

.REM !

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZTOD-C-0

PRODUCT NAME: TC4 - TC11 TEST 4

DATE: JUNE 15, 1973

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: L. R. KOLLER

THIS MAINDEC OBSOLETES MAINDEC-11-D3DC

COPYRIGHT 1972, 1973. DIGITAL EQUIPMENT CORP., MAYNARD, MASS.

TC4 - TC11 TEST 4 IS PART 4 OF A FIVE PROGRAM PACKAGE
USED TO TEST THE TC11 DECTAPE CONTROL.

1. ABSTRACT

TC4 - TC11 TEST 4 IS PART 4 OF A FIVE PROGRAM PACKAGE USED TO TEST THE TC11 DECTAPE CONTROL. TC4 TESTS AND EXERCISES THE TC11 CONTROL AND FROM ONE TO EIGHT SELECTED TRANSPORTS. TC4 CONCENTRATES ON TESTING FOR CORRECT OPERATION OF THE READ ALL AND WRITE ALL COMMANDS, AND CHECKS FOR CORRECT OPERATION OF THE PARITY CIRCUITS.

ALL EXECUTION TIMES QUOTED ARE TYPICAL OF A 11/20 SYSTEM.
EXECUTION TIMES IN OTHER PDP-11 SYSTEMS WILL VARY.

2. REQUIREMENTS2.1 EQUIPMENT

- A. PDP-11 SYSTEM (4K CORE).
- B. ASR33/35 TELETYPE.
- C. TC11 DECTAPE CONTROL AND AT LEAST ONE TUS6 DUAL TRANSPORT.
- D. AT LEAST ONE STANDARD PDP-11 FORMAT DECTAPE. THE GUARD AREAS OF THE TAPE BLOCKS MUST BE ZERO. IF NECESSARY, REFORMAT THE TAPE.

THE TELETYPE AND TC11 CONTROL MUST HAVE THEIR STANDARD PERIPHERAL ADDRESSES, INTERRUPT LEVELS, AND INTERRUPT VECTOR ADDRESSES. REFER TO SECTION 7.2 IF YOUR SYSTEM DOES NOT HAVE STANDARD PERIPHERAL ADDRESSES.

2.2 STORAGE

THIS PROGRAM USES LOCATIONS 000000 THROUGH 017500.

3. LOADING PROCEDURE

THIS PROGRAM'S OBJECT TAPE IS PUNCHED IN ABSOLUTE FORMAT.
THE ABS LOADER IS USED TO LOAD THE PROGRAM.

4. USE PROCEDURE

- A. LOAD UNITS TO BE TESTED WITH STANDARD FORMAT DECTAPE. SET TO REMOTE/WRITE ENABLE.
- B. SET WRTH SWITCH OFF; WALL SWITCH TO ON.
- C. LOAD ADDRESS 000200.
- D. PRESS START.
- E. THE PROGRAM IDENTIFIES ITSELF, TYPES SETUP INSTRUCTIONS, AND HALTS.
- F. PERFORM SETUP (STEPS A AND B) SET UNITS TO BE TESTED IN SR7 THROUGH SR0 AND PRESS CONT. (SR7 FOR UNIT7, SR6 FOR UNIT6, ETC.).
- G. THE PROGRAM TYPES SR OPTIONS MESSAGE. SET DESIRED SR OPTIONS IF ANY. NORMAL SR IS 000000. PRESS CONT.

THIS PROGRAM'S SR OPTIONS ARE:

SR15 = 1	HALT ON ERROR
SR14 = 1	ENTER SCOPE MODE
SR13 = 1	INHIBIT ERROR PRINTOUT
SR11 = 1	INHIBIT ITERATION
SR10 = 1	HALT AT END OF TEST CURRENTLY EXECUTING
SR9 = 1	SELECT THE TEST SPECIFIED BY SR7 THROUGH SR0
SR7 THROUGH SR0	- NUMBER OF TEST TO BE SELECTED

SECTION 7.1 GIVES A COMPLETE EXPLANATION OF SR OPTIONS.

- H. THE PROGRAM BEGINS EXECUTION.
- I. AT THE END OF EACH PASS THE TELETYPE BELL RINGS ONCE, AND THE CHARACTER "*" IS TYPED.
- J. REFER TO SECTION 6.2 IF ERROR PRINTOUTS OCCUR.

EXECUTION TIME:

- A. ONE NORMAL ERROR FREE PASS TAKES APPROXIMATELY 36 MINUTES.
- B. ONE SINGLE ITERATION PASS (SR11=1) TAKES ABOUT 7 MINUTES.

THE SINGLE ITERATION PASS IS A CONVENIENT WAY TO QUICKLY DETERMINE IF ANY SOLID PROBLEMS EXIST. FOR A THOROUGH TEST, THE NORMAL ITERATION PASS SHOULD BE RUN.

4.1 RESTART PROCEDURE

TO RESTART THE PROGRAM WITHOUT GENERATING THE INITIAL PRINTOUTS PROCEED AS FOLLOWS: (TRANSPORT UNDER TEST REMAINS THE SAME)

- A. LOAD ADDRESS 001000
- B. DO UNIT SETUP AS DESCRIBED IN STEPS A AND B OF USE PROCEDURE.
- C. SELECT ANY DESIRED OPTIONS.
- D. PRESS START.
- E. GO TO STEP H OF USE PROCEDURE.

5. PROGRAM AND/OR OPERATOR ACTION
-----5.1 NORMAL HALTS

LOC 002432 COMMON HALT. THIS HALT IS CONTAINED IN A SUBROUTINE THAT IS CALLED BY THOSE PARTS OF THE PROGRAM THAT REQUIRE THAT THE PROCESSOR STOP. THIS HALT NORMALLY OCCURS UPON COMPLETION OF NON-ERROR PRINTOUTS. THE CONSOLE DATA LIGHTS DISPLAY THE ADDRESS OF INSTRUCTION THAT GENERATED THE HALT REQUEST.

LOC 001764 ROUTINE END HALT. THIS HALT OCCURS UPON COMPLETION OF THE CURRENT TEST ROUTINE IF SR10 IS SET. THE CONSOLE DATA LIGHTS DISPLAY THE NUMBER OF THE TEST JUST COMPLETED.

5.2 NORMAL PRINTOUTS

ALL NON-ERROR PRINTOUTS ARE NORMAL PRINTOUTS. INSTRUCTION, TITLE, AND USER ERROR PRINTOUTS ARE NORMAL PRINTOUTS.

6. ERRORS

ERRORS ARE REPORTED IN THIS PROGRAM BY THE FOLLOWING METHODS:

- A. UNCONDITIONAL ERROR HALTS, OR
- B. ERROR PRINTOUT FOLLOWED BY OPTIONAL ERROR HALT.

6.1 UNCONDITIONAL ERROR HALTS

AN UNCONDITIONAL ERROR HALT WILL OCCUR AT THE ADDRESSES LISTED BELOW IF THROUGH HARDWARE OR SOFTWARE FAILURE, PROGRAM CONTROL IS TRANSFERRED TO AN UNEXPECTED AREA BETWEEN 000000 AND 000176.

000002 RESERVED AREA
 000006 ERROR TRAP
 000012 RESERVED INSTRUCTION TRAP
 000016 DEBUG TRAP
 000022 IOT TRAP
 000026 POWER FAIL TRAP
 000040 THROUGH 000176 - SYSTEM SOFTWARE AND INTERRUPT VECTOR AREA, EXCEPT FOR TC11 AND TTY VECTORS.

TO FIND OUT WHERE THE PROGRAM WAS AT THE TIME THE FAILURE OCCURRED.

- A. EXAMINE CONTENTS OF REGISTER 6. (ADDRESS 177706).
- B. TRANSFER THE CONTENTS OF REG 6 TO THE SR, LOAD ADDRESS AND EXAMINE.
- C. THE DATA SHOWN IN THE DATA LIGHTS IS THE VALUE OF THE PC WHEN THE FAILURE OCCURRED.
- D. LOCATE IN PROGRAM LISTING THE DISPLAYED PC VALUE.

(6.1 CONT'D)

E. THE INSTRUCTION THAT IMMEDIATELY PRECEDES THE ONE REFERENCED BY THE DISPLAYED PC VALUE IS THE INSTRUCTION THAT WAS/WAS BEING EXECUTED WHEN THE FAILURE OCCURRED.

AN UNCONDITIONAL ERROR HALT FAILURE IS AN ABNORMAL CONDITION INDICATING A HARDWARE FAILURE, OR MOST UNLIKELY, A PROGRAM FAILURE. THIS PROGRAM ASSUMES THAT THE PROCESSOR IS IN OPERATING CONDITION IN ORDER TO PERFORM ITS TESTS. ANY FURTHER STEPS REQUIRED TO DIAGNOSE AN UNCONDITIONAL ERROR HALT ARE NOT WITHIN THE SCOPE OF THIS PROGRAM.

E.2 ERROR PRINTOUTS

ERROR PRINTOUTS ARE GENERATED BY THE "ERRN" SUBROUTINE. THE "ERRN" SUBROUTINE IS CALLED BY AN "ERRORN" STATEMENT IN THE PROGRAM LISTING. AN ERROR PRINTOUT LOOKS AS FOLLOWS:

T XXX PC OYYYYY ICNT ZZZZ. UNIT W FPC OVVVVV
UP TO 2 ADDITIONAL LINES OF ERROR INFORMATION.

WHERE:

T XXX IS THE NUMBER OF FAILING ROUTINE (OCTAL),
PC OYYYYY IS THE ADDRESS OF ERROR CALL,
ICNT ZZZZ. IS THE ITERATION COUNT AT TIME OF FAILURE.
UNIT W IS THE UNIT IN USE AT TIME OF FAILURE.
FPC OVVVVV IS TYPED WHEN THE ERROR CALL IS GENERATED BY A
SUBROUTINE, AND IT IS NECESSARY TO INDICATE WHERE
THE SUBROUTINE WAS CALLED FROM.

AFTER THE PRINTOUT IS COMPLETED, THE PROGRAM WILL HALT AT COMMON ERROR HALT AT LOC 002446 IF SR15 IS SET.

WHEN AN ERROR PRINTOUT OCCURS:

- A. LOOK UP THE ADDRESS REFERENCED BY PC OYYYYY IN THE LISTING.
- B. OPPOSITE THE PC VALUE AN ERRORN STATEMENT WILL BE FOUND, AND IN THE COMMENTS SECTION, A DESCRIPTION OF THE ERROR.
- C. AT THE BEGINNING OF THE TEST ROUTINE A DESCRIPTION OF THE TEST WILL BE FOUND.

UP TO 2 LINES OF ADDITIONAL ERROR INFORMATION MAY APPEAR ON AN ERROR PRINTOUT. SOME OF THE ITEMS THAT MAY APPEAR ARE:

- A. BLKRQ XXXX. BLKRQ REPRESENTS THE INITIAL BLOCK NUMBER USED WHEN AN OPERATION WAS INITIATED. (IN A 2 OR MORE BLOCK TRANSFER, BLKRQ REPRESENTS THE INITIAL BLOCK NUMBER, EVEN THOUGH A FAILURE MAY NOT HAVE OCCURRED UNTIL A SUBSEQUENT BLOCK.
- B. IN A DATA ERROR PRINTOUT THE "WORD #" THAT FAILED REPRESENTS THE POSITION OF THE CORRECT WORD IN THE WRITE BUFFER, AND IT IS NOT MEANT TO DESCRIBE THE WORD'S POSITION IN A DECTAPE BLOCK.

7. MISCELLANEOUS

7.1 SR OPTIONS

THE STANDARD SR OPTIONS ARE DESCRIBED HERE.

SR15 HALT ON ERROR. WITH SR15 SET TO A 1, THE PROGRAM WILL HALT AFTER AN ERROR OCCURS. PRESSING CONT WILL CAUSE PROGRAM TO RESUME OPERATION.

SR14 SCOPE. THIS OPTION CAUSES THE PROGRAM TO REMAIN IN THE CURRENT TEST ROUTINE. WHEN THE OPTION IS REMOVED, THE PROGRAM WILL COMPLETE THE CURRENT ROUTINE, AND WILL THEN GO ON TO THE NEXT ROUTINE.

SR13 INHIBIT ERROR PRINTOUT. THIS OPTION IF SET, WILL REMOVE ALL ERROR PRINTOUTS.

*****NOTE*****

SCOPE MODE OPERATION IS ACHIEVED BY LOCKING THE PROGRAM IN THE CURRENT ROUTINE, INHIBITING ERROR PRINTOUTS, AND BYPASSING ERROR HALTS.

SR11 INHIBIT ITERATION. SETTING THIS OPTION WILL CAUSE THE PROGRAM TO EXECUTE EACH TEST ONLY ONCE, INSTEAD OF THE NORMAL NUMBER OF ITERATIONS SELECTED FOR EACH TEST. TWO POSSIBLE USES OF THIS OPTION ARE:

- A. QUICK PASS. EACH TEST IS RUN ONLY ONCE.
- B. TO SKIP OVER A FAILING ROUTINE.

SR10 HALT AT END OF CURRENT ROUTINE. WITH THE OPTION SET, THE PROGRAM WILL HALT AT THE END OF EACH TEST, AND DISPLAY IN DATA LIGHTS THE NUMBER OF THE TEST JUST COMPLETED. THREE POSSIBLE USES OF THIS OPTION ARE:

- A. TO STEP THROUGH THE PROGRAM ONE ROUTINE AT A TIME.
- B. WHEN THE PROGRAM HAS BEEN RUNNING FOR A WHILE, TO FIND OUT HOW FAR IT HAS PROGRESSED.
- C. IN CASE OF A BLOW UP, ETC., TO STEP THROUGH ONE TEST AT A TIME UNTIL THE FAILURE REOCCURS. THE ROUTINE FOLLOWING THE PREVIOUSLY COMPLETED ROUTINE WOULD BE THE FAILING ROUTINE.

SR9 SELECT ROUTINE. WITH SR9 SET, THE PROGRAM WILL GO AND EXECUTE THE ROUTINE INDICATED BY SR7 THROUGH SR0, AFTER THE CURRENT ROUTINE HAS BEEN COMPLETED. IF THE OPTION IS REMOVED, THE PROGRAM WILL PROCEED TO EXECUTE THE ROUTINES FOLLOWING THE SELECTED ROUTINE.

7.2 TESTING TC11 AT NON-STANDARD ADDRESSES AND/OR VECTORS

THIS PROGRAM CAN TEST THE TC11 AT NON-STANDARD ADDRESSES AND VECTORS PROVIDED THOSE ADDRESSES AND VECTORS ARE PROVIDED TO THE PROGRAM AS FOLLOWS:

- A. AFTER LOADING PROGRAM REFER TO PROGRAM LISTING AND CHANGE LOCATIONS 001004 THROUGH 001020 TO REFLECT THE NEW TC11 ADDRESSES AND VECTORS.
- B. IF THE TELETYPE IS ALSO AT NON STANDARD ADDRESSES, CHANGE LOCATIONS 001022 AND 001024 ALSO.
- C. PROCEED TO USE THE PROGRAM, OR
- D. USING STANDARD DUMP ROUTINES, DUMP OUT THE ENTIRE PROGRAM IN ABSOLUTE FORMAT TO HAVE AN OBJECT TAPE THAT REFLECTS YOUR SYSTEM, OR
- E. DUMP OUT ONLY LOCATIONS 001004 THROUGH 001024 IN ABSOLUTE FORMAT, AND LOAD IT ALSO AFTER LOADING THE MAIN PROGRAM.

8. DESCRIPTION

THIS PROGRAM IS ORGANIZED INTO THREE MAIN SECTIONS:

- A. CONTROL ROUTINE,
- B. TEST ROUTINES,
- C. COMMON SUBROUTINES

8.1 CONTROL ROUTINE

THE CONTROL ROUTINE ASSUMES CONTROL WHEN THE PROGRAM IS STARTED. IT HAS THE FOLLOWING FUNCTIONS:

- A. CONTROLS SEQUENCE OF TEST ROUTINES.
- B. HONORS AND ACTS ON SR OPTIONS.

THE CONTROL ROUTINE IS CALLED FROM A TEST ROUTINE BY THE "SCOPE" STATEMENT.

8.2 TEST ROUTINES

THE ACTUAL TESTING IS PERFORMED BY A SET OF TEST ROUTINES THAT ARE NUMBERED SEQUENTIALLY FROM 0 TO 7 (OCTAL). EACH TEST ROUTINE IS PRECEDED BY A TEST HEADER THAT IS USED BY THE CONTROL ROUTINE IN ORDER TO PROPERLY SEQUENCE THROUGH THE TESTS. THE HEADER LOOKS AS FOLLOWS: (EXAMPLE)

```

*****
T3:      3           :ROUTINE NUMBER 3.           *
         T4         :ADDRESS OF NEXT ROUTINE      *
         10.        :TEST ITERATION COUNT        *
         BAGA       :SCOPE ENTRY POINT           *
*****

```

THE FIRST 2 ITEMS ARE SELF EXPLANATORY. THE TEST ITERATION COUNT INDICATES TO THE CONTROL ROUTINE THE NUMBER OF TIMES THE TEST SHOULD BE PERFORMED BEFORE GOING ON TO THE NEXT ROUTINE.

THE SCOPE ENTRY POINT INDICATES TO THE CONTROL ROUTINE THE ADDRESS IT SHOULD RETURN TO AFTER THE FIRST ITERATION. THE ADDRESS MAY NOT NECESSARILY POINT TO THE FIRST INSTRUCTION OF THE TEST.

