

# TC11

TC3 BASIC FUNCTIONS  
MD-11-DZTCC-A

EP DZTCC A DL A

OCT 1976

COPYRIGHT ©1976



FICHE 1 OF 1

Made in U.S.A.

11



B01

TC3 - TC11 TEST 3  
DZTCCA.P11

MACY11 27(732) 10-SEP-76 15:51 PAGE 1

.REM !

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZTCC-A-D

PRODUCT NAME: TC3 - TC11 TEST 3

DATE: MAY 1, 1972

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: L. R. KOLLER

THIS MAINDEC OBSOLETES MAINDEC-11-D3CC

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

TC3 - TC11 TEST 3 IS PART 3 OF A FIVE PROGRAM PACKAGE  
USED TO TEST THE TC11 DECTAPE CONTROL.



1. ABSTRACT

TC3 - TC11 TEST 3 IS PART 3 OF A FIVE PROGRAM PACKAGE USED TO TEST THE TC11 DECTAPE CONTROL. TC3 TESTS AND EXERCISES THE TC11 CONTROL AND FROM ONE TO EIGHT SELECTED TRANSPORTS. TC3 CONCENTRATES ON TESTING FOR CORRECT OPERATION OF THE UP TO SPEED BIT (UPS), ABILITY TO READ BLOCK NUMBERS AND DETECT END ZONES, EXERCISES RNUM COMMAND (WRITE DATA) AND RDATA (READ DATA) COMMANDS BOTH FORWARD AND REVERSE, AND WITH SINGLE, DOUBLE, AND 4 BLOCK TRANSFERS.

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 TU56 DUAL TRANSPORT.

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. WRTM AND WALL SWITCHES MUST BE OFF.
- 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 43 MINUTES.
- B. ONE SINGLE ITERATION PASS (SR11=1) TAKES ABOUT 15 MINUTES.

\*\*\*\*\*NOTE\*\*\*\*\*

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. PERFORM STEP F OF PREVIOUS PROCEDURE.
- C. PRESS START.
- D. GO TO STEP H OF PREVIOUS PROCEDURE.



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

LOC 002444 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 001774 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  
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.

AFTER THE PRINTOUT IS COMPLETED, THE PROGRAM WILL HALT AT COMMON ERROR HALT AT LOC 002460 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. SUBROUTINE NAME. APPEARS WHEN ERROR CALL IS MADE FROM A SUBROUTINE.
- B. FPC XXXXXX. (FROM PC). ADDRESS THE SUBROUTINE WAS CALLED FROM.
- C. 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.
- D. IN A DATA ERROR PRINTOUT THE "WORD #" THAT FAILED REPRESENTS THE POSITION OF THE WORD IN THE READ BUFFER, AND IT IS NOT MEANT TO DESCRIBE THE WORD'S POSITION IN A DECTAPE BLOCK.



7. MISCELLANEOUS  
-----7.1 SR OPTIONS  
-----

THE STANDARD SR OPTIONS ARE DESCRIBED HERE.

- SR15 HALT ON ERROR. WITH SR15 SET TO A 1, THE PROGRAM WILL HALT AFTER AN ERROR OCCURS. PRESSING CONT WILL CAUSE PROGRAM TO RESUME OPERATION.
- SR14 SCOPE. THIS OPTION CAUSES THE PROGRAM TO REMAIN IN THE CURRENT TEST ROUTINE. WHEN THE OPTION IS REMOVED, THE PROGRAM WILL COMPLETE THE CURRENT ROUTINE, AND WILL THEN GO ON TO THE NEXT ROUTINE.
- SR13 INHIBIT ERROR PRINTOUT. THIS OPTION IF SET, WILL REMOVE ALL ERROR PRINTOUTS.

## \*\*\*\*\*NOTE\*\*\*\*\*

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

- SR11 INHIBIT ITERATION. SETTING THIS OPTION WILL CAUSE THE PROGRAM TO EXECUTE EACH TEST ONLY ONCE, INSTEAD OF THE NORMAL NUMBER OF ITERATIONS SELECTED FOR EACH TEST. TWO POSSIBLE USES OF THIS OPTION ARE:
- A. QUICK PASS. EACH TEST IS RUN ONLY ONCE.
  - B. TO SKIP OVER A FAILING ROUTINE.
- SR10 HALT AT END OF CURRENT ROUTINE. WITH THE OPTION SET, THE PROGRAM WILL HALT AT THE END OF EACH TEST, AND DISPLAY IN DATA LIGHTS THE NUMBER OF THE TEST JUST COMPLETED. THREE POSSIBLE USES OF THIS OPTION ARE:
- A. TO STEP THROUGH THE PROGRAM ONE ROUTINE AT A TIME.
  - B. WHEN THE PROGRAM HAS BEEN RUNNING FOR A WHILE, TO FIND OUT HOW FAR IT HAS PROGRESSED.
  - C. IN CASE OF A BLOW UP, ETC., TO STEP THROUGH ONE TEST AT A TIME UNTIL THE FAILURE REOCCURS. THE ROUTINE FOLLOWING THE PREVIOUSLY COMPLETED ROUTINE WOULD BE THE FAILING ROUTINE.
- SR9 SELECT ROUTINE. WITH SR9 SET, THE PROGRAM WILL GO AND EXECUTE THE ROUTINE INDICATED BY SR7 THROUGH SR9, AFTER THE CURRENT ROUTINE HAS BEEN COMPLETED. IF THE OPTION IS REMOVED, THE PROGRAM WILL PROCEED TO EXECUTE THE ROUTINES FOLLOWING THE SELECTED ROUTINE.



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

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

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

8. DESCRIPTION

THIS PROGRAM IS ORGANIZED INTO THREE MAIN SECTIONS:

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

8.1 CONTROL ROUTINE

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

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

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



8.2 TEST ROUTINES

THE ACTUAL TESTING IS PERFORMED BY A SET OF TEST ROUTINES THAT ARE NUMBERED SEQUENTIALLY FROM 0 TO 36 (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)

```

*****
T20:   20           ;ROUTINE NUMBER 20.          *
      T21         ;ADDRESS OF NEXT ROUTINE      *
      100.        ;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 "ERR " SUBROUTINE, WHICH IS CALLED BY AN "ERRORN" STATEMENT AND TYPES THE TEST NUMBER AND PC VALUE WHEN A FAILURE OCCURS.



```

379
380
381
382
383      000000
384      000000 000002
385      000002 000000
386      000004 000006
387      000006 000000
388      000010 000012
389      000012 000000
390      000014 002320
391      000016 000340
392      000020 002350
393      000022 000340
394      000024 000026
395      000026 000000
396      000030 002120
397      000032 000340
398      000034 000036
399      000036 000000
400
401
402
403      177570
404      177776
405      001000
406      000240
407      000000
408      100000
409      100000
410      040000
411      020000
412      010000
413      004000
414      002000
415      001000
416      000400
417      000200
418      000100
419      000040
420      000020
421      000010
422      000004
423      000002
424      000001
425      000000
426      000001
427      000002
428      000003
429      000004
430      000005
431      000006
432      000007

```

```

      .LIST SEQ,LD,BIN,ME
      .NLIST MC,MD
      .ABS
;
      .=0
      .+2 ;UNASSIGNED TRAP
MACHER: HALT
      .+2 ;SP OVERFLOW, BUS ERROR TRAP
      HALT
      .+2 ;RESERVED INSTRUCTION TRAP
TRCV: SVSS ;TRACE TRAP
      PRTY7
IOTV: RS5S ;TRAP TO CALL IOX
      PRTY7
      .+2 ;POWER FAIL TRAP
EMTV: HALT
      EMTINT ;EMT TRAP
      PRTY7
TRPV: .+2 ;TRAP TRAP. SIMILAR TO EMT
      HALT
;LOC 40 THROUGH 376 FILLED WITH .+2 AND HALT.
;EQUATE .LIST
      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

```



433	000007	PC=%7
434	005746	PUSH=005746
435	024646	PUSH2=024646
436	005726	POPSP=005726
437	022626	POPSP2=022626
438	000340	PRTY7=340
439	000300	PRTY6=300
440	000240	PRTY5=240
441	000200	PRTY4=200
442	000140	PRTY3=140
443	000100	PRTY2=100
444	000040	PRTY1=40
445	000000	PRTY0=0
446	000007	BELL=007
447	177777	TLAST=-1
448	000003	TRC=3
449	000040	I=40
450	177777	X=-1
451	100000	A=BIT15
452	040000	B=BIT14
453	020000	C=BIT13
454	000000	V0=0
455	000004	V1=4
456	000010	V2=10
457	000014	V3=14
458	000020	V4=20
459	000024	V5=24
460	000030	V6=30
461	000034	V7=34
462	020000	MAINT=BIT13
463	010000	DINH=BIT12
464	004000	REV=BIT11
465	000000	FWD=0
466	000000	U0=0
467	000400	U1=BIT8
468	001000	U2=BIT9
469	001400	U3=BIT9!BIT8
470	002000	U4=BIT10
471	002400	U5=BIT10!BIT8
472	003000	U6=BIT10!BIT9
473	003400	U7=BIT10!BIT9!BIT8
474	000100	IE=BIT6
475	000000	SAT=0
476	000002	RNUM=BIT1
477	000004	RDATA=BIT2
478	000006	RALL=BIT2!BIT1
479	000010	SST=BIT3
480	000012	WRM=BIT3!BIT1
481	000014	WDATA=BIT3!BIT2
482	000016	WALL=BIT3!BIT2!BIT1
483	000001	DO=BIT0
484	000200	UPS=BIT7
485	010000	ILO=BIT12
486	004000	SELE=BIT11
487	000000	EMTX=0
488	000003	SAVSS=3



