

TC4 - TC11 TEST 4
DZTCDC.P11

MACY11 27(732) 08-SEP-76 09:04 PAGE 1

.REM !

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZTCDC-C-D

PRODUCT NAME: TC4 - TC11 TEST 4

DATE: JUNE 15, 1973

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: L. R. KOLLER

THIS MAINDEC OBSOLETE 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. REQUIREMENTS
-----2.1 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.

6.2 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 QVVVVV
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.
UNITW IS THE UNIT IN USE AT TIME OF FAILURE.
FPC QVVVVV 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. MISCELLANEOUS7.1 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 TYPE 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.

B. DESCRIPTION

THIS PROGRAM IS ORGANIZED INTO THREE MAIN SECTIONS:

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

B.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 "ERRR" SUBROUTINE, WHICH IS CALLED BY AN "ERRORN" STATEMENT AND TYPES THE TEST NUMBER AND PC VALUE WHEN A FAILURE OCCURS.

377
378
379
380
381 000000 000000
382 000000 000002
383 000002 000000
384 000004 000006
385 000006 000000
386 000010 000012
387 000012 000000
388 000014 002310
389 000016 000340
390 000020 002340
391 000022 000340
392 000024 000026
393 000026 000000
394 000030 002110
395 000032 000340
396 000034 002756
397 000036 000000
398
399
400 177570
401 177776
402 001000
403 000240
404 000000
405 100000
406 100000
407 040000
408 020000
409 010000
410 004000
411 002000
412 001000
413 000400
414 000200
415 000100
416 000040
417 000020
418 000010
419 000004
420 000002
421 000001
422 000000
423 000001
424 000002
425 000003
426 000004
427 000005
428 000006
429 000007
430 000007

.LIST SEQ,BIN,ME
.NLIST MC,MD
.ABS
;
.=0
.+2 ;UNASSIGNED TRAP
MACHER: HALT ;SP OVERFLOW, BUS ERROR TRAP
.+2
HALT
.+2 ;RESERVED INSTRUCTION TRAP
TRCV: HALT ;TRACE TRAP
SVSS
PRTY7
IOTV: RSSS ;TRAP TO CALL IOX
PRTY7
.+2 ;POWER FAIL TRAP
HALT
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

431	005746	PUSH=005746
432	024646	PUSH2=024646
433	005726	POPSP=005726
434	022626	POPSP2=022626
435	000340	PRTY7=340
436	000300	PRTY6=300
437	000240	PRTY5=240
438	000200	PRTY4=200
439	000140	PRTY3=140
440	000100	PRTY2=100
441	000040	PRTY1=40
442	000000	PRTY0=0
443	000007	BELL=007
444	177777	TLAST=-1
445	000003	TRC=3
446	000040	I=40
447	177777	X=-1
448	100000	A=BIT15
449	040000	B=BIT14
450	020000	C=BIT13
451	000000	V0=0
452	000004	V1=4
453	000010	V2=10
454	000014	V3=14
455	000020	V4=20
456	000024	V5=24
457	000030	V6=30
458	000034	V7=34
459	020000	MAINT=BIT13
460	010000	DINH=BIT12
461	004000	REV=BIT11
462	000000	FWD=0
463	000000	U0=0
464	000400	U1=BIT8
465	001000	U2=BIT9
466	001400	U3=BIT9!BIT8
467	002000	U4=BIT10
468	002400	U5=BIT10!BIT8
469	003000	U6=BIT10!BIT9
470	003400	U7=BIT10!BIT9!BIT8
471	000100	IE=BIT6
472	000000	SAT=0
473	000002	RNUM=BIT1
474	000004	RDATA=BIT2
475	000006	RALL=BIT2!BIT1
476	000010	SST=BIT3
477	000012	WRTM=BIT3!BIT1
478	000014	WDATA=BIT3!BIT2
479	000016	WALL=BIT3!BIT2!BIT1
480	000001	DO=BIT0
481	000200	UPS=BIT7
482	010000	ILO=BIT12
483	004000	SELE=BIT11
484	000000	EMTX=0
485	000003	SAVSS=3
486	000004	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
 =1000
 JMP GETRODY
 TCST: 177340
 ICCM: 177342
 TCWC: 177344
 TCBA: 177346
 TCDT: 177350
 TCVTR: 214
 TCLVL: PRTY6
 TPS: 177564
 TPB: 177566
 ICTR: OPEN
 ICNT: OPEN
 KSTART: TO
 SCOPTR: OPEN
 RTNNO: OPEN
 NXTST: OPEN
 CURTST: OPEN
 CTRA: OPEN
 TCCMT: OPEN
 TCSTT: OPEN
 TCDTT: 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:

;GO TO START OF PROGRAM.
 ;BYPASS INITIAL PRINTOUTS.
 ;TC11 STATUS REGISTER.
 ;TC11 COMMAND REGISTER.
 ;TC11 WORD COUNT REGISTER.
 ;TC11 BUS ADDRESS REGISTER.
 ;TC11 DATA REGISTER.
 ;TC11 INTERRUPT VECTOR
 ;TC11 INTERRUPT PRIORITY LEVEL.
 ;LSP CSR
 ;LSP BUFFER
 ;CONTAINS CURRENT ITERATION COUNT
 ;CONTAINS ACCUMULATED ITERATION COUNT.
 ;CONTAINS STARTING ROUTINE ADDR.
 ;CONTAINS CURRENT SCOPE POINTER.

.WORD CHAINN ; POINTER FOR EMT CALL SCOPE
 .WORD SRSETT ; 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 ERRN ; 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	.WORD	EHLT	; POINTER FOR EMT CALL	EHALT
544	001152	002364	.WORD	STTCV	; POINTER FOR EMT CALL	SVECTR
545	001154	003350	.WORD	STCOM	; POINTER FOR EMT CALL	SETCOM
546	001156	003310	.WORD	STATS	; POINTER FOR EMT CALL	STATUS
547	001160	003444	.WORD	STPDT	; POINTER FOR EMT CALL	STOPDT
549	001162	003454	.WORD	CKERZ	; POINTER FOR EMT CALL	CKERRZ
549	001164	003544	.WORD	NOINTR	; POINTER FOR EMT CALL	NOINT
550	001166	004344	.WORD	SRCHFF	; POINTER FOR EMT CALL	SRCHF
551	001170	004352	.WORD	SRCHRR	; POINTER FOR EMT CALL	SRCHR
552	001172	005010	.WORD	WDAF	; POINTER FOR EMT CALL	WDAF
553	001174	005036	.WORD	WDATR	; POINTER FOR EMT CALL	WDATAR
554	001176	005020	.WORD	RDAF	; POINTER FOR EMT CALL	RDAF
555	001200	005046	.WORD	RDATR	; POINTER FOR EMT CALL	RDATAR
556	001202	005064	.WORD	RDTFSS	; POINTER FOR EMT CALL	RDTFSS
557	001204	004244	.WORD	CWCBA	; POINTER FOR EMT CALL	CKWCBA
558	001206	004176	.WORD	CLEARR	; POINTER FOR EMT CALL	CLEAR
559	001210	004216	.WORD	BINFLL	; POINTER FOR EMT CALL	BINFIL
560	001212	003642	.WORD	DATCK	; POINTER FOR EMT CALL	DATCHK
561	001214	003632	.WORD	DTCKI	; POINTER FOR EMT CALL	DATCKI
562	001216	003616	.WORD	ADATCK	; POINTER FOR EMT CALL	ADTCK
563	001220	003606	.WORD	ADATCI	; POINTER FOR EMT CALL	ADTCKI
564	001222	003010	.WORD	INBINN	; POINTER FOR EMT CALL	INBIN
565	001224	003042	.WORD	GTBINI	; POINTER FOR EMT CALL	GETBINI
566	001226	005110	.WORD	WALLFF	; POINTER FOR EMT CALL	WALLF
567	001230	005126	.WORD	WALLRR	; POINTER FOR EMT CALL	WALLR
568	001232	005264	.WORD	RALLFF	; POINTER FOR EMT CALL	RALLF
569	001234	005302	.WORD	RALLRR	; POINTER FOR EMT CALL	RALLR
570	001236	006560	.WORD	SQDRV	; POINTER FOR EMT CALL	SEQDRV
571	001240	006602	.WORD	SELDRR	; POINTER FOR EMT CALL	SELDRV
572						

```

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 061524 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 MOVB #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 #WBUFO,@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+F4D+D0
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 001334 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 #,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 MOVB GTAB(0),GTAPES ;GET ASCII # FOR GOOD TAPE.
627 001514 104010 TYPE ;TYPE # OF UNIT TO TEST.
628 001516 011251 GTAPES