8.3 COMMON SUBROUTINES

ALL SUBROUTINES NEEDED BY EITHER THE CONTROL ROUTINE OR TEST ROUTINES ARE GROUPED TOGETHER. THE MOST SIGNIFICANT SUBROUTINE IS THE "ERRN" SUBROUTINE, WHICH IS CALLED BY AN "ERRORN" STATEMENT AND TYPES THE TEST NUMBER AND PC VALUE WHEN A FAILURE OCCURS.


```

000000
000002
000004
000006
000010
000012
000014
000016
000020
000022
000024
000026
000030
000032
000034
000036
177570
177776
001000
000240
000000
100000
100000
040000
020000
010000
004000
002000
001000
000400
000200
000100
000040
000020
000010
000004
000002
000001
000000
000001
000002
000003
000004
000005
000006
000007
000007

```

```

      .LIST SEQ,BIN,ME
      .NLIST MC,MD
      .ABS
:
      .+0
      .+2 ;UNASSIGNED TRAP
MACHER: HALT ;SP OVERFLOW, BUS ERROR TRAP
      .+2
      HALT ;RESERVED INSTRUCTION TRAP
      .+2
TRCV: HALT ;TRACE TRAP
      SVSS
      PRTY7
IOTV: RSSS ;TRAP TO CALL IOX
      PRTY7
      .+2
      HALT ;POWER FAIL TRAP
EMTV: EMTINT ;EMT TRAP
      PRTY7
TRPV: DLY ;TRAP TRAP. SIMILAR TO EMT
      PRTY0
;LOC 40 THROUGH 376 ARE FILLED WITH .+2 AND HALT.
;EQUATE STATEMENTS
SR=177570
PSW=177776
SPBOT=1000
NOP=240
OPEN=0
MANUAL=BIT15
BIT15=100000
BIT14=40000
BIT13=20000
BIT12=10000
BIT11=4000
BIT10=2000
BIT9=1000
BIT8=400
BIT7=200
BIT6=100
BIT5=40
BIT4=20
BIT3=10
BIT2=4
BIT1=2
BIT0=1
R0=%0
R1=%1
R2=%2
R3=%3
R4=%4
R5=%5
R6=%6
R7=%7
PC=%7

```

401 005746
402 024646
403 005726
404 022626
405 000340
406 000300
407 000240
408 000200
409 000140
410 000100
411 000040
412 000000
413 000007
414 177777
415 000003
416 000040
417 177777
418 100000
419 040000
420 020000
421 000000
422 000004
423 000010
424 000014
425 000020
426 000024
427 000030
428 000034
429 020000
430 010000
431 004000
432 000000
433 000000
434 000400
435 001000
436 001400
437 002000
438 002400
439 003000
440 003400
441 000100
442 000000
443 000002
444 000004
445 000006
446 000010
447 000012
448 000014
449 000016
450 000001
451 000200
452 010000
453 004000
454 000000
455 000003
456 000004

PUSH=005746
PUSH2=024646
POPSP=005726
POPSP2=022626
PTY7=340
PTY6=300
PTY5=240
PTY4=200
PTY3=140
PTY2=100
PTY1=40
PTY0=0
BELL=007
TLAST=-1
TRC=3
I=40
X=-1
A=BIT15
B=BIT14
C=BIT13
V0=0
V1=4
V2=10
V3=14
V4=20
V5=24
V6=30
V7=34
MAINT=BIT13
DINH=BIT12
REV=BIT11
FWD=0
U0=0
U1=BIT8
U2=BIT9
U3=BIT9:BIT8
U4=BIT10
U5=BIT10:BIT8
U6=BIT10:BIT9
U7=BIT10:BIT9:BIT8
IE=BIT6
SAT=0
RNUM=BIT1
RDATA=BIT2
RALL=BIT2:BIT1
SST=BIT3
WRTM=BIT3:BIT1
WDATA=BIT3:BIT2
WALL=BIT3:BIT2:BIT1
DO=BIT0
UPS=BIT7
ILO=BIT12
SELE=BIT11
EMTX=0
SAVSS=3
RSTSS=4

487		104400	
488		000200	
489	000200	000167	001036
490		001000	
491	001000	000167	000554
492	001004	177340	
493	001006	177342	
494	001010	177344	
495	001012	177346	
496	001014	177350	
497	001016	000214	
498	001020	000300	
499	001022	177564	
500	001024	177566	
501	001026	000000	
502	001030	000000	
503	001032	006644	
504	001034	000000	
505	001036	000000	
506	001040	000000	
507	001042	000000	
508	001044	000000	
509	001046	000000	
510	001050	000000	
511	001052	000000	
512	001054	000000	
513	001056	000000	
514	001060	000000	
515	001062	000000	
516	001064	000000	
517	001066	000000	
518	001070	000000	
519	001072	000000	
520	001074	000000	
521	001076	000000	
522	001100	000000	
523	001102	000000	
524	001104	000000	
525	001106	000005	
526	001110	000310	
527	001112	000000	
528	001114		
529	001114	001674	
530	001116	002404	
531	001120	002130	
532	001122	002230	
533	001124	002160	
534	001126	002260	
535	001130	002150	
536	001132	002254	
537	001134	002660	
538	001136	002452	
539	001140	003106	
540	001142	003200	
541	001144	003156	
542	001146	002426	

```

DELAY=TRAP
.=200
JMP START ;GO TO START OF PROGRAM.
.=1000
JMP GETROY ;BYPASS INITIAL PRINTOUTS.
TCST: 177340 ;TC11 STATUS REGISTER.
TCOM: 177342 ;TC11 COMMAND REGISTER.
TCWC: 177344 ;TC11 WORD COUNT REGISTER.
TCBA: 177346 ;TC11 BUS ADDRESS REGISTER.
TCOT: 177350 ;TC11 DATA REGISTER.
TCVTR: 214 ;TC11 INTERRUPT VECTOR
TCLVL: PRTY6 ;TC11 INTERRUPT PRIORITY LEVEL.
TPS: 177564 ;LSP CSR
TPB: 177566 ;LSP BUFFER
ICTR: OPEN ;CONTAINS CURRENT ITERATION COUNT
ICNT: OPEN ;CONTAINS ACCUMULATED ITERATION COUNT.
KSTART: TO ;CONTAINS STARTING ROUTINE ADDR.
SCOPTR: OPEN ;CONTAINS CURRENT SCOPE POINTER.
RTNNO: OPEN
NXTST: OPEN
CURTST: OPEN
CTRA: OPEN
TCOMT: OPEN
TCSTT: OPEN
TCOTT: OPEN
TCWCT: OPEN
TCBAT: OPEN
BLKRQ: OPEN
UNIT: OPEN
UNITN: OPEN
UNITS: OPEN
COMND: OPEN
TEMP: OPEN
FPC: OPEN
LPBG: OPEN
LPBT: OPEN
LPB: OPEN
ELPB: OPEN
ERRLIM: 5
BLKNUM: 200.
ALLPAR: OPEN
EMTTAB:

```

```

.WORD CHAINN ; POINTER FOR EMT CALL SCOPE
.WORD SRETT ; POINTER FOR EMT CALL SRESET
.WORD SV03 ; POINTER FOR EMT CALL SAV03
.WORD RS03 ; POINTER FOR EMT CALL RST03
.WORD SV05 ; POINTER FOR EMT CALL SAV05
.WORD RS05 ; POINTER FOR EMT CALL RST05
.WORD SV05S ; POINTER FOR EMT CALL SAV05S
.WORD RS05S ; POINTER FOR EMT CALL RST05S
.WORD TYP ; POINTER FOR EMT CALL TYPE
.WORD ERN ; POINTER FOR EMT CALL ERRORN
.WORD OACNVV ; POINTER FOR EMT CALL OACNV
.WORD BDCNVV ; POINTER FOR EMT CALL BDCNV
.WORD BMOVV ; POINTER FOR EMT CALL BMOVE
.WORD CHLT ; POINTER FOR EMT CALL CHALT

```

543 001150 002440
544 001152 002364
545 001154 003350
546 001156 003310
547 001160 003444
549 001162 003454
549 001164 003544
550 001166 004344
551 001170 004352
552 001172 005010
553 001174 005036
554 001176 005020
555 001200 005046
556 001202 005064
557 001204 004244
558 001206 004176
559 001210 004216
560 001212 003642
561 001214 003632
562 001216 003616
563 001220 003606
564 001222 003010
565 001224 003042
566 001226 005110
567 001230 005126
568 001232 005264
569 001234 005302
570 001236 006560
571 001240 006602
572

.WORD EHLT
.WORD STTCV
.WORD STCOM
.WORD STATS
.WORD STPDT
.WORD CKERZ
.WORD NCIMTR
.WORD SRCHFF
.WORD SRCHAR
.WORD WDATAF
.WORD WDATR
.WORD RDATAF
.WORD RDATR
.WORD RDTFSS
.WORD CWCBA
.WORD CLEARR
.WORD BINFL
.WORD DATCK
.WORD DTCKI
.WORD ADATCK
.WORD ADATCI
.WORD INBINN
.WORD GTBNI
.WORD WALLFF
.WORD WALLRR
.WORD RALLFF
.WORD RALLRR
.WORD SQDRV
.WORD SELDRR

: POINTER FOR EMT CALL EHALT
: POINTER FOR EMT CALL SVECTR
: POINTER FOR EMT CALL SETCOM
: POINTER FOR EMT CALL STATUS
: POINTER FOR EMT CALL STOPDT
: POINTER FOR EMT CALL CKERFZ
: POINTER FOR EMT CALL NOINT
: POINTER FOR EMT CALL SRCHF
: POINTER FOR EMT CALL SRCHR
: POINTER FOR EMT CALL WDATAF
: POINTER FOR EMT CALL WDATAR
: POINTER FOR EMT CALL RDATAF
: POINTER FOR EMT CALL RDATAR
: POINTER FOR EMT CALL RDATAFS
: POINTER FOR EMT CALL CKWCBA
: POINTER FOR EMT CALL CLEAR
: POINTER FOR EMT CALL BINFIL
: POINTER FOR EMT CALL DATCHK
: POINTER FOR EMT CALL DATCKI
: POINTER FOR EMT CALL ADTCK
: POINTER FOR EMT CALL ADTCKI
: POINTER FOR EMT CALL INBIN
: POINTER FOR EMT CALL GETBNI
: POINTER FOR EMT CALL WALLF
: POINTER FOR EMT CALL WALLR
: POINTER FOR EMT CALL RALLF
: POINTER FOR EMT CALL RALLR
: POINTER FOR EMT CALL SEQDRV
: POINTER FOR EMT CALL SELDRV

```

573
574 001242 012706 001000 START: MOV #SPBOT,R6 ;SET BOTTOM OF SP STACK.
575 001246 005067 177564 CLR RTNNO
576 001252 104010 TYPE ;TYPE TITLE.
577 001254 010333 PGTIT
578 001256 005737 000042 TST @#42 ;PROGRAM LOADED VIA MONITOR?
579 001262 001524 BEQ STRTA ;BR IF NOT.
580 ;ROUTINE TO DETERMINE TRANSPORTS AVAILABLE FOR TEST.
581 001264 012767 000402 001336 MOV #402,ERRND ;DISABLE ERROR PRINTOUTS.
582 001272 112767 000376 177566 MOV#B #376,UNITS ;ASSUME DRIVES 1-7 AVAILABLE.
583 001300 012700 000010 MOV #8,RO ;SET UP TO TEST 8 TIMES.
584 001304 005267 000010 DTRMN: INC SQDRV1
585 001310 042767 177770 000002 BIC #177770,SQDRV1
586 001316 104052 SELDRV ;SELECT A TRANSPORT.
587 001320 000000 SQDRV1: OPEN ;TRANSPORT #.
588 001322 000431 BR DTRMNA ;UNIT NOT AVAILABLE RETURN.
589 001324 104020 SETCOM ;REWIND TO REVERSE END ZONE.
590 001326 004002 RNUM+REV
591 001330 000437 BR DTRMNB ;ERROR RETURN.
592 001332 005777 177450 TST @TCCM ;WAIT.
593 001336 100375 BPL -4
594 001340 005777 177440 TST @TCST ;END ZONE?
595 001344 100031 BPL DTRMNB ;BR IF NOT.
596 001346 012777 011276 177436 MOV #WBUF0,@TCBA ;SET CURRENT ADDR.
597 001354 012777 177777 177426 MOV #-1,@TCWC ;SET WORD COUNT.
598 001362 104020 SETCOM ;YES. ISSUE WRITE DATA COMMAND.
599 001364 000015 WDATA+FWD+DO
600 001366 000420 BR DTRMNB ;ERROR RETURN.
601 001370 032777 100200 177410 BIT #BIT15+BIT7,@TCCM ;WAIT FOR ERROR/READY.
602 001376 001774 BEQ -6
603 001400 005777 177402 TST @TCCM ;ERROR?
604 001404 100411 BMI DTRMNB ;BR IF YES.
605 001406 104022 DTRMNA: STOPDT ;STOP DECTAPE.
606 001410 005300 DEC RO ;DONE 8 TIMES?
607 001412 001374 BNE DTRMN ;BR IF NOT.
608 001414 105767 177446 TSTB UNITS ;ANY UNITS AVAILABLE?
609 001420 001015 BNE DTRMNC ;BR IF YES.
610 001422 104010 TYPE ;TYPE NON AVAILABLE MESSAGE.
611 001424 011060 NOUNIT
612 001426 000571 BR CHNC ;GO EXIT.
613 001430 032777 014000 177346 DTRMNB: BIT #BIT12+BIT11,@TCST ;ILO OR SELE ERROR?
614 001436 001763 BEQ DTRMNA ;BR IF NOT.
615 001440 016701 177654 MOV SQDRV1,R1
616 001444 146167 006634 177414 BICB UNTAB(1),UNITS ;DESELECT NON AVAILABLE TRANSPORT.
617 001452 000755 BR DTRMNA
618 001454 104010 DTRMNC: TYPE ;TYPE UNITS TO BE TESTED.
619 001456 011226 GOOD
620 001460 012767 000001 000010 MOV #1,CPENA
621 001466 012767 000007 177350 MOV #7,CTRA ;CHECK UP TO 7 UNITS.
622 001474 104052 SELDRV ;SELECT DRIVE.
623 001476 000000 CPENA: OPEN ;UNIT TO BE SELECTED.
624 001500 000407 BR CPENB ;UNIT NOT AVAILABLE.
625 001502 016700 177356 MOV UNITN,RO ;SUCCESS.
626 001506 116067 011254 007535 MOV#B GTAB(0),GTAPES ;GET ASCII # FOR GOOD TAPE.
627 001514 104010 TYPE ;TYPE # OF UNIT TO TEST.
628 001516 011251 GTAPES

```

0015267	177752	OPENB:	INC	CPENA	:UPDATE TO NEXT DRIVE.
0015367	177314		DEC	CTRA	:CHECKED ALL DRIVES?
001361			BNE	CPENA-2	:BR IF NOT.
00041			BR	GETRDY	:YES.
104010		STRTA:	TYPE		:TYPE UNIT SELECT INSTRUCTIONS.
010361			INST1		
104015			CHALT		:WAIT FOR USER.
116767	176022		MOVB	SR UNITS	:GET UNITS.
001771	177316		BEG	STRTA	:BR IF NON SELECTED.
104010			TYPE		:TYPE SR OPTION MESSAGE.
010541			ASETSR		
104015			CHALT		:COMMON HALT.
012767	001002	GETRDY:	MOV	#1002 ERRNO	:ENABLE ERROR PRINTOUTS.
016767	177240		MOV	KSTART,NXTST	:ADDR OF 1ST ROUTINE TO NXTST
012767	000340	GTRDYX:	MOV	#PRTY7,PSW	:SET PRIORITY 7.
012706	001000		MOV	#SPBOT,R6	:SET BOTTOM OF STACK.
104001			SRESET		:ISSUE RESET.
104043			INBIN		:INITIALIZE BINARY COUNT.
005067	177274		CLR	ALLPAR	:CLEAR PARITY ERROR ALLOWED INDICATOR.
104017			SVECTR		:PRESET DT INTERRUPT VECTOR TO 0.
000000			0		
104767	000214	GTRDYA:	JSR	R7 FORWD	:ROLL FORWARD TO "NEXT" ROUTINE.
032767	001000	GTRDYB:	BIT	#BIT9,SR	:CHECK SELECT ROUTINE SWITCH
001003	175734		BNE	GTRDYC	:BRANCH IF SELECT ROUTINE SWITCH IS SET.
104051		GORUN:	SEODRV		:SELECT SEQUENTIAL DRIVE.
000177	177176		JMP	@COURTST	:GO RUN CURRENT ROUTINE.
126767	177166	GTRDYC:	CMFB	RTNNO,SR	:COMPARE RTNNO TO SR.
001771			BEG	GORUN	:BR IF ROUTINE FOUND.
022767	177777	GTRDYD:	CMF	#-1,NXTST	:NO. CHECK FOR LAST ROUTINE.
001357			BNE	GTRDYA	:BRANCH IF NOT LAST ROUTINE.
104010			TYPE		:TYPE INCORRECT RTN SELECTED.
010314			AINCR		
104015			CHALT		:COMMON HALT.
000733			BR	GETRDY	:START OVER.
104022		CHAINN:	STOPDT		
012706	001000		MOV	#SPBOT,R6	:RESTORE STACK.
005267	177122		INC	ICNT	:INCREMENT ITERATION COUNT.
001003			BNE	CHNAC	:BR IF RESULT NOT 0.
005167	177114		COM	ICNT	:RESULT 0. RESET ICNT TO -1.
032767	040000	CHNAC:	BIT	#BIT14,SR	:CHECK FOR SCOPE OPTION.
001403			BEG	CHNA	:BRANCH IF SCOPE SW NOT SET.
104051		CHNAB:	SEODRV		:SELECT SEQUENTIAL DRIVE.
000177	177102		JMP	@SCOPTA	:RETURN TO ROUTINE.
032767	004000	CHNA:	BIT	#BIT11,SR	:TEST INHIBIT ITERATION SWITCH
001003	175630		BNE	CHNAA	:BRANCH IF INHIBIT ITERATION SW SET.
005367	177060		DEC	ICTR	:DECREMENT ITERATION COUNT.
001367			BNE	CHNAB	:BRANCH IF COUNT NOT 0.
032767	002000	CHNAA:	BIT	#BIT10,SR	:ROUTINE END HALT SW SET? (SR10)
001403			BEG	CHNB	:BRANCH IF NOT SET.
016700	177052		MOV	RTNNO,R0	:TEST # TO R0.
000000			HALT		:ROUTINE END HALT. TEST # IN LIGHTS.
032767	001000	CHNB:	BIT	#BIT9,SR	:CHECK SELECT ROUTINE SWITCH
001271			BNE	GETRDY	:BRANCH IF SELECT RTN SW SET
022767	177777		CMF	#-1,NXTST	:LAST TEST?
001273	177034		BNE	GTRDYX	:BRANCH IF NOT LAST TEST.
104010			TYPE		:TYPE PROGRAM END BELL.