```

489      000004
490      000200
491 000200 000167 001044
492      001000
493 001000 000167 000562
494 001004 177340
495 001006 177342
496 001010 177344
497 001012 177346
498 001014 177350
499 001016 000214
500 001020 000300
501 001022 177564
502 001024 177566
503 001026 000000
504 001030 000000
505 001032 007016
506 001034 000000
507 001036 000000
508 001040 000000
509 001042 000000
510 001044 000000
511 001046 000000
512 001050 000000
513 001052 000000
514 001054 000000
515 001056 000000
516 001060 000000
517 001062 000000
518 001064 000000
519 001066 000000
520 001070 000000
521 001072 000000
522 001074 000000
523 001076 000000
524 001100 000005
525 001102
526 001102 001704
527
528 001104 002414
529
530 001106 002140
531
532 001110 002240
533
534 001112 002170
535
536 001114 002270
537
538 001116 002160
539
540 001120 002264
541
542 001122 003044
543
544 001124 002464

```

```

RST55=4
.=200
JMP START
.=1000
JMP GETRODY
TCST: 177340
TCCM: 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
CTRB: 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
ERRLIM: 5
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
.LIST
.WORD SRSETT ;POINTER FOR EMT CALL SRESET
.LIST
.WORD SV03 ;POINTER FOR EMT CALL SAV03
.LIST
.WORD RS03 ;POINTER FOR EMT CALL RST03
.LIST
.WORD SV05 ;POINTER FOR EMT CALL SAV05
.LIST
.WORD RS05 ;POINTER FOR EMT CALL RST05
.LIST
.WORD SV05S ;POINTER FOR EMT CALL SAV05S
.LIST
.WORD RS05S ;POINTER FOR EMT CALL RST05S
.LIST
.WORD TYP ;POINTER FOR EMT CALL TYPE
.LIST
.WORD ERR ;POINTER FOR EMT CALL ERROR

```



545		.LIST		
546	001126 002474	.WORD	ERRN	; POINTER FOR EMT CALL ERRORN
547		.LIST		
548	001130 002436	.WORD	CHLT	; POINTER FOR EMT CALL CHALT
549		.LIST		
550	001132 002452	.WORD	EHLT	; POINTER FOR EMT CALL EHALT
551		.LIST		
552	001134 002374	.WORD	STTCV	; POINTER FOR EMT CALL SVECTR
553		.LIST		
554	001136 003142	.WORD	DLY	; POINTER FOR EMT CALL DELAY
555		.LIST		
556	001140 003532	.WORD	STCOM	; POINTER FOR EMT CALL SETCOM
557		.LIST		
558	001142 003514	.WORD	STATS	; POINTER FOR EMT CALL STATUS
559		.LIST		
560	001144 003626	.WORD	STPDT	; POINTER FOR EMT CALL STOPDT
561		.LIST		
562	001146 003636	.WORD	CKER	; POINTER FOR EMT CALL CKERR
563		.LIST		
564	001150 003654	.WORD	CKERZ	; POINTER FOR EMT CALL CKERRZ
565		.LIST		
566	001152 003720	.WORD	SDTRG	; POINTER FOR EMT CALL SAVDTR
567		.LIST		
568	001154 003772	.WORD	RWIND	; POINTER FOR EMT CALL REWIND
569		.LIST		
570	001156 003762	.WORD	WINDD	; POINTER FOR EMT CALL WIND
571		.LIST		
572	001160 004046	.WORD	NOINTR	; POINTER FOR EMT CALL NOINT
573		.LIST		
574	001162 004070	.WORD	RNDBK	; POINTER FOR EMT CALL RNDBLK
575		.LIST		
576	001164 004144	.WORD	RDBK1	; POINTER FOR EMT CALL RNDBK1
577		.LIST		
578	001166 004636	.WORD	SRCHFF	; POINTER FOR EMT CALL SRCHF
579		.LIST		
580	001170 004644	.WORD	SRCHRR	; POINTER FOR EMT CALL SRCHR
581		.LIST		
582	001172 004574	.WORD	WNGBK	; POINTER FOR EMT CALL WRNGBK
583		.LIST		
584	001174 002722	.WORD	RNGEN	; POINTER FOR EMT CALL RNDNUM
585		.LIST		
586	001176 004620	.WORD	CFPC	; POINTER FOR EMT CALL CNVFPC
587		.LIST		
588	001200 005314	.WORD	WDAF	; POINTER FOR EMT CALL WDATAF
589		.LIST		
590	001202 005342	.WORD	WDATR	; POINTER FOR EMT CALL WDATAR
591		.LIST		
592	001204 005324	.WORD	RDAF	; POINTER FOR EMT CALL RDATAF
593		.LIST		
594	001206 005352	.WORD	RDATR	; POINTER FOR EMT CALL RDATAR
595		.LIST		
596	001210 004472	.WORD	CWCBA	; POINTER FOR EMT CALL CKWCBA
597		.LIST		
598	001212 004402	.WORD	CLEAR	; POINTER FOR EMT CALL CLEAR
599		.LIST		
600	001214 004422	.WORD	FILL	; POINTER FOR EMT CALL FILL



```

601
602 001216 004444
603
604 001220 004250
605
606 001222 004240
607
608 001224 003200
609
610 001226 003232
611
612 001230 003744
613
614 001232 005370
615
616 001234 003000
617
618 001236 003016
619
620 001240 005400
621
622 001242 005422
623
624 001244 003276
625
626 001246 003404
627
628 001250 012706 001000
629 001254 005067 177556
630 001260 104010
631 001262 005551
632 001264 005737 000042
633 001270 001524
634
635 001272 012767 000402 001370
636 001300 112767 000376 177562
637 001306 012700 000010
638 001312 005267 000010
639 001316 042767 177770 000002
640 001324 104060
641 001326 000000
642 001330 000431
643 001332 104017
644 001334 004002
645 001336 000437
646 001340 005777 177442
647 001344 100375
648 001346 005777 177432
649 001352 100031
650 001354 012777 013350 177430
651 001362 012777 177777 177420
652 001370 104017
653 001372 000015
654 001374 000420
655 001376 032777 100200 177402
656 001404 001774

```