```

629	001520	005267	177752		CPENB:	INC	CPENA	:UPDATE TO NEXT DRIVE.
630	001524	005367	177314			DEC	CTRA	:CHECKED ALL DRIVES?
631	001530	001361				BNE	CPENA-2	:BR IF NOT.
632	001532	000412				BR	GETRDY	:YES.
633	001534	104010			STRTA:	TYPE		:TYPE UNIT SELECT INSTRUCTIONS.
634	001536	010361				INST1		
635	001540	104015				CHALT		:WAIT FOR USER.
636	001542	116767	176022	177316		MOV8	SR UNITS	:GET UNITS.
637	001550	001771				BEQ	STATA	:BR IF NON SELECTED.
638	001552	104010				TYPE		:TYPE SR OPTION MESSAGE.
639	001554	010541				ASETSR		
640	001556	104015				CHALT		:COMMON HALT.
641	001558	012767	001002	001042	GETRDY:	MOV	#1002,ERRNO	:ENABLE ERROR PRINTOUTS.
642	001556	016767	177240	177244		MOV	KSTART,NXTST	:ADDR OF 1ST ROUTINE TO NXTST
643	001574	012767	000340	176174	GTRDYX:	MOV	#PRTY7,PSW	:SET PRIORITY 7.
644	001602	012706	001000			MOV	#SPBOT,R6	:SET BOTTOM OF STACK.
645	001606	104001				SRESET		:ISSUE RESET.
646	001610	104043				INBIN		:INITIALIZE BINARY COUNT.
647	001612	005067	177274			CLR	ALLPAR	:CLEAR PARITY ERROR ALLOWED INDICATOR.
648	001616	104017				SVECTR		:PRESET DT INTERRUPT VECTOR TO 0.
649	001620	000000				0		
650	001622	004767	000214		GTRDYA:	JSR	R7,FORWC	:ROLL FORWARD TO "NEXT" ROUTINE.
651	001626	032767	001000	175734	GTRDYB:	BIT	#BIT9,SR	:CHECK SELECT ROUTINE SWITCH
652	001634	001003				BNE	GTRDYC	:BRANCH IF SELECT ROUTINE SWITCH IS SET.
653	001636	104051			GORUN:	SEQDRV		:SELECT SEQUENTIAL DRIVE.
654	001640	000177	177176			JMP	2COURTST	:GO RUN CURRENT ROUTINE.
655	001644	126767	177166	175716	GTRDYC:	CMPB	RTNNO,SR	:COMPARE RTNNO TO SR.
656	001652	001771				BEQ	GORUN	:BR IF ROUTINE FOUND.
657	001654	022767	177777	177156	GTRDYD:	CMP	#-1,NXTST	:NO. CHECK FOR LAST ROUTINE.
658	001662	001357				BNE	GTRDYA	:BRANCH IF NOT LAST ROUTINE.
659	001664	104010				TYPE		:TYPE INCORRECT RTN SELECTED.
660	001666	010314				AINCRT		
661	001670	104015				CHALT		:COMMON HALT.
662	001672	000732				BR	GETRDY	:START OVER.
663	001674	104022			CHAINN:	STOPDT		
664	001676	012706	001000			MOV	#SPBOT,R6	:RESTORE STACK.
665	001702	005267	177122			INC	ICNT	:INCREMENT ITERATION COUNT.
666	001706	001002				BNE	CHNAC	:BR IF RESULT NOT 0.
667	001710	005167	177114			COM	ICNT	:RESULT 0. RESET ICNT TO -1.
668	001714	032767	040000	175646	CHNAC:	BIT	#BIT14,SR	:CHECK FOR SCOPE OPTION.
669	001722	001403				BEQ	CHNA	:BRANCH IF SCOPE SW NOT SET.
670	001724	104051			CHNAB:	SEQDRV		:SELECT SEQUENTIAL DRIVE.
671	001726	000177	177102			JMP	2SCOPTR	:RETURN TO ROUTINE.
672	001732	032767	004000	175630	CHNA:	BIT	#BIT11,SR	:TEST INHIBIT ITERATION SWITCH
673	001740	001003				BNE	CHNAA	:BRANCH IF INHIBIT ITERATION SW SET.
674	001742	005367	177060			DEC	ICTR	:DECREMENT ITERATION COUNT.
675	001746	001366				BNE	CHNAB	:BRANCH IF COUNT NOT 0.
676	001750	032767	002000	175612	CHNAA:	BIT	#BIT10,SR	:ROUTINE END HALT SW SET? (SR!C)
677	001756	001403				BEQ	CHNB	:BRANCH IF NOT SET.
678	001760	016700	177052			MOV	RTNNO,R0	:TEST # TO R0.
679	001764	000000				HALT		:ROUTINE END HALT. TEST # IN LIGHTS.
680	001766	032767	001000	175574	CHNB:	BIT	#BIT9,SR	:CHECK SELECT ROUTINE SWITCH
681	001774	001271				BNE	GETRDY	:BRANCH IF SELECT RTN SW SET
682	001776	022767	177777	177034		JMP	#-1,NXTST	:LAST TEST?
683	002004	001273				BNE	GTRDYX	:BRANCH IF NOT LAST TEST.
684	002006	104010				TYPE		:TYPE PROGRAM END BELL.

685	002010	010576			APGEND		
686	002012	013700	000042		CHNC: MOV	3*42, R0	;GET CONTENTS OF 42.
687	002016	001410			BEQ	HERE	;BR IF 0.
688	002020	000005			RESET		;NOT 0. ISSUE RESET.
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	GETRDY	;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)+, SCOPTR	;GET SCOPE LOOP ENTRY POINTER.
699	002066	010567	176750		FORWDA: 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	R0, -(6)	;PUSH R0.
705	002112	016600	000002		MOV	2(6), R0	;GET EMT PC.
706	002116	014000			MOV	-(0), R0	;GET EMT CALL.
707	002120	006300			ASL	R0	;TIMES 2.
708	002122	016000	171114		MOV	EMTTAB-10000(0), R0	;FORM EMT ADDR.
709	002126	000200			RTS	R0	;GO TO EMT RTN. RESTORE R0.
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	R0, -(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			RS03: POPSP2		
736	002232	012600			MOV	(6)+, R0	;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), -(6)	;MOVE PC AND PSW DOWN STACK.


```

741 002246 016646 177766      MOV      -10.(6),-(6)
742 002252 000002      RTI      ;EXIT.
743 002254 010566 000020      :SUB TO SET R5 IN EMT PC AND RESTORE REGS 0 TO 5.
RSO5S: MOV      R5,16.(6)      ;SET EMT PC TO R5 CONTENTS.
744 002254 010566 000020      :SUB TO RESTORE REGS 0 TO 5.
RSO5S: POPSP2
745 002260 022626      MOV      (6)+,R0
746 002262 012600      MOV      (6)+,R1
747 002264 012601      MOV      (6)+,R2
748 002266 012602      MOV      (6)+,R3
749 002270 012603      MOV      (6)+,R4
750 002272 012604      MOV      (6)+,R5
751 002274 012605      MOV      -14.(6),-(6)      ;MOVE PC AND PSW DOWNSTACK.
752 002276 016646 177762      MOV      -14.(6),-(6)
753 002302 016646 177762      RTI      ;EXIT.
754 002306 000002      SV55S: MOV      (6)+,-6(6)      ;PC AND PSW UPSTACK.
755 002310 012666 177772      MOV      (6)+,-6(6)
756 002314 012666 177772      MOV      R5,-(6)      ;SAVE R5.
757 002320 010546      MOV      R4,-(6)      ;SAVE R4.
758 002322 010446      PUSH2
759 002324 024646      MOV      8.(6),R5      ;EMT PC TO R5.
760 002326 014605 000010      MOV      R5,R4      ;EMT PC TO R4.
761 002332 014504      TST      -(4)
762 002334 005744      RTI      ;EXIT EMT SUB.
763 002336 000002      RSS5S: MOV      R5,8.(6)      ;R5 TO EMT PC.
764 002340 010566 000010      POPSP2
765 002344 022626      MOV      (6)+,R4      ;RESTORE R4.
766 002346 012604      MOV      (6)+,R5      ;RESTORE R5.
767 002350 012605      MOV      -6(6),-(6)
768 002352 016646 177772      MOV      -6(6),-(6)
769 002356 016646 177772      RTI      ;EXIT.
770 002362 000002      :ROUTINE TO SET TC11 INTERRUPT VECTOR AND PRIORITY
771 002364 104006      STTCV: SAVO5S
772 002366 016701 176424      MOV      TCVTR,R1      ;VECTOR TO R1.
773 002372 012521      MOV      (5)+,(1)+      ;SET DESIRED VECTOR.
774 002374 016721 176420      MOV      TCLVL,(1)+      ;SET TC11 PRIORITY.
775 002400 104007      RSTOSS
776 002402 000002      RTI
777 002404 010046      :ROUTINE TO ISSUE RESET.
SRSETT: MOV      R0,-(6)      ;PUSH R0.
778 002406 012700 052525      MOV      #52525,R0      ;DATA TO R0.
779 002412 005100      COM      R0      ;COMPLEMENT (R0).
780 002414 010067 177770      MOV      R0,SRSETT+4      ;(R0) TO SRSETT+4.
781 002420 000005      RESET      ;ISSUE RESET. (R0) IS
782 002422 012600      MOV      (6)+,R0      ;RESTORE R0.
783 002424 000002      RTI      ;DISPLAYED. EXIT.
784 002426 104006      :COMMON HALT ROUTINE
CHLT: SAVO5S
785 002430 010400      MOV      R4,R0      ;DEVELOP ADDR OF CALLER.
786 002432 000000      HALT      ;HALT CALL ADDR IN DATA LIGTHS.
787 002434 104007      RSTOSS
788 002436 000002      RTI      ;EXIT.
789 002440 005767 175124      :CONDITIONAL ERROR HALT ROUTINE.
EHLT: TST      SR      ;CHECK FOR HALT ON ERROR.
790 002444 100001      BPL      EHLTA      ;BRANCH IF NO HALT DESIRED.
791 002446 000000      HALT      ;HALT.

```

797 002450 000002
798 002452 104022
799 002454 010467 176414
800 002460 104012
801 002462 001074
802 002464 011215
803 002466 000006
804 002470 000003
805 002472 010567 000136
806 002476 162767 000002 000130
807 002504 104012
808 002506 002634
809 002510 010260
810 002512 000006
811 002514 104012
812 002516 001036
813 002520 010250
814 002522 000003
815 002524 104013
816 002526 001030
817 002530 010275
818 002532 000005
819 002534 104012
820 002536 001064
821 002540 010312
822 002542 000001
823 002544 104012
824 002546 001060
825 002550 010724
826 002552 000006
827 002554 104012
828 002556 001050
829 002560 010656
830 002562 000006
831 002564 104012
832 002566 001046
833 002570 010642
834 002572 000006
835 002574 104012
836 002576 001054
837 002600 010610
838 002602 000006
839 002604 104012
840 002606 001056
841 002610 010625
842 002612 000006
843 002614 012767 010244 000012
844 002622 032767 020000 174740
845 002630 001002
846 002632 104010
847 002634 000000
848 002636 012567 177772
849 002642 022767 177777 177764
850 002650 001364
851 002652 104016
852 002654 000004

EHLTA: RFI
ERRN: STOPDT

R4,FPC

; IN DATA LIGHTS.
; ALL STOP.
; CONVERT CALL ADDR OF SUB CALLING.