685	002010	010576			APGEN0		
686	002012	013700	000042		CHNC: MOV 3042,RO		:GET CONTENTS OF 42.
687	002016	001410			BEQ HERE		:BR IF 0.
688	002020	000005			RESET		:NOT 0. ISSUE PESET.
689	002022	004710			LOGIC: JSR PC,(0)		:RETURN TO MONITOR.
690	002024	000240	000240	000240	WORD NOP,NOP,NOP		
691	002032	105767	177030		TSTB UNITS		:ANY UNITS AVAILABLE FOR TESTING?
692	002036	001765			BEQ CHNC		:BR IF NOT.
693	002040	000647			HERE: BR GETDY		:GO REPEAT PROGRAM.
694	002042	016705	176772		FORWD: MOV NXTST,R5		:ADDR OF NEXT ROUTINE TO R5.
695	002046	012567	176764		MOV (5)+,RTNNO		:GET NEXT ROUTINE NUMBER.
696	002052	012567	176762		MOV (5)+,NXTST		:GET ADDR OF NEXT "NEXT" ROUTINE.
697	002056	012567	176744		MOV (5)+,ICTR		:GET ITERATION COUNT.
698	002062	012567	176746		MOV (5)+,SCOPTA		:GET SCOPE LOOP ENTRY POINTER.
699	002066	010567	176750		FORWA: MOV R5,CURTST		:ADDR OF NOW CURRENT TEST TO CURTST.
700	002072	012767	000001	176730	MOV #1,ICNT		:PRESET ICNT TO 1.
701	002100	016767	176732	175462	MOV RTNNO,SR		:DISPLAY ROUTINE #.
702	002106	000207			RTS R7		:EXIT FORWD SUBROUTINE.
703					:EMT INTERPRETER ROUTINE.		
704	002110	010046			EMTINT: MOV RO,-(6)		:PUSH RO.
705	002112	016600	000002		MOV 2(6),RO		:GET EMT PC.
706	002116	014000			MOV -(0),RO		:GET EMT CALL.
707	002120	006300			RO		:TIMES 2.
708	002122	016000	171114		MOV EMTTAB-10000(0),RO		:FORM EMT ADDR.
709	002126	000200			RTS RO		:GO TO EMT RTN. RESTORE RO.
710					:SAVE REGS 0 TO 3 SUBROUTINE.		
711	002130	012666	177766		SV03: MOV (6)+,-10.(6)		:MOVE PC UPSTACK.
712	002134	012666	177766		MOV (6)+,-10.(6)		:MOVE STATUS UPSTACK.
713	002140	012767	000002	000046	MOV #RTI,SV05C		
714	002146	000415			BR SV05B		
715					:SUB TO SAVE REGS 0 TO 5 AND PLACE EMT PC IN R5.		
716	002150	012767	000240	000036	SV05S: MOV #NOP,SV05C		
717	002156	000403			BR SV05A		
718					:SUB TO SAVE REGS 0 TO 5.		
719	002160	012767	000002	000026	SV05: MOV #RTI,SV05C		
720	002166	012666	177762		SV05A: MOV (6)+,-14.(6)		:MOVE PC AND PSW UPSTACK.
721	002172	012666	177762		MOV (6)+,-14.(6)		
722	002176	010546			MOV R5,-(6)		
723	002200	010446			MOV R4,-(6)		
724	002202	010346			SV05B: MOV R3,-(6)		
725	002204	010246			MOV R2,-(6)		
726	002206	010146			MOV R1,-(6)		
727	002210	010046			MOV RO,-(6)		
728	002212	024646			PUSH2		
729	002214	000002			SV05C: RTI		:RTI OR NOP.
730	002216	016605	000020		MOV 16.(6),R5		:EMT PC TO R5.
731	002222	010504			MOV R5,R4		
732	002224	005744			TST -(4)		
733	002226	000002			RTI		:EXIT.
734					:RESTORE REGS 0 TO 3 SUBROUTINE.		
735	002230	022626			PS03: POPSP2		
736	002232	012600			MOV (6)+,RO		:RESTORE REGS 0 TO 4.
737	002234	012601			MOV (6)+,R1		
738	002236	012602			MOV (6)+,R2		
739	002240	012603			MOV (6)+,R3		
740	002242	016646	177766		MOV -10.(6),-16)		:MOVE PC AND PSW DOWN STACK.

```

741 002246 016646 177766      MOV      -10.(6),-(6)
742 002252 000002      RTI      :EXIT.
743      :SUB TO SET R5 IN EMT PC AND RESTORE REGS 0 TO 5.
744 002254 010566 000020      RS05S:  MOV      R5,16.(6)      :SET EMT PC TO R5 CONTENTS.
745      :SUB TO RESTORE REGS 0 TO 5.
746 002260 022626      RS05:   POPSP2
747 002262 012600      MOV      (6)+,R0
748 002264 012601      MOV      (6)+,R1
749 002266 012602      MOV      (6)+,R2
750 002270 012603      MOV      (6)+,R3
751 002272 012604      MOV      (6)+,R4
752 002274 012605      MOV      (6)+,R5
753 002276 016646 177762      MOV      -14.(6),-(6)      :MOVE PC AND PSW DOWNSTACK.
754 002302 016646 177762      MOV      -14.(6),-(6)
755 002306 000002      RTI      :EXIT.
756 002310 012666 177772      SVSS:   MOV      (6)+,-6(6)      :PC AND PSW UPSTACK.
757 002314 012666 177772      MOV      (6)+,-6(6)
758 002320 010546      MOV      R5,-(6)      :SAVE R5.
759 002322 010446      MOV      R4,-(6)      :SAVE R4.
760 002324 024646      PUSH2
761 002326 016605 000010      MOV      8.(6),R5      :EMT PC TO R5.
762 002332 010504      MOV      R5,R4      :EMT PC TO R4.
763 002334 005744      TST      -(4)
764 002336 000002      RTI      :EXIT EMT SUB.
765 002340 010566 000010      RSS5:   MOV      R5,8.(6)      :R5 TO EMT PC.
766 002344 022626      POPSP2
767 002346 012604      MOV      (6)+,R4      :RESTORE R4.
768 002350 012605      MOV      (6)+,R5      :RESTORE R5.
769 002352 016646 177772      MOV      -6(6),-(6)
770 002356 016646 177772      MOV      -6(6),-(6)
771 002362 000002      RTI      :EXIT.
772      :ROUTINE TO SET T011 INTERRUPT VECTOR AND PRIORITY
773 002364 104006      STTCV:  SAVO5S
774 002366 016701 176424      MOV      TCVTR,R1      :VECTOR TO R1.
775 002372 012521      MOV      (5)+,(1)+      :SET DESIRED VECTOR.
776 002374 016721 176420      MOV      TCVLVL,(1)+    :SET T011 PRIORITY.
777 002400 104007      RSTOSS
778 002402 000002      RTI
779      :ROUTINE TO ISSUE RESET.
780 002404 010046      SRSETT: MOV      R0,-(6)      :PUSH R0.
781 002406 012700 052525      MOV      #52525,R0      :DATA TO R0.
782 002412 005100      COM      R0      :COMPLEMENT (R0).
783 002414 010067 177770      MOV      R0,SRSETT+4    : (R0) TO SRSETT+4.
784 002420 000005      RESET    :ISSUE RESET. (R0) IS
785 002422 012600      MOV      (6)+,R0      :RESTORE R0.
786 002424 000002      RTI      :DISPLAYED. EXIT.
787      :COMMON HALT ROUTINE
788 002426 104006      CHLT:   SAVO5S
789 002430 010400      MOV      R4,R0      :DEVELOP ADDR OF CALLER.
790 002432 000000      HALT    :HALT CALL ADDR IN DATA LIGHTS.
791 002434 104007      RSTOSS
792 002436 000002      RTI      :EXIT.
793      :CONDITIONAL ERROR HALT ROUTINE.
794 002440 005767 175124      EHLT:   TST      SR      :CHECK FOR HALT ON ERROR.
795 002444 100001      BPL     EHLTA      :BRANCH IF NO HALT DESIRED.
796 002446 000000      HALT    :HALT.

```


853	002656	000002				RTI		:EXIT EMT SUB.
854						:SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.		
855	002660	104006				TYP: SAVOSS		
856	002662	012500				MOV (5)+,RO		:ADDRESS OF MESSAGE TO RO.
857	002664	112001				TYPA: MOV (10)+,R1		:GET CHARACTER
858	002666	001006				BNE TYP		:BRANCH IF NOT TERMINATOR..
859	002670	112701	000177			MOV #177,R1		:OUTPUT RUBOUT.
860	002674	004767	000020			JSR R7,TYP		
861	002700	104007				RSTOSS		
862	002702	000002				RTI		:TERMINATOR CHAR. DONE. EXIT.
863	002704	122701	000045			TYP: CMPB #45,R1		:CHECK FOR"%".
864	002710	001411				BEQ TYFF		:BRANCH IF"%".
865	002712	004767	000002			JSR R7,TYP		:TYPE CHAR IN R1
866	002716	000762				BR TYP		
867	002720	110177	176100			TYPD: MOV R1,PTP		:OUTPUT CHARACTER TO PRINTER
868	002724	105777	176072			TSTB STP		:WAIT FOR DONE FLAG.
869	002730	100375				BPL -4		
870	002732	000207				RTS R7		:EXIT
871	002734	112701	000015			TYPF: MOV #15,R1		:MOVE CARRIAGE RETURN CODE TO R1
872	002740	004767	177754			JSR R7,TYP		:GO TYPE CHAR.
873	002744	112701	000012			TYPG: MOV #12,R1		:MOVE LF CODE TO R1.
874	002750	004767	177744			JSR R7,TYP		:GO TYPE CHAR.
875	002754	000743				BR TYP		
876						:SUBROUTINE TO DELAY.		
877		002762				DLYR=DLY+4		
878		002770				DLYR1=DLYR+4		
879	002756	012727	011610	000000		DLY: MOV #5000.,#0		
880	002764	012727	001750	000000		DLYA: MOV #1000.,#0		:DELAY COUNT TO DLYR1.
881	002772	005367	177772			DLYB: DEC DLYR1		:DLYR1 -1.
882	002776	001375				BNE DLYB		
883	003000	005367	177756			DEC DLYR		:DEC DLYR.
884	003004	001367				BNE DLYA		
885	003006	000002				RTI		:EXIT.
886						:SUBROUTINE TO INITIALIZE BINARY COUNT PATTERNS		
887	003010	012767	177777	000016		INBINN: MOV #-1,RIND		:SET ALL VARIABLES
888	003016	016767	000012	000012		MOV RIND,PTO		
889	003024	016767	000004	000006		MOV RIND,PTI		
890	003032	000002				RTI		:EXIT.
891	003034	000000				RIND: OPEN		
892	003036	000000				PTO: OPEN		
893	003040	000000				PTI: OPEN		
894						:SPECIAL BINARY COUNT PATTERN SUBROUTINE		
895	003042	016767	177770	177770		GTBINI: MOV PTO,PTI		:PREVIOUS BIN CHAR TO PTI
896	003050	005167	177764			COM PTI		
897	003054	005167	177754			COM RIND		
898	003060	001002				BNE +6		
899	003062	005267	177752			INC PTI		
900	003066	016767	177746	177742		MOV PTI,PTO		:SAVE BIN CHAR IN PTO
901	003074	000003				SAVSS		
902	003076	016725	177736			MOV PTI,(5)+		
903	003102	000004				RSTSS		
904	003104	000002				RTI		:EXIT.
905						:EMT SUB TO CONVERT OCTAL TO ASCII.		
906	003106	104006				OACNVV: SAVOSS		:SAVE REGS.
907	003110	013500				MOV (5)+,RO		:GET OCTAL VALUE.
908	003112	012501				MOV (5)+,R1		:GET DESTINATION ADDR.

909	003114	012502				MOV	(5)+,R2		:GET CONVERT COUNT.
910	003116	060201				ADD	R2,R1		:DEVELOP ADDR TO STORE 1ST CHAR.
911	003120	010003			OACNVA:	MOV	R0,R3		
912	003122	042703	177770			BIC	#177770,R3		:ISOLATE LEAST SIGNIFICANT DIGIT.
913	003126	062703	000050			ADD	#60,R3		:CONVERT DIGIT TO ASCII.
914	003132	110341				MOVSB	R3,-(1)		:STORE ASCII CHARACTER.
915	003134	042700	000007			BIC	#7,R0		
916	003140	006000				ROR	R0		
917	003142	006000				ROR	R0		
918	003144	006000				ROR	R0		
919	003146	005302				DEC	R2		:DONE ALL DIGITS?
920	003150	001363				BNE	OACNVA		:BRANCH IF NOT DONE.
921	003152	104007				RSTOSS			:RESTORE REGS.
922	003154	000002				RTI			:DONE. EXIT.
923									:EMT SUB TO MOVE VARIABLE NUMBER OF BYTES.
924	003156	104006			BMOVV:	SAVOSS			:SAVE REGS.
925	003160	012501				MOV	(5)+,R1		:GET "FROM" ADDRESS
926	003162	012502				MOV	(5)+,R2		:GET "TO" ADDRESS
927	003164	012503				MOV	(5)+,R3		:GET COUNT
928	003166	112122			BMOVA:	MOVSB	(1)+,(2)+		:MOVE BYTE
929	003170	005303				DEC	R3		:DECREMENT COUNT
930	003172	001375				BNE	BMOVA		:BRANCH IF NOT DONE.
931	003174	104007				RSTOSS			:RESTORE REGS.
932	003176	000002				RTI			:DONE. EXIT.
933									:EMT SUB TO CONVERT BINARY TO DECIMAL ASCII.
934	003200	104006			BDCNVV:	SAVOSS			:SAVE REGS.
935	003202	013501				MOV	3(5)+,R1		:GET BINARY VALUE.
936	003204	012700	003302			MOV	#DECVAL,R0		:ADDR OF DECVAL TO R0.
937	003210	012702	003270			MOV	#TENPWR,R2		:ADDR OF 10 POWER TO R2.
938	003214	012703	000005			MOV	#5,R3		:SET UP FOR 5 CONVERSIONS.
939	003220	005004			BDCNVA:	CLR	R4		:CLEAR RESULT.
940	003222	161201			BDCNVB:	SUB	(2),R1		:10 POWER FROM VALUE.
941	003224	103402				BCS	BDCNVC		:BR IF UNSUCCESSFUL.
942	003226	005204				INC	R4		:1 TO RESULT.
943	003230	000774				BR	BDCNVB		:DO IT AGAIN.
944	003232	061201			BDCNVC:	ADD	(2),R1		:RESTORE SUBTRACTED VALUE.
945	003234	062704	000060			ADD	#60,R4	:CONVERT	:RESULT TO ASCII.
946	003240	110420				MOVSB	R4,(0)+		:STORE RESULT.
947	003242	005722				TST	(2)+		:UPDATE 10 POWER ADDR.
948	003244	005303				DEC	R3		:DONE 5 TIMES?
949	003246	001364				BNE	BDCNVA		:BR IF NOT.
950	003250	012501				MOV	(5)+,R1		:GET ADDR TO STORE ASCII.
951	003252	012502				MOV	(5)+,R2		:GET # OF DIGITS REQUIRED.
952	003254	060201				ADD	R2,R1		:START WITH LSD.
953	003256	114041			BDCNVD:	MOVSB	-(0),-(1)		:TRANSFER CHARACTER.
954	003260	005302				DEC	R2		:DONE?
955	003262	001375				BNE	BDCNVD		:BR IF NOT.
956	003264	104007				RSTOSS			:RESTORE REGS.
957	003266	000002				RTI			:EXIT.
958	003270	023420			TENPWR:	10000.			
959	003272	001750				1000.			
960	003274	000144				100.			
961	003276	000012				10.			
962	003300	000001				1			
963	003302	040	040	040	DECVAL:	.BYTE	040,040,040,040,040,040		
964	003305	040	040	040					

H02

TC4 - TC11 TEST 4
DZTCDC.P11

MACY11 27(732) 08-SEP-76 09:04 PAGE 20

```
965          :EMT SUB TO SAVE TCCM, TCST, TCDT, TCWC, TCBA.
966 003310 017767 175470 175532 STATS: MOV @TCST,TCSTT ;SAVE TCST.
967 003316 017767 175464 175522      MOV @TCCM,TCGMT ;SAVE TCCM.
968 003324 017767 175460 175522      MOV @TCWC,TCWCT ;SAVE TCWC.
969 003332 017767 175456 175512      MOV @TCDT,TCDTT ;SAVE TCDT.
970 003340 017767 175446 175510      MOV @TCBA,TCBAT ;SAVE TCBA.
971 003346 000002          RTI ;EXIT EMT SUB.
972          :EMT SUB TO ISSUE DT COMMAND SPECIFIED AT CALL+2.
973 003350 005067 175514 STCOM: CLR COMND ;CLEAR PREVIOUS COMMAND
974 003354 016767 175502 175506      MOV UNIT,COMND ;UNIT # TO COMND.
975 003362 057667 000000 175500      BIS @(&),COMND ;SET DESIRED COMMAND IN COMND.
976 003370 016777 175474 175410      MOV COMND,@TCCM ;ISSUE COMMAND.
977 003376 032777 100200 175402      BIT #BIT15!BIT7,@TCCM ;READY AND ERROR BIT CLEAR?
978 003404 001414          BEQ STCOMB ;BR IF YES.
979 003406 032767 000001 175454      BIT #BIT0,COMND ;WAS THE DO BIT SET IN COMND?
980 003414 001410          BEQ STCOMB ;BR IF NOT.
981 003416 000003          SAVSS
982 003420 104021          STATUS ;SAVE STATUS.
983 003422 104011          ERRORN ;ERROR. DO BIT FAILED TO CAUSE CLEARING
984 003424 011207          FPCMSG
985 003426 011105          STCMSG ;OF READY AND/OR ERROR BIT(S), OR ILO,
986 003430 010634          STAT ;BLOCK MISS, OR DATA MISS ERROR OCCURRED.
987 003432 177777          -1
988 003434 104000          SCOPE
989 003436 062716 000002 STCOMB: ADD #2,(&) ;SET UP RETURN.
990 003442 000002          RTI ;EXIT STCOM SUB.
991          :EMT SUB TO STOP ALL DECTAPES.
992 003444 042777 000116 175334 STPDT: BIC #116,@TCCM ;ISSUE SAT COMMAND.
993 003452 000002          RTI ;EXIT EMT SUB.
994          :EMT SUB TO CHECK FOR DECTAPE ERROR OR END ZONE.
995 003454 000003          CKERZ: SAVSS
996 003456 005777 175324          TST @TCCM ;ERROR BIT SET?
997 003462 100406          BMI CKERZC ;BR IF YES.
998 003464 005725          TST (5)+ ;NO. SET UP OK EXIT.
999 003466 005725          CKERZA: TST (5)+
1000 003470 000004          CKERZB: RSTSS
1001 003472 005067 175414          CLR ALLPAR ;CLEAR PARITY ERR ALLOWED INDICATOR.
1002 003476 000002          RTI ;EXIT EMT SUB.
1003 003500 005777 175300          CKERZC: TST @TCST ;ENDZ BIT SET?
1004 003504 100770          BMI CKERZA ;BR IF YES.
1005 003506 005767 175400          TST ALLPAR ;PARITY ERR ALLOWED?
1006 003512 001404          BEQ CKERZD ;PARITY ERR NOT ALLOWED.
1007 003514 032777 040000 175262      BIT #BIT14,@TCST ;PARITY ERR SET?
1008 003522 001360          BNE CKERZA-2 ;BR IF YES.
1009 003524 104021          CKERZD: STATUS
1010 003526 104011          ERRORN ;. CTAPE ERROR.
1011 003530 011207          FPCMSG
1012 003532 010704          DTERR
1013 003534 010634          STAT
1014 003536 010715          BLKSB
1015 003540 177777          -1
1016 003542 000752          BR CKERZB
1017          :EMT SUB TO HANDLE FAILURE TO INTERRUPT.
1018 003544 000003          NOINTR: SAVSS
1019 003546 104021          STATUS ;SAVE STATUS.
1020 003550 104011          ERRORN ;DECTAPE FAIL TO INTERRUPT.
```