```

.LIST
.WORD BINFLL ; POINTER FOR EMT CALL BINFIL
.LIST
.WORD DATCK ; POINTER FOR EMT CALL DATCHK
.LIST
.WORD DTCKI ; POINTER FOR EMT CALL DATCKI
.LIST
.WORD INBINN ; POINTER FOR EMT CALL INBIN
.LIST
.WORD GTBIN1 ; POINTER FOR EMT CALL GETBN1
.LIST
.WORD PRENDZ ; POINTER FOR EMT CALL PRMEND
.LIST
.WORD RDTFX ; POINTER FOR EMT CALL RDATAFX
.LIST
.WORD INRNDN ; POINTER FOR EMT CALL INRND
.LIST
.WORD RNDFLL ; POINTER FOR EMT CALL RNDFIL
.LIST
.WORD SQDRV ; POINTER FOR EMT CALL SEQDRV
.LIST
.WORD SELDRR ; POINTER FOR EMT CALL SELDRV
.LIST
.WORD OACNVV ; POINTER FOR EMT CALL OACNV
.LIST
.WORD BDCNVV ; POINTER FOR EMT CALL BDCNV
.LIST
START: MOV #SPBOT,R6 ; SET BOTTOM OF SP STACK.
CLR RTNNO
TYPE ; TYPE TITLE.
PGTIT
TST @#42 ; PROGRAM LOADED VIA MONITOR?
BEQ STRTA ; BR IF NOT.
; ROUTINE TO DETERMINE TRANSPORTS AVAILABLE FOR TEST.
MOV #402,ERRND
MOVB #376,UNITS ; ASSUME DRIVES 1-7 AVAILABLE.
MOV #8,R0 ; SET UP TO TEST 8 TIMES.
DTRMN: INC SQDRV1
BIC #177770,SQDRV1
SELDRV ; SELECT A TRANSPORT.
SQDRV1: OPEN ; TRANSPORT #.
BR DTRMNA ; UNIT NOT AVAILABLE RETURN.
SETCOM ; REWIND TO REVERSE END ZONE.
RNUM+REV
BR DTRMNB ; ERROR RETURN.
TST @TCM ; WAIT.
BPL -4
TST @TCST ; END ZONE?
BPL DTRMNB ; BR IF NOT.
MOV #WBUF,@TCBA ; SET CURRENT ADDR.
MOV #-1,@TCWC ; SET WORD COUNT.
SETCOM ; YES. ISSUE WRITE DATA COMMAND.
WDATA+FWD+DO
BR DTRMNB ; ERROR RETURN.
BIT #BIT15+BIT7,@TCCM ; WAIT FOR ERROR/READY.
BEQ -6

```



657	001406	005777	177374		TST	BTCCM		:ERROR?
658	001412	100411			BMI	DTRMNB		:BR IF YES.
659	001414	104021		DTRMNA:	STOPDT			:STOP DECTAPE.
660	001416	005300			DEC	RD		:DONE 8 TIMES?
661	001420	001334			BNE	DTRMN		:BR IF NOT.
662	001422	105767	177442		TSTB	UNITS		:ANY UNITS AVAILABLE?
663	001426	001015			BNE	DTRMNC		:BR IF YES.
664	001430	104010			TYPE			:TYPE NON AVAILABLE MESSAGE.
665	001432	006733			NOUNIT			
666	001434	000572			BR	CHNC		:GO EXIT.
667	001436	032777	014000	177340	DTRMNB:	#BIT12+BIT11, BTCCST		:ILO OR SELE ERROR?
668	001444	001763			BEQ	DTRMNA		:BR IF NOT.
669	001446	016701	177654		MOV	SQDRV1, RI		
670	001452	146167	005454	177410	BICB	UNTAB(1), UNITS		:DESELECT NON AVAILABLE TRANSPORT.
671	001460	000755			BR	DTRMNA		
672	001462	104010		DTRMNC:	TYPE			:TYPE UNITS TO BE TESTED.
673	001464	006760			GOOD			
674	001466	012767	000001	000010	MOV	#1, CPENA		
675	001474	012767	000007	177344	MOV	#7, CTRB		:CHECK UP TO 7 UNITS.
676	001502	104060			SELDIV			:SELECT DRIVE.
677	001504	000000		CPENA:	OPEN			:UNIT TO BE SELECTED.
678	001506	000407			BR	CPENB		:UNIT NOT AVAILABLE.
679	001510	016700	177352		MOV	UNITN, RD		:SUCCESS.
680	001514	116067	007006	005261	MOV	GTAB(0), GTAPES		:GET ASCII # FOR GOOD TAPE.
681	001522	104010			TYPE			:TYPE # OF UNIT TO TEST.
682	001524	007003			GTAPES			
683	001526	005267	177752		CPENB:	INC	CPENA	:UPDATE TO NEXT DRIVE.
684	001532	005367	177310		DEC	CTRB		:CHECKED ALL DRIVES?
685	001536	001361			BNE	CPENA-2		:BR IF NOT.
686	001540	000412			BR	GETRDY		:YES.
687	001542	104010		STRTA:	TYPE			:TYPE UNIT SELECT INSTRUCTIONS.
688	001544	005577			INST1			
689	001546	104013			CHALT			:WAIT FOR USER.
690	001550	116767	176014	177312	MOV	SR, UNITS		:GET UNITS TO TEST.
691	001556	001771			BEQ	STRTA		:BR IF NO UNITS SELECTED.
692	001560	104010			TYPE			:TYPE SR OPTION MESSAGE.
693	001562	005762			ASETSR			
694	001564	104013			CHALT			:COMMON HALT.
695	001566	012767	001002	001074	GETRDY:	MOV	#1002, ERRND	:ALLOW ERROR PRINTOUTS.
696	001574	016767	177232	177236	MOV	KSTART, NXTST		:ADDR OF 1ST ROUTINE TO NXTST
697	001602	012767	000006	176174	GTRDYX:	MOV	#6, MACHER	:RESET MACHER TRAP.
698	001610	012767	000340	176160	MOV	#PTY7, PSW		:SET PRIORITY 7.
699	001616	012706	001000		MOV	#SPBOT, R6		:SET BOTTOM OF STACK.
700	001622	104001			SRESET			:ISSUE RESET.
701	001624	104051			INBIN			:INITIALIZE BINARY COUNT.
702	001626	104015			SVECTR			:PRESET DT INTERRUPT VECTOR TO 0.
703	001630	000000			0			
704	001632	004767	000214		GTRDYA:	JSR	R7, FORWD	:ROLL FORWARD TO "NEXT" ROUTINE.
705	001636	032767	001000	175724	GTRDYB:	BIT	#BIT9, SR	:CHECK SELECT ROUTINE SWITCH
706	001644	001003			BNE	GTRDYC		:BRANCH IF SELECT ROUTINE SWITCH IS SET.
707	001646	104057			GORUN:	SEQDRV		:SELECT SEQUENTIAL DRIVE.
708	001650	000177	177166		JMP	JCURTST		:GO RUN CURRENT ROUTINE.
709	001654	126767	177156	175706	GTRDYC:	CMPB	RTNNO, SR	:COMPARE RTNNO TO SR.
710	001662	001771			BEQ	GORUN		:BRANCH IF ROUTINE FOUND.
711	001664	022767	177777	177146	GTRDYD:	CMP	#-1, NXTST	:NO. CHECK FOR LAST ROUTINE.
712	001672	001357			BNE	GTRDYA		:BRANCH IF NOT LAST ROUTINE.



Line	Address	Code	Label	Instruction	Comments	Remarks	Assembly	Description
713	001674	104010		TYPE				;TYPE INCORRECT RTN SELECTED.
714	001676	005531		AINCRT				
715	001700	104013		CHALT				;COMMON HALT.
716	001702	000731		BR	GETRDY			;START OVER.
717	001704	104021	CHAINN:	STOPDT				
718	001706	012706	001000	MOV	#SPBOT,R6			;RESET STACK.
719	001712	005267	177112	INC	ICNT			;INCREMENT ITERATION COUNT.
720	001716	001002		BNE	CHNAC			;BR IF RESULT NOT 0.
721	001720	005167	177104	COM	ICNT			;RESULT 0. RESET ICNT TO -1.
722	001724	032767	040000	CHNAC:	BIT #BIT14,SR	175636		;CHECK FOR SCOPE OPTION.
723	001732	001403		BEQ	CHNA			;BRANCH IF SCOPE SW NOT SET.
724	001734	104057		CHNAB:	SEQDRV			;SELECT SEQUENTIAL DRIVE.
725	001736	000177	177072	JMP	2SCOPT			;RETURN TO ROUTINE.
726	001742	032767	004000	CHNA:	BIT #BIT11,SR	175620		;TEST INHIBIT ITERATION SWITCH
727	001750	001003		BNE	CHNAA			;BRANCH IF INHIBIT ITERATION SW SET.
728	001752	005367	177050	DEC	ICTR			;DECREMENT ITERATION COUNT.
729	001756	001366		BNE	CHNAB			;BRANCH IF COUNT NOT 0.
730	001760	032767	002000	CHNAA:	BIT #BIT10,SR	175602		;ROUTINE END HALT SW SET? (SR10)
731	001766	001403		BEQ	CHNB			;BRANCH IF NOT SET.
732	001770	016700	177042	MOV	RTNNO,RO			;TEST # TO RO.
733	001774	000000		HALT				;ROUTINE END HALT. TEST # IN LIGHTS.
734	001776	032767	001000	CHNB:	BIT #BIT9,SR	175564		;CHECK SELECT ROUTINE SWITCH
735	002004	001270		BNE	GETRDY			;BRANCH IF SELECT RTN SW SET
736	002006	022767	177777	177024	CMP #1,NXTST			;LAST TEST?
737	002014	001272		BNE	GTRDYX			;BRANCH IF NOT LAST TEST.
738	002016	104010		TYPE				;TYPE PROGRAM END BELL.
739	002020	006017		APGEND				
740	002022	013700	000042	CHNC:	MOV 2#42,RO			;GET CONTENTS OF 42.
741	002026	001410		BEQ	HERE			;BR IF ZERO.
742	002030	000005		RESET				;NON-ZERO. ISSUE RESET.
743	002032	004710		LOGIC:	JSR PC,(0)			;RETURN TO MONITOR.
744	002034	000240	000240	WORD	NOP,NOP,NOP			
745	002042	105767	177022	TSTB	UNITS			;ANY UNITS AVAILABLE FOR TESTING?
746	002046	001765		BEQ	CHNC			;BR IF NOT.
747	002050	000646		HERE:	BR			;GO REPEAT PROGRAM.
748	002052	016705	176762	FORWD:	MOV NXTST,R5			;ADDR OF NEXT ROUTINE TO R5.
749	002056	012567	176754	MOV	(5)+,RTNNO			;GET NEXT ROUTINE NUMBER.
750	002062	012567	176752	MOV	(5)+,NXTST			;GET ADDR OF NEXT "NEXT" ROUTINE.
751	002066	012567	176734	MOV	(5)+,ICTR			;GET ITERATION COUNT.
752	002072	012567	176736	MOV	(5)+,SCOPT			;GET SCOPE LOOP ENTRY POINTER.
753	002076	010567	176740	FORWDA:	MOV R5,CURTST			;ADDR OF NOW CURRENT TEST TO CURTST.
754	002102	012767	000001	176720	MOV #1,ICNT			;PRESET ICNT TO 1.
755	002110	016767	176722	175452	MOV RTNNO,SR			;DISPLAY ROUTINE #.
756	002116	000207		RTS	R7			;EXIT FORWD SUBROUTINE.
757				;EMT INTERPRETER ROUTINE.				
758	002120	010046		EMTINT:	MOV RO,-(6)			;PUSH RO.
759	002122	016600	000002	MOV	2(6),RO			;GET EMT PC.
760	002126	014000		MOV	-(0),RO			;GET EMT CALL
761	002130	006300		ASL	RO			;TIMES 2.
762	002132	016000	171102	MOV	EMTTAB-10000(0),RO			;FORM EMT ADDR.
763	002136	000200		RTS	RO			;GO TO EMT RTN. RESTORE RO.
764				;SAVE REGS 0 TO 3 SUBROUTINE.				
765	002140	012666	177766	SVO3:	MOV (6)+,-10.(6)			;MOVE PC UPSTACK.
766	002144	012666	177766	MOV	(6)+,-10.(6)			;MOVE STATUS UPSTACK.
767	002150	012767	000002	000046	MOV #RTI,SVOSC			
768	002156	000415		BR	SVO5B			