OACNV
FPC
AFPC
6

000136
000002 000130

R5,ERRB
#2,ERRB

; SAVE REG 55
; DETERMINE CALLING ADDR.
; CONVERT CALLING ADDR TO ASCII.

OACNV
ERRB
APC
6

; CONVERT TEST # TO ASCII.

OACNV
RTNNO
ATNUMB

; CONVERT ICNT TO DECIMAL ASCII.

3
BDCNV
ICNT
AICNT

; CONVERT UNIT NUMBER TO ASCII.

5
OACNV
UNITN
AUNIT

; CONVERT BLKRQ TO ASCII.

1
OACNV
BLKRQ
ABLKRO

; CONVERT TCST TO ASCII.

6
OACNV
TCSTT
ATCST

; CONVERT TCCM TO ASCII.

6
OACNV
TCCMT
ATCCM

; CONVERT TCWC TO ASCII.

6
OACNV
TCWCT
ATCWC

; CONVERT TCBA TO ASCII.

6
OACNV
TCBAT
ATCBA

ERRNA: #EMD,ERRB
ERRND: #BIT13,SR
ERRNB

; TYPE ERR HEADER MSG IF NOT INHIBITED.
; INHIBIT ERR PRINT?
; BR TO INHIBIT.

ERRB: TYPE
ERRNB: OPEN
MOV (5)+,ERRB
CMP #-1,ERRB
BNE ERRNA

; TYPE MSG.
; DESIRED MSG ADDR GOES HERE.
; GET ADDR OF NEXT MSG.
; TERMINATOR?
; GO TYPE IF NOT TERMINATOR.
; END OF MSGS. HALT IF REQUIRED.
; RESTORE REG 55.

ERRNC: EHALT
RST55


```

909 003114 012502
910 003116 060201
911 003120 010003
912 003122 042703 177770
913 003126 062703 000060
914 003132 110341
915 003134 042700 000007
916 003140 006000
917 003142 006000
918 003144 006000
919 003146 005302
920 003150 001363
921 003152 104007
922 003154 000002
923
924 003156 104006
925 003160 012501
926 003162 012502
927 003164 012503
928 003166 112122
929 003170 005303
930 003172 001375
931 003174 104007
932 003176 000002
933
934 003200 104006
935 003202 013501 003302
936 003204 012700 003270
937 003210 012702 000005
938 003214 012703
939 003220 005004
940 003222 161201
941 003224 103402
942 003226 005204
943 003230 000774
944 003232 061201 000060
945 003234 062704
946 003240 110420
947 003242 005722
948 003244 005303
949 003246 001364
950 003250 012501
951 003252 012502
952 003254 060201
953 003256 114041
954 003260 005302
955 003262 001375
956 003264 104007
957 003266 000002
958 003270 023420
959 003272 001750
960 003274 000144
961 003276 000012
962 003300 000001
963 003302 040 040
964 003305 040 040

```

```

MOV (5)+,R2 ;GET CONVERT COUNT.
ADD R2,R1 ;DEVELOP ADDR TO STORE 1ST CHAR.
JACNVA: MOV R0,R3
BIC #177770,R3 ;ISOLATE LEAST SIGNIFICANT DIGIT.
ADD #60,R3 ;CONVERT DIGIT TO ASCII.
MOVB R3,-(1) ;STORE ASCII CHARACTER.
SIC #7,R0
ROR R0
ROR R0
ROR R0
DEC R2 ;DONE ALL DIGITS?
BNE OACNVA ;BRANCH IF NOT DONE.
RSTOSS ;RESTORE REGS.
RTI ;DONE. EXIT.
;EMT SUB TO MOVE VARIABLE NUMBER OF BYTES.
BMOVV: SAVOSS ;SAVE REGS.
MOV (5)+,R1 ;GET "FROM" ADDRESS
MOV (5)+,R2 ;GET "TO" ADDRESS
MOV (5)+,R3 ;GET COUNT
BMOVA: MOVB (1)+,(2)+ ;MOVE BYTE
DEC R3 ;DECREMENT COUNT
BNE BMOVA ;BRANCH IF NOT DONE.
RSTOSS ;RESTORE REGS.
RTI ;DONE. EXIT.
;EMT SUB TO CONVERT BINARY TO DECIMAL ASCII.
BDCNVV: SAVOSS ;SAVE REGS.
MOV @((5)+,R1) ;GET BINARY VALUE.
MOV #DECVAL,R0 ;ADDR OF DECVAL TO R0.
MOV #TENPWR,R2 ;ADDR OF 10 POWER TO R2.
MOV #5,R3 ;SET UP FOR 5 CONVERSIONS.
BDCNVA: CLR R4 ;CLEAR RESULT.
BDCNVB: SUB (2),R1 ;10 POWER FROM VALUE.
BCS BDCNVC ;BR IF UNSUCCESSFUL.
INC R4 ;+1 TO RESULT.
BR BDCNVB ;DO IT AGAIN.
BDCNVC: ADD (2),R1 ;RESTORE SUBTRACTED VALUE.
ADD #60,R4 ;CONVERT RESULT TO ASCII.
M.VB R4,(0)+ ;STORE RESULT.
TST (2)+ ;UPDATE 10 POWER ADDR.
DEC R3 ;DONE 5 TIMES?
BNE BDCNVA ;BR IF NOT.
MOV (5)+,R1 ;GET ADDR TO STORE ASCII.
MOV (5)+,R2 ;GET # OF DIGITS REQUIRED.
ADD R2,R1 ;START WITH LSD.
BDCNVD: MOVB -(0),-(1) ;TRANSFER CHARACTER.
DEC R2 ;DONE?
BNE BDCNVD ;BR IF NOT.
RSTOSS ;RESTORE REGS.
RTI ;EXIT.
TENPWR: 10000.
1000.
100.
10.
1
DECVAL: .BYTE 040,040,040,040,040,040

```

```

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,TCCT ;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 @6,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 STCMMSG ;OF READY AND/OR ERROR BIT(S). OR ILO.
986 003430 010634 STAT ;BLOCK MISS, OR DATA MISS ERROR OCCURRED.
987 003432 77777 -1
988 003434 04000 SCOPE
989 003436 062716 000002 STCOMB: ADD #2,(6) ;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: RST55
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 ;DECTAPE 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		;EXIT EMT SUB.
1027					;EMT SUB TO CHECK EXPECTED DATA AGAINST ACTUAL DATA AND REPORT ERRORS.		
1028	003566	000000			DATKNT: 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: MOV	#-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	MOV	#-1,CKINDB	;INDICATE 18 BIT DATA CHECK.
1041	003630	000410			BR	DATCKK	
1042	003632	112767	177777	177744	DTCKI: MOV	#-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	DATCKB: MOV	ERRLIM,ERRCTR	;ERR LIMIT TO ERROR COUNTER.
1056	003712	005067	177656		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	177535	ROR	ACTDAT	
1080	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	WRDCNT
1116	004170	001250		BNE	DATCKB	; 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	004216	104006		; EMT SUB TO FILL AREA WITH BINARY COUNT PATTERN.		
1130	004220	012500		BINFLL:	SAVOSS	
1131	004222	012501		MOV	(5)+, R0	; GET STARTING ADDR.
1132	004224	104044		MOV	(5)+, R1	; GET COUNT.
				BINFLA:	GETBNI	; GET BINARY WORD.

