I ERR4 /--RETURN--
```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 8

```

00336 0000 QUEST, 0
00337 1736 TAD I QUEST
00340 3343 DCA MNUM
00341 2336 ISZ QUEST
00342 4577 JMS I [MESSGE
00343 0000 MNUM, 0
00344 4575 JMS I [ANSWER
00345 4325 QUEST1, JMS ERR4
00346 1040 TAD COUNT
00347 7640 SZA CLA
00350 5345 JMP QUEST1
00351 5736 JMP I QUEST /--RETURN--

```

```

/CONVERT UNIT NUMBER TO A WORD OF THE FORM
/000 XXX XXX 000 OR
/100 XXX XXX 000
/WHERE XY0 IS THE THIRD DIGIT OF THE IOT
/AND 0 OR 1 REFLECTS THE TD8E UNIT NUMBER
/ENTER WITH THE UNIT NUMBER IN THE AC
/EXIT WITH SPECIAL CODE IN AC

```

```

00352 0000 UNITNO, 0
00353 0173 AND [7 /MASK OUT ALL EXTRANEIOUS BITS
00354 7110 CLL RAR /SAVE 0/1 BIT IN LINK
00355 7421 MQL /STORE ROTATED WORD, CLEAR AC
00356 7010 RAR
00357 7521 SWP /PRESERVE 0/1 BIT IN MQ
00360 1365 TAD TABX /GET DEVICE NUMBER CORRECTLY
00361 3317 DCA CRLF
00362 1717 TAD I CRLF
00363 7501 MQA /OR IN 0/1 BIT
00364 5752 JMP I UNITNO /--RETURN--

```

```

00365 2435 TABX, UNITS
/SKIP 4 LINES AND FETCH MARK TRACK

```

```

00366 0000 SKIPQ, 0
00367 6773 IOTR5, SDSQ
00370 5367 JMP .-1
00371 6776 IOTR6, SDRC
00372 5766 JMP I SKIPQ /--RETURN--

```


/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 9

00376 0215
 00377 0600
 0400 *400

/USER RESPONSE HANDLER
 /USES MQ FOR TEMPORARY STORAGE
 /EXIT WITH RESPONSE IN MQ
 /EXIT TO CALL+1 IF JUST CARRIAGE RETURN
 /OR ILLEGAL CHARACTER, CARRIAGE RETURN
 /OR ;,CARRIAGE RETURN
 /EXIT TO CALL+2 IF GOOD DATA, CARRIAGE RETURN
 /INCREMENT COUNT AND EXIT TO CALL+2 IF GOOD DATA;
 /ILLEGAL CHARACTERS CAUSE WHOLE ANSWER TO BE IGNORED
 /AND EXIT TO CALL+1

00400 0000 ANSWER, 0
 00401 7621 CAM /CLEAR AC AND MQ
 00402 1236 TAD CLEAR
 00403 3212 DCA SWITCH
 00404 4250 MORE, JMS LISTEN /FETCH A CHARACTER
 00405 1377 TAD (-215
 00406 7440 SZA /IS IT A CARRIAGE RETURN?
 00407 5214 JMP .+5 /NO
 00410 1171 TAD [212 /YES--OUTPUT LINE FEED
 00411 4327 JMS TYPE
 00412 7000 SWITCH, NOP /SET UP EXIT ADDRESS
 00413 5600 JMP I ANSWER /--RETURN--
 00414 1376 TAD (215-260
 00415 7510 SPA /IS CHARACTER LESS THAN 260?
 00416 5240 JMP BAD /YES--ILLEGAL CHARACTER
 00417 1174 TAD [260-270 /NO
 00420 7500 SMA /IS IT MORE THAN 269?
 00421 5243 JMP SEMI /YES--CHECK FOR SEMICOLON
 00422 1375 TAD (270 /RESTORE CHARACTER
 00423 0173 AND [7 /MASK OUT EXTRANEIOUS BITS
 00424 7100 CLL
 00425 7521 SWP
 00426 0374 AND (777 /MASK OUT FIRST DIGIT IF THERE ARE 4
 00427 7004 RAL /ROTATE 3 LEFT
 00430 7006 RTL
 00431 7501 MQA /FETCH NEW CHARACTER
 00432 7421 MQL /STORE RESULT IN MQ
 00433 1237 TAD SKIP /SET UP TO SKIP ON RETURN
 00434 3212 DCA SWITCH
 00435 5204 JMP MORE /FETCH ANOTHER

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 10

```

00436 7000 CLEAR, NOP
00437 2200 SKIP, ISZ ANSWER

00440 7200 BAD, CLA /ILLEGAL CHARACTER
00441 4567 JMS I [CRLF
00442 5600 JMP I ANSWER /--RETURN--

/TEST FOR SEMICOLON

00443 1373 SEMI, TAD (270-273
00444 7640 SZA CLA /IS CHARACTER A SEMICOLON?
00445 5240 JMP BAD /NO--ILLEGAL CHARACTER
00446 2040 ISZ COUNT /YES--INCREMENT COUNTER
00447 5212 JMP SWITCH /EXIT FROM ANSWER ROUTINE

/TELETYPE INPUT AND ECHO HANDLER

00450 0000 LISTEN, 0
00451 6031 KSF
00452 5251 JMP .-1
00453 4566 JMS I [PARITY
00454 6046 TLS /ECHO CHARACTER
00455 4257 JMS CHECK /CHECK FOR CTRL/C AND CTRL/S
00456 5650 JMP I LISTEN /--RETURN--

/CHECK FOR CTRL/C AND CTRL/S
/ENTER WITH INPUT CHARACTER IN AC
/EXIT TO HANDLER OR WITH CHARACTER IN AC

00457 0000 CHECK, 0
00460 1372 TAD (-203
00461 7450 SNA /IS IT CTRL/C?
00462 5670 JMP I CTRLC /YES--HANDLE IT
00463 1371 TAD (203-223
00464 7450 SNA /IS IT CTRL/S?
00465 5565 JMP I [REPEAT /YES--HANDLE IT
00466 1370 TAD (223 /RESTORE CHARACTER
00467 5657 JMP I CHECK /--RETURN--

00470 7600 CTRLC, LIMIT

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 11

/MESSAGE OUTPUT HANDLER

/EXPECTS MESSAGE ADDRESS TO BE IN LOCATION AFTER CALL
/EXITS TO CALL+2

```

00471 0000 MESSGE, 0
00472 1671      TAD I MESSGE
00473 3324      DCA FINDER      /SET UP POINTER
00474 2271      ISZ MESSGE
00475 3325      DCA LOC      /SET L/R SWITCH TO L (EVEN)
00476 1724 LNEXT, TAD I FINDER /GET WORD
00477 7002      BSW
00500 0164 RHALF, AND [77
00501 7450      SNA      /IS CHARACTER 0 (TERMINATOR)?
00502 5671      JMP I MESSGE /YES--RETURN--
00503 3326      DCA CHAR
00504 1326      TAD CHAR
00505 0367      AND (40
00506 7650      SNA CLA      /IS IT A LETTER?
00507 1163      TAD [100     /YES--301-337
00510 1162      TAD [200     /NO--240-277
00511 1326      TAD CHAR      /RESTORE CHARACTER
00512 4327      JMS TYPE     /OUTPUT IT
00513 2325      ISZ LOC
00514 1325      TAD LOC
00515 7010      RAR
00516 7630      SZL CLA      /WHICH HALF WAS THAT?
00517 5322      JMP .+3
00520 2324      ISZ FINDER   /RIGHT
00521 5276      JMP LNEXT
00522 1724      TAD I FINDER   /LEFT
00523 5300      JMP RHALF
00524 0000 FINDER, 0
00525 0000 LOC, 0
00526 0000 CHAR, 0

```

/TELETYPE OUTPUT ROUTINE

```

00527 0000 TYPE, 0
00530 6041      TSF
00531 5330      JMP .-1
00532 6046      TLS
00533 7200      CLA
00534 5727      JMP I TYPE     /--RETURN--

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 12

/INSERT IOT'S ACCORDING TO TABLES
 /UNIT CONTAINS APPROPRIATE UNIT CODE
 /COUNT CONTAINS -(NUMBER OF IOT'S TO TRANSFER)
 /COUNT1 CONTAINS ADDRESS OF ADDRESS TABLE
 /UNIT CONTAINS UNIT CODE OF CURRENT UNIT

```

00535 0000 INSERT, 0
00536 1441      TAD I COUNT1
00537 3043      DCA COUNT3
00540 1045      TAD UNIT
00541 7421      MQL
00542 1443      TAD I COUNT3      /MAKE NEW IOT
00543 0366      AND (7007
00544 7501      MQA
00545 7041      CIA
00546 1443      TAD I COUNT3      /COMPARE WITH IOT FROM PROGRAM
00547 7650      SNA CLA      /ARE THE IOT'S THE SAME AS THE LAST UNIT?
00550 5735      JMP I INSERT      /YES--RETURN--
00551 1441      INS1, TAD I COUNT1
00552 3043      DCA COUNT3
00553 1443      TAD I COUNT3      /GET IOT FROM PROGRAM
00554 0366      AND (7007      /RETAIN ONLY SIGNIFICANT BITS
00555 7501      MQA      /OR IN UNIT NUMBER
00556 3443      DCA I COUNT3      /PUT IT IN PROGRAM
00557 2041      ISZ COUNT1      /BUMP COUNTERS
00560 2040      ISZ COUNT      /DONE YET?
00561 5351      JMP INS1      /NO
00562 5735      JMP I INSERT      /YES--RETURN--