```

769      :SUB TO SAVE REGS 0 TO 5 AND PLACE EMT PC IN R5.
770 002160 012767 000240 000036 SV05S: MOV      #NOP,SV05C
771 002166 000403                BR        SV05A
772      :SUB TO SAVE REGS 0 TO 5.
773 002170 012767 000002 000026 SV05:  MOV      #RTI,SV05C
774 002176 012666 177762                SV05A: MOV      (6)+,-14.(6) ;MOVE PC AND PSW UPSTACK.
775 002202 012666 177762                MOV      (6)+,-14.(6)
776 002206 010546                MOV      R5,-(6)
777 002210 010446                MOV      R4,-(6)
778 002212 010346                SV05B: MOV      R3,-(6)
779 002214 010246                MOV      R2,-(6)
780 002216 010146                MOV      R1,-(6)
781 002220 010046                MOV      R0,-(6)
782 002222 024646                PUSH2
783 002224 000002                SV05C: RTI
784 002226 016605 000020                MOV      16.(6),R5 ;RTI OR NOP.
785 002232 010504                MOV      R5,R4 ;EMT PC TO R5.
786 002234 005744                TST      -(4)
787 002236 000002                RTI ;EXIT.
788      :RESTORE REGS 0 TO 3 SUBROUTINE.
789 002240 022626                RS03: POPSP2
790 002242 012600                MOV      (6)+,R0 ;RESTORE REGS 0 TO 4.
791 002244 012601                MOV      (6)+,R1
792 002246 012602                MOV      (6)+,R2
793 002250 012603                MOV      (6)+,R3
794 002252 016646 177766                MOV      -10.(6),-(6) ;MOVE PC AND PSW DOWN STACK.
795 002256 016646 177766                MOV      -10.(6),-(6)
796 002262 000002                RTI ;EXIT.
797      :SUB TO SET R5 IN EMT PC AND RESTORE REGS 0 TO 5.
798 002264 010566 000020                RS05S: MOV      R5,16.(6) ;SET EMT PC TO R5 CONTENTS.
799      :SUB TO RESTORE REGS 0 TO 5.
800 002270 022626                RS05: POPSP2
801 002272 012600                MOV      (6)+,R0
802 002274 012601                MOV      (6)+,R1
803 002276 012602                MOV      (6)+,R2
804 002300 012603                MOV      (6)+,R3
805 002302 012604                MOV      (6)+,R4
806 002304 012605                MOV      (6)+,R5
807 002306 016646 177762                MOV      -14.(6),-(6) ;MOVE PC AND PSW DOWNSTACK.
808 002312 016646 177762                MOV      -14.(6),-(6)
809 002316 000002                RTI ;EXIT.
810 002320 012666 177772                SV5S: MOV      (6)+,-6(6) ;PC AND PSW UPSTACK.
811 002324 012666 177772                MOV      (6)+,-6(6)
812 002330 010546                MOV      R5,-(6) ;SAVE R5.
813 002332 010446                MOV      R4,-(6) ;SAVE R4.
814 002334 024646                PUSH2
815 002336 016605 000010                MOV      8.(6),R5 ;EMT PC TO R5.
816 002342 010504                MOV      R5,R4 ;EMT PC TO R4.
817 002344 005744                TST      -(4)
818 002346 000002                RTI ;EXIT EMT SUB.
819 002350 010566 000010                RSS5: MOV      R5,8.(6) ;R5 TO EMT PC.
820 002354 022626                POPSP2
821 002356 012604                MOV      (6)+,R4 ;RESTORE R4.
822 002360 012605                MOV      (6)+,R5 ;RESTORE R5.
823 002362 016646 177772                MOV      -6(6),-(6)
824 002366 016646 177772                MOV      -6(6),-(6)

```



825	002372	000002			RTI			:EXIT.
826					:ROUTINE TO SET TC11 INTERRUPT VECTOR AND PRIORITY			
827	002374	104006			STTCV: SAVOSS			
828	002376	016701	176414		MOV TCVTR,R1			:VECTOR TO R1.
829	002402	012521			MOV (5)+,(1)+			:SET DESIRED VECTOR.
830	002404	016721	176410		MOV TCLVL,(1)+			:SET TC11 PRIORITY.
831	002410	104007			RSTOSS			
832	002412	000002			RTI			
833					:ROUTINE TO ISSUE RESET.			
834	002414	010046			SRSETT: MOV R0,-(6)			:PUSH R0.
835	002416	012700	052525		MOV #52525,R0			:DATA TO R0.
836	002422	005100			COM R0			:COMPLEMENT (R0).
837	002424	010067	177770		MOV R0,SRSETT+4			:(R0) TO SRSETT+4.
838	002430	000005			RESET			:ISSUE RESET. (R0)
839	002432	012600			MOV (6)+,R0			:RESTORE R0.
840	002434	000002			RTI			:DISPLAYED. EXIT.
841					:COMMON HALT ROUTINE			
842	002436	104006			CHLT: SAVOSS			
843	002440	010500			MOV R5,R0			:DEVELOP ADDR OF CALLER.
844	002442	005740			TST -(0)			
845	002444	000000			HALT			:HALT CALL ADDR IN DATA LIGTHS.
846	002446	104007			RSTOSS			
847	002450	000002			RTI			:EXIT.
848					:CONDITIONAL ERROR HALT ROUTINE.			
849	002452	005767	175112		EHLT: TST SR			:CHECK FOR HALT ON ERROR.
850	002456	100001			BPL EHLTA			:BRANCH IF NO HALT DESIRED.
851	002460	000000			HALT			:HALT.
852	002462	000002			EHLTA: RTI			:IN DATA LIGHTS.
853	002464	012767	000406	000204	ERR: MOV #406,ERRNB			:SET UP FOR SINGLE ERR MSG.
854	002472	000403			BR ERRN+6			
855	002474	012767	000240	000174	ERRN: MOV #NOP,ERRNB			:SET UP FOR MULTIPLE ERR MSGS.
856	002502	104021			STOPDT			:ALL STOP.
857	002504	010467	176366		MOV R4,FPC			:CONVERT CALL ADDR OF SUB CALLING.
858	002510	104061			OACNV			:CONVERT IT TO ASCII.
859	002512	001076			FPC			
860	002514	006720			AFPC			
861	002516	000006			6			
862	002520	000003			SAVSS			:SAVE REG 55
863	002522	010567	000146		MOV R5,ERRB			:DETERMINE CALLING ADDR.
864	002526	162767	000002	000140	SUB #2,ERRB			
865	002534	104061			OACNV			:CONVERT CALLING ADDR TO ASCII.
866	002536	002674			ERRB			
867	002540	005477			APC			
868	002542	000006			6			
869	002544	104061			OACNV			:CONVERT TEST # TO ASCII.
870	002546	001036			RTNNO			
871	002550	005467			ATNUMB			
872	002552	000003			3			
873	002554	104062			BDCNV			:CONVERT ICNT TO DECIMAL ASCII.
874	002556	001030			ICNT			
875	002560	005513			AICNT			
876	002562	000005			5			
877	002564	104061			OACNV			:CONVERT UNIT # TO ASCII.
878	002566	001066			UNITN			
879	002570	005527			AUNIT			
880	002572	000001			1			