```

1133 004226 000000          BINFLB: OPEN          ;BINARY WCRD IS STORED HERE.
1134 004230 016720 177772      MOV      BINFLB,(0)+  ;STORE WORD.
1135 004234 005301          DEC      RI           ;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.
1141 004244 000003          CWCBA: SAVSS
1142 004246 012567 174620      MOV      (5)+,TEMP  ;GET EXPECTED TCBA CONTENTS.
1143 004252 104021          STATUS         ;SAVE TCWC AND TCBA.
1144 004254 005777 174530      TST      @TCWC     ;WORD COUNT 0?
1145 004260 001407          BEQ      CWCBB     ;BR IF 0 (OK).
1146 004262 104011          ERRORN        ;WORD COUNT NOT 0. TYPE
1147 004264 011207          FPCMSG        ;CONTENTS OF TCWC AND TCBA.
1148 004266 010733          WCNOTO
1149 004270 010602          CTCWC
1150 004272 010617          CTCBA
1151 004274 177777          -1
1152 004276 104000          SCOPE
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
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          INTCB        ;ACTUAL TCBA, AND TCWC.
1162 004326 010763          TCBASB
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.
1171 004344 105067 000315      SRCHFF: CLRB     DIRIND ;SET FORWARD INDICATOR.
1172 004350 000403          BR      SRCHA
1173 004352 112767 177777 000305 SRCHRR: MOVB    #-1,DIRIND ;SET REVERSE INDICATOR.
1174 004360 012777 004440 174430 SRCHA:  MOV     #SRCHC,@TCVTR ;SET INTERRUPT VECTOR TO SRCHC.
1175 004366 112767 000005 000270      MOVB    #5,REV CNT ;SET MAX # OF REVERSALS ALLOWED.
1176 004374 052767 004000 000020      BIS     #REV,SRCHM ;SET REV BIT IN SRCHM.
1177 004402 032777 004000 174376      BIT     #REV,@TCM   ;REV BIT SET IN TCCM?
1178 004410 001003          BNE     SRCHAA     ;BR IF YES.
1179 004412 042767 004000 000002      BIC     #REV,SRCHM ;NO. CLEAR REV BIT FROM SRCHM.
1180 004420 104020          SRCHAA: SETCOM  ;START SEARCH.
1181 004422 000103          SRCHM: RNUM!IE!DO
1182 004424 000402          BR      SRCHB
1183 004426 005277 174354          SRCCON: INC     @TCCM ;ISSUE DO TO ENABLE RNUM.
1184 004432 104400          SRCHB: DELAY   ;TIME OUT INTERRUPT.
1185 004434 104024          NOINT  ;FAILURE TO INTERRUPT.
1186 004436 104000          SCOPE
1187 004440 012716 004446          SRCHC: MOV     #SRCHD,(6) ;HERE WHEN INTERRUPT OCCURS.
1188 004444 000002          RTI           ;EXIT TO SRCHD.

```


1189	004446	022626			SRCHD:	POVSP2			;RESTORE STACK.
1190	004450	005777	174332			TST	@TCCM		;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, @TCCM		;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	SRCHE:	BIT	#BIT11, @TCCM		;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, @TCCM		;TCDT EQUAL. CHECK DIRECTION.
1211	004564	001004				BNE	SRCHG		;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, @TCCM		;REV BIT SET?
1219	004614	001404				BEQ	SRCRVB		;BR IF NOT.
1220	004616	042777	004000	174162	SRCRVA:	BIC	#BIT11, @TCCM		;IN REVERSE. SET TO FORWARD.
1221	004624	000403				BR	SRCRVC		
1222	004626	052777	004000	174152	SRCRVB:	BIS	#BIT11, @TCCM		;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 @TCWM ;ERROR BIT SET?
1255 004764 100005 BPL WRDFRF ;BR IF NOT.
1256 004766 005067 174120 WRDFRZ: CLR ALLPAR
1257 004772 104023 CKERRZ ;CHECK FOR ERRORS.
1258 004774 104000 SCOPE ;ERROR RETURN.
1259 004776 000240 NOP ;ENDZ RETURN.
1260 005000 104034 WRDFRF: CKWCBA ;CHECK WORD COUNT AND CURRENT ADDR.
1261 005002 000000 WRDFRG: OPEN ;TCBA SHOULD EQUAL THIS.
1262 005004 000004 WRDFRE: RSTSS
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 #5167, 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: SAVOSS
1289 005144 005077 173640 CLR @TCWC ;ZERO TCWC
1290 005150 005077 173636 CLR @TCBA ;ZERO TCBA
1291 005154 012500 MOV (5)+, R0 ;TRANSFER COUNT TO R0.
1292 005156 012501 MOV (5)+, R1 ;DATA ADDR TO R1.
1293 005160 012502 MOV (5)+, 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
1302 005210 012177 173600
1303 005214 005777 173566
1304 005220 100003
1305 005222 104023
1306 005224 104000
1307 005226 000240
1308 005230 005300
1309 005232 001401
1310 005234 000002
1311 005236 042777 000100 173542
1312 005244 032777 001000 173532
1313 005252 001774
1314 005254 022626
1315 005256 022626
1316 005260 104007
1317 005262 000002
    
```

```

CWALLC: MOVB (2)+,@TCST
         MOV (1)+,@TCDT
         TST @TCCM
         BPL CWALLD
         CKERRZ
         SCOPE
         NOP
CWALLD: DEC RD
         BEQ CWALLE
         RTI
CWALLE: BIC #IE,@TCCM
         IS: BIT #1000,@TCST
         BEQ IS
CWALLF: POPSP2
         POPSP2
         RSTOSS
         RTI
    
```

```

; HERE ON INTERRUPT. LOAD EXTENDED DATA BITS IN TCST
; LOAD DATA IN TCDT
; ERROR BIT SET?
; BR IF NO ERROR.
; GO CHECK ON ERROR.
; ERROR RETURN.
; ENDZ RETURN.
; ALL TRANSFERS DONE?
; BR IF YES.
; NO. EXIT INTERRUPT
; DISABLE INTERRUPT.
; WAIT FOR DATA MISS.
; BR IF NONE YET.
; RESTORE STACK TO STATE BEFORE DELAY.

; RESTORE REGS.
; EXIT WALL SUBROUTINE
    
```

```

:318 :READ ALL SUBROUTINE. FORWARD OR REVERSE.
:319 :CALL: RALLF OR RALLR :READ ALL FORWARD OR REVERSE
:320 :COUNT :TRANSFER COUNT
:321 :DADDR :DATA ADDR
:322 :EDADDR :EXTENDED DATA ADDR.
:323
:324 005264 012767 000107 000056 RALLFF: MOV #RALL!FWD!IE!DO,CRALLB ;SETUP READ ALL FORWARD.
:325 005272 012767 104025 000036 MOV #SRCHF,CRALLA
:326 005300 000406 ER CRALL
:327 005302 012767 004107 000040 RALLRR: MOV #RALL!REV!IE!DO,CRALLB ;SETUP READ ALL REVERSE.
:328 005310 012767 104026 000020 MOV #SRCHR,CRALLA
:329 005316 104006 CRALL: SAVJSS
:330 005320 005077 173464 CLR @TCWC ;ZERO TCWC
:331 005324 005077 173462 CLR @TCBA ;ZERO TCBA
:332 005330 012500 MOV (S)+,R0 ;TRANSFER COUNT TO R0.
:333 005332 012501 MOV (S)+,R1 ;DATA ADDR TO R1
:334 005334 012502 MOV (S)+,R2 ;EXTENDED DATA ADDR TO R2.
:335 005336 000000 CRALLA: OPEN ;SRCHF OR SRCHR CALL.
:336 005340 012777 005360 173450 MOV #CRALLC,@TCVTR ;SET INTERRUPT VECTOR TO CRALLC.
:337 005346 104020 SETCOM ;ISSUE RALLF OR RALLR
:338 005350 000000 CRALLB: OPEN ;COMMAND GOES HERE.
:339 005352 104400 DELAY ;WAIT FOR INTERRUPT
:340 005354 104024 NOINT ;FAILURE TO INTERRUPT.
:341 005356 104000 SCOPE
:342 005360 117722 173420 CRALLC: MOV @TCST,(2)+ ;STORE EXTENDED DATA BITS
:343 005364 017721 173424 MOV @TCDT,(1)+ ;STORE DATA
:344 005370 005777 173412 TST @TCOM ;ERROR BIT SET?
:345 005374 100003 BPL CRALLD ;BR IF NO ERROR.
:346 005376 104023 CKERRZ ;GO CHECK ON ERROR.
:347 005400 104000 SCOPE ;ERROR RETURN.
:348 005402 000240 NOP ;ENDZ RETURN.
:349 005404 005300 CRALLD: DEC R0 ;ALL TRANSFERS DONE?
:350 005406 001401 BEQ CRALLE ;BR IF YES.
:351 005410 000002 RTI ;NO. EXIT INTERRUPT
:352 005412 112777 000002 173356 CRALLE: MOV @RNUM,@TCOM ;STOP RALL BY SWITCHING TO RNUM COMMAND.
:353 005420 022626 CRALLF: POPSP2 ;RESTORE STACK TO STATE BEFORE DELAY.
:354 005422 022626 POPSP2
:355 005424 104007 RSTOSS
:356 005426 000002 RTI ;RESTORE REGS.
:EXIT RALL SUBROUTINE

```

```

1357
1358
1359
1360
1361
1362
1363
1364 005430 104002
1365 005432 005067 173444
1366 005436 005067 173442
1367 005442 012500
1368 005444 012502
1369 005446 012767 000400 000166
1370 005454 012067 000160
1371 005460 016701 000154
1372 005464 112203
1373 005466 006003
1374 005470 006001
1375 005472 006003
1376 005474 006001
1377 005476 000301
1378 005500 004767 000150
1379 005504 042701 177700
1380 005510 004767 000146
1381 005514 016701 000120
1382 005520 004767 000120
1383 005524 042701 177700
1384 005530 004767 000126
1385 005534 016701 000100
1386 005540 042701 177700
1387 005544 004767 000112
1388 005550 005367 000066
1389 005554 001337
1390 005556 104003
1391 005560 005167 173316
1392 005564 042767 177700 173310
1393 005572 016767 173304 173300
1394 005600 016767 173276 173270
1395 005606 000367 173270
1396 005612 012767 000004 000022
1397 005620 006367 173256
1398 005624 006167 173254
1399 005630 005367 000006
1400 005634 001371
1401 005636 000205
1402 005640 000000
1403 005642 000000
1404 005644 006201
1405 005646 006201
1406 005650 006201
1407 005652 006201
1408 005654 006201
1409 005656 006201
1410 005660 000207
1411 005662 016767 173214 000040
1412 005670 016767 173206 000034

```

```

:SUBROUTINE TO CALCULATE FORWARD CHECKSUM FOR 256 WORD DATA BLOCK.
:2 MSB BITS OF 6 BIT CHKSUM ARE STORED AT LOC ELPB (RIGHT JUSTIFIED)
:THE OTHER 4 BITS ARE STORED AT LOC LPB (LEFT JUSTIFIED)
:SUBROUTINE CALL IS: JSR R5,PARITY ;CALL TO PARITY SUBROUTINE
: ADDR ;ADDR OF DATA STRING
: EADR ;ADDR OF EXTENDED DATA STRING.
PARITY: SAV03
CLR LPB ;CLEAR CHECKSUM WORDS.
CLR ELPB
MOV (5)+,R0 ;DATA STRING ADDR TO R0.
MOV (5)+,R2 ;EXTENDED DATA STRING ADDR TO R2.
MOV #256,PARCTR ;SETUP TO COMPUTE PARITY FOR 256 WORDS.
PARTYA: MOV (0)+,PWORD ;MOVE DATA WORD TO PWORD
MOV PWORD,R1 ;AND TO R1.
MOV#B (2)+,R3 ;EXTENDED DATA BYTE TO R3.
ROR R3
ROR R1
ROR R3
ROR R1
SWAB R1 ;PREPARE TO COMPUTE PARITY ON 6 MOST SIGNIFICANT
JSR PC,ASR2 ;BITS. INCLUDES EXTENDED BITS 16 AND 17.
BIC #177700,R1
JSR PC,BITCOM ;GO COMPUTE PARITY.
MOV PWORD,R1 ;PREPARE CENTER 6 BITS FOR PARITY COMPUTATION
JSR PC,ASR6
BIC #177700,R1
JSR PC,BITCOM ;GO COMPUTE PARITY
MOV PWORD,R1 ;PREPARE 6 LEAST SIGNIFICANT BITS FOR PARITY
BIC #177700,R1 ;COMPUTATION
JSR PC,BITCOM ;GO COMPUTE PARITY.
DEC PARCTR ;DONE 256 TIMES?
BNE PARTYA ;BR IF NOT.
RST03
COM LPB
BIC #177700,LPB ;MOVE 2 MOST SIGNIFICANT BITS OF
MOV LPB,LPBT ;SAVE COMPUTED PARITY.
MOV LPB,LPBG
XPARTY: SWAB LPB ;COMPUTED PARITY TO ELPB (RIGHT JUSTIFIED)
MOV #4,PARCTR ;AND 4 LEAST SIGNIFICANT TO LPB (LEFT JUSTIFIED)
PARTYB: ASL LPB
ROL ELPB
DEC PARCTR
BNE PARTYB
RTS R5 ;EXIT.
PWORD: OPEN
PARCTR: OPEN
ASR6: ASR R1 ;ENTRY TO SHIFT RIGHT 6 REG 1 SUB.
ASR R1
ASR R1 ;ENTRY TO SHIFT RIGHT 4 REG 1 SUB
ASR R1
ASR R1
ASR2: ASR R1
ASR R1
RTS PC ;EXIT SHIFT RIGHT SUB.
BITCOM: MOV LPB,LPBY ;SUBROUTINE TO COMPUTE 6 BIT PARITY
MOV LPB,LPBZ