1021	003552	011207				FPCMSG			
1022	003554	010665				INTFAI			
1023	003556	010634				STAT			
1024	003560	177777				-1			
1025	003562	000004				RSTSS			
1026	003564	000002				RTI			
1027						:EMT SUB TO CHECK EXPECTED DATA		:EXIT EMT SUB.	
1028	003566	000000				DATEKT: OPEN		:CURRENT WORD NUMBER.	
1029	003570	000000				ERRCTR: OPEN		:ERROR COUNTER.	
1030	003572	000000				WRDCNT: OPEN		:# OF WORDS TO CHECK.	
1031	003574	000000				BEXPDT: OPEN			
1032	003576	000000				EXPDAT: OPEN			
1033	003600	000000				BACTDT: OPEN			
1034	003602	000000				ACTDAT: OPEN			
1035	003604	000				CKINDA: .BYTE	OPEN	:INCR/DECR INDICATOR.	
1036	003605	000				CKINDB: .BYTE	OPEN	:16/18 BIT DATA INDICATOR	
1037	003606	112767	177777	177770		ADATCI: MOVB	#-1,CKINDA	:INDICATE DECREMENT OF ACT DATA.	
1038	003614	000402				BR	ADATCK+4		
1039	003616	105067	177762			ADATCK: CLRB	CKINDA	:INDICATE INCREMENT OF ACT DATA.	
1040	003622	112767	177777	177755		MOVB	#-1,CKINDB	:INDICATE 18 BIT DATA CHECK.	
1041	003630	000410				BR	DATCKK		
1042	003632	112767	177777	177744		DTCKI: MOVB	#-1,CKINDA	:INDICATE DECREMENT OF ACT DATA.	
1043	003640	000402				BR	DATCK+4		
1044	003642	105067	177736			DATCK: CLRB	CKINDA	:INDICATE INCREMENT OF ACT DATA.	
1045	003646	105067	177733			CLRB	CKINDB	:INDICATE 16 BIT DATA CHECK.	
1046	003652	104006				DATCKK: SAVOSS			
1047	003654	012500				MOV	(5)+,R0	:GET EXP DATA ADDR.	
1048	003656	012501				MOV	(5)+,R1	:GET ACT DATA ADDR.	
1049	003660	105767	177721			TSTB	CKINDB	:16 OR 18 BIT DATA?	
1050	003664	001402				BEQ	DATCKA	:BR IF 16 BIT DATA.	
1051	003666	012502				MOV	(5)+,R2	:GET BEXP DATA ADDR.	
1052	003670	012503				MOV	(5)+,R3	:GET BACT DATA ADDR.	
1053	003672	012567	177674			DATCKA: MOV	(5)+,WRDCNT	:GET # OF WORDS TO CHECK.	
1054	003676	012767	000001	177662		MOV	#1,DATKNT	:SET CURRENT WORD # TO 1.	
1055	003704	016767	175176	177656		MOV	ERRLIM,ERRCTR	:ERR LIMIT TO ERROR COUNTER.	
1056	003712	005067	177656			DATCKB: CLR	BEXPDT		
1057	003716	005067	177656			CLR	BACTDT		
1058	003722	011067	177650			MOV	(0),EXPDAT	:GET EXP DATA WORD.	
1059	003726	011167	177650			MOV	(1),ACTDAT	:GET ACT DATA WORD.	
1060	003732	105767	177647			TSTB	CKINDB	:16 OR 18 BIT DATA?	
1061	003736	001412				BEQ	DATCKC	:BR IF 16 BIT DATA.	
1062	003740	111267	177630			MOVB	(2),BEXPDT	:GET BEXP DATA BYTE.	
1063	003744	111367	177630			MOVB	(3),BACTDT	:GET BACT DATA BYTE.	
1064	003750	042767	177774	177616		BIC	#177774,BEXPDT	:ISOLATE 2 LSD IN BEXPDT AND	
1065	003756	042767	177774	177614		BIC	#177774,BACTDT	:BACTDT.	
1066	003764	026767	177606	177610		DATCKC: CMP	EXPDAT,ACTDAT	:COMPARE ACT DATA AND EXP DATA.	
1067	003772	001004				BNE	DATCKD	:BR IF NOT SAME.	
1068	003774	026767	177574	177576		CMP	BEXPDT,BACTDT	:SAME. COMPARE BACT AND BEXP DATA.	
1069	004002	001450				BEQ	DATCKE	:BR IF SAME.	
1070	004004	104013				DATCKD: BDCNV		:DATA NOT SAME. CONVERT WORD # TO DECIMAL ASCII.	
1071	004006	003566				DATKNT			
1072	004010	011022				AWDCNT			
1073	004012	000004				4			
1074	004014	006167	177556			ROL	EXPDAT	:SET UP DATA FOR CONVERSION AND TYPEOUT.	
1075	004020	006167	177550			ROL	BEXPDT		
1076	004024	006067	177546			ROR	EXPDAT		

1077	004030	006167	177546	RUL	ACTDAT	
1078	004034	006167	177540	ROL	BACTDT	
1079	004040	006067	177536	ROR	ACTDAT	
1090	004044	104012		OACNV		; CONVERT BEXP DATA TO ASCII.
1081	004046	003574		BEXPDT		
1082	004050	011035		ADATSB		
1083	004052	000001		1		
1084	004054	104012		OACNV		; CONVERT EXP DATA TO ASCII.
1085	004056	003576		EXPDAT		
1086	004060	011036		ADATSB+1		
1087	004062	000005		5		
1088	004064	104012		OACNV		; CONVERT BACT DATA TO ASCII.
1089	004066	003600		BACTDT		
1090	004070	011051		ADATWS		
1091	004072	000001		1		
1092	004074	104012		OACNV		; CONVERT ACT DATA TO ASCII.
1093	004076	003602		ACTDAT		
1094	004100	011052		ADATWS+1		
1095	004102	000005		5		
1096	004104	104011		ERRORN		; TYPE DATA ERROR MESSAGE.
1097	004106	011207		FPCMSG		
1098	004110	010715		BLKSB		
1099	004112	011002		DATERR		
1100	004114	177777		-1		
1101	004116	005367	177446	DEC	ERRCTR	; NTH ERROR?
1102	004122	001423		BEQ	DATCKH	; BR IF YES.
1103	004124	005267	177436	INC	DATKNT	; INCREMENT WORD #
1104	004130	105767	177450	TSTB	CKINDA	; INCR/DECR?
1105	004134	001406		BEQ	DATCKF	; BR TO INCR.
1106	004136	105767	177443	TSTB	CKINDB	
1107	004142	001401		BEQ	.+4	
1108	004144	122243		CMPB	(2)+, -(3)	; INCR-DECR DATA ADDRESSES.
1109	004146	022041		CMP	(0)+, -(1)	
1110	004150	000405		BR	DATCKG	
1111	004152	105767	177427	TSTB	CKINDB	
1112	004156	001401		BEQ	.+4	
1113	004160	122223		CMPB	(2)+, (3)+	; INCR-INCR DATA ADDRESSES.
1114	004162	022021		CMP	(0)+, (1)+	
1115	004164	005367	177402	DATCKG:	DEC	WRDCN1
1116	004170	001250		BNE	DATCKF	; DONE CHECKING?
1117	004172	104007		DATCKH:	RSTOSS	; BR IF NOT.
1118	004174	000002		RTI		; DONE.
1119						; EXIT.
1120	004176	104006				; EMT SUB TO CLEAR SPECIFIED AREA TO 0'S.
1121	004200	012500		CLEAR:	SAVOSS	
1122	004202	012501		MOV	(5)+, R0	; GET STARTING ADDR.
1123	004204	005020		MOV	(5)+, R1	; GET COUNT.
1124	004206	005301		CLR	(0)+	; CLEAR WORD.
1125	004210	001375		DEC	R1	; DONE?
1126	004212	104007		BNE	.-4	; BR IF NOT DONE.
1127	004214	000002		RSTOSS		; DONE
1128				RTI		; EXIT.
1129						; EMT SUB TO FILL AREA WITH BINARY COUNT PATTERN.
1130	004216	104006		BINFLL:	SAVOSS	
1131	004220	012500		MOV	(5)+, R0	; GET STARTING ADDR.
1132	004222	012501		MOV	(5)+, R1	; GET COUNT.
1133	004224	104044		BINFLA:	GETBN1	; GET BINARY WORD.

```

1133 004226 000000          BINFLB: OPEN          ; BINARY WORD IS STORED HERE.
1134 004230 016720 177772      MOV      BINFLB,(0)+  ; STORE WORD.
1135 004234 005301          DEC      R1           ; DONE?
1136 004236 001372          BNE     BINFLA       ; BR IF NOT DONE.
1137 004240 104007          RSTOSS             ; DONE.
1138 004242 000002          RTI              ; EXIT.
1139
1140          ; EMT SUB TO CHECK THAT WORD COUNT IS 0, AND THAT TCBA CONTENTS
1141          ; MATCH THE EXPECTED CONTENTS.
1142 004244 000003          CWCBA: SAVSS
1143 004246 012567 174620      MOV      (5)+,TEMP  ; GET EXPECTED TCBA CONTENTS.
1144 004252 104021          STATUS           ; SAVE TCWC AND TCBA.
1145 004254 005777 174500      TST     @TCWC      ; WORD COUNT 0?
1146 004260 001407          BEQ     CWCBB      ; BR IF 0 (OK).
1147 004262 104011          ERRORN          ; WORD COUNT NOT 0. TYPE
1148 004264 011107          FPCMSG          ; CONTENTS OF TCWC AND TCBA.
1149 004270 010602          WCNOTO
1150 004272 010617          CTCWC
1151 004274 177777          CTCBA
1152 004276 104000          -1
1153 004300 026777 174566 174504 CWCBB: CMP     TEMP,@TCBA  ; TCBA AND EXPECTED TCBA SAME?
1154 004306 001414          BEQ     CWCBC      ; BR IF YES (OK).
1155 004310 104012          OACNV          ; NO. CONVERT EXPECTED TCBA TO ASCII.
1156 004312 001072          TEMP
1157 004314 010771          ATCBAS
1158 004316 000006          6
1159 004320 104011          ERRORN          ; TCBA DOES NOT MATCH EXPECTED
1160 004322 011207          FPCMSG          ; TCBA CONTENTS. TYPE EXPECTED TCBA,
1161 004324 010747          INCTCB         ; ACTUAL TCBA, AND TCWC.
1162 004326 010763          TCBA5B
1163 004330 010617          CTCBA
1164 004332 010602          CTCWC
1165 004334 177777          -1
1166 004336 104000          SCOPE
1167 004340 000004          CWCBC: RSTSS
1168 004342 000002          RTI              ; EXIT.
1169
1170          ; EMT SUBS TO SEARCH FOR DESIRED BLOCK NUMBER. SRCHFF GETS FORWARD
1171          ; BLOCK NUMBERS. SRCHRR GETS REVERSE BLOCK NUMBERS.
1172 004344 105067 000315      SRCHFF: CLRB     DIRIND ; SET FORWARD INDICATOR.
1173 004350 000403          BR      SRCHA
1174 004352 112767 177777 000305 SRCHRR: MOVB    #-1,DIRIND ; SET REVERSE INDICATOR.
1175 004360 012777 004440 174430 SRCHA:  MOV     #SRCHC,@TCVTR ; SET INTERRUPT VECTOR TO SRCHC.
1176 004366 112767 000005 000270      MOVB    #5,REVCNT  ; SET MAX # OF REVERSALS ALLOWED.
1177 004374 052767 004000 000020      BIS     #REV,SRCHM ; SET REV BIT IN SRCHM.
1178 004402 032777 004000 174376      BIT     #REV,@TCCM ; REV BIT SET IN TCCM?
1179 004410 001003          BNE     SRCHAA     ; BR IF YES.
1180 004412 042767 004000 000002      BIC     #REV,SRCHM ; NO. CLEAR REV BIT FROM SRCHM.
1181 004420 104020          SRCHAA: SETCOM
1182 004422 000103          SRCHM: RNUM! IE! DO ; START SEARCH.
1183 004424 000402          BR      SRCHB
1184 004426 005277 174354      SRCON: INC     @TCCM ; ISSUE DO TO ENABLE RNUM.
1185 004432 104400          SRCHB: DELAY    ; TIME OUT INTERRUPT.
1186 004434 104024          NOINT
1187 004436 104000          SCOPE
1188 004440 012716 004446      SRCHC: MOV     #SRCHD,(6) ; HERE WHEN INTERRUPT OCCURS.
1189 004444 000002          RTI              ; EXIT TO SRCHD.