00566 7007
00567 0040
00570 0223
00571 7760
00572 7575
00573 7775
00574 0777
00575 0270
00576 7735
00577 7563
0600 PAGE

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 13

/COUNT THE NUMBER OF WORDS PER BLOCK
 /PLACE IT IN MWORDS
 /BE SURE ALL TAPES MATCH INPUT FORMAT

```

00600 1032 SETUP,  TAD LIST
00601 3031          DCA OPOINT      /SET POINTER TO I/O LIST
00602 3042          DCA COUNT2     /CLEAR COUNTER
00603 1030          TAD OCOUNT
00604 7040          CMA
00605 3033          DCA OUTNUM     /SET # OF UNITS
00606 1377 SET4,   TAD (TABLE1-END1-1 /SET UP COUNTERS FOR IOT FIX
00607 3040          DCA COUNT
00610 1376          TAD (TABLE1
00611 3041          DCA COUNT1
00612 1431          TAD I OPOINT
00613 3045          DCA UNIT
00614 4561 JMS I [INSERT /PUT THE PROPER IOT'S IN THE FOLLOWING
ROUTINE
00615 3047          DCA WDCNT     /CLEAR WORD COUNT
00616 1045          TAD UNIT
00617 0160          AND [4000
00620 1375          TAD (2000
00621 6774 IOTX7,  SDLC
00622 7200          CLA
00623 6776 IOTX8,  SDRC
00624 7006          RTL
00625 7430          SZL           /DOES UNIT EXIST?
00626 5231          JMP .+3       /YES
00627 4557 SELERR, JMS I [ERR3    /@SELECT ERROR UNIT N@
00630 5200          JMP SETUP
00631 0374          AND (400
00632 7640          SZA CLA       /TURNED ON?
00633 5227          JMP SELERR    /NO
00634 1045          TAD UNIT      /GET 0 OR 1 UNIT BIT (0 OR 4000)
00635 0160          AND [4000
00636 1156          TAD [1000    /GET GO BIT
00637 6774 IOTX1,  SDLC         /START READING FORWARD
00640 4353          JMS SKIP4     /SKIP 8 LINES TO AVOID GARBAGE
00641 4353          JMS SKIP4
00642 6771 IOTX3,  SDSS         /LOOK FOR FORWARD BLOCK NUMBER (26)
00643 5242          JMP .-1
00644 6776 IOTX4,  SDRC
00645 0164          AND [77
00646 1373          TAD (-26
00647 7640          SZA CLA       /FOUND YET?
00650 5242          JMP IOTX3     /NO--KEEP LOOKING

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 14

```

00651 4353 SET2, JMS SKIP4 /YES--START COUNTING LINES BY FOURS
00652 2047 ISZ WDCNT
00653 7000 NOP
00654 0164 AND [77
00655 1372 TAD (-51
00656 7640 SZA CLA /FOUND GUARD YET?
00657 5251 JMP SET2 /NO
00660 1045 TAD UNIT /YES
00661 0160 AND [4000
00662 6774 IOTX2, SDLC /STOP UNIT
00663 7200 CLA
00664 1042 TAD COUNT2
00665 7440 SZA /IS THIS THE INPUT UNIT?
00666 5274 JMP SET5 /NO
00667 1371 TAD (-11 /YES--SAVE THE COUNT
00670 1047 TAD WDCNT
00671 7041 CIA
00672 3042 DCA COUNT2
00673 5300 JMP SET3 /FIRST OUTPUT UNIT
00674 1371 SET5, TAD (-11 /NOT INPUT UNIT
00675 1047 TAD WDCNT
00676 7640 SZA CLA /SAME NUMBER OF WORDS AS INPUT UNIT?
00677 5344 JMP ERR5 /NO*ILLEGAL FORMAT*
00700 2031 SET3, ISZ OPOINT /NEXT UNIT
00701 2033 ISZ OUTNUM /DONE YET?
00702 5206 JMP SET4 /NO
00703 1042 TAD COUNT2 /YES--PRINT MESSAGE
00704 3555 DCA I [MWORDS /SET UP NUMBER OF WORDS PER BLOCK
00705 1555 TAD I [MWORDS
00706 7041 CIA
00707 4554 JMS I [PRINT /PRINT 4 DIGIT NUMBER OF BLOCKS
00710 4577 JMS I [MESSGE /YES--PRINT REST OF MESSAGE
00711 0076 MESSG3
00712 4567 JMS I [CRLF

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 15

/IF WHOLE TAPE IS TO BE COPIED, IT IS NECESSARY TO
 /COMPUTE THE NUMBER OF BLOCKS ON THE TAPE (NB)
 /USING THE NUMBER OF WORDS PER BLOCK (WB)
 /AND THE FORMULA:
 /OCTAL: NB=[63 6160/(WB+17)]+2
 /DECIMAL: NB=[212,080/(WB+15)]+2

```

00713 1036          TAD NUMBER
00714 7640          SZA CLA           /COPY WHOLE TAPE?
00715 5770          JMP VERFQ         /NO--
00716 3040          DCA COUNT         /YES--COMPUTE NUMBER OF BLOCKS ON TAPE
00717 1555          TAD I [MWORDS
00720 7041          CIA
00721 1367          TAD (17
00722 3041          DCA COUNT1        /GET NUMBER OF WORDS PER BLOCK+17
00723 1366          TAD (-64
00724 3042          DCA COUNT2
00725 1365          TAD (-6160
00726 7100 SUB,     CLL
00727 1041          TAD COUNT1
00730 2040          ISZ COUNT         /COUNT A BLOCK--TOO MANY?
00731 7410          SKP               /NO
00732 5344          JMP ERR5         /YES--ERROR
00733 7430          SZL
00734 2042          ISZ COUNT2
00735 5326          JMP SUB
00736 7300          CLA CLL
00737 1040          TAD COUNT         /COUNT IS [63 6160/WB+17]+1
00740 7001          IAC               /ADD 1 MORE
00741 3036          DCA NUMBER        /STORE AS # OF BLOCKS TO TRANSFER
00742 5743          JMP I .+1
00743 1000          VERFQ

00744 7200 ERR5,   CLA
00745 4577          JMS I [MESSGE
00746 2472          ERROR5           /*ILLEGAL FORMAT UNIT*
00747 4553          JMS I [DECODE     /PRINT UNIT NUMBER
00750 4552          JMS I [CTRLR     /WAIT FOR CTRL/R
00751 4567          JMS I [CRLF      /CARRIAGE RETURN/LINE FEED
00752 5200          JMP SETUP        /TRY AGAIN

/READ FOUR LINES AND FETCH MARK TRACK

00753 0000 SKIP4,  0
00754 6773 IOTX5,  SDSQ
00755 5354          JMP .-1
00756 6776 IOTX6,  SDRC
00757 5753          JMP I SKIP4      /--RETURN--

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 16

00765 1620
 00766 7714
 00767 0017
 00770 1000
 00771 7767
 00772 7727
 00773 7752
 00774 0400
 00775 2000
 00776 2447
 00777 7770

1000 PAGE

/IS TAPE TO BE VERIFIED?
 /SET UP DEPENDING ON RESPONSE

01000 1046 VERFQ, TAD VERF
 01001 7640 SZA CLA /VERIFY?
 01002 5210 JMP YES /YES--
 01003 1377 TAD (NOP /NO--
 01004 3616 DCA I VERF1A
 01005 1376 TAD (OUTN
 01006 3617 DCA I VERF2A
 01007 5214 JMP CONT
 01010 1375 YES, TAD (RAR
 01011 3616 DCA I VERF1A
 01012 1374 TAD (VERIFY
 01013 3617 DCA I VERF2A
 01014 5615 CONT, JMP I .+1
 01015 1200 DOIT

01016 1311 VERF1A, VERF1
 01017 1274 VERF2A, VERF2

/WAIT FOR CTRL/R

01020 0000 CTRLR, 0
 01021 4551 JMS I [LISTEN /FETCH CHARACTER
 01022 1150 TAD [-222
 01023 7640 SZA CLA /IT IT CTRL/R?
 01024 5221 JMP .-3 /NO--WAIT FOR ONE
 01025 4567 JMS I [CRLF /CARRIAGE RETURN/LINE FEED
 01026 5620 JMP I CTRLR /--RETURN--

01027 3040 REPEAT, DCA COUNT
 01030 4567 JMS I [CRLF
 01031 4547 JMS I [QUEST /@REPEAT (YES=1;NO=0):@
 01032 1122 MESS11
 01033 7701 ACL
 01034 0173 AND [7
 01035 7640 SZA CLA
 01036 5546 JMP I [CLEAN /YES
 01037 5545 JMP I [START+4 /NO--RESTART

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE

PAGE 17