```

```

1413 005676 005167 000030          CUM      LPBZ
1414 005702 040167 000022          BIC      R1,LPBY
1415 005706 040167 000020          BIC      R1,LPBZ
1416 005712 046767 000012 173162          BIC      LPBY,LPB
1417 005720 056767 000006 173154          BIS      LPBZ,LPB
1418          RTS      PC
1419 005726 000207
1419 005730 000000          LPBY:   OPEN
1420 005732 000000          LPBZ:   OPEN
1421          :SUBROUTINE TO PERFORM COMPLEMENT OBVERSE ON DATA SPECIFIED.
1422          :SUBROUTINE CALL:          JSR      R5,OBVERS ;CALL TO SUBROUTINE
1423          :                          ADDR      ;ADDR OF DATA STRING
1424          :                          EADR      ;ADDR OF EXTENDED DATA STRING
1425          :                          COUNT     ;NUMBER OF WORDS TO PROCESS.
1426 005734 104002          OBVERS: SAVO3
1427 005736 012500          MOV      (5)+,R0          ;GET ADDR OF DATA STRING TO R0
1428 005740 012501          MOV      (5)+,R1          ;ADDR OF EXTENDED DATA TO R1.
1429 005742 012567 000102          MOV      (5)+,OBVCNT     ;COUNT TO OBVCNT
1430 005746 012767 000006 000076 OBVA:   MOV      #6,OBVCTR
1431 005754 011002          MOV      (0),R2          ;DATA WORD TO R2
1432 005756 111103          MOV      (1),R3          ;EXTENDED DATA BYTE TO R3.
1433 005760 005102          COM      R2
1434 005762 005103          COM      R3
1435 005764 005010          CLR      (0)          ;CLEAR DESTINATION WORD.
1436 005766 105011          CLRB    (1)
1437 005770 006003          ROR      R3
1438 005772 006002          CBVB:   ROR      R2
1439 005774 006003          ROR      R3
1440 005776 006002          ROR      R2
1441 006000 006003          ROR      R3
1442 006002 006002          ROR      R2
1443 006004 006110          ROL      (0)
1444 006006 106111          ROLB    (1)
1445 006010 006103          ROL      R3
1446 006012 006110          ROL      (0)
1447 006014 106111          ROLB    (1)
1448 006016 006103          ROL      R3
1449 006020 006110          ROL      (0)
1450 006022 106111          ROLB    (1)
1451 006024 005367 000022          DEC      OBVCTR          ;DONE 6 TIMES?
1452 006030 001360          BNE      OBVB          ;BR IF NOT DONE.
1453 006032 005720          TST      (0)+          ;UPDATE DATA ADDRESSES.
1454 006034 005201          INC      R1
1455 006036 005367 000006          DEC      OBVCNT          ; DONE?
1456 006042 001341          BNE      OBVA          ;BR IF NOT.
1457 006044 104003          RSTO3
1458 006046 000205          RTS      R5          ;EXIT
1459 006050 000000          OBVCNT: OPEN
1460 006052 000000          OBVCTR: OPEN
1461          :SUBROUTINE WRITE DATA, READ DATA, SINGLE BLOCK, BINARY COUNT.
1462 006054 005067 000112          RWFBK1: CLR      RWFIND          ;SET SINGLE BLOCK INDICATOR.
1463 006060 000403          BR      RWFBKA
1464          :SUBROUTINE TO WRITE DATA, READ DATA, 2 BLOCKS, BINARY COUNT.
1465 006062 012767 177777 000102          RWFBK2: MOV      #-1,RWFIND          ;SET 2 BLOCK INDICATOR.
1466 006070 004767 000312          RWFBKA: JSR      PC,BINFLO          ;BIN FILL 256 WORD WRITE BUFFER 0.
1467 006074 016767 173010 172756          MOV      BLKNUM,BLKRQ
1468 006102 104027          WDATAF          ;CALL WDATAF SUB TO WRITE FWD 256. WORDS

```

```

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          BEQ      RWFBKE ; BR IF YES.
1480 006134 004767 000272  JSR      PC,BINFLI ; NO. DOUBLE. BIN FILL WBUF1.
1481 006140 005267 172714  INC      BLKRQ
1482 006144 104027          WDATAF          ; CALL WDATAF 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          RWFBKE: RTS    PC
1493 006172 000000          RWFIND: 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 177777 000156 SETWBR: MOV      *-1,SETIND       ; SET REV INDICATOR.
1499 006210 004567 177214  SETWBA: JSR      R5,PARITY       ; COMPUTE PARITY FOR WBUFO.
1500 006214 011276          WBUFO
1501 006216 015411          EWBUFFO
1502 006220 016767 172656 004050 MOV      LPB,FWCKSO          ; MOVE PARITY TO FWD CHECKSUM IN WBUFO.
1503 006226 116767 172652 007555 MOVB     ELPB,EWFCKO
1504 006234 004567 177170          JSR      R5,PARITY          ; COMPUTE PARITY FOR WBUF1.
1505 006240 012322          WBUF1
1506 006242 016023          EWBUFF1
1507 006244 016767 172632 005050 MOV      LPB,FWCKSI          ; MOVE PARITY TO FWD CHECKSUM IN WBUF1.
1508 006252 116767 172626 010143 MOVB     ELPB,EWFCKI
1509 006260 016767 172574 003000 MOV      BLKRQ,WFBKLO        ; SET UP FORWARD AND REVERSE BLOCK NUMBERS
1510 006266 016767 172566 004016 MOV      BLKRQ,WFBKLI        ; IN WRITE BUFFERS.
1511 006274 016767 172560 004002 MOV      BLKRQ,WBBLKO
1512 006302 016767 172552 005020 MOV      BLKRQ,WBBLKI
1513 006310 005267 003776          INC      WFBKLI
1514 006314 005267 005010          INC      WBBLKI
1515 006320 004567 177410          JSR      R5,OBVERS          ;
1516 006324 012304          WRBLKO
1517 006326 016014          EWRBKO
1518 006330 000001          I
1519 006332 004567 177376          JSR      R5,OBVERS          ;
1520 006336 013330          WRBLKI
1521 006340 016426          EWRBKI
1522 006342 000001          I
1523 006344 005767 000016  TST      SETIND          ; REVERSE SET UP?
1524 006350 001405          BEQ      STWBE           ; BR IF NOT.