```

L02

TC4 - TC11 TEST 4
DZTCDC.P11

MACY11 27(732) 08-SEP-76 09:04 PAGE 24

1189	004446	022626			SRCHD: POPSP2				;RESTORE STACK.
1190	004450	005777	174332		TST	@TCOM			;ERROR BIT SET?
1191	004454	100003			BPL	SRCHDA			;BR IF NOT.
1192	004456	104023			CKERRZ				;CHECK FOR ERROR/ENDZ.
1193	004460	104000			SCOPE				;ERROR RETURN. SCOPE.
1194	004462	000451			BR	SRCREV			;ENDZ. GO REVERSE DIRECTION.
1195	004464	027767	174324	174366	SRCHDA: CMP	@TCDT, BLKRQ			;COMPARE BLK# IN TCDT TO REQUIRED BLK.
1196	004472	001431			BEQ	SRCF			;BR IF BLK FOUND.
1197	004474	003014			BGT	SRCHE			;BR IF TCDT HIGH.
1198	004476	032777	004000	174302	BIT	#BIT11, @TCOM			;TCDT LOW. CHECK DIRECTION.
1199	004504	001750			BEQ	SRCCON			;BR IF GOING FWD. CONTINUE SAME DIRECTION.
1200	004506	062777	000003	174300	ADD	#3, @TCDT			;ADD 3 TO TCDT.
1201	004514	027767	174274	174336	CMP	@TCDT, BLKRQ			;LOWER BY 3 OR MORE?
1202	004522	101435			BLOS	SRCRVA			;GO REVERSE IF LOWER BY 3 OR MORE.
1203	004524	000740			BR	SRCCON			;NOT LOW ENOUGH. CONTINUE SAME DIRECTION.
1204	004526	032777	004000	174252	SRCHD: BIT	#BIT11, @TCOM			;TCDT HIGH. CHECK DIRECTION.
1205	004534	001334			BNE	SRCCON			;BR IF IN REVERSE. CONTINUE DIRECTION.
1206	004536	162777	000003	174250	SUB	#3, @TCDT			;SUBTRACT 3 FROM TCDT.
1207	004544	026777	174310	174242	CMP	BLKRQ, @TCDT			;HIGHER BY 3 OR MORE?
1208	004552	003425			BLE	SRCRVB			;GO REVERSE IF HIGHER BY 3 OR MORE.
1209	004554	000724			BR	SRCCON			;NOT HIGH ENOUGH. CONTINUE DIRECTION.
1210	004556	032777	004000	174222	SRCF: BIT	#BIT11, @TCOM			;TCDT EQUAL. CHECK DIRECTION.
1211	004564	001004			BNE	SRCCHG			;BR IF IN REVERSE.
1212	004566	105767	000073		TSTB	DIRIND			;GOING FORWARD. FWD BLK# WANTED?
1213	004572	001315			BNE	SRCCON			;BR IF FWD BLK# NOT WANTED.
1214	004574	000002			RTI				;EXIT EMT SUB.
1215	004576	105767	000063		SRCHG: TSTB	DIRIND			;GOING REV. REV BLK# WANTED?
1216	004602	001711			BEQ	SRCCON			;BR IF REV BLK# NOT WANTED.
1217	004604	000002			RTI				;REV BLK# WANTED. EXIT.
1218	004606	032777	004000	174172	SRCREV: BIT	#BIT11, @TCOM			;REV BIT SET?
1219	004614	001404			BEQ	SRCRVB			;BR IF NOT.
1220	004616	042777	004000	174162	SRCRVA: BIC	#BIT11, @TCOM			;IN REVERSE. SET TO FORWARD.
1221	004624	000403			BR	SRCRVC			
1222	004626	052777	004000	174152	SRCRVB: BIS	#BIT11, @TCOM			;FORWARD. SET TO REVERSE.
1223	004634	105367	000024		SRCRVC: DECB	REVCNT			;FIFTH REVERSAL ISSUED?
1224	004640	001272			BNE	SRCCON			;BR IF NOT.
1225	004642	104021			STATUS				;YES. ERROR. SAVE STATUS.
1226	004644	000003			SAVSS				
1227	004646	104011			ERRORN				;BLK# NOT FOUND WITHIN 5 TAPE
1228	004650	011207			FPCMSG				;REVERSALS.
1229	004652	010715			BLKSB				
1230	004654	011134			SRCHER				
1231	004656	010634			STAT				
1232	004660	177777			-1				
1233	004662	104000			SCOPE				
1234	004664	000			REVCNT: .BYTE	OPEN			
1235	004665	000			DIRIND: .BYTE	OPEN			
1236					;EMT SUBS TO WDATA, RDATA, FORWARD OR REVERSE.				
1237	004666	012767	005067	000072	WRDFR: MOV	#5067, WRDFRZ			
1238	004674	000003			WRDFRN: SAVSS				
1239	004676	012577	174106		MOV	(5)+, @TCWC			;GET WORD COUNT AND SET IN TCWC
1240	004702	017767	174102	000072	MOV	@TCWC, WRDFRG			;2(WORD COUNT) TO WRDFRG.
1241	004710	006367	000066		ASL	WRDFRG			
1242	004714	005477	174070		NEG	@TCWC			;IN 2'S COMPLEMENT FORM.
1243	004720	012577	174066		MOV	(5)+, @TCBA			;SET ADDR IN TCBA.
1244	004724	067767	174062	000050	ADD	@TCBA, WRDFRG			;2(WORD COUNT)+TCBA=FINAL TCBA CONTENTS.


```

1245 004732 000000 WRDFRA: OPEN ;SRCHF OR SRCHR CALL GOES HERE.
1246 004734 012777 004754 174054 MOV #WRDFRC, @TCVTR ;SET INTERRUPT VECTOR TO WRDFRC.
1247 004742 104020 SETCOM ;ISSUE WDATA OR RDATA.
1248 004744 000000 WRDFRB: OPEN ;COMMAND GOES HERE.
1249 004746 104400 DELAY ;TIMEOUT INTERRUPT.
1250 004750 104024 NOINT ;FAILURE TO INTERRUPT.
1251 004752 104000 SCOPE
1252 004754 022626 WRDFRC: POPSP2 ;HERE WHEN INTERRUPT OCCURS.
1253 004756 022626 WRDFRD: POPSP2 ;RESTORE STACK.
1254 004760 005777 174022 TST @TCWC ;ERROR BIT SET?
1255 004764 100005 BPL WRDFRF ;BR IF NOT.
1256 004766 005067 174120 WRDFRZ: CLR ALLPAR
1257 004772 104023 CKERRZ
1258 004774 104000 SCOPE ;CHECK FOR ERRORS.
1259 004776 000240 NOP ;ERROR RETURN.
1260 005000 104034 WRDFRF: CKWCBA ;ENDZ RETURN.
1261 005002 000000 WRDFRG: OPEN ;CHECK WORD COUNT AND CURRENT ADDR.
1262 005004 000004 WRDFRE: RSTSS ;TCBA SHOULD EQUAL THIS.
1263 005006 000002 RTI ;EXIT.
1264 005010 012767 000115 177726 WDATF: MOV #WDATA!FWD!IE!DO, WRDFRB
1265 005016 000403 BR RDATA+6
1266 005020 012767 000105 177716 RDATAF: MOV #RDATA!FWD!IE!DO, WRDFRB
1267 005026 012767 104025 177676 MOV #SRCHF, WRDFRA
1268 005034 000714 BR WRDFR
1269 005036 012767 004115 177700 WDATR: MOV #WDATA!REV!IE!DO, WRDFRB
1270 005044 000403 BR RDATA+6
1271 005046 012767 004105 177670 RDATA: MOV #RDATA!REV!IE!DO, WRDFRB
1272 005054 012767 104026 177650 MOV #SRCHR, WRDFRA
1273 005062 000701 BR WRDFR
1274 005064 012767 005167 177674 RDTFSS: MOV #S167, WRDFRZ
1275 005072 012767 000105 177644 MOV #RDATA!FWD!IE!DO, WRDFRB
1276 005100 012767 104025 177624 MOV #SRCHF, WRDFRA
1277 005106 000672 BR WRDFRN
1278 ;WRITE ALL SUBROUTINE. FORWARD OR REVERSE.
1279 ;CALL: WALLF OR WALLR ;WRITE ALL FORWARD OR REVERSE
1280 ;COUNT ;TRANSFER COUNT
1281 ;DADDR ;DATA ADDR
1282 ;EDADDR ;EXTENDED DATA ADDR.
1283 005110 012767 000117 000056 WALLFF: MOV #WALL!FWD!IE!DO, CWALLB ;SETUP WRITE ALL FORWARD
1284 005116 012767 104025 000036 MOV #SRCHF, CWALLA
1285 005124 000406 BR CWALL
1286 005126 012767 004117 000040 WALLRR: MOV #WALL!REV!IE!DO, CWALLB ;SETUP WRITE ALL REVERSE
1287 005134 012767 104026 000020 MOV #SRCHR, CWALLA
1288 005142 104006 CWALL: SAVO55
1289 005144 005077 173640 CLR @TCWC ;ZERO TCWC
1290 005150 005077 173636 CLR @TCBA ;ZERO TCBA
1291 005154 012500 MOV (S)+, R0 ;TRANSFER COUNT TO R0.
1292 005156 012501 MOV (S)+, R1 ;DATA ADDR TO R1.
1293 005160 012502 MOV (S)+, R2 ;EXTENDED DATA ADDR TO R2.
1294 005162 000000 CWALLA: OPEN ;SRCHF OR SRCHR CALL.
1295 005164 012777 005204 173624 MOV #CWALLC, @TCVTR ;SET INTERRUPT VECTOR TO CWALLC.
1296 005172 104020 SETCOM ;ISSUE WALLF OR WALLR
1297 005174 000000 CWALLB: OPEN ;COMMAND GOES HERE.
1298 005176 104400 DELAY ;WAIT FOR INTERRUPT
1299 005200 104024 NOINT ;FAILURE TO INTERRUPT.
1300 005202 104000 SCOPE

```

1301	005204	112277	173574	CWALLC:	MOVB	(2)+, @TCST	; HERE ON INTERRUPT. LOAD EXTENDED DATA BITS IN TCST
1302	005210	012177	173600		MOV	(1)+, @TCDT	; LOAD DATA IN TCDT
1303	005214	005777	173566		TST	@TCOM	; ERROR BIT SET?
1304	005220	100003			BPL	CWALLD	; BR IF NO ERROR.
1305	005222	104023			CKERRZ		; GO CHECK ON ERROR.
1306	005224	104000			SCOPE		; ERROR RETURN.
1307	005226	000240			NOP		; ENDZ RETURN.
1308	005230	005300		CWALLD:	DEC	RD	; ALL TRANSFERS DONE?
1309	005232	001401			BEQ	CWALLE	; BR IF YES.
1310	005234	000002			RTI		; NO. EXIT INTERRUPT
1311	005236	042777	000100 173542	CWALLE:	BIC	#IE, @TCOM	; DISABLE INTERRUPT.
1312	005244	032777	001000 173532	IS:	BIT	#1000, @TCST	; WAIT FOR DATA MISS.
1313	005252	001774			BEQ	IS	; BR IF NONE YET.
1314	005254	022626		CWALLF:	POPSP2		; RESTORE STACK TO STATE BEFORE DELAY.
1315	005256	022626			POPSP2		
1316	005260	104007			RSTOSS		; RESTORE REGS.
1317	005262	000002			RTI		; EXIT WALL SUBROUTINE

```

1318 :READ ALL SUBROUTINE. FORWARD OR REVERSE.
1319 :CALL: RALLF OR RALLR :READ ALL FORWARD OR REVERSE
1320 :COUNT :TRANSFER COUNT
1321 :DADDR :DATA ADDR
1322 :EDADDR :EXTENDED DATA ADDR.
1323
1324 005264 012767 000107 000056 RALLFF: MOV @RALL!FWD!IE!DO,CRALLB ;SETUP READ ALL FORWARD.
1325 005272 012767 104025 000036 MOV @SRCHF,CALLA
1326 005300 000406 BR CALL
1327 005302 012767 004107 000040 RALLRR: MOV @RALL!REV!IE!DO,CRALLB ;SETUP READ ALL REVERSE.
1328 005310 012767 104026 000020 MOV @SRCHR,CALLA
1329 005316 104006 CRALL: SAVOSS
1330 005320 005077 173464 CLR @TCWC ;ZERO TCWC
1331 005324 005077 173462 CLR @TCBA ;ZERO TCBA
1332 005330 012500 MOV (5)+,R0 ;TRANSFER COUNT TO R0.
1333 005332 012501 MOV (5)+,R1 ;DATA ADDR TO R1
1334 005334 012502 MOV (5)+,R2 ;EXTENDED DATA ADDR TO R2.
1335 005336 000000 CRALLA: OPEN ;SRCHF OR SRCHR CALL.
1336 005340 012777 005360 173450 MOV @CRALLC,@TCVTR ;SET INTERRUPT VECTOR TO CRALLC.
1337 005346 104020 SETCOM ;ISSUE RALLF OR RALLR
1338 005350 000000 CRALLB: OPEN ;COMMAND GOES HERE.
1339 005352 104400 DELAY ;WAIT FOR INTERRUPT
1340 005354 104024 NOINT ;FAILURE TO INTERRUPT.
1341 005356 104000 SCOPE
1342 005360 117722 173420 CRALLC: MOV @TCST,(2)+ ;STORE EXTENDED DATA BITS
1343 005364 017721 173424 MOV @TCDT,(1)+ ;STORE DATA
1344 005370 005777 173412 TST @TCOM ;ERROR BIT SET?
1345 005374 100003 BPL CRALLD ;BR IF NO ERROR.
1346 005376 104023 CKERRZ ;GO CHECK ON ERROR.
1347 005400 104000 SCOPE ;ERROR RETURN.
1348 005402 000240 NOP ;ENDZ RETURN.
1349 005404 005300 CRALLD: DEC R0 ;ALL TRANSFERS DONE?
1350 005406 001401 BEQ CRALLE ;BR IF YES.
1351 005410 000002 RTI ;NO. EXIT INTERRUPT
1352 005412 112777 000002 173366 CRALLE: MOV @RNUM,@TCOM ;STOP RALL BY SWITCHING TO RNUM COMMAND.
1353 005420 022626 CRALLF: POPSP2 ;RESTORE STACK TO STATE BEFORE DELAY.
1354 005422 022626 POPSP2
1355 005424 104007 RSTOSS ;RESTORE REGS.
1356 005426 000002 RTI ;EXIT RALL SUBROUTINE

```

```

1355 :SUBROUTINE TO CALCULATE FORWARD CHECKSUM FOR 256 WORD DATA BLOCK.
1356 : 2 MSB BITS OF 6 BIT CHKSUM ARE STORED AT LOC ELPB (RIGHT JUSTIFIED)
1357 : THE OTHER 4 BITS ARE STORED AT LOC LPB (LEFT JUSTIFIED)
1358 :SUBROUTINE CALL IS: JSR R5,PARITY ;CALL TO PARITY SUBROUTINE
1359 : ADDR ;ADDR OF DATA STRING
1360 : EADR ;ADDR OF EXTENDED DATA STRING.
1361
1362 PARITY: SAVO3
1363 CLR LPB ;CLEAR CHECKSUM WORDS.
1364 CLR ELPB
1365 MOV (5)+,R0 ;DATA STRING ADDR TO R0.
1366 MOV (5)+,R2 ;EXTENDED DATA STRING ADDR TO R2.
1367 MOV #256,PARCTR ;SETUP TO COMPUTE PARITY FOR 256 WORDS.
1368 MOV (0)+,PWORD ;MOVE DATA WORD TO PWORD
1369 MOV PWORD,R1 ;AND TO R1.
1370 MOVB (2)+,R3 ;EXTENDED DATA BYTE TO R3.
1371 ROR R3
1372 ROR R1
1373 ROR R3
1374 ROR R1
1375 SWAB R1 ;PREPARE TO COMPUTE PARITY ON 6 MOST SIGNIFICANT
1376 JSR PC,ASR2 ;BITS. INCLUDES EXTENDED BITS 16 AND 17.
1377 BIC #177700,R1 ;GO COMPUTE PARITY.
1378 JSR PC,BITCOM ;PREPARE CENTER 6 BITS FOR PARITY COMPUTATION
1379 MOV PWORD,R1 ;GO COMPUTE PARITY
1380 JSR PC,ASR6 ;PREPARE 6 LEAST SIGNIFICANT BITS FOR PARITY
1381 BIC #177700,R1 ;COMPUTATION
1382 JSR PC,BITCOM ;GO COMPUTE PARITY.
1383 MOV PWORD,R1 ;DONE 256 TIMES?
1384 BIC #177700,R1 ;BR IF NOT.
1385 JSR PC,BITCOM
1386 DEC PARCTR
1387 BNE PARTYA
1388 RSTO3
1389 COM LPB
1390 BIC #177700,LPB ;MOVE 2 MOST SIGNIFICANT BITS OF
1391 MOV LPB,LPB1 ;SAVE COMPUTED PARITY.
1392 MOV LPB,LPB2
1393 SWAB LPB ;COMPUTED PARITY TO ELPB (RIGHT JUSTIFIED)
1394 MOV #4,PARCTR ;AND 4 LEAST SIGNIFICANT TO LPB (LEFT JUSTIFIED)
1395 ASL LPB
1396 ROL ELPB
1397 DEC PARCTR
1398 BNE PARTYB
1399 RTS R5 ;EXIT.
1400
1401 PWORD: OPEN
1402 PARCTR: OPEN
1403 ASR6: ASR R1 ;ENTRY TO SHIFT RIGHT 6 REG 1 SUB.
1404 ASR R1
1405 ASR R1
1406 ASR R1 ;ENTRY TO SHIFT RIGHT 4 REG 1 SUB
1407 ASR R1
1408 ASR R1
1409 RTS PC ;EXIT SHIFT RIGHT SUB.
1410 BITCOM: MOV LPB,LPB1 ;SUBROUTINE TO COMPUTE 6 BIT PARITY
1411 MOV LPB,LPB2

```

```

1355 005430 104002
1356 005432 005067 173444
1357 005436 005067 173442
1358 005442 012500
1359 005444 012502
1360 005446 012767 000400 000166
1361 005454 012067 000160
1362 005460 016701 000154
1363 005464 112203
1364 005466 006003
1365 005470 006001
1366 005472 006003
1367 005474 006001
1368 005476 000301
1369 005500 004767 000150
1370 005504 042701 177700
1371 005510 004767 000146
1372 005514 016701 000120
1373 005520 004767 000120
1374 005524 042701 177700
1375 005530 004767 000126
1376 005534 016701 000100
1377 005540 042701 177700
1378 005544 004767 000112
1379 005550 005367 000066
1380 005554 001337
1381 005556 104003
1382 005560 005167 173316
1383 005564 042767 177700 173310
1384 005572 016767 173304 173300
1385 005600 016767 173276 173270
1386 005606 000367 173270
1387 005612 012767 000004 000022
1388 005620 006367 173256
1389 005624 006167 173254
1390 005630 005367 000006
1391 005634 001371
1392 005636 000205
1393 005640 000000
1394 005642 000000
1395 005644 006201
1396 005646 006201
1397 005650 006201
1398 005652 006201
1399 005654 006201
1400 005656 006201
1401 005660 000207
1402 005662 016767 173214 000040
1403 005670 016767 173206 000034

```

```

005706 005167 000030 CUM LPBZ
005708 040167 000022 BIC R1,LPBY
005706 040167 000020 BIC R1,LPBZ
005712 046767 000012 BIC LPBY,LPB
005720 056767 000006 BIS LPBZ,LPB
005726 000207 RTS PC
005730 000000 LPBY: OPEN
005732 000000 LPBZ: OPEN
: SUBROUTINE TO PERFORM COMPLEMENT OBERSE ON DATA SPECIFIED.
: SUBROUTINE CALL: JSR R5,OBVERS :CALL TO SUBROUTINE
: ADDR :ADDR OF DATA STRING
: EADR :ADDR OF EXTENDED DATA STRING
: COUNT :NUMBER OF WORDS TO PROCESS.
OBVERS: SAVO3
005734 104002 MOV (S)+,R0 :GET ADDR OF DATA STRING TO R0
005736 012500 MOV (S)+,R1 :ADDR OF EXTENDED DATA TO R1.
005740 012501 MOV (S)+,OBVCT :COUNT TO OBVCT
005742 000102 000006 000076 OBVA: MOV #6,OBVCTR
005746 012767 MOV (0),R2 :DATA WORD TO R2
005754 011002 MOV (1),R3 :EXTENDED DATA BYTE TO R3.
005756 111103 MOV# COM
005760 005102 COM R0,R0
005762 005103 COM R1,R1
005764 005010 CLR (0) :CLEAR DESTINATION WORD.
005766 105011 CLR# (1)
005770 006003 OBVB: ROR R0,R0
005772 006002 ROR R1,R1
005774 006003 ROR R2,R2
005776 006002 ROR R3,R3
006000 006003 ROR R0,R0
006002 006002 ROR R1,R1
006004 006110 ROR R2,R2
006006 106111 ROR# (1)
006010 006103 ROR R0,R0
006012 006110 ROR R1,R1
006014 106111 ROR# (1)
006016 006103 ROR R2,R2
006020 006110 ROR R3,R3
006022 106111 ROR# (1)
006024 005367 000022 DEC OBVCTR :DONE 6 TIMES?
006030 001360 BNE OBVB :BR IF NOT DONE.
006032 005720 TST (0)+ :UPDATE DATA ADDRESSES.
006034 005201 INC R1
006036 005367 000006 DEC OBVCT :DONE?
006042 001341 BNE OBVA :BR IF NOT.
006044 104003 RSTO3
006046 000205 RTS R5 :EXIT
006050 000000 OBVCNT: OPEN
006052 000000 OBVCTR: OPEN
: SUBROUTINE WRITE DATA, READ DATA, SINGLE BLOCK, BINARY COUNT.
RWFBK1: CLR RWFINO :SET SINGLE BLOCK INDICATOR.
BR RWFBKA
: SUBROUTINE TO WRITE DATA, READ DATA, 2 BLOCKS, BINARY COUNT.
006062 012767 177777 000102 RWFBK2: MOV #1,RWFINO :SET 2 BLOCK INDICATOR.
006070 004767 000312 RWFBKA: JSR PC,BINFLO :BIN FILL 256 WORD WRITE BUFFER 0.
006074 016767 173010 172756 MOV BLKNUM,BLKRO
006102 104027 WDATAF :CALL WDATAF SUB TO WRITE FWD 256. WORDS