```

881 002574 104061 OACNV ;CONVERT BLKRQ TO ASCII.
882 002576 001062 BLKRQ
883 002600 006207 ABLKRQ
884 002602 000006 6
885 002604 104061 OACNV ;CONVERT TCST TO ASCII.
886 002606 001052 TCSTT
887 002610 006140 ATCST
888 002612 000006 6
889 002614 104061 OACNV ;CONVERT TCCM TO ASCII.
890 002616 001050 TCCMT
891 002620 006123 ATCCM
892 002622 000006 6
893 002624 104061 OACNV ;CONVERT TCDT TO ASCII.
894 002626 001054 TCDTT
895 002630 006065 ATCDT
896 002632 000006 6
897 002634 104061 OACNV ;CONVERT TCWC TO ASCII.
898 002636 001056 TCWCT
899 002640 006031 ATCWC
900 002642 000006 6
901 002644 104061 OACNV ;CONVERT TCBA TO ASCII.
902 002646 001060 TCBAT
903 002650 006047 ATCBA
904 002652 000006 6
905 002654 012767 005464 000012 MOV #EMD,ERRB ;TYPE ERR HEADER MSG IF NOT INHIBITED.
906 002662 032767 020000 174700 ERRNA: BIT #BIT13,SR ;INHIBIT ERR PRINT?
907 002670 001002 ERRND: BNE ERRNB ;BR TO INHIBIT.
908 002672 104010 TYPE ;TYPE MSG.
909 002674 000000 ERRB: OPEN ;DESIRED MSG ADDR GOES HERE.
910 002676 000240 ERRNB: NOP ;NOP FOR MULT MSG,OR 406 FOR SINGLE MSG.
911 002700 012567 177770 MOV (5)+,ERRB ;GET ADDR OF NEXT MSG.
912 002704 022767 177777 177762 CMP #-1,ERRB ;TERMINATOR?
913 002712 001363 BNE ERRNA ;GO TYPE IF NOT TERMINATOR.
914 002714 104014 ERRNC: EHALT ;END OF MSGS. HALT IF REQUIRED.
915 002716 000004 RSTSS ;RESTORE REG 55.
916 002720 000002 RTI ;EXIT EMT SUB.
917 ;RANDOM NUMBER GENERATOR. ROUTINE EXITS WITH NUMBER IN REGISTER 0.
918 RNGEN: SAVO55 ;SAVE REGS 0-55.
919 002722 104006 MOV RP1,RO
920 002724 016700 000044 ROL RO
921 002730 006100 ROL RO
922 002732 006100 ROL RO
923 002734 066700 000036 ADD RP2,RO
924 002740 010067 000030 MOV RO,RP1
925 002744 006100 ROL RO
926 002746 006100 ROL RO
927 002750 066700 000022 ADD RP2,RO
928 002754 006100 ROL RO
929 002756 006100 ROL RO
930 002760 010067 000012 MOV RO,RP2
931 002764 016725 000004 MOV RP1,(5)+ ;STORE # AT LOC FOLLOWING SUB CALL.
932 002770 104007 RSTO55 ;RESTORE REGS 0-55.
933 002772 000002 RTI ;EXIT EMT SUB.
934 002774 001233 RP1: 1233
935 002776 007622 RP2: 7622
936 003000 012767 001233 177766 ;EMT SUB TO INITIALIZE RANDOM NUMBER SUBROUTINE.
INRNDN: MOV #1233,RP1

```