```
01040 0611 MESSG6, TEXT @FIRST INPUT BLOCK:@
01041 2223
01042 2440
01043 1116
01044 2025
01045 2440
01046 0214
01047 1703
01050 1372
01051 0000
01052 0611 MESSG7, TEXT @FIRST OUTPUT BLOCK:@
01053 2223
01054 2440
01055 1725
01056 2420
01057 2524
01060 4002
01061 1417
01062 0313
01063 7200
01064 1625 MESSG8, TEXT @NUMBER OF BLOCKS TO COPY:@
01065 1502
01066 0522
01067 4017
01070 0640
01071 0214
01072 1703
01073 1323
01074 4024
01075 1740
01076 0317
01077 2031
01100 7200
01101 2605 MESSG9, TEXT @VERIFY OUTPUT (YES=1,NO=0):@
01102 2211
01103 0631
01104 4017
01105 2524
01106 2025
01107 2440
01110 5031
01111 0523
01112 7561
01113 5416
01114 1775
01115 6051
01116 7200
01117 0417 MESS10, TEXT @DONE@
01120 1605
01121 0000
01122 2205 MESS11, TEXT @REPEAT (YES=1,NO=0):@
01123 2005
01124 0124
01125 4050
```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE

PAGE 17-1

```
01126 3105
01127 2375
01130 6154
01131 1617
01132 7560
01133 5172
01134 0000
01135 2605 ERROR1, TEXT @VERIFY ERROR BLOCK @
01136 2211
01137 0631
01140 4005
01141 2222
01142 1722
01143 4002
01144 1417
01145 0313
01146 4000
01147 2401 ERROR2, TEXT @TAPE ERROR BLOCK @
01150 2005
01151 4005
01152 2222
01153 1722
01154 4002
01155 1417
01156 0313
01157 4000
01160 2305 ERROR3, TEXT @SELECT ERROR UNIT @
01161 1405
01162 0324
01163 4005
01164 2222
01165 1722
01166 4025
01167 1611
01170 2440
01171 0000
```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 18

```

01174 1400
01175 7010
01176 1246
01177 7000
      1200 PAGE

      /SETUP FOR ACTUAL READ/WRITE/VERIFY OPERATION
01200 1144 DOIT, TAD [LIMIT-END /SET UP NUMBER OF BLOCKS
01201 4275 JMS DIV1 /IN FIELD 0 BUFFER
01202 3051 DCA FLD0
01203 1371 TAD M200
01204 4275 JMS DIV1 /AND IN FIELD N BUFFERS
01205 3052 DCA FLDN
01206 1034 TAD IBLOCK /SET UP RUNNING COUNTERS AND POINTERS
01207 3064 DCA INB /FOR NEXT INPUT BLOCK
01210 1035 TAD OBLOCK
01211 3065 DCA OUTB /FOR NEXT OUTPUT BLOCK
01212 1036 TAD NUMBER
01213 3057 DCA NUMB1 /FOR NUMBER OF BLOCKS LEFT TO TRANSFER
01214 5220 JMP .+4
01215 1055 ALLDUN, TAD XNUMB
01216 7650 SNA CLA /DONE WITH ALL BLOCKS YET?
01217 5313 JMP REWIND /YES

      /READ---
01220 1032 READX, TAD LIST /NO--SET UP POINTER TO OUTPUT UNITS
01221 3031 DCA OPOINT
01222 1030 TAD OCOUNT
01223 7040 CMA
01224 3033 DCA OUTNUM
01225 1064 TAD INB
01226 3056 DCA BLOCKN
01227 1057 TAD NUMB1 /SET POINTERS FOR TRANSFER
01230 3055 DCA XNUMB
01231 1057 TAD NUMB1
01232 3060 DCA NUMB2 /SAVE COUNTER FOR WRITE
01233 1020 TAD INPUT /SELECT INPUT UNIT
01234 3045 DCA UNIT
01235 3050 DCA RW /SET R/W BIT TO READ

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 19

```

01236 1176          TAD [END          /SET START OF BUFFERS IN CASE
01237 3053          DCA BUF0          /THEY WERE CHANGED BY VERIFY
01240 3054          DCA BUFN
01241 4543          JMS I [READY      /FILL THE BUFFERS
01242 1055          TAD XNUMB        /SAVE THE POINTERS
01243 3057          DCA NUMB1
01244 1056          TAD BLOCKN
01245 3064          DCA INB

01246 2031  OUTN,   ISZ OPOINT
01247 2033          ISZ OUTNUM        /DONE WITH ALL UNITS YET?
01250 5254          JMP .+4          /NO--CONTINUE WRITING
01251 1066          TAD OHOLD        /YES
01252 3065          DCA OUTB
01253 5215          JMP ALLDUN        /READ ANOTHER BUFFER LOAD
01254 1065  WRITEX, TAD OUTB
01255 3066          DCA OHOLD
01256 1066          TAD OHOLD        /WRITE
01257 3056          DCA BLOCKN       /RESET POINTERS
01260 1065          TAD OUTB
01261 3061          DCA VB          /SAVE COUNTER FOR VERIFY
01262 1060          TAD NUMB2
01263 3055          DCA XNUMB
01264 1431          TAD I OPOINT     /SELECT OUTPUT UNIT
01265 3045          DCA UNIT
01266 7330          CLA CLL CML RAR  /AC=4000
01267 3050          DCA RW          /SET R/W BIT TO WRITE
01270 4543          JMS I [READY
01271 1056          TAD BLOCKN
01272 3066          DCA OHOLD
01273 5674          JMP I .+1
01274 1400  VERF2,  VERIFY

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 20

/SEE HOW MANY BLOCKS WILL FIT INTO BUFFER
 /ENTER WITH BUFFER SIZE IN AC
 /EXIT WITH # OF BLOCKS IN AC

```

01275 0000 DIV1, 0
01276 3041 DCA COUNT1
01277 3040 DCA COUNT
01300 1041 TAD COUNT1 /TOTAL WORDS
01301 7100 DIV2, CLL
01302 1555 TAD I [MWORDS /-NUMBER OF WORDS PER BLOCK
01303 7420 SNL /RUN OUT OF ROOM?
01304 5307 JMP .+3 /YES--
01305 2040 ISZ COUNT /NO--COUNT A BLOCK
01306 5301 JMP DIV2
01307 7300 CLA CLL /IGNORE LESS THAN A BLOCK LEFT
01310 1040 TAD COUNT
01311 7010 VERF1, RAR /DIVIDE BY 2 IF VERIFY (NOP IF NO VERIFY)
01312 5675 JMP I DIV1 /--RETURN--

```

/END OF OPERATION
 /REWIND TAPES TO INITIAL END ZONE

```

01313 1030 REWIND, TAD OCOUNT
01314 7040 CMA
01315 3042 DCA COUNT2 /SET NUMBER OF TAPES STILL SPINNING
01316 7240 RLIST, CLA CMA
01317 1032 TAD LIST
01320 3031 DCA OPOINT /SET POINTER TO UNIT LIST
01321 1030 TAD OCOUNT
01322 7001 IAC
01323 7040 CMA
01324 3033 DCA OUTNUM /SET NUMBER OF UNITS IN LIST
01325 4566 RUNIT, JMS I [PARITY
01326 4542 JMS I [CHECK /CHECK TTY FOR CTRL/S OR CTRL/C
01327 2033 ISZ OUTNUM /DONE WITH WHOLE LIST YET?
01330 7610 SKP CLA /NO
01331 5316 JMP RLIST /YES--START THROUGH LIST AGAIN
01332 2031 ISZ OPOINT
01333 1431 TAD I OPOINT /GET UNIT CODE
01334 7006 RTL
01335 7630 SZL CLA /STILL SPINNING?
01336 5325 JMP RUNIT /NO--TRY NEXT TAPE
01337 1431 TAD I OPOINT /YES
01340 3045 DCA UNIT

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 21

```

01341 1141          TAD [-6
01342 3040          DCA COUNT
01343 1140          TAD [RTAB
01344 3041          DCA COUNT1
01345 4561          JMS I [INSERT /PUT PROPER IOT'S IN THIS ROUTINE
01346 1431          TAD I OPOINT
01347 0160          AND [4000 /UNIT/READ
01350 1137          TAD [3000 /REVERSE/GO
01351 6774 IOTR1,   SDLC
01352 4536          JMS I [SKIPQ
01353 4536          JMS I [SKIPQ /WAIT FOR DRIVE TO GET UP TO SPEED
01354 6771 IOTR2,   SDSS
01355 5354          JMP .-1
01356 6776 IOTR3,   SDRC /GET MARK TRACK BITS
01357 0164          AND [77
01360 1135          TAD [-22
01361 7640          SZA CLA /END ZONE?
01362 5325          JMP RUNIT /NO--NEXT UNIT
01363 7332          CLA CLL CML RTR /AC=2000
01364 7501          MQA /UNIT CODE STILL IN MQ FROM INSERT
01365 3431          DCA I OPOINT /SET STOPPED BIT
01366 1431          TAD I OPOINT
01367 0134          AND [6000
01370 6774 IOTR4,   SDLC /STOP UNIT
01371 7600 M200,   7600 /CLA
01372 2042          ISZ COUNT2 /ALL TAPES STOPPED?
01373 5325          JMP RUNIT /NO--NEXT UNIT
01374 4577          JMS I [MESSGE /YES
01375 1117          MESS10 /@DONE@
01376 5565          JMP I [REPEAT

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 22

1400 PAGE

/VERIFICATION ROUTINES