```

1525	006352	004567	177356	JSR	R5,OBVERS	;REVERSE SET UP. COMPLEMENT OBVERSE
1526	006356	011264		AWBUFO		;ENTIRE WRITE BUFFER.
1527	006360	015404		EAWBFO		
1528	006362	001020		528.		
1529	006364	000207		STWBE: RTS	PC	;EXIT.
1530	006366	000000		SETIND: OPEN		
1531				;SUBROUTINE TO		CLEAR ENTIRE READ BUFFER.
1532	006370	104035		CLRRBF: CLEAR		
1533	006372	013334		ARBUFO		
1534	006374	001024		532.		
1535	006376	104035		CLEAR		
1536	006400	016430		EARBFO		
1537	006402	000412		266.		
1538	006404	000207		RTS	PC	;EXIT.
1539	006406	104035		BINFLO: CLEAR		;CLEAR ENTIRE WBUFO.
1540	006410	011264		AWBUFO		
1541	006412	000412		266.		
1542	006414	104035		CLEAR		;CLEAR EXTENDED WBUFO.
1543	006416	015404		EAWBFO		
1544	006420	000205		133.		
1545	006422	104036		BINFIL		;FILL WRITE BUFFER 0 WITH BINARY COUNT.
1546	006424	011276		WBUFO		
1547	006426	000400		256.		
1548	006430	000207		RTS	PC	;EXIT.
1549	006432	104035		BINFL1: CLEAR		;CLEAR ENTIRE WBUF1.
1550	006434	012310		AWBUF1		
1551	006436	000412		266.		
1552	006440	104035		CLEAR		;CLEAR EXTENDED WBUF1.
1553	006442	016016		EAWBF1		
1554	006444	000205		133.		
1555	006446	104036		BINFIL		;FILL WRITE BUFFER 1 WITH BINARY COUNT.
1556	006450	012322		WBUF1		
1557	006452	000400		256.		
1558	006454	000207		RTS	PC	;EXIT.
1559				;SUBROUTINE TO		REVERSE SEQUENCE OF 2 DATA STRINGS. 1ST STRING IS A WORD STRING,
1560				;2ND STRING IS		A BYTE STRING. BOTH STRINGS MUST HAVE SAME NUMBER OF ELEMENTS.
1561				:CALL: JSR	R5,REVERS	;CALL REVERS SUB.
1562				:ADR		;ADDR OF WORD STRING.
1563				:BADR		;ADDR OF BYTE STRING.
1564				:CNT		;NUMBER OF ELEMENTS TO REVERSE.
1565	006456	104002		REVERS: SAVO3		
1566	006460	012500		MOV	(5)+,R0	;WORD STRING ADDR TO R0.
1567	006462	012501		MOV	(5)+,R1	;BYTE STRING ADDR TO R1.
1568	006464	012502		MOV	(5)+,R2	;COUNT TO R2.
1569	006466	005302		DEC	R2	;DECREMENT R2 BY 1 TO FIGURE ADDR OF LAST
1570	006470	010203		MOV	R2,R3	;ELEMENT FOR BOTH STRINGS.
1571	006472	006302		ASL	R2	
1572	006474	060002		ADD	R0,R2	;ADDR OF END DATA WORD IN R2.
1573	006476	060103		ADD	R1,R3	;ADDR OF END DATA BYTE IN R3.
1574	006500	011067	000050	RVERSA: MOV	(0),RVERSA	
1575	006504	011267	000046	MOV	(2),RVERSB	
1576	006510	016712	000040	MOV	RVERSA,(2)	
1577	006514	016720	000036	MOV	RVERSB,(0)+	
1578	006520	111167	000030	MOVB	(1),BRVERSA	
1579	006524	111367	000025	MOVB	(3),BRVERSB	
1580	006530	116713	000020	MOVB	BRVERSA,(3)	


```

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   RVRSA         ;BR IF YES. REVERSAL NOT COMPLETE.
1586 006550 104003              RSTO3
1587 006552 000205              RTS    RS           ;YES. DONE.
1588 006554                    ;EXIT.
1589 006554      000          RVRSA:
1590 006555      000          BRVRSB: .BYTE  OPEN
1591 006556 000000          BRVRSB: .BYTE  OPEN
1592                    RVRSB:  OPEN
1593 006560 005267 000010      ;EMT SUB TO SELECT SEQUENTIAL DECTAPE UNIT.
1594 006564 042767 177770 000002  SQRV:  INC    SQRVA
1595 006572 104052              BIC   #177770,SQRVA
1596 006574 000000              SELDRV
1597 006576 000770          SQRVA: OPEN          ;DESIRED UNIT NUMBER.
1598 006600 000002              BR    SQRV         ;UNIT NOT AVAILABLE RETURN.
1599                    RTI          ;UNIT SELECTED. EXIT.
1600 006602 104006          ;EMT SUB TO SELECT DECTAPE UNIT SPECIFIED.
1601 006604 012500          SELDRR: SAVOSS
1602 006606 136067 006634 172252  MOV   (5)+,R0      ;GET NUMBER OF UNIT TO BE SELECTED.
1603 006614 001405          BITB  UNTAB(0),UNITS ;SEE IF UNIT AVAILABLE FOR TESTING.
1604 006616 010067 172242          BEQ   SELDRA       ;BR IF UNIT NOT AVAILABLE.
1605 006622 110067 172235          MOV   R0,UNITN    ;AVAI. SELECT UNIT.
1606 006626 005725          MOVB  R0,UNIT+1
1607 006630 104007          TST   (5)+        ;SET UP SELECTED EXIT.
1608 006632 000302          SELDRA: RSTOSS
1609 006634      001      002      004          RTI          ;EXIT.
1610 006637      010      020      040          UNTAB: .BYTE  BIT0,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7
1611 006642      100      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 006660 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,BLKRG
     JSR PC,SETWBF ;SET UP WRITE BUFFER FWD.
     JSR PC,CLRBUF ;CLEAR READ BUFFERS.
     RALLF ;READ ALL FWD 258. WORDS INTO
     258. ;ADDR RACKSO AND UP. EXTENDED DATA BITS
     RACKSO ;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

```

```

:*****
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,BLKRG
     JSR PC,SETWBF ;SET UP WRITE BUFFER REV.
     JSR PC,CLRBUF ;CLEAR READ BUFFERS.
     RALLR ;READ ALL REV 258. WORDS INTO
     258. ;ADDR RACKSO AND UP. EXTENDED DATA BITS ARE
     RACKSO ;STORED IN CONSECUTIVE BYTES STARTING
     ERRCKO ;AT ADDRESS ERRCKO
     ADTKI ;CALL ADTKI 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 ERFKO-1. ACTUAL
     EWBUFF ;IS CHECKED IN DESCENDING ORDER. FIFTH ARGUMENT
     ERFKO-1 ;REPRESENTS # OF 18 BIT ELEMENTS TO BE CHECKED.
     256.
     SCOPE

```

```

1664
1665 007004 000002
1666 007006 007120
1667 007010 000012
1668 007012 007014
1669
1670
1671
1672
1673
1674
1675
1676 007014 004767 177366
1677 007020 016767 172064 172032
1678 007026 004767 177142
1679 007032 004045
1680 007034 000402
1681 007036 011274
1682 007040 015410
1683 007042 104031
1684 007044 000400
1685 007046 013346
1686 007050 104037
1687 007052 011276
1688 007054 013346
1689 007056 000400
1690 007060 004567 176650
1691 007064 011264
1692 007066 015404
1693 007070 000412
1694 007072 104050
1695 007074 000402
1696 007076 013344
1697 007100 016434
1698 007102 104042
1699 007104 011274
1700 007106 014346
1701 007110 015410
1702 007112 017035
1703 007114 000402
1704 007116 104000

```

```

*****
t2: 2 ;ROUTINE NUMBER 2 *
    3 ;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 ;WRITE ALL FWD 258. WORDS STARTING
           258. ;FROM ADDR RWCKSO . EXTENDED DATA BITS ARE
           ;TAKEN FROM CONSECUTIVE BYTES STARTING
           EWRCKO ;AT ADDRESS EWRCKO .
           RDATAF ;CALL RDATAF SUB TO READ FWD 256. WORDS
           256. ;AND STORE AT ADDR STARTING AT RBUFO
           RBUFO
           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.
           JSR RS,OBVERS
           RACKSO
           ERWBFO
           266.
           RALLR ;READ ALL REV 258. WORDS INTO
           258. ;ADDR RACKSO AND UP. EXTENDED DATA BITS ARE
           RACKSO ;STORED IN CONSECUTIVE BYTES STARTING
           ERACKO ;AT ADDRESS ERACKO .
           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

```

```

1705
1706 007120 000003
1707 007122 007246
1708 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 *
    4 ;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,BLKRQ
     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
     ERCKO ;AT ADDRESS ERCKO .
     ADTKI ;CALL ADTKI 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 171616 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 *
    TS ;ADDRESS OF NEXT ROUTINE *
    10. ;TEST ITERATION COUNT *
    IA ;SCOPE ENTRY POINT *
*****

```

```

*****
:WALL FWD TEST. 2 BLOCKS. BINARY COUNT.
:TEST SEQUENCE: 1. BINARY FILL WBUFO 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 WBUFO.
     JSR PC,BINFL1 ;BINFIL WBUF1.
     MOV BLKNUM,BLKRG
     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 RBUFO
     RBUFO
     RALLF ;READ ALL FWD 528. WORDS INTO
     528. ;ADDR RACKSO 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 RACKSO 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 WBUFO 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 WBUFO.
     JSR PC,BINFL1 ;BINFIL WBUF1.
     MOV BLKNUM,BLKRG
     JSR PC,SETWBR ;SET UP WRITE BUFFER REVERSE.
     JSR R5,REVERS ;REVERSC 528 WORDS OF WRITE DATA.
     AWBUFO
     EAWBFO
     528.
     INC BLKRG

```

```

1908 007414 104046
1909 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 00476 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 AWBUFO. EXTENDED DATA BITS ARE
AWBUFO ;TAKEN FROM CONSECUTIVE BYTES STARTING
EAWBFO ;AT ADDRESS EAWBFO.
RDATAR ;CALL RDATAR SUB TO READ REV 512. WORDS
512. ;STARTING AT ADDR RBUFO
RBUFO
RALLR ;READ ALL REV 528. WORDS INTO
528. ;ADDR ARBUFO AND UP. EXTENDED DATA BITS ARE
ARBUFO ;STORED IN CONSECUTIVE BYTES STARTING
EARBFO ;AT ADDRESS EARBFO.
ADTCK ;CALL ADTCK SUB TO CHECK 18 BIT DATA STARTING
AWBUFO ;AT ADDR AWBUFO AND EAWBFO. AGAINST 18 BIT DATA
ARBUFO ;STARTING AT ADDR ARBUFO 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 WBUFO.
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. OBVERSE WRITE DATA TO MATCH RALL REV DATA.
7. RALL REV 258 WORDS EVERY OTHER BLOCK. VERIFY DATA.
8. REOBVERSE WRITE DATA TO MATCH RALL FWD DATA.
9. RALL FWD 260 WORDS EVERY OTHER BLOCK. VERIFY DATA.
KA: JSR PC,BINFLO ;BINFIL WBUFO.
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 RWCKSO. EXTENDED DATA BITS ARE
RWCKSO ;TAKEN FROM CONSECUTIVE BYTES STARTING
EWRCKO ;AT ADDRESS EWRCKO.
DEC RO ;DONE ALL BLOCKS?
BNE KB ;BR IF NOT DONE.
MOV #299.,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 RBUFO
RBUFO
DATCKI ;CALL DATCKI TO CHECK DATA STORED AT
WBUFO ;WBUFO AGAINST DATA STORED AT RBUFO+510.
RBUFO+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      #299.,RO    ;SET UP TO READ DATA FWD EVERY OTHER BLOCK.
1867 007604 012767 177776 171246   MOV      #-2,BLKRQ
1868 007612 062767 000002 171240   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      KC          ;BR IF NOT DONE.
1878 007642 004567 176066   JSR      RS,OBVERS ;OBVERSE 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 171172   MOV      #579.,BLKRQ
1884 007666 162767 000002 171164   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          RWCKSO        ;AT ADDR RWCKSO AND EWRCKO , AGAINST 18 BIT DATA
1891 007710 014346          FRCKSO        ;STARTING AT ADDR FRCKSO AND ERFCKO . 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 ;REOBVERSE 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 171110   MOV      #-2,BLKRQ
1903 007750 062767 000002 171102   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          ;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
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 R5,LSUBA ;RUN TEST.
BITS
JSR R5,LSUBA ;RUN TEST.
BIT4
JSR R5,LSUBA ;RUN TEST.
BIT3
JSR R5,LSUBA ;RUN TEST.
BIT2
JSR R5,LSUBA ;RUN TEST.
BIT1
JSR R5,LSUBA ;RUN TEST.
BIT0
SCOPE ;SCOPE.
LSUBA: MOV BLKNUM,BLKRG ;COMPUTE PARITY FOR WBUFO.
JSR R5,PARITY
WBUFO
EWBUFO
BIC (5)+,LPBT ;CLEAR SPECIFIED BIT FROM CALCULATED PARITY.
MOV LPBT,LPB
JSR R5,XPARTY
MOV LPB,FWCKSO ;MOVE BAD PARITY TO WBUFO.
ELPB,EWFCKO
WALLF ;WRITE ALL FWD 258. WORDS STARTING
258. ;FROM ADDR RWCKSO . EXTENDED DATA BITS ARE
RWCKSO ;TAKEN FROM CONSECUTIVE BYTES STARTING
EWCKO ;AT ADDRESS EWCKO .
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
RWCKSO ;AT ADDR RWCKSO AND EWCKO , AGAINST 18 BIT DATA
RRCKSO ;STARTING AT ADDR RRCKSO AND ERRCKO . FIFTH ARGUMENT
EWCKO ;REPRESENTS # OF 18 BIT ELEMENTS TO CHECK.
ERRCKO ;REPORT ERRORS.
258.
RDATAFS ;CALL RDATAFS SUB TO READ DATA FWD 256. WORDS
256. ;AND STORE AT ADDR STARTING AT RBLFC .
RBUFO ;ALLOW PARITY ERROR.
BIT ;PARITY ERROR?
BNE #BIT14,ATCST ;BR IF PARITY ERROR SET.
LSUBBA