```

937 003006 012767 007622 177762      MOV      #7622,RP2:
938 003014 000002                    RTI      ;EXIT.
939                                ;EMT SUB TO FILL AREA WITH RANDOM NUMBERS.
940 003016 104006      RNDFLL: SAVOSS
941 003020 012500      MOV      (5)+,RO      ;STARTING ADDR TO RO.
942 003022 012501      MOV      (5)+,R1      ;COUNT TO R1.
943 003024 104035      RNDFLA: RNDNUM      ;GET RANDOM NUMBER.
944 003026 000000      RNDFLB: OPEN      ;RANDOM NUMBER IS STORED HERE.
945 003030 016720 177772      MOV      RNDFLB,(0)+ ;STORE NUMBER PER RO.
946 003034 005301      DEC      R1          ;DONE?
947 003036 001372      BNE     RNDFLA      ;BR IF NOT.
948 003040 104007      RSTOSS
949 003042 000002      RTI      ;EXIT EMT SUB.
950                                ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.
951 003044 104006      TYP:   SAVOSS
952 003046 012500      MOV      (5)+,RO      ;ADDRESS OF MESSAGE TO RO.
953 003050 112001      TYPA:  MOVB     (0)+,R1  ;GET CHARACTER
954 003052 001006      BNE     TYPC          ;BRANCH IF NOT TERMINATOR..
955 003054 112701 000177      MOVB     #177,R1      ;OUTPUT RUBOUT.
956 003060 004767 000020      JSR     %7,TYPD
957 003064 104007      RSTOSS
958 003066 000002      RTI
959 003070 122701 000045      TYPC:  CMPB     #45,R1  ;TERMINATOR CHAR. DONE. EXIT.
960 003074 001411      BEQ     TYPF          ;CHECK FOR"%".
961 003076 004767 000002      JSR     R7,TYPD      ;BRANCH IF"%".
962 003102 000762      BR      TYPD          ;TYPE CHAR IN TYPDAT
963 003104 110177 175714      TYPD:  MOVB     R1,@TPB ;OUTPUT CHARACTER TO PRINTER
964 003110 105777 175706      TSTB     @TPS        ;WAIT FOR DONE FLAG.
965 003114 100375      BPL     -4
966 003116 000207      RTS     R7
967 003120 112701 000015      TYPF:  MOVB     #15,R1  ;EXIT
968 003124 004767 177754      JSR     R7,TYPD      ;MOVE CARRIAGE RETURN CODE TO TYPDAT
969 003130 112701 000012      TYPG:  MOVB     #12,R1  ;GO TYPE CHAR.
970 003134 004767 177744      JSR     R7,TYPD      ;MOVE LF CODE TO TYPDAT.
971 003140 000743      BR      TYPD          ;GO TYPE CHAR.
972                                ;SUBROUTINE TO DELAY
973                                DLYRO=DLY+4
974                                DLYR1=DLYA+4
975 003142 012727 000372 000000      DLY:   MOV      #250.,#0 ;DELAY COUNT TO DLYRO.
976 003150 005067 174622      CLR     PSW          ;SET PRIORITY 0.
977 003154 012727 001750 000000      DLYA:  MOV      #1000.,#0 ;1 MSEC COUNT TO DLYR1.
978 003162 005367 177772      DLYB:  DEC      DLYR1   ;DECREMENT 1 MSEC COUNT.
979 003166 001375      BNE     DLYB        ;BR IF NOT 0.
980 003170 005367 177752      DEC     DLYRO       ;DECREMENT DELAY COUNT.
981 003174 001367      BNE     DLYA        ;BR IF NOT DONE DELAYING.
982 003176 000002      RTI      ;EXIT.
983                                ;SUBROUTINE TO INITIALIZE BINARY COUNT PATTERNS
984 003200 012767 177777 000016      INBINN: MOV     #-1,RIND ;SET ALL VARIABLES
985 003206 016767 000012 000012      MOV     RIND,PTO
986 003214 016767 000004 000006      MOV     RIND,PT1
987 003222 000002      RTI
988 003224 000000      RIND:  OPEN
989 003226 000000      PTO:   OPEN
990 003230 000000      PT1:   OPEN
991                                ;SPECIAL BINARY COUNT PATTERN SUBROUTINE. EXITS WITH BIN CHAR IN RO
992 003232 016767 177770 177770      GTBIN1: MOV     PTO,PT1 ;PREVIOUS BIN CHAR TO PT1

```



993	003240	005167	177764	COM	PT1	
994	003244	005167	177754	COM	RIND	
995	003250	001002		BNE	+6	
996	003252	005267	177752	INC	PT1	
997	003256	016767	177746	MOV	PT1,PTO	;SAVE BIN CHAR IN PTO
998	003264	000003		SAVSS		
999	003266	016725	177736	MOV	PT1,(5)+	;
1000	003272	000004		RSTSS		
1001	003274	000002		RTI		;EXIT.
1002				:OCTAL TO ASCII CONVERT ROUTINE		
1003	003276	104006		OACNVV: SAVO55		;SAVE REGS.
1004		003302		OACNVX=.	+2	
1005	003300	013527	000000	MOV	2(5)+,#0	;GET OCTAL VALUE.
1006	003304	012501		MOV	(5)+,R1	;GET DESTINATION ADDR.
1007	003306	012502		MOV	(5)+,R2	;GET CONVERT COUNT.
1008	003310	060201		ADD	R2,R1	;DEVELOP ADDR TO STORE 1ST CHAR.
1009	003312	016703	177764	OACNVA: MOV	OACNVX,R3	
1010	003316	042703	177770	BIC	#177770,R3	;ISOLATE LEAST SIGNIFICANT DIGIT.
1011	003322	062703	000060	ADD	#60,R3	;CONVERT DIGIT TO ASCII.
1012	003326	110341		MOVB	R3,-(1)	;STORE ASCII CHARACTER.
1013	003330	042767	000007	BIC	#7,OACNVX	
1014	003336	006067	177740	ROR	OACNVX	
1015	003342	006067	177734	ROR	OACNVX	
1016	003346	006067	177730	ROR	OACNVX	
1017	003352	005302		DEC	R2	;DONE ALL DIGITS?
1018	003354	001356		BNE	OACNVA	;BRANCH IF NOT DONE.
1019	003356	104007		RSTO55		;RESTORE REGS.
1020	003360	000002		RTI		;DONE. EXIT.
1021				:SUBROUTINE TO MOVE A VARIABLE NUMBER OF BYTES.		
1022	003362	104002		BMOVE: SAVO3		;SAVE REGS.
1023	003364	012501		MOV	(5)+,R1	;GET "FROM" ADDRESS
1024	003366	012502		MOV	(5)+,R2	;GET "TO" ADDRESS
1025	003370	012503		MOV	(5)+,R3	;GET COUNT
1026	003372	112122		BMOVA: MOVB	(1)+,(2)+	;MOVE BYTE
1027	003374	005303		DEC	R3	;DECREMENT COUNT
1028	003376	001375		BNE	BMOVA	;BRANCH IF NOT DONE.
1029	003400	104003		RSTO3		;RESTORE REGS.
1030	003402	000205		RTS	R5	;DONE EXIT
1031				:BINARY TO DECIMAL ASCII CONVERT SUBROUTINE.		
1032	003404	104006		BDCNVV: SAVO55		;SAVE REGS.
1033	003406	013501		MOV	2(5)+,R1	;GET BINARY VALUE.
1034	003410	012700	003506	MOV	#DECVAL,R0	;ADDR OF DECVAL TO R0.
1035	003414	012702	003474	MOV	#TENPWR,R2	;ADDR OF 10 POWER TO R2.
1036	003420	012703	000005	MOV	#5,R3	;SET UP FOR 5 CONVERSIONS.
1037	003424	005004		BDCNVA: CLR	R4	;CLEAR RESULT.
1038	003426	161201		BDCNVB: SUB	(2),R1	;10 POWER FROM VALUE.
1039	003430	103402		BCS	BDCNVC	;BR IF UNSUCCESSFUL.
1040	003432	005204		INC	R4	;+1 TO RESULT.
1041	003434	000774		BR	BDCNVB	;DO IT AGAIN.
1042	003436	061201		BDCNVC: ADD	(2),R1	;RESTORE SUBTRACTED VALUE.
1043	003440	062704	000060	ADD	#60,R4	;CONVERT RESULT TO ASCII.
1044	003444	110420		MOVB	R4,(0)+	;STORE RESULT.
1045	003446	005722		TST	(2)+	;UPDATE 10 POWER ADDR.
1046	003450	005303		DEC	R3	;DONE 5 TIMES?
1047	003452	001364		BNE	BDCNVA	;BR IF NOT.
1048	003454	012501		MOV	(5)+,R1	;GET ADDR TO STORE ASCII.



1049	003456	012502				MOV	(5)+,R2		;GET # OF DIGITS REQUIRED.
1050	003460	060201				ADD	R2,R1		;START WITH LSD.
1051	003462	114041				BDCNVD:	MOV	-(0),-(1)	;TRANSFER CHARACTER.
1052	003464	005302				DEC	R2		;DONE?
1053	003466	001375				BNE	BDCNVD		;BR IF NOT.
1054	003470	104007				RSTOSS			;RESTORE REGS.
1055	003472	000002				RTI			;EXIT.
1056	003474	023420				TENPWR:	10000.		
1057	003476	001750					1000.		
1058	003500	000144					100.		
1059	003502	000012					10.		
1060	003504	000001					1		
1061	003506	040	040	040		DECVAL:	.BYTE	040,040,040,040,040,040	
1062	003511	040	040	040					
1063						:EMT SUB	TO SAVE TCCM, TCST		
1064	003514	017767	175264	175330		STATS:	MOV	@TCST,TCST	;SAVE TCST.
1065	003522	017767	175260	175320			MOV	@TCCM,TCCM	;SAVE TCCM.
1066	003530	000002					RTI		;EXIT EMT SUB.
1067						:EMT SUB	TO ISSUE DT COMMAND SPECIFIED AT CALL+2.		
1068	003532	000003				STCOM:	SAVSS		
1069	003534	016767	175324	175330			MOV	UNIT,COMND	;UNIT # TO COMND.
1070	003542	042767	174377	175322			BIC	#174377,COMND	
1071	003550	052567	175316				BIS	(5)+,COMND	;SET DESIRED COMMAND IN COMND.
1072	003554	016777	175312	175224			MOV	COMND,@TCCM	;ISSUE COMMAND.
1073	003562	032777	100200	175216			BIT	#BIT15:BIT7,@TCCM	;READY AND ERROR BIT CLEAR?
1074	003570	001414					BEQ	STCOMB	;BR IF YES.
1075	003572	032767	000001	175272			BIT	#BIT0,COMND	;WAS THE DO BIT SET IN COMND?
1076	003600	001410					BEQ	STCOMB	;BR IF NOT.
1077	003602	104020					STATUS		;SAVE STATUS.
1078	003604	104012					ERRORN		;ERROR. DO BIT FAILED TO CAUSE CLEARING
1079	003606	006713					FPCMSG		
1080	003610	006346					STCMSG		;OF READY AND/OR ERROR BIT(S).
1081	003612	006115					STAT		
1082	003614	177777					-1		
1083	003616	000004				STCOMA:	RSTSS		
1084	003620	000002					RTI		;EXIT EMT SUB.
1085	003622	005725				STCOMB:	TST	(5)+	;SET UP OK EXIT.
1086	003624	000774					BR	STCOMA	
1087						:EMT SUB	TO STOP ALL DECTAPES.		
1088	003626	042777	000116	175152		STPDT:	BIC	#116,@TCCM	;ISSUE SAT COMMAND.
1089	003634	000002					RTI		;EXIT EMT SUB.
1090						:EMT SUB	TO CHECK FOR ERROR. (ENDZ IS CONSIDERED AN ERROR).		
1091	003636	000003				CKER:	SAVSS		
1092	003640	104020					STATUS		;SAVE STATUS.
1093	003642	005777	175140				TST	@TCCM	;ERROR BIT SET?
1094	003646	100416					BMI	CKERZD	;BR IF YES.
1095	003650	005725					TST	(5)+	;NO. SET UP OK EXIT.
1096	003652	000407					BR	CKERZB	
1097						:EMT SUB	TO CHECK FOR DECTAPE ERROR OR END ZONE.		
1098	003654	000003				CKERZ:	SAVSS		
1099	003656	104020					STATUS		;SAVE STATUS.
1100	003660	005777	175122				TST	@TCCM	;ERROR BIT SET?
1101	003664	100404					BMI	CKERZC	;BR IF YES.
1102	003666	005725					TST	(5)+	;NO. SET UP OK EXIT.
1103	003670	005725				CKERZA:	TST	(5)+	
1104	003672	000004				CKERZB:	RSTSS		