```

01400 1061 VERIFY, TAD VB          /SET POINTERS AND COUNTERS FOR TRANSFER
01401 3056          DCA BLOCKN
01402 1060          TAD NUMB2
01403 3055          DCA XNUMB
01404 3050          DCA RW
01405 1062          TAD END0      /SET BEGINNINGS OF VERIFY BUFFERS
01406 3053          DCA BUF0
01407 1377          TAD (3700
01410 3054          DCA BUFN
01411 1275          TAD CDF0
01412 3271          DCA COMP2
01413 4543          JMS I [READY  /READ VERIFY BUFFERS FULL
01414 1041          TAD COUNT1    /GET # OF BLOCKS IN LAST BUFFER FILLED
01415 3043          DCA COUNT3
01416 7040          CMA           /SET AUTOINDEX POINTERS TO BUFFERS
01417 1176          TAD [END
01420 3011          DCA X11
01421 7140          CMA CLL
01422 1062          TAD END0
01423 3012          DCA X12
01424 1040          TAD COUNT
01425 7040          CMA
01426 1037          TAD FIELDS
01427 3040          DCA COUNT      /SET NUMBER OF FIELDS WHICH WERE FILLED
01430 4255          JMS COMP4     /GET NUMBER OF BLOCKS
01431 1051          TAD FLD0
01432 7041          CIA
01433 3044          DCA COUNT4    /SET COUNTER
01434 4266          JMS COMP      /COMPARE THE BUFFERS

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 23

```

01435 1040 COMP3, TAD COUNT
01436 7650 SNA CLA
01437 5533 JMP I [OUTN
01440 4255 JMS COMP4 /GET NUMBER OF BLOCKS
01441 1052 TAD FLDN
01442 7041 CIA
01443 3044 DCA COUNT4
01444 1271 TAD COMP2 /EACH FIELD-----
01445 1376 TAD (10
01446 3271 DCA COMP2 /SET CDF INSTRUCTION PROPERLY
01447 7140 CMA CLL /SET AUTOINDEX POINTERS TO BUFFERS
01450 3011 DCA X11
01451 1375 TAD (3677
01452 3012 DCA X12
01453 4266 JMS COMP
01454 5235 JMP COMP3 /DO THE NEXT FIELD

```

```

/ENTER WITH AC CLEAR
/EXIT TO CALL+1 WITH AC CLEAR IF
/NORMAL BUFFER FILL
/EXIT TO CALL+2 WITH # OF BLOCKS IN AC IF
/LAST BUFFER

```

```

01455 0000 COMP4, 0
01456 2040 ISZ COUNT /LAST FIELD FILLED?
01457 5655 JMP I COMP4 /NO--RETURN--
01460 1055 TAD XNUMB /YES--OUT OF BLOCKS?
01461 7640 SZA CLA
01462 5655 JMP I COMP4 /NO--RETURN--
01463 1043 TAD COUNT3 /YES--GET ACTUAL # OF BLOCKS
01464 2255 ISZ COMP4 /INCREMENT RETURN ADDRESS
01465 5655 JMP I COMP4 /--RETURN--

```


/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 24

/COMPARE PORTION OF VERIFY ROUTINE

```

01466 0000 COMP, 0
01467 1555 TAD I [MWORDS /SET NUMBER OF WORDS PER BLOCK COUNTER
01470 3042 DCA COUNT2
01471 7402 COMP2, HALT /SHOULD CONTAIN CDF N
01472 1411 TAD I X11 /GET CORRESPONDING WORDS FROM EACH BUFFER
01473 7041 CIA
01474 1412 TAD I X12
01475 6201 CDF0, CDF 0
01476 7640 SZA CLA /DO WORDS MATCH?
01477 5305 JMP ERR1 /NO--VERIFY ERROR
01500 2042 TRY, ISZ COUNT2 /DONE WITH BLOCK?
01501 5271 JMP COMP2 /NO--CONTINUE
01502 2044 ISZ COUNT4 /DONE WITH ALL BLOCKS?
01503 5267 JMP COMP+1 /NO
01504 5666 JMP I COMP /YES--RETURN--

01505 4577 ERR1, JMS I [MESSGE
01506 1135 ERROR1 /*VERIFY ERROR BLOCK *
01507 1044 TAD COUNT4 /GET CURRENT BLOCK NUMBER
01510 7041 CIA
01511 1774 TAD I (BLOCKS /FROM BLOCK THIS OPERATION STARTED WITH
01512 4331 JMS PRINT /PRINT 4 DIGIT BLOCK NUMBER
01513 4577 JMS I [MESSGE
01514 2505 ERROR6 /*UNIT *
01515 4553 JMS I [DECODE /PRINT UNIT NUMBER
01516 4551 WAIT, JMS I [LISTEN /WAIT FOR RESPONSE
01517 3331 DCA PRINT
01520 4567 JMS I [CRLF
01521 1331 TAD PRINT
01522 1132 TAD [-224
01523 7450 SNA /WAS IT CTRL/T?
01524 5531 JMP I [WRITEX /YES--TRY AGAIN
01525 1130 TAD [2
01526 7640 SZA CLA /WAS IT CTRL/R?
01527 5316 JMP WAIT /NO--WAIT FOR A GOOD RESPONSE
01530 5300 JMP TRY /YES--IGNORE AND CONTINUE

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 25

/PRINT A 4 DIGIT OCTAL NUMBER
/ENTER WITH NUMBER IN AC

```

01531 0000 PRINT, 0
01532 3577 DCA I [MESSGE /TEMPORARY STORAGE
01533 1170 TAD [-4
01534 3575 DCA I [ANSWER /SET DIGIT COUNTER
01535 1577 TAD I [MESSGE
01536 7004 RAL
01537 3567 DCA I [CRLF
01540 1567 FOUR, TAD I [CRLF
01541 7004 RAL
01542 7006 RTL
01543 3567 DCA I [CRLF
01544 1567 TAD I [CRLF
01545 0173 AND [7
01546 1127 TAD [260
01547 4572 JMS I [TYPE /PRINT ONE DIGIT
01550 2575 ISZ I [ANSWER /DONE YET?
01551 5340 JMP FOUR /NO
01552 5731 JMP I PRINT /YES--RETURN--

```

/CLEAN UP UNIT TABLES AFTER REWIND

```

01553 1032 CLEAN, TAD LIST
01554 3031 DCA OPOINT
01555 1030 TAD OCOUNT
01556 7040 CMA
01557 3033 DCA OUTNUM /SET POINTER AND COUNTER
01560 1431 CLEAN1, TAD I OPOINT /GET UNIT CODE
01561 0373 AND (4770 /MASK OUT EXTRANEIOUS BITS
01562 3431 DCA I OPOINT /REPLACE IT
01563 2031 ISZ OPOINT
01564 2033 ISZ OUTNUM /DONE YET?
01565 5360 JMP CLEAN1 /NO
01566 5526 JMP I [DOIT /YES--NEXT OPERATION

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 26

01573 4770
 01574 1715
 01575 3677
 01576 0010
 01577 3700
 1600 PAGE

/FILL ALL N FIELDS ONCE
 /ENTER WITH AC CLEAR
 /# OF BLOCKS FOR FIELD 0 IN FLD0
 /# OF BLOCKS FOR OTHERS IN FLDN
 /ADDRESSES OF BUFFERS IN BUF0, BUFN
 /R/W BIT (0 OR 4000) IN RW