```

E03

T04 - T011 TEST 4
DZTC00.P11

MACY11 27(732) 09-SEP-76 09:04 PAGE 30

1469	006104	000400		256.		: STARTING AT ADDR WBUFO
1470	006106	011276		WBUFO		
1471	006110	104032		RDATAR		: CALL RDATAR SUB TO READ REV 256. WORDS
1472	006112	000400		256.		: STARTING AT ADDR RBUFO
1473	006114	013346		RBUFO		
1474	006116	104040		DATCKI		: CALL DATCKI TO CHECK DATA STORED AT
1475	006120	011276		WBUFO		: WBUFO AGAINST DATA STORED AT RBUFO+510.
1476	006122	014344		RBUFO+510.		: CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
1477	006124	000400		256.		: ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
1478	006126	005767	000040	TST	RWFIND	: SINGLE BLOCK ONLY?
1479	006132	001416		BEG	RWFBKE	: BR IF YES.
1480	006134	004767	000272	JSR	PC, BINFL1	: NO. DOUBLE. BIN FILL WBUF1.
1481	006140	005267	172714	INC	BLKRO	
1482	006144	104027		WDATAR		: CALL WDATAR SUB TO WRITE FWD 256. WORDS
1483	006146	000400		256.		: STARTING AT ADDR WBUF1
1484	006150	012322		WBUF1		
1485	006152	104032		RDATAR		: CALL RDATAR SUB TO READ REV 256. WORDS
1486	006154	000400		256.		: STARTING AT ADDR RBUF1
1487	006156	014372		RBUF1		
1488	006160	104040		DATCKI		: CALL DATCKI TO CHECK DATA STORED AT
1489	006162	012322		WBUF1		: WBUF1 AGAINST DATA STORED AT RBUF1+510.
1490	006164	015370		RBUF1+510.		: CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
1491	006166	000400		256.		: ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
1492	006170	000207		RTS	PC	
1493	006172	000000		RWFBKE: OPEN		
1494				: SUBROUTINE TO SET UP WRITE BUFFER(S) TO MATCH DATA EXPECTED		
1495				: AFTER RALL OPERATION.		
1496	006174	005067	000166	SETWBF: CLR	SETIND	: SET FWD INDICATOR.
1497	006200	000403		BR	SETWBA	
1498	006202	012767	172777	SETWBR: MOV	#-1, SETIND	: SET REV INDICATOR.
1499	006210	004567	172214	SETWBA: JSR	RS, PARITY	: COMPUTE PARITY FOR WBUFO.
1500	006214	011276		WBUFO		
1501	006216	015411		EWBUFO		
1502	006220	016767	172656	MOV	LPB, FWCKSO	: MOVE PARITY TO FWD CHECKSUM IN WBUFO.
1503	006226	116767	172652	MOV	ELPB, EWFKO	
1504	006234	004567	177170	JSR	RS, PARITY	: COMPUTE PARITY FOR WBUF1.
1505	006240	012322		WBUF1		
1506	006242	016023		EWBUF1		
1507	006244	016767	172632	MOV	LPB, FWCKS1	: MOVE PARITY TO FWD CHECKSUM IN WBUF1.
1508	006252	116767	172626	MOV	ELPB, EWFK1	
1509	006260	016767	172574	MOV	BLKRO, WFBLKO	: SET UP FORWARD AND REVERSE BLOCK NUMBERS
1510	006266	016767	172566	MOV	BLKRO, WRBLKI	: IN WRITE BUFFERS.
1511	006274	016767	172560	MOV	BLKRO, WRBLKO	
1512	006302	016767	172552	MOV	BLKRO, WRBLKI	
1513	006310	005267	003776	INC	WFBLKI	
1514	006314	005267	005010	INC	WRBLKI	
1515	006320	004567	177410	JSR	RS, OBVERS	
1516	006324	012304		WRBLKO		
1517	006326	016014		EWBKO		
1518	006330	000001		JSR	RS, OBVERS	
1519	006332	004567	177376	WRBLKI		
1520	006336	013330		EWBKI		
1521	006340	016426		JSR	RS, OBVERS	
1522	006342	000001		TST	SETIND	: REVERSE SET UP?
1523	006344	005767	000016	BEG	STWBE	: BR IF NOT.
1524	006350	001405				

```

1525 006352 004567 177356 JSR R5,OBVERS :REVERSE SET UP. COMPLEMENT OBVERSE
1526 006356 011264 EAWBFO :ENTIRE WRITE BUFFER.
1527 006360 015404 EAWBFO
1528 006362 001020 528.
1529 006364 000207 STWBE: RTS PC :EXIT.
1530 006366 000000 SETIND: OPEN
:SUBROUTINE TO CLEAR ENTIRE READ BUFFER.
1531 006370 104035 CLRARB: CLEAR
1532 006372 013334 ARBFO
1533 006374 001024 532.
1534 006376 104035 CLEAR
1535 006400 016430 EARBFO
1536 006402 000412 266.
1537 006404 000207 RTS PC :EXIT.
1538 006406 104035 BINFLO: CLEAR :CLEAR ENTIRE WBUFO.
1539 006410 011264 WBUFO
1540 006412 000412 266.
1541 006414 104035 CLEAR :CLEAR EXTENDED WBUFO.
1542 006416 015404 EAWBFO
1543 006420 000205 133.
1544 006422 104036 BINFIL :FILL WRITE BUFFER 0 WITH BINARY COUNT.
1545 006424 011276 WBUFO
1546 006426 000400 256.
1547 006430 000207 RTS PC :EXIT.
1548 006432 104035 BINFL1: CLEAR :CLEAR ENTIRE WBUF1.
1549 006434 012310 WBUF1
1550 006436 000412 266.
1551 006440 104035 CLEAR :CLEAR EXTENDED WBUF1.
1552 006442 016016 EAWBF1
1553 006444 000205 133.
1554 006446 104036 BINFIL :FILL WRITE BUFFER 1 WITH BINARY COUNT.
1555 006450 012322 WBUF1
1556 006452 000400 256.
1557 006454 000207 RTS PC :EXIT.
1558 :SUBROUTINE TO REVERSE SEQUENCE OF 2 DATA STRINGS. 1ST STRING IS A WORD STRING.
1559 :2ND STRING IS A BYTE STRING. BOTH STRINGS MUST HAVE SAME NUMBER OF ELEMENTS.
1560 :CALL: JSR R5,REVERS :CALL REVERS SUB.
1561 : :ADDR OF WORD STRING.
1562 : :ADDR OF BYTE STRING.
1563 : :NUMBER OF ELEMENTS TO REVERSE.
1564 :
1565 REVERS: SAVO3
1566 006456 104002 MOV (5)+,R0 :WORD STRING ADDR TO R0.
1567 006460 012500 MOV (5)+,R1 :BYTE STRING ADDR TO R1.
1568 006462 012501 MOV (5)+,R2 :COUNT TO R2.
1569 006464 012502 DEC R2 :DECREMENT R2 BY 1 TO FIGURE ADDR OF LAST
1570 006466 005302 MOV R2,R3 :ELEMENT FOR BOTH STRINGS.
1571 006470 010203 MOV R2,R3
1572 006472 006302 ASL R2
1573 006474 060002 ADD R0,R2 :ADDR OF END DATA WORD IN R2.
1574 006476 060103 ADD R1,R3 :ADDR OF END DATA BYTE IN R3.
1575 006500 011067 000050 RVERSA: MOV (0),RVERSA
1576 006504 011267 000046 MOV (2),RVERSB
1577 006510 016712 000040 MOV RVERSA,(2)
1578 006514 016720 000036 MOV RVERSB,(0)+
1579 006520 111167 000030 MOVB (1),BRVERSA
1580 006524 111367 000025 MOVB (3),BRVERSB
1581 006530 116713 000020 MOVE BRVERSA,(3)

```

G03

T04 - T011 TEST 4
DET000.P11

MACY11 27(732) 09-SEP-76 09:04 PAGE 32

```

1581 006534 116721 000015      MOVB  BRVRSB,(1)+
1582 006540 005742      TST   -(2)          ;COMPLETE DATA ADDR UPDATE.
1583 006542 105743      TSTB  -(3)
1584 006544 020200      CMP   R2,R0        ;R2 LARGER THAN R0?
1585 006546 101354      BHI   RVERSA       ;BR IF YES. REVERSAL NOT COMPLETE.
1586 006550 104003      RSTO3
1587 006552 000205      RTS    R5          ;YES. DONE.
1588 006554      RVERSA:           ;EXIT.
1589 006554      BRVRSB: .BYTE  OPEN
1590 006555      BRVRSB: .BYTE  OPEN
1591 006556 000000      RVERSB: OPEN
1592 006556 000000      ;EMT SUB TO SELECT SEQUENTIAL DECTAPE UNIT.
1593 006560 005267 000010      SQDRV: INC  SQDRVA
1594 006564 042767 177770 000002      BIC   #177770,SQDRVA
1595 006572 104052      SELDRV
1596 006574 000000      SQDRVA: OPEN      ;DESIRED UNIT NUMBER.
1597 006576 000770      BR    SQDRV       ;UNIT NOT AVAILABLE RETURN.
1598 006600 000002      RTI             ;UNIT SELECTED. EXIT.
1599 006602 104006      ;EMT SUB TO SELECT DECTAPE UNIT SPECIFIED.
1600 006602 104006      SELDRR: SAVO55
1601 006604 012500      MOV   (5)+,R0    ;GET NUMBER OF UNIT TO BE SELECTED.
1602 006606 136067 006634 172252      BITB  UNTAB(0),UNITS ;SEE IF UNIT AVAILABLE FOR TESTING.
1603 006614 001405      BEQ   SELDRA     ;BR IF UNIT NOT AVAILABLE.
1604 006616 010067 172242      MOV   R0,UNITN  ;AVAIL. SELECT UNIT.
1605 006622 110067 172235      MOVB  R0,UNIT+1
1606 006626 005725      TST   (5)+      ;SET UP SELECTED EXIT.
1607 006630 104007      SELDRA: RSTOSS
1608 006632 000002      RTI             ;EXIT.
1609 006634 001      UNTAB: .BYTE  BIT0,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7
1610 006637 010      003      004
1611 006642 100      020      040
           200

```



```

1612
1613 006644 000000
1614 006646 006724
1615 006650 000012
1616 006652 006654
1617
1618
1619
1620
1621
1622
1623 006654 004767 177174
1624 006650 016767 172224 172172
1625 006666 004767 177302
1626 006672 004767 177472
1627 006676 104047
1628 006700 000402
1629 006702 013344
1630 006704 016434
1631 006706 104041
1632 006710 011276
1633 006712 013346
1634 006714 015411
1635 006716 016435
1636 006720 000400
1637 006722 104000
1638
1639 006724 000001
1640 006726 007004
1641 006730 000012
1642 006732 006734
1643
1644
1645
1646
1647
1648
1649 006734 004567 177114
1650 006740 016767 172144 172112
1651 006746 004767 177230
1652 006752 004767 177412
1653 006756 104050
1654 006760 000402
1655 006762 013344
1656 006764 016434
1657 006766 104042
1658 006770 011276
1659 006772 014344
1660 006774 015411
1661 006776 017034
1662 007000 000400
1663 007002 104000

```

```

*****
TO: 0 ;ROUTINE NUMBER 0 *
    T1 ;ADDRESS OF NEXT ROUTINE *
    10. ;TEST ITERATION COUNT *
    CA ;SCOPE ENTRY POINT *
*****
:RALL FORWARD TEST. SINGLE BLOCK. BINARY COUNT.
:TEST SEQUENCE: 1. WDATA FWD 1 BLOCK.
                 2. RDATA REV 1 BLOCK. VERIFY DATA.
                 3. RALL FWD 258 WORDS. VERIFY DATA INCLUDING REVERSE
                   AND FORWARD CHECKSUMS.
CA: JSR PC,RWFBK1 ;WRITE/READ SINGLE BLOCK.
     MOV BLKNUM,BLKRO
     JSR PC,SETWBF ;SET UP WRITE BUFFER FWD.
     JSR PC,CLRBUF ;CLEAR READ BUFFERS.
     RALLF ;READ ALL FWD 258. WORDS INTO
           258. ;ADDR RRCKSO AND UP. EXTENDED DATA BITS
           RRCKSO ;ARE STORED IN CONSECUTIVE BYTES STARTING
           ERRCKO ;AT ADDRESS ERRCKO
           ADTCK ;CALL ADTCK SUB TO CHECK 18 BIT DATA STARTING
           WBUFO ;AT ADDR WBUFO AND EWBUFF, AGAINST 18 BIT DATA
           RBUFO ;STARTING AT ADDR RBUFO AND ERBUFO. FIFTH ARGUMENT
           EWBUFF ;REPRESENTS # OF 18 BIT ELEMENTS TO CHECK.
           ERBUFO ;REPORT ERRORS.
           256.
           SCOPE ;SCOPE.
*****
T1: 1 ;ROUTINE NUMBER 1 *
    T2 ;ADDRESS OF NEXT ROUTINE *
    10. ;TEST ITERATION COUNT *
    DA ;SCOPE ENTRY POINT *
*****
:RALL REV TEST. SINGLE BLOCK. BINARY COUNT.
:TEST SEQUENCE: 1. WDATA FWD 1 BLOCK.
                 2. RDATA REV 1 BLOCK. VERIFY DATA.
                 3. COMPLEMENT OBVERSE WRITE BUFFER.
                 4. RALL REV 258 WORDS. VERIFY DATA.
DA: JSR R5,RWFBK1 ;WRITE/READ SINGLE BLOCK (SEQUENCE 1 AND 2).
     MOV BLKNUM,BLKRO
     JSR PC,SETWBR ;SET UP WRITE BUFFER REV.
     JSR PC,CLRBUF ;CLEAR READ BUFFERS.
     RALLR ;READ ALL REV 258. WORDS INTO
           258. ;ADDR RRCKSO AND UP. EXTENDED DATA BITS ARE
           RRCKSO ;STORED IN CONSECUTIVE BYTES STARTING
           ERRCKO ;AT ADDRESS ERRCKO
           ADTCKI ;CALL ADTCKI SUB TO CHECK 18 BIT DATA STARTING
           WBUFO ;AT ADDR WBUFO AND EWBUFF, AGAINST 18 BIT DATA
           FRCKSO-2 ;STARTING AT ADDR FRCKSO-2 AND ERCKO-1. ACTUAL
           EWBUFF ;IS CHECKED IN DESCENDING ORDER. FIFTH ARGUMENT
           ERCKO-1 ;REPRESENTS # OF 18 BIT ELEMENTS TO BE CHECKED.
           256. ;REPORT ERRORS.
           SCOPE

```

```

1665 007004 000002
1666 007006 007120
1667 007010 000012
1668 007012 007014
1669
1670
1671
1672
1673
1674
1675 007014 004767 177366
1676 007020 016767 172064 172032
1677 007026 004767 177142
1678 007032 104045
1679 007034 000402
1680 007036 011274
1681 007040 015410
1682 007042 104031
1683 007044 000400
1684 007046 013346
1685 007050 104037
1686 007052 011276
1687 007054 013346
1688 007056 000400
1689 007060 004567 176650
1690 007064 011264
1691 007066 015404
1692 007070 000412
1693 007072 104050
1694 007074 000402
1695 007076 013344
1696 007100 016434
1697 007102 104042
1698 007104 011274
1699 007106 014346
1700 007110 015410
1701 007112 017035
1702 007114 000402
1703 007116 104000

```

```

*****
t2: 2 ;ROUTINE NUMBER 2 *
    T3 ;ADDRESS OF NEXT ROUTINE *
    10. ;TEST ITERATION COUNT *
    GA ;SCOPE ENTRY POINT *
*****
:WALL FORWARD TEST. SINGLE BLOCK. BINARY COUNT.
:TEST SEQUENCE: 1. BINARY FILL WRITE BUFFER 0.
                 2. FILL IN REV AND FWD CHECKSUMS.
                 3. WALL FWD 258 WORDS ONTO BLOCK 200.
                 4. RDATA FWD 256 WORDS. VERIFY DATA.
                 5. RALL REV 258 WORDS. VERIFY 258 WORDS INCLUDING EXTENDED BITS.
:
:
:
GA: JSR PC,BINFLO ;BINFIL WBUFO.
     MOV BLKNUM,BLKRO
     JSR PC,SETWBF ;SET UP WRITE BUFFER FWD.
     WALLF 258. ;WRITE ALL FWD 258. WORDS STARTING
     RWCKSO ;FROM ADDR RWCKSO. EXTENDED DATA BITS ARE
     ERCKO ;TAKEN FROM CONSECUTIVE BYTES STARTING
     RDATAF 256. ;AT ADDRESS ERCKO.
     RBUFO ;CALL RDATAF SUB TO READ FWD 256. WORDS
     DATCHK ;AND STORE AT ADDR STARTING AT RBUFO
     WBUFO ;CALL DATCHK SUB TO CHECK DATA STORED AT
     RBUFO ;WBUFO AGAINST DATA STORED AT RBUFO
     256. ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
     JSR R5,OBVERS ;ERRORS.
     AWBUFO
     EAWBFO
     266.
     RALLR 258. ;READ ALL REV 258. WORDS INTO
     RWCKSO ;ADDR RWCKSO AND UP. EXTENDED DATA BITS ARE
     ERCKO ;STORED IN CONSECUTIVE BYTES STARTING
     ADCKI ;AT ADDRESS ERCKO.
     RWCKSO ;CALL ADCKI SUB TO CHECK 18 BIT DATA STARTING
     ERCKO ;AT ADDR RWCKSO AND ERCKO. AGAINST 18 BIT DATA
     258. ;STARTING AT ADDR RWCKSO AND ERCKO. ACTUAL DATA
     SCOPE ;IS CHECKED IN DESCENDING ORDER. FIFTH ARGUMENT
           ;REPRESENTS # OF 18 BIT ELEMENTS TO BE CHECKED.
           ;REPORT ERRORS.