```


1976	010214	104012			OACNV			;NO. ERROR. CONVERT INCORRECT PARITY TO ASCII.
1977	010216	001100			L'PBT			
1978	010220	011204			ABPAR			
1979	010222	000002			2			
1980	010224	104012			OACNV			;CONVERT GOOD PARITY TO ASCII.
1981	010226	001076			L'PBG			
1982	010230	011174			AGPAR			
1983	010232	000002			2			
1984	010234	104011			ERRORN			;FAILURE TO DETECT PARITY ERROR.
1985	010236	011153			NPARE			
1986	010240	177777			-1			
1987	010242	000205			LSUBBA:	RTS	R5	;EXIT.
1988								
1989	010244	022445	020124		EMO:	.ASCII	'%%T'	
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	'%%TCH - T011 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	044124				
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	RESETSR:	.ASCIZ	'%SET SR OPTIONS. NORMAL SR=0'	
2029	010546	051123	047440	052120				
2030	010554	047511	051516	020056				
2031	010562	047516	046522	046101				

2032	010570	051440	036522	000060			
2033	010576	000			APGEN:	.BYTE	007
2034	010577	045	000052			.ASCIZ	'%#'
2035	010602	052040	053503	020103	CTCWC:	.ASCII	'TCWC'
2036	010610	020040	020040	020040	ATCWC:	.ASCIZ	'
2037	010616	000					
2038	010617	040	041524	040502	CTCBA:	.ASCII	'TCBA'
2039	010624	040					
2040	010625	040	020040	020040	ATCBA:	.ASCIZ	'
2041	010632	000040					
2042	010634	052040	041503	020115	STAT:	.ASCII	'TCCM'
2043	010642	020040	020040	020040	ATCCM:	.ASCII	'TCST'
2044	010650	052040	051503	020124			
2045	010656	020040	020040	020040	ATCST:	.ASCIZ	'
2046	010664	000					
2047	010665	040	047516	042040	INTFAI:	.ASCIZ	'NC DT INTRPT'
2048	010672	020124	047111	051124			
2049	010700	052120	000040				
2050							
2051	010704	042040	020124	051105	DTERR:	.ASCIZ	'DT ERR'
2052	010712	020122	000				
2053	010715	040	046102	051113	BLKSB:	.ASCII	'BLKRQ'
2054	010722	020121					
2055	010724	020040	020040	020040	ABLKRO:	.ASCIZ	'
2056	010732	000					
2057	010733	124	053503	020103	WCNOTO:	.ASCIZ	'TCWC NOT O'
2058	010740	047516	020124	020060			
2059	010746	000					
2060	010747	124	041103	020101	INOTCB:	.ASCIZ	'TCBA WRONG'
2061	010754	051127	047117	020107			
2062	010762	000					
2063	010763	040	041524	040502	TCBASB:	.ASCII	'TCBA'
2064	010770	040					
2065	010771	040	020040	020040	ATCBAS:	.ASCIZ	'
2066	010776	020040	000040				
2067	011002	042040	052101	020101	DATERR:	.ASCII	'DATA ERR WORD'
2068	011010	051105	020122	053440			
2069	011016	051117	020104				
2070	011027	020040	020040	020056	AWDCNT:	.ASCII	'S/B'
2071	011030	051440	041057	040			
2072	011035	040	020040	020040	ADATSB:	.ASCII	'WAS'
2073	011042	020040	053440	051501			
2074	011050	040					
2075	011051	040	020040	020040	ADATWS:	.ASCIZ	'
2076	011056	000040					
2077	011060	047045	020117	047125	NOUNIT:	.ASCIZ	'%NO UNITS AVAILABLE.'
2078	011066	052111	020123	053101			
2079	011074	044501	040514	046102			
2080	011102	027105	000				
2081	011105	122	054504	042457	STCMSG:	.ASCIZ	'RDY/ERR NOT O AFTER DO'
2082	011112	051122	047040	052117			
2083	011120	030040	040440	052106			
2084	011126	051105	042040	000117			
2085	011134	041040	045514	047040	SRCHER:	.ASCIZ	'BLK NOT FOUND'
2086	011142	052117	043040	052517			
2087	011150	042116	000				

2098	011153	045	047516	050040	NPARE:	.ASCII	'%NO PAR ERR GOOD'
2099	011160	051101	042440	051122			
2090	011166	043440	047517	020104			
2091	011174	020040	020040	040502	AGPAR:	.ASCII	'BAD'
2092	011202	020104					
2093	011204	020040	000		ABPAR:	.ASCIZ	' '
2094	011207	040	043040	041520	FPCMSG:	.ASCII	'FPC'
2095	011214	040					
2096	011215	040	020040	020040	AFPC:	.ASCIZ	'%'
2097	011222	020040	000045				
2098	011226	053445	046111	020114	GOOD:	.ASCIZ	'%WILL TEST UNITS:'
2099	011234	042524	052123	052440			
2100	011242	044516	051524	020072			
2101	011250	000					
2102	011251	040	000054		GTAPES:	.ASCIZ	'0','1','2','3','4','5','6','7'
2103	011254	060	061	062	GTAB:	.BYTE	
2104	011257	063	064	065			
2105	011262	066	067				
2106						.EVEN	
2107							
2108	011264	000000			AWBUFO:	OPEN	;WRITE BUFFER 0
2109	011266	000000			WFBLKO:	OPEN	
2110	011270	000000				OPEN	
2111	011272	000000				OPEN	
2112	011274	000000			RWCKSO:	OPEN	
2113	011276	000000			WBUFO:	OPEN	
2114		012276				. = +510.	
2115	012276	000000			FWCKSO:	OPEN	
2116	012300	000000				OPEN	
2117	012302	000000				OPEN	
2118	012304	000000			WRBLKO:	OPEN	
2119	012306	000000				OPEN	
2120	012310	000000			AWBUF1:	OPEN	;WRITE BUFFER 1
2121	012312	000000			WFBLK1:	OPEN	
2122	012314	000000				OPEN	
2123	012316	000000				OPEN	
2124	012320	000000			RWCKS1:	OPEN	
2125	012322	000000			WBUF1:	OPEN	
2126		013322				. = +510.	
2127	013322	000000			FWCKS1:	OPEN	
2128	013324	000000				OPEN	
2129	013326	000000				OPEN	
2130	013330	000000			WRBLK1:	OPEN	
2131	013332	000000				OPEN	
2132	013334	000000			ARBUFO:	OPEN	;READ BUFFER 0
2133	013336	000000			RFBLKO:	OPEN	
2134	013340	000000				OPEN	
2135	013342	000000				OPEN	
2136	013344	000000			RRCKSO:	OPEN	
2137	013346	000000			RBUFO:	OPEN	
2138		014346				. = +510.	
2139	014346	000000			FRCKSO:	OPEN	
2140	014350	000000				OPEN	
2141	014352	000000				OPEN	
2142	014354	000000			RRBLKO:	OPEN	
2143	014356	000000				OPEN	

```

2144 014360 000000 ARBUF1: OPEN ;READ BUFFER 1
2145 014362 000000 RFBLK1: OPEN
2146 014364 000000 OPEN
2147 014366 000000 OPEN
2148 014370 000000 RRCKS1: OPEN
2149 014372 000000 RBUF1: OPEN
2150 015372 015372 .=. +510.
2151 015372 000000 FRCKS1: OPEN
2152 015374 000000 OPEN
2153 015376 000000 OPEN
2154 015400 000000 RRBLK1: OPEN
2155 015402 000000 OPEN
2156 015404 000 EAWBFO: .BYTE OPEN ;EXTENDED WRITE BUFFER 0
2157 015405 000 000 000 EWFCKO: .BYTE OPEN,OPEN,OPEN
2158 015410 000 EWRCKO: .BYTE OPEN
2159 015411 000 EWBUFO: .BYTE OPEN
2160 016011 016011 .=. +255.
2161 016011 000 000 000 EWFCKO: .BYTE OPEN,OPEN,OPEN
2162 016014 000 000 000 EWRBKO: .BYTE OPEN,OPEN
2163 016016 000 000 000 EAWBF1: .BYTE OPEN ;EXTENDED WRITE BUFFER 1
2164 016017 000 000 000 EWFBK1: .BYTE OPEN,OPEN,OPEN
2165 016022 000 000 000 EWRCK1: .BYTE OPEN
2166 016023 000 000 000 EWBUF1: .BYTE OPEN
2167 016423 016423 .=. +255.
2168 016423 000 000 000 EWFCK1: .BYTE OPEN,OPEN,OPEN
2169 016426 000 000 000 EWRBK1: .BYTE OPEN,OPEN
2170 016430 000 000 000 EARBFO: .BYTE OPEN,OPEN,OPEN,OPEN ;EXTENDED READ BUFFER 0
2171 016433 000 000 000 ERRCKO: .BYTE OPEN
2172 016434 000 000 000 ERBUFO: .BYTE OPEN
2173 016435 000 000 000 .=. +255.
2174 017035 017035 .=. +255.
2175 017035 000 000 000 ERFCKO: .BYTE OPEN,OPEN,OPEN,OPEN,OPEN
2176 017040 000 000 000 EARBF1: .BYTE OPEN,OPEN,OPEN,OPEN ;EXTENDED READ BUFFER 1
2177 017042 000 000 000 ERRCK1: .BYTE OPEN
2178 017045 000 000 000 ERBUF1: .BYTE OPEN
2179 017046 000 000 000 .=. +255.
2180 017047 000 000 000 ERFCK1: .BYTE OPEN,OPEN,OPEN,OPEN,OPEN
2181 017447 017447 .=. +255.
2182 017447 000 000 000 :
2183 017452 000 000 000 .END
2184
2185 000001