01600 0000 READY, 0
 01601 1125 TAD [IOTLOC-TABEND-1
 01602 3040 DCA COUNT
 01603 1124 TAD [IOTLOC
 01604 3041 DCA COUNT1
 01605 4561 JMS I [INSERT /PUT PROPER IOT'S IN HANDLER
 01606 1045 TAD UNIT
 01607 7710 SPA CLA /EVEN OR ODD UNIT NUMBER?
 01610 7107 CLL IAC RTL /ODD
 01611 1123 TAD [ORIGIN /EVEN
 01612 3063 DCA ENTRY /SET UP ENTRY TO HANDLER
 01613 1050 TAD RW
 01614 7421 MQL /STORE UNIT BIT FOR LATER
 01615 7040 CMA
 01616 1037 TAD FIELDS /SET COUNTER FOR # OF FIELDS
 01617 3040 DCA COUNT
 01620 7100 CLL
 01621 1051 TAD FLD0 /ADJUST NUMBER OF BLOCKS TO
 01622 4257 JMS SUB1 /TRANSFER DEPENDING ON NUMBER
 01623 1051 TAD FLD0 /LEFT TO BE TRANSFERRED
 01624 4266 JMS SUB2 /RESET FUNCTION WORD
 01625 1053 TAD BUF0 /SET UP BUFFER POINTERS
 01626 4277 JMS SUB3
 01627 4307 JMS TRANS /TRANSFER DATA--FIELD 0
 01630 2040 ZOOM, ISZ COUNT /BEGINNING OF LOOP FOR EACH FIELD ABOVE 0
 01631 7410 SKP /DONE YET?
 01632 5600 JMP I READY /YES--RETURN--

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 27

```

01633 1037      TAD FIELDS
01634 7141      CIA CLL
01635 1040      TAD COUNT
01636 7001      IAC
01637 7004      RAL
01640 7006      RTL          /GET FIELD SETTING READY
01641 7421      MQL          /STORE IN MQ
01642 1313      TAD FUNCTN   /GET PREVIOUS FUNCTION WORD
01643 0160      AND [4000    /GET R/W BIT
01644 7501      MQA          /OR IN FIELD SETTING
01645 7421      MQL          /STORE
01646 7100      CLL
01647 1052      TAD FLDN     /ADJUST NUMBER OF BLOCKS TO TRANSFER
01650 4257      JMS SUB1
01651 1052      TAD FLDN
01652 4266      JMS SUB2     /AND RESET FUNCTION WORD
01653 1054      TAD BUFN
01654 4277      JMS SUB3     /SET UP BUFFER POINTERS
01655 4307      JMS TRANS    /TRANSFER DATA--FIELDS 1-N
01656 5230      JMP ZOOM     /FILL ANOTHER FIELD

01657 0000      SUB1,      0
01660 7041      CIA
01661 1055      TAD XNUMB
01662 7120      CLL CML     /SET LINK=1
01663 7500      SMA          /ARE THERE LESS BLOCKS LEFT THAN A FIELD
FULL?
01664 3055      DCA XNUMB    /NO--REDUCE COUNT OF BLOCKS LEFT
01665 5657      JMP I SUB1   /YES-TRANSFER BLOCKS LEFT--RETURN--

01666 0000      SUB2,      0
01667 3041      DCA COUNT1   /LINK=1 IF BLOCKS LEFT, 0 IF NONE
01670 7420      SNL          /DONE WITH ALL BLOCKS YET?
01671 3055      DCA XNUMB    /YES--BUMP SWITCH
01672 1041      TAD COUNT1   /NO
01673 7002      BSW
01674 7501      MQA          /PUT # OF BLOCKS INTO FUNCTION WORD
01675 3313      DCA FUNCTN   /START REVERSE
01676 5666      JMP I SUB2   /--RETURN--

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 28

```

01677 0000 SUB3, 0
01700 3314 DCA BUFADD
01701 1056 TAD BLOCKN /SET STARTING BLOCK NUMBER
01702 3315 DCA BLOCKS
01703 1041 TAD COUNT1
01704 1056 TAD BLOCKN
01705 3056 DCA BLOCKN /RESET STARTING BLOCK FOR NEXT TIME
01706 5677 JMP I SUB3 /--RETURN--

```

/CALL TO THE HANDLER

```

01707 0000 TRANS, 0
01710 4566 JMS I [PARITY /CHECK TELETYPE
01711 4542 JMS I [CHECK /WAS ^C OR ^S TYPED?
01712 4463 JMS I ENTRY
01713 0000 FUNCTN, 0 /FUNCTION WORD
01714 0000 BUFADD, 0 /BUFFER ADDRESS
01715 0000 BLOCKS, 0 /STARTING BLOCK NUMBER
01716 5330 JMP ERR /ERROR RETURN
01717 4566 JMS I [PARITY /CHECK TELETYPE
01720 4542 JMS I [CHECK /WAS ^C OR ^S TYPED?
01721 7200 CLA
01722 1055 TAD XNUMB
01723 7640 SZA CLA /DONE YET?
01724 5707 JMP I TRANS /NO--RETURN--
01725 2040 ISZ COUNT
01726 5600 JMP I READY /--RETURN--
01727 5600 JMP I READY /--RETURN--

```

/TRANSFER ERROR HANDLER

```

01730 7650 ERR, SNA CLA /FATAL ERROR?
01731 5343 JMP SELECT /NO
01732 4577 JMS I [MESSGE /YES
01733 1147 ERROR2 /*TAPE ERROR BLOCK *
01734 1777 TAD I (BLOCK
01735 4554 JMS I [PRINT /PRINT BLOCK NUMBER
01736 4577 JMS I [MESSGE
01737 2505 ERROR6 /*UNIT *
01740 4354 JMS DECODE /PRINT UNIT NUMBER
01741 4567 JMS I [CRLF
01742 5522 JMP I [REWIND
01743 4346 SELECT, JMS ERR3
01744 2313 ISZ FUNCTN /TURN AROUND AND TRY AGAIN
01745 5312 JMP FUNCTN-1

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 29

```

01746 0000 ERR3, 0
01747 4577 JMS I [MESSGE
01750 1160 ERROR3 /*SELECT ERROR UNIT *
01751 4354 JMS DECODE /PRINT UNIT NUMBER
01752 4776 JMS CTRLR /WAIT FOR CTRL/R
01753 5746 JMP I ERR3 /--RETURN--

```

```

/DECODE UNIT NUMBER FOR PRINTING
/PRINT UNIT NUMBER BEFORE RETURNING

```

```

01754 0000 DECODE, 0
01755 7100 CLL
01756 1045 TAD UNIT
01757 7004 RAL
01760 7421 MQL /SAVE ROTATED CODE IN MQ
01761 7004 RAL
01762 7521 SWP /SAVE EVEN/ODD BIT IN MQ
01763 7010 RAR /WORK ON IOT CODE
01764 7012 RTR
01765 7001 IAC
01766 7040 CMA
01767 0173 AND [7
01770 7501 MQA /INCLUDE EVEN/ODD BIT
01771 1127 TAD [260 /MAKE ASCII DIGIT
01772 4572 JMS I [TYPE
01773 5754 JMP I DECODE /--RETURN--

```

```

01776 1020
01777 2004
2000 PAGE

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 30

/TD8E DECTAPE HANDLER

/SLIGHTLY MODIFIED VERSION OF DEC-E8-UZTA-D
/COPYRIGHT 1971 DIGITAL EQUIPMENT CORPORATION
/ MAYNARD, MASSACHUSETTS 01754

/THE CALLING SEQUENCE IS:

/ JMS ENTRY
/ FUNCTION WORD
/ BUFFER ADDRESS
/ STARTING BLOCK
/ ERROR RETURN
/ NORMAL RETURN (AC CLEAR)

/FUNCTION WORD:

/ BIT 0: 0=READ, 1=WRITE
/ BITS 1-5: # OF BLOCKS TO BE TRANSFERRED
/ BITS 6-8: FIELD OF BUFFER AREA
/ BITS 9-10: UNUSED
/ BIT 11: 1=START FORWARD, 0=START REVERSE

/ERRORS:

/THE HANDLER DETECTS TWO TYPES OF ERRORS:

/FATAL ERRORS:

/ PARITY ERROR
/ TIMING ERROR
/ TOO GREAT A BLOCK NUMBER

/FATAL ERRORS TAKE ERROR RETURN WITH AC=4000

/NON-FATAL ERROR:

/ SELECT ERROR (IMPROPER UNIT NUMBER OR NO UNIT NUMBER)
/NON-FATAL ERROR TAKES ERROR RETURN WITH AC=0

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 31

2000 PAGE

0000 MFIELD=0

2000 ORIGIN=.