```

```

1705
1706 007120 000003
1707 007122 007246
1709 007124 000012
1709 007126 007130
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719 007130 004767 177252
1720 007134 016767 171750 171716
1721 007142 004767 177034
1722 007146 004567 177304
1723 007152 011274
1724 007154 015410
1725 007156 000402
1726 007160 104046
1727 007162 000402
1728 007164 011274
1729 007166 015410
1730 007170 104032
1731 007172 000400
1732 007174 013346
1733 007176 004567 176532
1734 007202 011274
1735 007204 015410
1736 007206 000402
1737 007210 104037
1738 007212 011276
1739 007214 013346
1740 007216 000400
1741 007220 104047
1742 007222 000402
1743 007224 013344
1744 007226 016434
1745 007230 104042
1746 007232 011274
1747 007234 014346
1748 007236 015410
1749 007240 017035
1750 007242 000402
1751 007244 104000

```

```

*****
T3:      3      ;ROUTINE NUMBER 3 *
        T4      ;ADDRESS OF NEXT ROUTINE *
        10.     ;TEST ITERATION COUNT *
        HA      ;SCOPE ENTRY POINT *
*****
:WALL REV TEST. SINGLE BLOCK. BINARY COUNT.
:TEST SEQUENCE: 1. BINARY FILL WRITE BUFFER 0.
                 2. FILL IN REV AND FWD CHECKSUMS.
                 3. OBVERSE 258 WORDS TO BE WRITTEN.
                 4. WALL REV 258 WORDS ONTO BLOCK 200.
                 5. RDATA REV 256 WORDS.
                 6. OBVERSE DATA READ IN STEP 5 AND VERIFY DATA.
                 7. RALL FWD 258 WORDS. VERIFY 258 WORDS INCLUDING EXTENDED BITS.
HA:      JSR     PC,BINFLO ;BINFIL WBUFO.
        MOV     BLKNUM,BLKRG
        JSR     PC,SETWBR ;SET UP WRITE BUFFER REV.
        JSR     RS,REVERS ;REVERSE WRITE BUFFER DATA.
        RWCKSO
        EWRCKO
        258.
        WALLR ;WRITE ALL REV 258. WORDS STARTING
        258. ;FROM ADDR RWCKSO . EXTENDED DATA BITS ARE
        RWCKSO ;TAKEN FROM CONSECUTIVE BYTES STARTING
        EWRCKO ;AT ADDRESS EWRCKO .
        RDATAR ;CALL RDATAR SUB TO READ REV 256. WORDS
        256. ;STARTING AT ADDR RBUFO
        RBUFO
        JSR     RS,OBVERS ;OBVERSE DATA READ.
        RWCKSO
        EWRCKO
        258.
        DATCHK ;CALL DATCHK SUB TO CHECK DATA STORED AT
        WBUFO ;WBUFO AGAINST DATA STORED AT RBUFO
        RBUFO ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
        256. ;ERRORS.
        RALLF ;READ ALL FWD 258. WORDS INTO
        258. ;ADDR RWCKSO AND UP. EXTENDED DATA BITS
        RWCKSO ;ARE STORED IN CONSECUTIVE BYTES STARTING
        ERRCKO ;AT ADDRESS ERRCKO .
        ADTCKI ;CALL ADTCKI SUB TO CHECK 18 BIT DATA STARTING
        RWCKSO ;AT ADDR RWCKSO AND EWRCKO . AGAINST 18 BIT DATA
        FRCKSO ;STARTING AT ADDR FRCKSO AND ERCKO . ACTUAL DATA
        EWRCKO ;IS CHECKED IN DESCENDING ORDER. FIFTH ARGUMENT
        ERCKO ;REPRESENTS # OF 18 BIT ELEMENTS TO BE CHECKED.
        258. ;REPORT ERRORS.
        SCOPE

```

```

1752
1753 007246 000004
1754 007250 007344
1755 007252 000012
1756 007254 007256
1757
1758
1759
1760
1761
1762
1763
1764 007256 004767 177124
1765 007262 004767 177144
1766 007266 016767 171564
1767 007274 004767 176674
1768 007300 104045
1769 007302 001020
1770 007304 011274
1771 007306 015410
1772 007310 104031
1773 007312 001000
1774 007314 013346
1775 007316 104047
1776 007320 001020
1777 007322 013344
1778 007324 016434
1779 007326 104041
1780 007330 011274
1781 007332 013344
1782 007334 015410
1783 007336 016434
1784 007340 001020
1785 007342 104000
1786
1787 007344 000005
1788 007346 007460
1789 007350 000012
1790 007352 007354
1791
1792
1793
1794
1795
1796
1797
1798
1799 007354 004767 177026
1800 007360 004767 177046
1801 007364 016767 171520 171466
1802 007372 004767 176604
1803 007376 004567 177054
1804 007402 011264
1805 007404 015404
1806 007406 001020
1807 007410 005267 171444

```

```

*****
t4: 4 ;ROUTINE NUMBER 4 *
    T5 ;ADDRESS OF NEXT ROUTINE *
    10. ;TEST ITERATION COUNT *
    IA ;SCOPE ENTRY POINT *
*****
:WALL FWD TEST. 2 BLOCKS. BINARY COUNT.
:TEST SEQUENCE: 1. BINARY FILL WBUF0 AND WBUF1.
                 2. FILL IN REV AND FWD CHECKSUMS.
                 3. WALL FWD 528 WORDS ONTO BLOCK 200 AND 201.
                 4. READ DATA FWD TO CHECK FOR PARITY ERRORS.
                 5. RALL FWD 528 WORDS. VERIFY DATA AND EXTENDED DATA.
IA: JSR PC,BINFLO ;BINFIL WBUF0.
     JSR PC,BINFL1 ;BINFIL WBUF1.
     MOV DLKNUM,BLKRQ
     JSR PC,SETWSF ;SET UP WRITE BUFFER FWD.
     WALLF ;WRITE ALL FWD 528. WORDS STARTING
           528. ;FROM ADDR RWCKSO . EXTENDED DATA BITS ARE
           RWCKSO ;TAKEN FROM CONSECUTIVE BYTES STARTING
           EWRCKO ;AT ADDRESS EWRCKO .
           RDATAF ;CALL RDATAF SUB TO READ FWD 512. WORDS
           512. ;AND STORE AT ADDR STARTING AT RBUF0
           RBUF0
           RALLF ;READ ALL FWD 528. WORDS INTO
           528. ;ADDR RWCKSO AND UP. EXTENDED DATA BITS
           RRCKSO ;ARE STORED IN CONSECUTIVE BYTES STARTING
           ERRCKO ;AT ADDRESS ERRCKO
           ADTCK ;CALL ADTCK SUB TO CHECK 18 BIT DATA STARTING
           RWCKSO ;AT ADDR RWCKSO AND EWRCKO . AGAINST 18 BIT DATA
           RRCKSO ;STARTING AT ADDR RRCKSO AND ERRCKO . FIFTH ARGUMENT
           EWRCKO ;REPRESENTS # OF 18 BIT ELEMENTS TO CHECK.
           ERRCKO ;REPORT ERRORS.
           528.
           SCOPE
*****
t5: 5 ;ROUTINE NUMBER 5 *
    T6 ;ADDRESS OF NEXT ROUTINE *
    10. ;TEST ITERATION COUNT *
    JA ;SCOPE ENTRY POINT *
*****
:WALL REV TEST. 2 BLOCKS. BINARY COUNT.
:TEST SEQUENCE: 1. BINARY FILL WBUF0 AND WBUF1.
                 2. FILL IN REV AND FWD CHECKSUMS.
                 3. OBVERSE AND REVERSE 528 WORDS OF DATA TO BE WRITTEN.
                 4. WALL REV 528 WORDS ONTO BLOCK 201 AND 200.
                 5. READ DATA REV 512 WORDS TO CHECK FOR PARITY ERRORS.
                 6. RALL REV 528 WORDS. VERIFY DATA AND EXTENDED DATA.
JA: JSR PC,BINFLO ;BINFIL WBUF0.
     JSR PC,BINFL1 ;BINFIL WBUF1.
     MOV BLKNUM,BLKRQ
     JSR PC,SETWR ;SET UP WRITE BUFFER REVERSE.
     JSR RS,REVER ;REVERSE 528 WORDS OF WRITE DATA.
     AWBUF0
     EAWBFO
     528.
     INC BLKRQ

```

1808 007414 104046
 1809 007416 001020
 1810 007420 011264
 1811 007422 015404
 1812 007424 104032
 1813 007426 001000
 1814 007430 013346
 1815 007432 104050
 1816 007434 001020
 1817 007436 013334
 1818 007440 016430
 1819 007442 104041
 1820 007444 011264
 1821 007446 013334
 1822 007450 015404
 1823 007452 016430
 1824 007454 001020
 1825 007456 104000
 1826
 1827 007460 000006
 1828 007462 010010
 1829 007464 000012
 1830 007466 007470
 1831
 1832
 1833
 1834
 1835
 1836
 1837
 1838
 1839
 1840
 1841
 1842 007470 004767 176712
 1843 007474 005067 171360
 1844 007500 004767 176470
 1845 007504 012700 001102
 1846 007510 012767 177777 171342
 1847 007516 005267 171336
 1848 007522 104045
 1849 007524 000403
 1850 007526 011274
 1851 007530 015410
 1852 007532 005300
 1853 007534 001370
 1854 007536 012700 000441
 1855 007542 012767 001103 171310
 1856 007550 162767 000002 171302
 1857 007556 104032
 1858 007560 000400
 1859 007562 013346
 1860 007564 104040
 1861 007566 011276
 1862 007570 014344
 1863 007572 000400

```

WALLR                               ;WRITE ALL REV 528. WORDS STARTING
528.                                ;FROM ADDR AUBUFD. EXTENDED DATA BITS ARE
AUBUFD                              ;TAKEN FROM CONSECUTIVE BYTES STARTING
EAWBFO                              ;AT ADDRESS EAWBFO.
RDATAR                              ;CALL RDATAR SUB TO READ REV 512. WORDS
512.                                ;STARTING AT ADDR RBUFD
RBUFD
RALLR                               ;READ ALL REV 528. WORDS
528.                                ;ADDR ARBUFD AND UP. EXTEN DATA BITS ARE
ARBUFD                              ;STORED IN CONSECUTIVE BYTES STARTING
EARBFO                              ;AT ADDRESS EARBFO.
ADTCK                               ;CALL ADTCK SUB TO CHECK 18 BIT DATA STARTING
AUBUFD                              ;AT ADDR AUBUFD AND EAWBFO. AGAINST 18 BIT DATA
ARBUFD                              ;STARTING AT ADDR ARBUFD AND EARBFO. FIFTH ARGUMENT
EAWBFO                              ;REPRESENTS # OF 18 BIT ELEMENTS TO CHECK.
EARBFO                              ;REPORT ERRORS.
528.
SCOPE
*****
ts: 6                                ;ROUTINE NUMBER 6 *
    T7                              ;ADDRESS OF NEXT ROUTINE *
    10.                             ;TEST ITERATION COUNT *
    KA                              ;SCOPE ENTRY POINT *
*****
:WALL-RALL TEST. ALL BLOCKS BINARY COUNT.
:TEST SEQUENCE: 1. BINARY FILL WBUFD.
                2. FILL IN REV AND FORWARD CHECKSUMS.
                3. WALL FWD 260 WORDS IN EACH BLOCK, STARTING WITH 0.
                4. RDATA REV EVERY OTHER BLOCK. VERIFY DATA.
                5. RDATA FWD EVERY OTHER BLOCK. VERIFY DATA.
                6. OBERVERSE WRITE DATA TO MATCH RALL REV DATA.
                7. RALL REV 258 WORDS EVERY OTHER BLOCK. VERIFY DATA.
                8. REOBERVERSE WRITE DATA TO MATCH RALL FWD DATA.
                9. RALL FWD 260 WORDS EVERY OTHER BLOCK. VERIFY DATA.
KA: JSR PC,BINFLO ;BINFIL WBUFD.
    CLR BLKRQ
    JSR PC,SETWBF ;SET UP WRITE BUFFER FWD.
    MOV #578.,RO ;SET UP TO WALL 578 BLOCKS STARTING
    MOV #-1,BLKRQ ;WITH BLOCK 0.
KB: INC BLKRQ
    WALLF ;WRITE ALL FWD 259. WORDS STARTING
    259. ;FROM ADDR RWCKSD. EXTENDED DATA BITS ARE
    RWCKSD ;TAKEN FROM CONSECUTIVE BYTES STARTING
    EWRCKD ;AT ADDRESS EWRCKD.
    DEC RO ;DONE ALL BLOCKS?
    BNE KB ;BR IF NOT DONE.
    MOV #289.,RO ;SET UP TO READ DATA REV EVERY OTHER BLOCK.
    MOV #579.,BLKRQ
    SUB #2,BLKRQ
KC: RDATAR ;CALL RDATAR SUB TO READ REV 256. WORDS
    256. ;STARTING AT ADDR RBUFD
    RBUFD
    DATCKI ;CALL DATCKI TO CHECK DATA STORED AT
    WBUFD ;WBUFD AGAINST DATA STORED AT RBUFD+510.
    RBUFD+510. ;CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
    256. ;ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
  
```

1864	007574	005300		DEC	RO				; DONE?
1865	007576	001364		BNE	KC				; BR IF NOT DONE.
1866	007600	012700	000441	MOV	#289.,RO				; SET UP TO READ DATA FWD EVERY OTHER BLOCK.
1867	007604	012767	177776	MOV	#-2,BLKRQ				
1868	007612	062767	000002	ADD	#2,BLKRQ				
1869	007620	104031		RDATAF					; CALL RDATAF SUB TO READ FWD 256. WORDS
1870	007622	000400		256.					; AND STORE AT ADDR STARTING AT RBUFO
1871	007624	013346		RBUFO					
1872	007626	104037		DATCHK					; CALL DATCHK SUB TO CHECK DATA STORED AT
1873	007630	011276		WBUFO					; WBUFO AGAINST DATA STORED AT RBUFO
1874	007632	013346		RBUFO					; CHECK NUMBER OF WORDS SPECIFIED. REPORT
1875	007634	000400		256.					; ERRORS.
1876	007636	005300		DEC	RO				; DONE?
1877	007640	001364		BNE	KD				; BR IF NOT DONE.
1878	007642	004567	176066	JSR	RS,OBVERS				; OBLVERSE WRITE DATA TO MATCH RALL REV DATA.
1879	007646	011264		AWBUFO					
1880	007650	015404		EAWBFO					
1881	007652	000412		266.					
1882	007654	012700	000441	MOV	#289.,RO				; SET UP TO RALL REV EVERY OTHER BLOCK.
1883	007660	012767	001103	MOV	#579.,BLKRQ				
1884	007666	162767	000002	SUB	#2,BLKRQ				
1885	007674	104050		RALLR					; READ ALL REV 258. WORDS INTO
1886	007676	000402		258.					; ADDR RRCKSO AND UP. EXTENDED DATA BITS ARE
1887	007700	013344		RRCKSO					; STORED IN CONSECUTIVE BYTES STARTING
1888	007702	016434		ERRCKO					; AT ADDRESS ERRCKO .
1889	007704	104042		ADTCKI					; CALL ADTCKI SUB TO CHECK 18 BIT DATA STARTING
1890	007706	011274		RRCKSO					; AT ADDR RWCKSO AND EWRCKO , AGAINST 18 BIT DATA
1891	007710	014346		FRCKSO					; STARTING AT ADDR FRCKSO AND ERCKO . ACTUAL DATA
1892	007712	015410		EWRCKO					; IS CHECKED IN DESCENDING ORDER. FIFTH ARGUMENT
1893	007714	017035		ERFCKO					; REPRESENTS # OF 18 BIT ELEMENTS TO BE CHECKED.
1894	007716	000400		256.					; REPORT ERRORS.
1895	007720	005300		DEC	RO				; DONE?
1896	007722	001361		BNE	KE				; BR IF NOT DONE.
1897	007724	004567	176004	JSR	RS,OBVERS				; REOBLVERSE WRITE DATA TO MATCH RALL FWD DATA.
1898	007730	011264		AWBUFO					
1899	007732	015404		EAWBFO					
1900	007734	000412		266.					
1901	007736	012700	000441	MOV	#289.,RO				; SET UP TO RALL FWD EVERY OTHER BLOCK.
1902	007742	012767	177776	MOV	#-2,BLKRQ				
1903	007750	062767	000002	ADD	#2,BLKRQ				
1904	007756	104047		RALLF					; READ ALL FWD 258. WORDS INTO
1905	007760	000402		258.					; ADDR RRCKSO AND UP. EXTENDED DATA BITS
1906	007762	013344		RRCKSO					; ARE STORED IN CONSECUTIVE BYTES STARTING
1907	007764	016434		ERRCKO					; AT ADDRESS ERRCKO .
1908	007766	104041		ADTCK					; CALL ADTCK SUB TO CHECK 18 BIT DATA STARTING
1909	007770	011274		RWCKSO					; AT ADDR RWCKSO AND EWRCKO , AGAINST 18 BIT DATA
1910	007772	013344		RRCKSO					; STARTING AT ADDR RRCKSO AND ERRCKO . FIFTH ARGUMENT
1911	007774	015410		EWRCKO					; REPRESENTS # OF 18 BIT ELEMENTS TO CHECK.
1912	007776	016434		ERRCKO					; REPORT ERRORS.
1913	010000	000402		258.					
1914	010002	005300		DEC	RO				; DONE?
1915	010004	001361		BNE	KF				; BR IF NOT DONE.
1916	010006	104000		SCOPE					
1917									
1918	010010	000007	↑7:	7					; ROUTINE NUMBER 7 *
1919	010012	177777		TLAST					; ADDRESS OF NEXT ROUTINE *