```

1105 003674 000002
1106 003676 005777 175102
1107 003702 100772
1108 003704 104012
1109 003706 006713
1110 003710 006167
1111 003712 006115
1112 003714 177777
1113 003716 000765
1114
1115 003720 017767 175064 175130
1116 003726 017767 175062 175120
1117 003734 017767 175052 175116
1118 003742 000002
1119
1120 003744 000003
1121 003746 104012
1122 003750 006565
1123 003752 006200
1124 003754 006115
1125 003756 177777
1126 003760 104000
1127
1128 003762 012767 000002 000012
1129 003770 000403
1130
1131 003772 012767 004002 000002
1132 004000 104017
1133 004002 000000
1134 004004 104000
1135 004006 005777 174774
1136 004012 100375
1137 004014 104020
1138 004016 104021
1139 004020 005767 175026
1140 004024 100407
1141 004026 000003
1142 004030 104012
1143 004032 006713
1144 004034 006167
1145 004036 006115
1146 004040 177777
1147 004042 104000
1148 004044 000002
1149
1150 004046 000003
1151 004050 104020
1152 004052 104012
1153 004054 006713
1154 004056 006150
1155 004060 006115
1156 004062 177777
1157 004064 000004
1158 004066 000002
1159
1160 004070 026727 174766 001101

```

```

RTI
:EXIT EMT SUB.
:ENDZ BIT SET?
:BR IF YES.
:DECTAPE ERROR.
:EMT SUB TO SAVE TCDT, TCBA, TCWC REGS.
SDTRG: MOV @TCWC,TCWCT ;SAVE TCWC.
MOV @TCDT,TCDTT ;SAVE TCDT.
MOV @TCBA,TCBAT ;SAVE TCBA.
RTI ;EXIT EMT SUB.
:EMT SUB TO REPORT PREMATURE ENDZ DETECTION.
PRENDZ: SAV55
ERRORN ;PREMATURE ENDZ DETECTED.
EMSG6
BLKSB
STAT
-1
SCOPE
:EMT SUB TO MOVE TAPE TO FORWARD END ZONE.
WINDD: MOV #RNUM!FWD,RWINDV ;SET UP WIND.
BR RWINDA
:EMT SUB TO MOVE TAPE TO REVERSE END ZONE.
RWIND: MOV #RNUM!REV,RWINDV ;SET UP REWIND.
RWINDA: SETCOM ;ISSUE WIND/REWIND.
RWINDV: OPEN
SCOPE ;SETCOM ERROR.
TST @TCCM ;ERROR BIT SET?
BPL .-4 ;BR IF NOT.
STATUS ;YES. GET STATUS AND STOP.
STOPDT
TST TCSTT ;WAS ERROR DUE TO ENDZ?
BMI RWINDC ;BR IF YES.
SAV55
ERRORN ;NO. DECTAPE ERROR.
FPCMSG
DTERR
STAT
-1
SCOPE
RWINDC: RTI ;EXIT.
:EMT SUB TO HANDLE FAILURE TO INTERRUPT.
NOINTR: SAV55
STATUS ;SAVE STATUS.
ERRORN ;DECTAPE FAIL TO INTERRUPT.
FPCMSG
INTFAI
STAT
-1
RSTSS
RTI
:EMT SUB TO GENERATE RANDOM BLOCK #, AND STORE AT BLKRQ.
RANDBK: CMP BLKRQ,#577. ;VALID # IN BLKRQ?

```



1161	004076	101005			BHI	RNDBKK		;BR IF NOT.
1162	004100	032767	004000	173462	BIT	#BIT11,SR		;RANDOM # DESIRED?
1163	004106	001401			BEQ	RNDBKK		;BR IF YES.
1164	004110	000002			RTI			;NO. RETAIN SAME BLOCK#.
1165	004112	104035			RNDBKK: RNDNUM			;GET RANDOM NUMBER.
1166	004114	000000			RNDBKA: OPEN			;RANDOM NUMBER GOES HERE.
1167	004116	042767	176000	177770	BIC	#176000,RNDBKA		
1168	004124	026727	177764	001101	CMP	RNDBKA,#1101		;NUMBER A VALID BLOCK NUMBER?
1169	004132	101356			BHI	RNDBK		;BR IF NOT.
1170	004134	016767	177754	174720	MOV	RNDBKA,BLKRG		;YES. NUMBER TO BLKRG.
1171	004142	000002			RTI			;EXIT EMT SUB.
1172					;EMT SUB TO GENERATE RANDOM BLOCK# AND STORE AT BLKRG.			
1173					;THE NUMBER MUST NOT BE >574, OR <3.			
1174	004144	026727	174712	001076	RDBK1: CMP	BLKRG,#574.		;SEE IF CURRENT BLK# IS VALID.
1175	004152	101011			BHI	RDBK1A		;IF # IS HIGH GENERATE ANOTHER.
1176	004154	026727	174702	000002	CMP	BLKRG,#2		;IS # <3?
1177	004162	101405			BLOS	RDBK1A		;IF # IS <3 GENERATE ANOTHER.
1178	004164	032767	004000	173376	BIT	#BIT11,SR		;VALID #. RANDOM NUMBER CHANGE?
1179	004172	001401			BEQ	RDBK1A		;BR IF RANDOM # IS DESIRED.
1180	004174	000002			RTI			;NO. KEEP SAME NUMBER.
1181	004176	104035			RDBK1A: RNDNUM			;GET RANDOM NUMBER.
1182	004200	000000			RDBK1B: OPEN			;RANDOM # IS STORED HERE.
1183	004202	042767	176000	177770	BIC	#176000,RDBK1B		
1184	004210	026727	177764	000002	CMP	RDBK1B,#2		;IS # <3?
1185	004216	101767			BLOS	RDBK1A		;IF # IS <3 GET ANOTHER.
1186	004220	026727	177754	001076	CMP	RDBK1B,#574.		;# LARGER THAN 574.?
1187	004226	101363			BHI	RDBK1A		;BR IF LARGER.
1188	004230	016767	177744	174624	MOV	RDBK1B,BLKRG		;NEW NUMBER TO BLKRG.
1189	004236	000002			RTI			;EXIT.
1190					;EMT SUB TO CHECK EXPECTED DATA AGAINST ACTUAL DATA AND REPORT ERRORS.			
1191	004240	012767	022041	000120	DTCKI: MOV	#022041,DTCKBA		;SET UP TO INCR EXP ADDR, DECR ACT ADDR.
1192	004246	000403			BR	DATCKK		
1193	004250	012767	022021	000110	DATCK: MOV	#022021,DTCKBA		;SET UP TO INCR BOTH DATA ADDRESSES.
1194	004256	104006			DATCKK: SAVOSS			
1195	004260	012500			MOV	(5)+,R0		;GET EXPECTED DATA ADDR.
1196	004262	012501			MOV	(5)+,R1		;GET ACTUAL DATA ADDR.
1197	004264	012502			MOV	(5)+,R2		;GET CHECK COUNT.
1198	004266	012767	000001	000104	MOV	#1,DATKNT		;SET CURRENT WORD # TO 1.
1199	004274	016703	174600		MOV	ERRLIM,R3		;ERROR LIMIT TO R3.
1200	004300	021011			DATCKA: CMP	(0),(1)		;COMPARE ACTUAL AND EXPECTED WORD.
1201	004302	001427			BEQ	DATCKB		;BR IF EQUAL (OK).
1202	004304	010067	000006		MOV	R0,EXPD		
1203	004310	010167	000012		MOV	R1,ACTD		
1204	004314	104061			OACNV			;CONVERT EXPECTED DATA TO ASCII.
1205	004316	000000			EXPD: OPEN			
1206	004320	006323			ADATSB			
1207	004322	000006			6			
1208	004324	104061			OACNV			;CONVERT ACTUAL DATA TO ASCII.
1209	004326	000000			ACTD: OPEN			
1210	004330	006337			ADATWS			
1211	004332	000006			6			
1212	004334	104062			BDCNV			;CONVERT WORD # TO DECIMAL ASCII.
1213	004336	004400			DATKNT			
1214	004340	006310			AWDCNT			
1215	004342	000004			4			
1216	004344	104012			ERRORN			;DATA ERROR.



1217	004346	006713		FPCMSG		
1218	004350	006200		BLKSB		
1219	004352	006270		DATERR		
1220	004354	177777		-1		
1221	004356	005303		DEC	R3	;NTH ERROR?
1222	004360	001405		BEQ	DATCKC	;BR IF YES.
1223	004362	005267	000012	DATCKB: INC	DATKNT	;INCREMENT WORD NUMBER.
1224	004366	000000		DTCKBA: OPEN		;UPDATE DATA ADDRESSES.
1225	004370	005302		DEC	R2	;DONE?
1226	004372	001342		BNE	DATCKA	;BR IF NOT DONE.
1227	004374	104007		DATCKC: RSTOSS		;DONE.
1228	004376	000002		RTI		;EXIT.
1229	004400	000000		DATKNT: OPEN		
1230				;EMT SUB TO CLEAR SPECIFIED AREA TO 0'S		
1231	004402	104006		CLEAR: SAVOSS		
1232	004404	012500		MOV	(5)+,R0	;GET STARTING ADDR.
1233	004406	012501		MOV	(5)+,R1	;GET COUNT.
1234	004410	005020		CLR	(0)+	;CLEAR WORD.
1235	004412	005301		DEC	R1	;DONE?
1236	004414	001375		BNE	.-4	;BR IF NOT DONE.
1237	004416	104007		RSTOSS		;DONE.
1238	004420	000002		RTI		;EXIT.
1239				;EMT SUB TO FILL AREA WITH SPECIFIED VALUE.		
1240	004422	104006		FILL: SAVOSS		
1241	004424	012500		MOV	(5)+,R0	;GET STARTING ADDR.
1242	004426	012502		MOV	(5)+,R2	;GET VALUE.
1243	004430	012501		MOV	(5)+,R1	;GET COUNT.
1244	004432	010220		MOV	R2,(0)+	;SET VALUE IN WORD.
1245	004434	005301		DEC	R1	;DONE?
1246	004436	001375		BNE	.-4	;BR IF NOT DONE.
1247	004440	104007		RSTOSS		;DONE.
1248	004442	000002		RTI		;EXIT.
1249				;EMT SUB TO FILL AREA WITH BINARY COUNT PATTERN.		
1250	004444	104006		BINFLL: SAVOSS		
1251	004446	012500		MOV	(5)+,R0	;GET STARTING ADDR.
1252	004450	012501		MOV	(5)+,R1	;GET COUNT.
1253	004452	104052		BINFLA: GETBNI		;GET BINARY WORD.
1254	004454	000000		BINFLB: OPEN		;BINARY WORD IS STORED HERE.
1255	004456	016720	177772	MOV	BINFLB,(0)+	;STORE WORD.
1256	004462	005301		DEC	R1	;DONE?
1257	004464	001372		BNE	BINFLA	;BR IF NOT DONE.
1258	004466	104007		RSTOSS		;DONE.
1259	004470	000002		RTI		;EXIT.
1260				;EMT SUB TO CHECK THAT WORD COUNT IS 0, AND THAT TCBA CONTENTS		
1261				;MATCH THE EXPECTED CONTENTS.		
1262	004472	000003		CWCBA: SAVSS		
1263	004474	012567	174374	MOV	(5)+,TEMP	;GET EXPECTED TCBA CONTENTS.
1264	004500	104024		SAVDTR		;SAVE TCWC AND TCBA.
1265	004502	005777	174302	TST	@TCWC	;WORD COUNT 0?
1266	004506	001407		BEQ	CWCBB	;BR IF 0 (OK).
1267	004510	104012		ERRORN		;WORD COUNT NOT 0. TYPE
1268	004512	006713		FPCMSG		;CONTENTS OF TCWC AND TCBA.
1269	004514	006217		WCNOTO		
1270	004516	006023		CTCWC		
1271	004520	006041		CTCBA		
1272	004522	177777		-1		