```

02000 0000 DTA0, 0 /ENTRY POINT FROM UNIT 0
02001 7300 CLA CLL /0 TO LINK
02002 5210 JMP DTA1X
02003 1000 C1000, 1000
02004 0000 DTA1, 0 /UNIT 2 ENTRY
02005 7320 CLA CLL CML /1 TO LINK
02006 1204 TAD DTA1
02007 3200 DCA DTA0 /PICK UP ARGS AT DTA0
02010 7010 DTA1X, RAR
02011 3361 DCA YUNIT /LINK TO UNIT POSITION
02012 6214 RDF
02013 1354 TAD C6203 /GET DATA FIELD AND SETUP RETURN
02014 3351 DCA LEAVE
02015 1361 TAD YUNIT /GET FUNCTION WORD
02016 6774 IOT4, SDLC /PUT FUNCTION INTO DATA REGISTER
02017 1600 TAD I DTA0
02020 6775 IOT1, SDLD
02021 7200 CLA
02022 1360 TAD MWORDS
02023 3356 DCA WCOUNT /STORE MASTER WORD COUNT
02024 2200 ISZ DTA0 /TO BUFFER
02025 1600 TAD I DTA0
02026 3357 DCA BUFF
02027 2200 ISZ DTA0 /TO BLOCK NUMBER
02030 1600 TAD I DTA0
02031 3204 DCA BLOCK
02032 2200 ISZ DTA0 /POINT TO ERROR EXIT
02033 6203 CIF CDF MFIELD /TO ROUTINES DATA FIELD
02034 6777 IOT2, SDRD /GET FUNCTION INTO AC
02035 7104 CLL RAL
02036 0305 AND CM200 /GET # PAGES TO XFER
02037 3364 DCA PGCT
02040 6777 IOT3, SDRD
02041 0374 C374, AND C70 /GET FIELD FOR XFER
02042 1353 TAD C6201 /FORM CDF N
02043 3377 DCA XFIELD /IF=0 AND DF=N AT XFER.
02044 7346 CLA CLL CMA RTL
02045 3367 DCA TRYCNT /3 ERROR TRIES
02046 6776 IOT5, SDRC
02047 0366 AND C100
02050 7640 SZA CLA
02051 5345 JMP FATAL-1

```


/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 32

```

02052 6777 IOT6, SDRD /PUT FUNCT INTO XFUNCT IN SECOND PG.
02053 3762 DCA I CXFUN
02054 1356 TAD WCOUNT
02055 3765 DCA I CXWCT
02056 6777 IOT7, SDRD /GET MOTION BIT TO LINK
02057 7110 CLL RAR
02060 5277 JMP GO /AND START THE MOTION.
02061 6772 RWCOM, SDST /ANY CHECKSUM ERRORS?
02062 7640 SZA CLA /OR CHECKSUM ERRORS?
02063 5337 JMP TRY3 /PLEASE NOTE THAT THE LINK IS ALWAYS
/SET AT RWCOM. GETCHK SETS IT.
/NO ERROR..FINISHED XFER?

02064 1364 TAD PGCT
02065 1305 TAD CM200
02066 7450 SNA
02067 5344 JMP EXIT /ALL DONE. GET OUT
02070 3364 DCA PGCT /NEW PAGE COUNT
02071 2204 ISZ BLOCK /NEXT BLOCK TO XFER
02072 1356 TAD WCOUNT /FORM NEXT BUFFER ADDRESS
02073 7041 CIA
02074 1357 TAD BUFF
02075 3357 DCA BUFF
02076 7120 CLL CML /FORCES MOTION FORWARD
02077 7232 GO, CLA CML RTR /LINK BECOMES MOTION BIT
02100 1203 TAD C1000
02101 1361 TAD YUNIT /PUT IN 'GO' AND UNIT #
02102 6774 IOT8, SDLC /LOOK FOR BLOCK NO.
02103 4755 JMS I CRDQUD /WAIT AT LEAST 6 LINES TO LOOK
02104 4755 JMS I CRDQUD
02105 7600 CM200, /COULD HAVE SAVED A LOC. HERE
02106 6771 SRCH, SDSS
02107 5306 JMP .-1 /WAIT FOR SINGLE LINE FLAG
02110 6776 IOT9, SDRC
02111 7106 CLL RTL /DIRECTION TO LINK. INFO BITS
/ARE SHIFTED.
/ISOLATE MARK TRACK BITS
02112 0241 AND C374 /IS IT END ZONE?
02113 1331 TAD M110 /THE LINK STAYS SAME THRU THIS
02114 7450 SNA
02115 5333 JMP ENDZ
02116 1363 TAD M20 /CHECK FOR BLOCK MARK
02117 7640 SZA CLA
02120 5306 JMP SRCH
02121 6777 IOT10, SDRD /GET THE BLOCK NUMBER
02122 7430 SZL /IF WE ARE IN REVERSE, LOOK FOR 3
/BLOCKS BEFORE TARGET BLOCK. THIS
/ALLOWS TURNAROUND AND UP TO SPEED.
/REVERSE

02123 1375 TAD C3
02124 7040 CMA
02125 1204 TAD BLOCK
02126 7040 CMA /IS IT RIGHT BLOCK?
02127 7450 SNA
02130 5370 JMP FOUND /YES..HOORAY!
02131 7670 M110, SZL SNA CLA /NO, BUT ARE WE HEADED FOR IT?
/ABOVE SNA IS SUPERFLUOUS.
02132 5306 JMP SRCH /YES

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 32-1

```
02133 6776 ENDZ,   SDRC           /WE ARE IN THE END ZONE
02134 7106        CLL RTL        /DIRECTION TO LINK
02135 7200        CLA           /ARE WE IN REVERSE?
02136 5277        JMP GO         /YES..TURN US AROUND
      /IF WE ARE IN THE END ZONE GOING FORWARD, IT IS AN ERROR
02137 7300 TRY3,   CLL CLA
02140 2367        ISZ TRYCNT
02141 5277        JMP GO         /TRY 3 TIMES
02142 7300        CLL CLA
02143 5346        JMP FATAL      /LINK OFF MEANS AC=4000 ON RETURN
02144 2200 EXIT,  ISZ DTA0
02145 7120        CLL CML       /AC=0 ON NORMAL RETURN
02146 1361 FATAL, TAD YUNIT
02147 6774        SDLC          /STOP THE UNIT
02150 7230        CLA CML RAR
02151 7402 LEAVE, HLT
02152 5600        JMP I DTA0    /--RETURN--
```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 33

```

02153 6201 C6201, 6201
02154 6203 C6203, 6203
02155 2314 CRDQUD, RDQUAD
02156 0000 WCOUNT, 0
02157 0000 BUFF, 0
02160 0000 MWORDS, 0
02161 0000 YUNIT, 0
02162 2354 CXFUN, XFUNCT
02163 7760 M20, -20
02164 0000 PGCT, 0
02165 2362 CXWCT, XWCT
02166 0100 C100, 100
02167 7775 TRYCNT, -3
      2004 BLOCK=DTA1

      2170 *ORIGIN+170
02170 7630 FOUND, SZL CLA /RIGHT BLOCK. HOW ABOUT DIRECTION?
02171 5277 JMP GO /WRONG..TURN AROUND
02172 1361 TAD YUNIT /PUT UNIT INTO LINK
02173 7104 CLL RAL /AC IS NOW 0
02174 0070 C70, 70 /*****DON'T MOVE THIS!!!!****
02175 0003 C3, 3
02176 1357 TAD BUFF /GET BUFFER ADDRESS
02177 7402 XFIELD, HLT /INTO NEXT PAGE

      2200 *ORIGIN+200
      2321 XUNIT=EQUFUN

02200 3361 DCA XBUFF
02201 6776 IOT16, SDRC
02202 6774 IOT17, SDLC
02203 7010 RAR /NOW GET UNIT #
02204 3321 DCA XUNIT
02205 6771 REVGRD, SDSS
02206 5205 JMP REVGRD /LOOK FOR REVERSE GUARD
02207 6776 IOT11, SDRC
02210 0242 AND K77
02211 1355 TAD CM32 /IS IT REVERSE GUARD?
02212 7640 SZA CLA
02213 5205 JMP REVGRD /NO.KEEP LOOKING
02214 1362 TAD XWCT
02215 3360 DCA WORDS /WORD COUNTER
02216 1354 TAD XFUNCT /GET FUNCTION READ OR WRITE
02217 7700 K7700, SMA CLA
02220 5255 JMP READ /NEG. IS WRITE
02221 6776 WRITE, SDRC
02222 0267 AND C300 /CHECK FOR WRITE LOCK AND SELECT ERROR
02223 7120 CLL CML /LOCK OUT AND SELECT ARE AC 0 ERRORS
02224 7640 SZA CLA
02225 5752 JMP I CFATAL /FATAL ERROR. LINK MUST BE ON
02226 4314 JMS RDQUAD /NO ONE EVER USES THIS WORD!
02227 7600 C7600, 7600
02230 1356 TAD C1400

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 33-1

| | | | | |
|-------|------|--------|-------------|---|
| 02231 | 1321 | | TAD XUNIT | /INITIATE WRITE MODE |
| 02232 | 6774 | IOT12, | SDLC | |
| 02233 | 7240 | | CLA CMA | |
| 02234 | 4305 | | JMS WRQUAD | /PUT 77 IN REVERSE CHECKSUM |
| 02235 | 7240 | | CLA CMA | |
| 02236 | 3357 | | DCA CHKSUM | |
| 02237 | 1761 | WRLP, | TAD I XBUFF | /GLORY BE! THE ACTUAL WRITE! |
| 02240 | 4305 | | JMS WRQUAD | |
| 02241 | 2361 | | ISZ XBUFF | /BUMP CORE POINTER |
| 02242 | 0077 | K77, | 77 | /ABOVE MAY SKIP |
| 02243 | 2360 | | ISZ WORDS | /DONE THIS BLOCK? |
| 02244 | 5237 | | JMP WRLP | /NOT YET..LOOP A WHILE |
| 02245 | 1354 | | TAD XFUNCT | /IS THE OPERATION FOR WDSBLK PER BLOCK? |
| 02246 | 7112 | | CLL RTR | /IF NO, WRITE A 0 WORD |
| 02247 | 7630 | | SZL CLA | |

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 34