```


RTNNO	001036	505*	575*	655	678	695*	701	812						
RVERSA	006500	1574*	1585											
RVRSA	006554	1574*	1576	1588*										
RVRSB	006556	1575*	1577	1591*										
RWCKSO	011274	1681	1699	1723	1728	1734	1746	1770	1780	1850	1890	1909	1959	1966
		2112*												
RWCKS1	012320	2124*												
RWFBKA	006070	1463	1466*											
RWFBKE	006170	1479	1492*											
RWFBK1	006054	1462*	1623	1649										
RWFBK2	006062	1465*												
RWFIND	006172	1462*	1465*	1478	1493*									
RO	=%:000000	422*	583*	606*	625*	678*	686*	704	705*	706*	707*	708*	709*	727
		736*	747*	780	781*	782*	783	785*	789*	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*	909*
		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*	1449*	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	1288	1329	1600
SAV5S	= 000003	485*	804	901	981	995	1018	1141	1226	1238				
SCOPE	= 104000	530*	988	1152	1166	1186	1193	1233	1251	1258	1300	1306	1341	1347
		1637	1663	1704	1751	1785	1825	1916	1947					
SCOPTR	001034	504*	671	698*										
SELDRA	006630	1603	1607*											
SELDRR	006602	571	1600*											
SELDRV	= 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	844		
SRCCON	004426	1183*	1199	1203	1205	1209	1213	1216	1224					

ADD	910	913	944	945	952	989	1200	1244	1572	1573	1868	1903			
ASL	707	1241	1397	1571											
ASR	1404	1405	1406	1407	1408	1409									
BCCS	941														
BEG	579	602	614	637	656	669	677	687	692	864	978	980	1006	1050	1061
	1069	1102	1105	1107	1112	1145	1154	1196	1199	1216	1219	1309	1313	1350	1479
	1524	1603													
BGT	1197														
BHI	1585														
BIC	585	912	915	992	1064	1065	1179	1220	1311	1379	1383	1386	1392	1414	1415
	1416	1594	1952												
BICB	616														
BIS	975	1176	1222	1417											
BIT	601	613	651	668	672	676	680	844	977	979	1007	1177	1198	1204	1210
	1218	1312	1974												
BITB	1602														
BLE	1208														
BLOS	1202														
BMI	604	997	1004												
BNE	607	609	631	652	658	666	673	675	681	683	845	850	858	892	894
	898	920	930	949	955	1008	1067	1116	1125	1135	1178	1205	1211	1213	1224
	1389	1400	1452	1456	1853	1865	1877	1896	1915	1975					
BFL	593	595	795	869	1191	1255	1304	1345							
BR	588	591	600	612	617	624	632	662	693	714	717	866	875	943	1016
	1038	1041	1043	1110	1172	1182	1194	1203	1209	1221	1265	1268	1270	1273	1277
	1285	1326	1463	1497	1597										
CLR	575	647	939	973	1001	1056	1057	1123	1256	1289	1290	1330	1331	1365	1366
	1435	1462	1496	1843											
CLRB	1039	1044	1045	1171	1436										
CMP	657	682	849	1066	1068	1109	1114	1153	1195	1201	1207	1584			
CMPB	655	863	1108	1113											
COM	657	782	896	897	1391	1413	1433	1434							
DEC	606	630	674	881	883	919	929	948	954	1101	1115	1124	1135	1303	1349
	1388	1399	1451	1455	1569	1852	1864	1875	1895	1914					
DECB	1223														
EMT	530	531	532	533	534	535	536	537	538	539	540	541	542	543	544
	545	546	547	548	549	550	551	552	553	554	555	556	557	558	559
	560	561	562	563	564	565	566	567	568	569	570	571	572		
HALT	383	385	387	393	399	679	790	796							
INC	584	629	665	899	942	1103	1183	1454	1481	1513	1514	1593	1807	1847	
JMP	489	491	654	671											
JSR	650	689	860	865	872	874	1378	1380	1382	1384	1387	1466	1480	1499	1504
	1515	1519	1525	1623	1625	1626	1649	1651	1652	1676	1678	1690	1719	1721	1722
	1733	1764	1765	1767	1799	1800	1802	1803	1842	1844	1878	1897	1934	1935	1937
	1939	1941	1943	1945	1949	1954									
MOV	574	581	583	596	597	615	620	621	625	641	642	643	644	664	678
	686	694	695	696	697	698	699	700	701	704	705	706	708	711	712
	713	716	719	720	721	722	723	724	725	726	727	730	731	736	737
	738	739	740	741	744	747	748	749	750	751	752	753	754	756	757
	758	759	761	762	765	767	768	769	770	774	775	776	780	781	783
	785	789	799	805	843	848	856	879	880	887	888	889	895	900	902
	907	908	909	911	925	926	927	935	936	937	938	950	951	966	967
	968	969	970	974	976	1047	1048	1051	1052	1053	1054	1055	1058	1059	1121
	1122	1130	1131	1134	1142	1174	1187	1237	1239	1240	1243	1246	1264	1266	1267
	1269	1271	1272	1274	1275	1276	1263	1284	1286	1287	1291	1292	1293	1295	1302
	1324	1325	1327	1328	1332	1333	1334	1336	1343	1367	1368	1369	1370	1371	1391

	1385	1393	1394	1396	1411	1412	1427	1428	1429	1430	1431	1465	1467	1498	1502
	1507	1509	1510	1511	1512	1566	1567	1568	1570	1574	1575	1576	1577	1601	1604
	1624	1650	1677	1720	1766	1801	1845	1846	1854	1855	1866	1867	1882	1883	1901
	1902	1948	1953	1955											
MOV8	582	626	636	857	859	867	871	873	914	928	946	953	1037	1040	1042
	1062	1063	1173	1175	1301	1342	1352	1372	1432	1503	1508	1578	1579	1580	1591
	1605	1956													
NEG	1242														
NOP	1259	1307	1348												
RESET	688	794													
ROL	1074	1075	1077	1079	1398	1443	1445	1446	1448	1449					
ROLB	1444	1447	1450												
ROR	916	917	918	1076	1079	1373	1374	1375	1376	1437	1438	1439	1440	1441	1442
RTI	713	719	729	733	742	755	764	771	778	786	792	797	853	862	885
	890	904	922	932	957	971	990	993	1002	1026	1118	1127	1138	1168	1188
	1214	1217	1263	1310	1317	1351	1356	1598	1608						
RTS	702	709	870	1401	1410	1418	1458	1492	1529	1538	1548	1558	1587	1987	
SUB	806	940	1206	1856	1884										
SWAB	1377	1395													
TRAP	487														
TST	578	592	594	603	732	763	794	947	996	998	999	1003	1005	1144	1190
	1254	1303	1344	1453	1478	1523	1582	1606							
TSTB	608	691	868	1049	1060	1104	1106	1111	1212	1215	1583				
.ABS	379														
.ASCII	1989	1990	1992	1995	1997	2007	2012	2016	20	2035	2038	2042	2043	2053	2063
	2067	2070	2072	2088	2091	2094									
.ASCIZ	1999	2000	2003	2025	2028	2034	2036	2040	2045	2047	2051	2055	2057	2060	2065
	2075	2077	2081	2085	2093	2096	2099	2102							
.BYTE	963	1035	1036	1234	1235	1583	1590	1609	2033	2103	2156	2157	2158	2159	2161
	2162	2163	2164	2165	2166	2168	2169	2170	2172	2173	2175	2177	2179	2180	2182
.END	2185														
.EVEN	2106														
.LIST	1	377	399	530	531	532	533	534	535	536	537	538	539	540	541
	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556
	557	558	559	560	561	562	563	564	565	566	567	568	569	570	571
	572	1617	1643	1669	1710	1757	1791	1831	1922						
.MACR	381														
.MLIST	1	378	399	530	531	532	533	534	535	536	537	538	539	540	541
	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556
	557	558	559	560	561	562	563	564	565	566	567	568	569	570	571
	572	1617	1643	1669	1710	1757	1791	1831	1922						
.PAGE	1318	1617	1664	1705	1752										
.REM	1														
.REPT	399														
.TITLE	1														
.WORD	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543
	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558
	559	560	561	562	563	564	565	566	567	568	569	570	571	572	

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

F05

TC4 - TC11 TEST 4 MACY11 27(732) 08-SEP-76 09:04 PAGE 60
DZTCDC.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

RUN-TIME: 8 15 4 SECONDS
RUN-TIME RATIO: 78/28=2.7
CORE USED: 10K (20 PAGES)