1920	010014	000012			
1921	010016	010020			
1922					
1923					
1924					
1925					
1926					
1927					
1928					
1929					
1930					
1931					
1932					
1933					
1934	010020	004767	176362		
1935	010024	004567	000042		
1936	010030	000040			
1937	010032	004567	000034		
1938	010036	000020			
1939	010040	004567	000026		
1940	010044	000010			
1941	010046	004567	000020		
1942	010052	000004			
1943	010054	004567	000012		
1944	010060	000002			
1945	010062	004567	000004		
1946	010066	000001			
1947	010070	104000			
1948	010072	016767	171012	170760	LSUBA:
1949	010100	004567	175324		
1950	010104	011276			
1951	010106	015411			
1952	010110	042567	170764		
1953	010114	016767	170760	170760	
1954	010122	004567	175460		
1955	010126	016767	170750	002142	
1956	010134	116767	170744	005647	
1957	010142	104045			
1958	010144	000402			
1959	010146	011274			
1960	010150	015410			
1961	010152	104047			
1962	010154	000402			
1963	010156	013344			
1964	010160	016434			
1965	010162	104041			
1966	010164	011274			
1967	010166	013344			
1968	010170	015410			
1969	010172	016434			
1970	010174	000402			
1971	010176	104033			
1972	010200	000400			
1973	010202	013346			
1974	010204	032777	040000	170572	
1975	010212	001013			

```

10. ;TEST ITERATION COUNT *
LA ;SCOPE ENTRY POINT *
*****
PARITY TEST.
TEST SEQUENCE: 1. BINARY FILL WBUFO.
                2. FILL IN REVERSE AND FORWARD CHECKSUMS. THE RESULTING
                FORWARD CHECKSUM WILL BE 77.
                3. CLEAR ONE BIT OF FWD CHECKSUM.
                4. WALL FWD 258 WORDS. INCLUDES INCORRECT PARITY.
                5. RALL FWD 258 WORDS TO VERIFY DATA WRITTEN.
                6. RDATA FWD 256 WORDS. PARITY ERROR SHOULD OCCUR. IF NO
                ERROR OCCURS, TYPE OUT THE CORRECT PARITY AND THE INCORRECT
                PARITY WRITTEN, TO INDICATE THE BIT FAILING TO CAUSE ERROR.
                7. REPEAT STEPS 2 THROUGH 6 FOR EACH PARITY BIT.
LA: JSR PC,BINFLO ;BINARY FILL WBUFO.
     JSR RS,LSUBA ;RUN TEST.
     BITS
     JSR RS,LSUBA ;RUN TEST.
     BIT4
     JSR RS,LSUBA ;RUN TEST.
     BIT3
     JSR RS,LSUBA ;RUN TEST.
     BIT2
     JSR RS,LSUBA ;RUN TEST.
     BIT1
     JSR RS,LSUBA ;RUN TEST.
     BIT0
     SCOPE ;SCOPE.
     MOV BLKNUM,BLKRQ
     JSR RS,PARITY ;COMPUTE PARITY FOR WBUFO.
     WBUFO
     EWBUFO
     BIC (5)+,LPBT ;CLEAR SPECIFIED BIT FROM CALCULATED PARITY.
     MOV LPBT,LPB
     JSR RS,XPARTY
     MOV LPB,FWCKSO ;MOVE BAD PARITY TO WBUFO.
     MOVB ELPB,EWFKO
     WALLF ;WRITE ALL FWD 258. WORDS STARTING
     258. ;FROM ADDR RWCKSO. EXTENDED DATA BITS ARE
     RWCKSO ;TAKEN FROM CONSECUTIVE BYTES STARTING
     ERCKO ;AT ADDRESS ERCKO.
     RALLF ;READ ALL FWD 258. WORDS INTO
     258. ;ADDR RWCKSO AND UP. EXTENDED DATA BITS
     RRCKSO ;ARE STORED IN CONSECUTIVE BYTES STARTING
     ERCKO ;AT ADDRESS ERCKO.
     ADTCK ;CALL ADTCK SUB TO CHECK 18 BIT DATA STARTING
     RWCKSO ;AT ADDR RWCKSO AND ERCKO. AGAINST 18 BIT DATA
     RRCKSO ;STARTING AT ADDR RWCKSO AND ERCKO. FIFTH ARGUMENT
     EWCKO ;REPRESENTS # OF 18 BIT ELEMENTS TO CHECK.
     ERCKO ;REPORT ERRORS.
     258.
     RDATAFS ;CALL RDATAFS SUB TO READ DATA FWD 256. WORDS
     256. ;AND STORE AT ADDR STARTING AT RBUFO.
     RBUFO ;ALLOW PARITY ERROR.
     BIT #BIT14,DTCT ;PARITY ERROR?
     BNE LSUBBA ;BR IF PARITY ERROR SET.

```

1976	010214	104012			OACNV		;NO. ERROR. CONVERT INCORRECT PARITY TO ASCII.
1977	010216	001100			LPBT		
1978	010220	011204			ABPAR		
1979	010222	000002			2		
1980	010224	104012			OACNV		;CONVERT GOOD PARITY TO ASCII.
1981	010226	001076			LPBG		
1982	010230	011174			AGPAR		
1983	010232	000002			2		
1984	010234	104011			ERRORN		;FAILURE TO DETECT PARITY
1985	010236	011153			NPARE		
1986	010240	177777			-1		
1987	010242	000205			LSUBBA: RTS	R5	;EXIT.
1988							
1989	010244	022445	020124		EMO: .ASCII	'%AT '	
1990	010250	020040	020040	050040	ATNUMB: .ASCII	' PC '	
1991	010256	020103					
1992	010260	020040	020040	020040	APC: .ASCII	' ICNT '	
1993	010266	020040	041511	052116			
1994	010274	040					
1995	010275	040	020040	020040	AICNT: .ASCII	' '	
1996	010302	056					
1997	010303	040	052440	044516	.ASCII	' UNIT '	
1998	010310	020124					
1999	010312	000040			AUNIT: .ASCIZ	' '	
2000	010314	022445	047111	040526	AINCRT: .ASCIZ	'%INVALID TEST'	
2001	010322	044514	020104	042524			
2002	010330	052123	000				
2003	010333	045	052045	032103	PGTIT: .ASCIZ	'%TC4 - TC11 TEST 4%'	
2004	010340	026440	052040	030503			
2005	010346	020061	042524	052123			
2006	010354	032040	022445	000			
2007	010361	045	052123	047101	INST1: .ASCII	'%STANDARD TAPES ON UNITS'	
2008	010366	040504	042122	052040			
2009	010374	050101	051505	047440			
2010	010402	020116	047125	052111			
2011	010410	123					
2012	010411	045	042522	047515	.ASCII	'%REMOTE, WRITE ENABLE'	
2013	010416	042524	020054	051127			
2014	010424	052111	020105	047105			
2015	010432	041101	042514				
2016	010436	053445	046101	051514	.ASCII	'%WALLSW: ON, WRTMSW: OFF'	
2017	010444	035127	047440	026116			
2018	010452	053440	052122	051515			
2019	010460	035127	047440	043106			
2020	010466	051445	046105	041505	.ASCII	'%SELECT UNITS WITH SR7 - SR0.'	
2021	010474	020124	047125	052111			
2022	010502	020123	044527	047124			
2023	010510	051440	033522	026440			
2024	010516	051440	030122	020056			
2025	010524	051120	051505	020123	.ASCIZ	'PRESS CONT.%'	
2026	010532	047503	052116	022456			
2027	010540	000					
2028	010541	045	042523	020124	ASETSR: .ASCIZ	'%SET SR OPTIONS. NORMAL SR=0'	
2029	010546	051123	047440	052120			
2030	010554	047511	051516	020056			
2031	010562	047516	046522	046101			

0032	010570	051440	036522	000060					
0033	010576	007			APGEN0:	.BYTE	007		
0034	010577	045	000052			.ASCIZ	'%*'		
0035	010602	052040	053503	020103	CTCWC:	.ASCII	'TCWC'		
0036	010610	020040	020040	020040	ATCWC:	.ASCIZ			
0037	010616	000							
0038	010617	040	041524	040502	CTCBA:	.ASCII	'TCBA'		
0039	010624	040							
0040	010625	040	020040	020040	ATCBA:	.ASCIZ			
0041	010632	000040							
0042	010634	052040	041503	020115	STAT:	.ASCII	'TCOM'		
0043	010642	020040	020040	020040	ATCOM:	.ASCII	'TCST'		
0044	010650	052040	051503	020124					
0045	010656	020040	020040	020040	ATCST:	.ASCIZ			
0046	010664	000							
0047	010665	040	047516	042040	INTFAI:	.ASCIZ	'NO DT INTRPT'		
0048	010672	020124	047111	051124					
0049	010700	052120	000040						
0050									
0051	010704	042040	020124	051105	DTERR:	.ASCIZ	'DT ERR'		
0052	010712	020122	000						
0053	010715	040	046102	051113	BLKSB:	.ASCII	'BLKRQ'		
0054	010722	020121							
0055	010724	020040	020040	020040	ABLKRQ:	.ASCIZ			
0056	010732	000							
0057	010733	124	053503	020103	WCNOTC:	.ASCIZ	'TCWC NOT B'		
0058	010740	047516	020124	020060					
0059	010746	000							
0060	010747	124	041103	020101	INCTCB:	.ASCIZ	'TCBA WRONG'		
0061	010754	051127	047117	020107					
0062	010762	000							
0063	010763	040	041524	040502	TCBASB:	.ASCII	'TCBA'		
0064	010770	040							
0065	010771	040	020040	020040	ATCBAS:	.ASCIZ			
0066	010776	020040	000040						
0067	011002	042040	052101	020101	DATERR:	.ASCII	'DATA ERR WORD'		
0068	011010	051105	020122	053440					
0069	011016	051117	020104						
0070	011022	020040	020040	020056	AWDCNT:	.ASCII	'SAB'		
0071	011030	051440	041057	040					
0072	011035	040	020040	020040	ADATSB:	.ASCII	'WAS'		
0073	011042	020040	053440	051501					
0074	011050	040							
0075	011051	040	020040	020040	DATWS:	.ASCIZ			
0076	011056	000040							
0077	011060	047045	020117	047125	NOUNIT:	.ASCIZ	'NO UNITS AVAILABLE.'		
0078	011066	052111	020123	053101					
0079	011074	044501	040514	046102					
0080	011102	027105	000						
0081	011105	122	054504	042457	STCMMSG:	.ASCIZ	'RDY ERR NOT B AFTER DO'		
0082	011112	051122	047040	052117					
0083	011120	030040	040440	052106					
0084	011126	051105	042040	000117					
0085	011134	041040	045514	047040	SROHER:	.ASCIZ	'BLK NOT FOUND'		
0086	011142	052117	043040	052517					
0087	011150	042116	000						

0111153	045	047516	050040
0111160	051101	042440	051122
0111166	043440	047517	020104
0111174	020040	020040	040502
0111180	020104		
0111187	020040	000	
0111194	040	043040	041520
0111201	040		
0111208	040	020040	020040
0111215	020040	000045	
0111222	053445	046111	020114
0111229	042524	052123	052440
0111236	044516	051524	020072
0111243	000		
0111250	040	000054	
0111257	060	061	
0111264	063	062	
0111271	066	065	
0111278		067	
0111285			
0111292			
0111299			
0111306			
0111313			
0111320			
0111327			
0111334			
0111341			
0111348			
0111355			
0111362			
0111369			
0111376			
0111383			
0111390			
0111397			
0111404			
0111411			
0111418			
0111425			
0111432			
0111439			
0111446			
0111453			
0111460			
0111467			
0111474			
0111481			
0111488			
0111495			
0111502			
0111509			
0111516			
0111523			
0111530			
0111537			
0111544			
0111551			
0111558			
0111565			
0111572			
0111579			
0111586			
0111593			
0111600			
0111607			
0111614			
0111621			
0111628			
0111635			
0111642			
0111649			
0111656			
0111663			
0111670			
0111677			
0111684			
0111691			
0111698			
0111705			
0111712			
0111719			
0111726			
0111733			
0111740			
0111747			
0111754			
0111761			
0111768			
0111775			
0111782			
0111789			
0111796			
0111803			
0111810			
0111817			
0111824			
0111831			
0111838			
0111845			
0111852			
0111859			
0111866			
0111873			
0111880			
0111887			
0111894			
0111901			
0111908			
0111915			
0111922			
0111929			
0111936			
0111943			
0111950			
0111957			
0111964			
0111971			
0111978			
0111985			
0111992			
0111999			

```

NFARE: .ASCII 'KNO PAR ERR GOOD '
AGPAR: .ASCII ' BAD '
APAR: .ASCIZ ' '
FPCMSG: .ASCII ' FPC '
AFPC: .ASCIZ ' '
GOOD: .ASCIZ 'WILL TEST UNITS: '
GTAPES: .ASCIZ ' '
GTAB: .BYTE '0','1','2','3','4','5','6','7'

```

```

.EVEN
AWBUFO: OPEN ;WRITE BUFFER 0
WFBLKO: OPEN
RWCKSO: OPEN
WBUFO: OPEN
FWCKSO: OPEN
WRBLKO: OPEN
AWBUF1: OPEN ;WRITE BUFFER 1
WFBLK1: OPEN
RWCKS1: OPEN
WBUF1: OPEN
FWCKS1: OPEN
WRBLK1: OPEN
ARBUFO: OPEN ;READ BUFFER 0
RFBLKO: OPEN
RRCKSO: OPEN
RBUFO: OPEN
FRCKSO: OPEN
RRBLKO: OPEN

```

014360	000000			RRBUF1:	OPEN				; READ BUFFER 1
014362	000000			RFBLK1:	OPEN				
014364	000000				OPEN				
014366	000000				OPEN				
014370	000000			RRCKS1:	OPEN				
014372	000000			RBUF1:	OPEN				
	015372				. = +510.				
015372	000000			FRCKS1:	OPEN				
015374	000000				OPEN				
015376	000000				OPEN				
015400	000000			RRBLK1:	OPEN				
015402	000000				OPEN				
015404	000			EARBFO:	.BYTE	OPEN			; EXTENDED WRITE BUFFER 0
015405	000	000	000	EWFBKO:	.BYTE	OPEN, OPEN, OPEN			
015410	000			EWCKO:	.BYTE	OPEN			
015411	000			EWBUFO:	.BYTE	OPEN			
	016011				. = +255.				
016011	000	000	000	EWFKO:	.BYTE	OPEN, OPEN, OPEN			
016014	000	000		EWBKO:	.BYTE	OPEN, OPEN			
016016	000			EARBF1:	.BYTE	OPEN			; EXTENDED WRITE BUFFER 1
016017	000	000	000	EWFBK1:	.BYTE	OPEN, OPEN, OPEN			
016022	000			EWCK1:	.BYTE	OPEN			
016023	000			EWBUF1:	.BYTE	OPEN			
	016423				. = +255.				
016423	000	000	000	EWFK1:	.BYTE	OPEN, OPEN, OPEN			
016426	000	000		EWBK1:	.BYTE	OPEN, OPEN			
016430	000	000	000	EARBFO:	.BYTE	OPEN, OPEN, OPEN, OPEN			; EXTENDED READ BUFFER 0
016433	000								
016434	000			ERRCKO:	.BYTE	OPEN			
016435	000			ERBUFO:	.BYTE	OPEN			
	017035				. = +255.				
017035	000	000	000	ERFKO:	.BYTE	OPEN, OPEN, OPEN, OPEN, OPEN			
017040	000	000							
017042	000	000	000	EARBF1:	.BYTE	OPEN, OPEN, OPEN, OPEN			; EXTENDED READ BUFFER 1
017045	000								
017046	000			ERRCK1:	.BYTE	OPEN			
017047	000			ERBUF1:	.BYTE	OPEN			
	017447				. = +255.				
017447	000	000	000	ERFK1:	.BYTE	OPEN, OPEN, OPEN, OPEN, OPEN			
017452	000	000							
	000001			:					.END

RTNNO	001036	505*	575*	655	678	695*	701	812						
RVERSA	006500	1574*	1585											
RVERSA	006554	1574*	1576	1598*										
RVERSA	006556	1575*	1577	1591*										
RWCKSO	011274	1681	1699	1723	1728	1734	1746	1770	1780	1850	1890	1909	1959	1966
		2112*												
		2124*												
RWCKS1	012320	1463	1466*											
RWFBKA	006070	1479	1492*											
RWFBKE	006170	1462*	1623	1649										
RWFBK1	006054	1465*												
RWFBK2	006062	1462*	1465*	1478	1493*									
RWFIND	006172	422*	583*	606*	625*	678*	686*	704	705*	706*	707*	708*	709*	727
RO	=%000000	736*	747*	780	781*	782*	783	795*	799*	856*	907*	911	915*	916*
		917*	918*	936*	1047*	1121*	1130*	1291*	1308*	1332*	1349*	1367*	1427*	1566*
		1572	1584	1601*	1604	1605	1845*	1852*	1854*	1864*	1866*	1876*	1882*	1895*
		1901*	1914*											
R1	=%000001	423*	615*	726	737*	748*	774*	857*	859*	863	867	871*	873*	908*
		910*	925*	935*	940*	944*	950*	952*	1048*	1122*	1124*	1131*	1135*	1292*
		1333*	1371*	1374*	1376*	1377*	1379*	1381*	1383*	1385*	1386*	1404*	1405*	1406*
		1407*	1408*	1409*	1414	1415	1428*	1454*	1567*	1573				
R2	=%000002	424*	725	738*	749*	909*	910	919*	926*	937*	951*	952	954*	1051*
		1293*	1334*	1368*	1431*	1433*	1438*	1440*	1442*	1568*	1569*	1570	1571*	1572*
		1584												
R3	=%000003	425*	724	739*	750*	911*	912*	913*	914	927*	929*	938*	948*	1052*
		1372*	1373*	1375*	1432*	1434*	1437*	1439*	1441*	1445*	1448*	1570*	1573*	
R4	=%000004	426*	723	731*	751*	759	762*	767*	789	799	939*	942*	945*	946
R5	=%000005	427*	694*	699	722	730*	731	744	752*	758	761*	762	765	768*
		805	1401*	1458*	1499*	1504*	1515*	1519*	1525*	1587*	1649*	1690*	1722*	1733*
		1803*	1878*	1897*	1935*	1937*	1939*	1941*	1943*	1945*	1949*	1954*	1987*	
R6	=%000006	428*	574*	644*	664*									
R7	=%000007	429*	650*	702*	860*	865*	870*	872*	874*					
SAT	= 000000	472*												
SAV03	= 104002	532*	1364	1426	1565									
SAV05	= 104004	534*												
SAV05S	= 104006	536*	773	788	855	906	924	934	1046	1120	1129	1298	1329	1600
SAV5S	= 000003	485*	804	901	981	995	1018	1141	1226	1238				
SCOPE	= 104000	530*	988	1152	1166	1196	1193	1233	1251	1258	1300	1306	1341	1347
		1637	1663	1704	1751	1785	1825	1916	1947					
SCOPTA	001034	504*	671	698*										
SELDRA	006630	1603	1607*											
SELDRA	006602	571	1600*											
SELDRAV	= 104052	572*	586	622	1595									
SELE	= 004000	483*												
SEQDRV	= 104051	571*	653	670										
SETCOM	= 104020	546*	589	598	1180	1247	1296	1337						
SETIND	006366	1496*	1498*	1523	1530*									
SETWBA	006210	1497	1499*											
SETWBF	006174	1496*	1625	1678	1767	1844								
SETWBR	006202	1498*	1651	1721	1802									
SPBOT	= 001000	402*	574	644	664									
SQDRV	006560	570	1593*	1597										
SQDRVA	006574	1593*	1594*	1596*										
SQDRV1	001320	584*	585*	587*	615									
SR	= 177570	400*	636	651	655	668	672	676	680	701*	794	944		
SRCOON	004426	1183*	1199	1203	1205	1209	1213	1216	1224					