```

02250 4305      JMS WRQUAD      /WRITE A WORD OF 0
02251 4336      JMS GETCHK      /DO THE CHECK SUM
02252 4305      JMS WRQUAD      /WRITE FORWARD CHECKSUM
02253 4305      JMS WRQUAD      /ALLOW CHECKSUM TO BE WRITTEN
02254 5753      JMP I CRWCOM

02255 4314 READ,  JMS RDQUAD
02256 4314      JMS RDQUAD
02257 4314      JMS RDQUAD      /SKIP CONTROL WORDS
02260 0242      AND K77
02261 1217      TAD K7700      /TACK 7700 ONTO CHECKSUM.
02262 3357      DCA CHKSUM      /CHECKSUM ONLY LOW 6 BITS ANYWAY
02263 4314 RDLP,  JMS RDQUAD
02264 4321      JMS EQUFUN      /COMPUT CHECKSUM AS WE GO
02265 3761      DCA I XBUFF      /IT GETS CONDENSED LATER
02266 2361      ISZ XBUFF
02267 0300 C300,  300          /PROTECTION
02270 2360      ISZ WORDS      /DONE THIS OP?
02271 5263      JMP RDLP        /NO SUCH LUCK
02272 1354      TAD XFUNCT      /IF OP WAS FOR WDSBLK-1, READ AND
02273 7112      CLL RTR        /CHECKSUM THE LAST TAPE WORD
02274 7620      SNL CLA
02275 5300      JMP RDLP2
02276 4314      JMS RDQUAD      /NOT NEEDED FOR WDSBLK/BLOCK
02277 4321      JMS EQUFUN      /CHECKSUM IT
02300 4314 RDLP2, JMS RDQUAD      /READ CHECKSUM
02301 0217      AND K7700
02302 4321      JMS EQUFUN
02303 4336      JMS GETCHK      /GET SIX BIT CHECKSUM
02304 5753      JMP I CRWCOM

02305 0000 WRQUAD, 0          /WRITE OUT A 12 BIT WORD
02306 4321      JMS EQUFUN      /ADD THIS TO CHECKSUM
02307 6773 IOT13, SDSQ        /SKIP ON QUADLINE FLAG
02310 5307      JMP .-1
02311 6775 IOT14, SDLD        /LOAD DATA ONTO BUS
02312 7200      CLA          /SDLD DOESN'T CLEAR AC
02313 5705      JMP I WRQUAD

02314 0000 RDQUAD, 0         /READ A 12 BIT WORD
02315 6773      SDSQ
02316 5315      JMP .-1
02317 6777 IOT15, SDRD        /READ DATA
02320 5714      JMP I RDQUAD

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 35

```

02321 0000 EQUFUN, 0 /COMPUTE EQUIVALENCE CHECKSUM
02322 7040 CMA
02323 3363 DCA EQU TMP /ACTUALLY CHECKSUMS ON DECTAPE ARE
02324 1363 TAD EQU TMP /EQUIVALENCE OF ALL WORDS IN A RECORD
02325 0357 AND CHKSUM /SIX BITS AT A TIME. BUT SINCE EQUIVALENCE
02326 7041 CIA /IS ASSOCIATIVE, WE CAN DO IT 12
02327 7104 CLL RAL /BITS AT A TIME AND CONDENSE LATER.
02330 1363 TAD EQU TMP /THIS ROUTINE USES THESE IDENTITIES:
02331 1357 TAD CHKSUM /A+B=(A.XOR.B)+2*(A.AND.B)
02332 3357 DCA CHKSUM /A.EQU.B=.NOT.(A.XOR.B)=A.XOR.(.NOT.B)
02333 1363 TAD EQU TMP /A.EQU.B=(A+(.NOT.B))-2*(A.AND.(.NOT.B))
02334 7040 CMA
02335 5721 JMP I EQUFUN

02336 0000 GETCHK, 0 /FORM 6 BIT CHECKSUM
02337 7200 CLA
02340 1357 TAD CHKSUM
02341 7040 CMA
02342 7106 CLL RTL
02343 7006 RTL
02344 7006 RTL
02345 4321 JMS EQUFUN
02346 7320 CLA CLL CML /FORCES LINK ON AT RWCOM
02347 1357 TAD CHKSUM
02350 0217 AND K7700
02351 5736 JMP I GETCHK

02352 2146 CFATAL, FATAL
02353 2061 CRWCOM, RWCOM
02354 0000 XFUNCT, 0
02355 7746 CM32, -32
02356 1400 C1400, 1400
02357 0000 CHKSUM, 0
02360 0000 WORDS, 0
02361 0000 XBUFF, 0
02362 0000 XWCT, 0
02363 0000 EQU TMP, 0

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE

PAGE 36

```

      2400 PAGE
      /
      /
02400 0000 PARITY, 0
02401 6036      KRB
02402 0121      AND   [177
02403 1162      TAD   [200
02404 5600      JMP I  PARITY

```

/IOT TABLES FOR TD8E SUBROUTINE

```

02405 2020 IOTLOC, IOT1
02406 2034      IOT2
02407 2040      IOT3
02410 2016      IOT4
02411 2046      IOT5
02412 2052      IOT6
02413 2056      IOT7
02414 2061      RWCOM
02415 2102      IOT8
02416 2106      SRCH
02417 2110      IOT9
02420 2121      IOT10
02421 2133      ENDZ
02422 2147      FATAL+1
02423 2205      REVGRD
02424 2207      IOT11
02425 2221      WRITE
02426 2232      IOT12
02427 2307      IOT13
02430 2311      IOT14
02431 2315      RDQUAD+1
02432 2317      IOT15
02433 2201      IOT16
02434 2202 TABEND, IOT17

```

```

      2435 UNITS=.
02435 0770      UNIT01
02436 0760      UNIT23
02437 0750      UNIT45
02440 0740      UNIT67

```

```

02441 1351 RTAB,  IOTR1
02442 1354      IOTR2
02443 1356      IOTR3
02444 1370      IOTR4
02445 0367      IOTR5
02446 0371      IOTR6

```

/IOT TABLES FOR WORDS PER BLOCK ROUTINE

```

02447 0637 TABLE1, IOTX1
02450 0662      IOTX2
02451 0642      IOTX3

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE

PAGE 36-1

| | | | |
|-------|------|-------|-------|
| 02452 | 0644 | | IOTX4 |
| 02453 | 0754 | | IOTX5 |
| 02454 | 0756 | | IOTX6 |
| 02455 | 0621 | | IOTX7 |
| 02456 | 0623 | END1, | IOTX8 |

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 37

```
02457 0622 MESSG4, TEXT @FROM UNIT:@
02460 1715
02461 4025
02462 1611
02463 2472
02464 0000
02465 2417 MESSG5, TEXT @TO UNITS:@
02466 4025
02467 1611
02470 2423
02471 7200
02472 1114 ERROR5, TEXT @ILLEGAL FORMAT UNIT @
02473 1405
02474 0701
02475 1440
02476 0617
02477 2215
02500 0124
02501 4025
02502 1611
02503 2440
02504 0000
02505 4025 ERROR6, TEXT @ UNIT @
02506 1611
02507 2440
02510 0000
02511 1114 ERROR4, TEXT @ILLEGAL RESPONSE@
02512 1405
02513 0701
02514 1440
02515 2205
02516 2320
02517 1716
02520 2305
02521 0000
```

```

2600 PAGE
      /ONCE ONLY CODE

02600 4777 END,      JMS I (QUEST
02601 2646          MESSG2      /@HIGHEST FIELD AVAILABLE:@
02602 7701          ACL
02603 0173          AND [7
02604 7041          CIA
02605 3037          DCA FIELDS
02606 1376          TAD (CDF
02607 3220          DCA CDF00
02610 1037          TAD FIELDS
02611 7450          SNA          /MORE THAN 1 FIELD??
02612 5237          JMP LIM      /NO--NO PROBLEM
02613 3041          DCA COUNT1   /YES--ARE THEY ALL PRESENT?
02614 1220 NEXT,    TAD CDF00
02615 1375          TAD (10
02616 3220          DCA CDF00   /SET FOR DATA FIELD CHANGE
02617 1374          TAD (HLT
02620 6201 CDF00,  CDF
02621 3775          DCA I (10    /TRY LOCATION 10
02622 1775          TAD I (10
02623 6201          CDF 0
02624 7041          CIA
02625 1374          TAD (HLT
02626 7650          SNA CLA      /IS FIELD THERE?
02627 5235          JMP NEXT1   /YES--TRY NEXT ONE
02630 4577          JMS I [MESSGE /NO
02631 2511          ERROR4      /ILLEGAL RESPONSE
02632 4567          JMS I [CRLF   /CARRIAGE RETURN/LINE FEED
02633 3040          DCA COUNT   /CLEAR COUNT
02634 5200          JMP END      /TRY AGAIN
02635 2041 NEXT1,  ISZ COUNT1  /DONE YET?
02636 5214          JMP NEXT    /NO

02637 1373 LIM,    TAD (LIMIT-END /SET BEGINNING OF VERIFY BUFFER
02640 7110          CLL RAR
02641 1176          TAD [END
02642 3062          DCA END0
02643 1372          TAD (NOP
02644 3771          DCA I (START1-1
02645 5770          JMP I (START1
02646 1011 MESSG2, TEXT @HIGHEST FIELD AVAILABLE:@
02647 0710
02650 0523
02651 2440
02652 0611
02653 0514
02654 0440
02655 0126
02656 0111
02657 1401
02660 0214
02661 0572

```

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE

PAGE 38-1

02662 0000
02770 0207
02771 0206
02772 7000
02773 5000
02774 7402
02775 0010
02776 6201
02777 0336
00121 0177
00122 1313
00123 2000
00124 2405
00125 7750
00126 1200
00127 0260
00130 0002
00131 1254
00132 7554
00133 1246
00134 6000
00135 7756
00136 0366
00137 3000
00140 2441
00141 7772
00142 0457
00143 1600
00144 5000
00145 0204
00146 1553
00147 0336
00150 7556
00151 0450
00152 1020
00153 1754
00154 1531
00155 2160
00156 1000
00157 1746
00160 4000
00161 0535
00162 0200
00163 0100
00164 0077
00165 1027
00166 2400
00167 0317
00170 7774
00171 0212
00172 0527
00173 0007
00174 7770
00175 0400

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 38-2

00176 2600
00177 0471
0000 FIELD 0
0200 *200

\$

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE PAGE 39

| | | | | | | | |
|--------|------|--------|------|--------|------|--------|------|
| AGAIN | 0217 | DTA1X | 2010 | IOT13 | 2307 | OUTNUM | 0033 |
| ALLDUN | 1215 | END | 2600 | IOT14 | 2311 | OUTPUT | 0021 |
| ANSWER | 0400 | ENDZ | 2133 | IOT15 | 2317 | PARITY | 2400 |
| BAD | 0440 | END0 | 0062 | IOT16 | 2201 | PGCT | 2164 |
| BLOCK | 2004 | END1 | 2456 | IOT17 | 2202 | PRINT | 1531 |
| BLOCKN | 0056 | ENTRY | 0063 | IOT2 | 2034 | QUEST | 0336 |
| BLOCKS | 1715 | EQUFUN | 2321 | IOT3 | 2040 | QUEST1 | 0345 |
| BUFADD | 1714 | EQUTMP | 2363 | IOT4 | 2016 | RDLP | 2263 |
| BUFF | 2157 | ERR | 1730 | IOT5 | 2046 | RDLP2 | 2300 |
| BUFN | 0054 | ERROR1 | 1135 | IOT6 | 2052 | RDQUAD | 2314 |
| BUF0 | 0053 | ERROR2 | 1147 | IOT7 | 2056 | READ | 2255 |
| CDF0 | 1475 | ERROR3 | 1160 | IOT8 | 2102 | READX | 1220 |
| CDF00 | 2620 | ERROR4 | 2511 | IOT9 | 2110 | READY | 1600 |
| CFATAL | 2352 | ERROR5 | 2472 | K77 | 2242 | REPEAT | 1027 |
| CHAR | 0526 | ERROR6 | 2505 | K7700 | 2217 | REVGRD | 2205 |
| CHECK | 0457 | ERR1 | 1505 | LEAVE | 2151 | REWIND | 1313 |
| CHKSUM | 2357 | ERR3 | 1746 | LIM | 2637 | RHALF | 0500 |
| CLEAN | 1553 | ERR4 | 0325 | LIMIT | 7600 | RLIST | 1316 |
| CLEAN1 | 1560 | ERR5 | 0744 | LIST | 0032 | RTAB | 2441 |
| CLEAR | 0436 | EXIT | 2144 | LISTEN | 0450 | RUNIT | 1325 |
| CM200 | 2105 | FATAL | 2146 | LNEXT | 0476 | RW | 0050 |
| CM32 | 2355 | FIELDS | 0037 | LOC | 0525 | RWCOM | 2061 |
| COMP | 1466 | FINDER | 0524 | MESSGE | 0471 | SELECT | 1743 |
| COMP2 | 1471 | FLDN | 0052 | MESSG1 | 0067 | SELERR | 0627 |
| COMP3 | 1435 | FLD0 | 0051 | MESSG2 | 2646 | SEMI | 0443 |
| COMP4 | 1455 | FOUND | 2170 | MESSG3 | 0076 | SETUP | 0600 |
| CONT | 1014 | FOUR | 1540 | MESSG4 | 2457 | SET2 | 0651 |
| COUNT | 0040 | FUNCTN | 1713 | MESSG5 | 2465 | SET3 | 0700 |
| COUNT1 | 0041 | GETCHK | 2336 | MESSG6 | 1040 | SET4 | 0606 |
| COUNT2 | 0042 | GO | 2077 | MESSG7 | 1052 | SET5 | 0674 |
| COUNT3 | 0043 | HALT | 7402 | MESSG8 | 1064 | SKIP | 0437 |
| COUNT4 | 0044 | IBLOCK | 0034 | MESSG9 | 1101 | SKIPQ | 0366 |
| CRDQUD | 2155 | INB | 0064 | MESS10 | 1117 | SKIP4 | 0753 |
| CRLF | 0317 | INPUT | 0020 | MESS11 | 1122 | SRCH | 2106 |
| CRWCOM | 2353 | INSERT | 0535 | MFIELD | 0000 | START | 0200 |
| CTRLC | 0470 | INS1 | 0551 | MNUM | 0343 | START1 | 0207 |
| CTRLR | 1020 | IOTLOC | 2405 | MORE | 0404 | SUB | 0726 |
| CXFUN | 2162 | IOTR1 | 1351 | MORE2 | 0224 | SUB1 | 1657 |
| CXWCT | 2165 | IOTR2 | 1354 | MWORDS | 2160 | SUB2 | 1666 |
| C100 | 2166 | IOTR3 | 1356 | M110 | 2131 | SUB3 | 1677 |
| C1000 | 2003 | IOTR4 | 1370 | M20 | 2163 | SWITCH | 0412 |
| C1400 | 2356 | IOTR5 | 0367 | M200 | 1371 | TABEND | 2434 |
| C3 | 2175 | IOTR6 | 0371 | NEXT | 2614 | TABLE1 | 2447 |
| C300 | 2267 | IOTX1 | 0637 | NEXT1 | 2635 | TABX | 0365 |
| C374 | 2041 | IOTX2 | 0662 | NUMBER | 0036 | TRANS | 1707 |
| C6201 | 2153 | IOTX3 | 0642 | NUMB1 | 0057 | TRY | 1500 |
| C6203 | 2154 | IOTX4 | 0644 | NUMB2 | 0060 | TRYCNT | 2167 |
| C70 | 2174 | IOTX5 | 0754 | OALL | 0255 | TRY3 | 2137 |
| C7600 | 2227 | IOTX6 | 0756 | OBLOCK | 0035 | TYPE | 0527 |
| DECODE | 1754 | IOTX7 | 0621 | OCOUNT | 0030 | UNIT | 0045 |
| DIV1 | 1275 | IOTX8 | 0623 | OHOLD | 0066 | UNITNO | 0352 |
| DIV2 | 1301 | IOT1 | 2020 | OPOINT | 0031 | UNITS | 2435 |
| DOIT | 1200 | IOT10 | 2121 | ORIGIN | 2000 | UNIT01 | 0770 |
| DTA0 | 2000 | IOT11 | 2207 | OUTB | 0065 | UNIT23 | 0760 |
| DTA1 | 2004 | IOT12 | 2232 | OUTN | 1246 | UNIT45 | 0750 |

/TD8E DECTAPE COPY, V4

PAL8-V10A NO DATE

PAGE 39-1

| | |
|--------|------|
| UNIT67 | 0740 |
| VB | 0061 |
| VERF | 0046 |
| VERFQ | 1000 |
| VERF1 | 1311 |
| VERF1A | 1016 |
| VERF2 | 1274 |
| VERF2A | 1017 |
| VERIFY | 1400 |
| WAIT | 1516 |
| WCOUNT | 2156 |
| WDCNT | 0047 |
| WHOLE | 0306 |
| WORDS | 2360 |
| WRITE | 2221 |
| WRITEX | 1254 |
| WRLP | 2237 |
| WRQUAD | 2305 |
| XBUFF | 2361 |
| XFIELD | 2177 |
| XFUNCT | 2354 |
| XNUMB | 0055 |
| XUNIT | 2321 |
| XWCT | 2362 |
| X11 | 0011 |
| X12 | 0012 |
| YES | 1010 |
| YUNIT | 2161 |
| ZOOM | 1630 |

ERRORS DETECTED: 0
LINKS GENERATED: 2