```

1385 005206 006367 000074          ASL      WRDFRG
1386 005212 005477 173572          NEG      @TCWC          ; IN 2'S COMPLEMENT FORM.
1387 005216 012577 173570          MOV      (5)+,@TCBA    ; SET ADDR IN TCBA.
1388 005222 067767 173564 000056          ADD      @TCBA,WRDFRG ; 2(WORD COUNT)+TCBA=FINAL TCBA CONTENTS.
1389 005230 000000          WRDFRA: OPEN          ; SRCHF OR SRCHR CALL GOES HERE.
1390 005232 104000          SCOPE
1391 005234 012777 005260 173554          MOV      @WRDFRC,@TCVTR ; ERROR IN SRCHF OR SRCHR.
1392 005242 104017          SETCOM          ; SET INTERRUPT VECTOR TO WRDFRC.
1393 005244 000000          WRDFRB: OPEN          ; ISSUE WDATA OR RDATA.
1394 005246 104000          SCOPE          ; COMMAND GOES HERE.
1395 005250 104016          DELAY          ; SETCOM ERROR.
1396 005252 104016          DELAY          ; TIMEOUT INTERRUPT.
1397 005254 104027          NOINT
1398 005256 104000          SCOPE          ; FAILURE TO INTERRUPT.
1399 005260 012716 005266          WRDFRC: MOV      @WRDFRD,(6) ; HERE WHEN INTERRUPT OCCURS.
1400 005264 000002          RTI
1401 005266 022626          WRDFRD: POPSP2      ; EXIT TO WRDFRD.
1402 005270 005777 173512          TST      @TCOM        ; RESTORE STACK.
1403 005274 100003          BPL      WRDFRF       ; ERROR BIT SET?
1404 005276 104023          CKERRZ        ; BR IF NOT.
1405 005300 104000          SCOPE          ; CHECK FOR ERRORS.
1406 005302 000240          NOP           ; ERROR RETURN.
1407 005304 104043          WRDFRF: CKWCBA      ; ENDZ RETURN.
1408 005306 000000          WRDFRG: OPEN      ; CHECK WORD COUNT AND CURRENT ADDR.
1409 005310 000004          WRDFRE: RST55     ; TCBA SHOULD EQUAL THIS.
1410 005312 000002          RTI
1411 005314 012767 000115 177722          WDATAF: MOV      @WDATA!FWD!IE!DO,WRDFRB ; EXIT.
1412 005322 000403          BR
1413 005324 012767 000105 177712          RDATAF: MOV      RDATAF+6,WRDFRB ; @WDATA!FWD!IE!DO,WRDFRB
1414 005332 012767 104032 177670          MOV      @SRCHF,WRDFRA ; @RDATA!FWD!IE!DO,WRDFRB
1415 005340 000714          BR          ; @SRCHF,WRDFRA
1416 005342 012767 004115 177674          WDATAF: MOV      @WDATA!REV!IE!DO,WRDFRB ; @WDATA!REV!IE!DO,WRDFRB
1417 005350 000403          BR          ; @WDATA!REV!IE!DO,WRDFRB
1418 005352 012767 004105 177664          RDATAF: MOV      RDATAF+6,WRDFRB ; @RDATA!REV!IE!DO,WRDFRB
1419 005360 012767 104033 177642          MOV      @SRCHR,WRDFRA ; @SRCHR,WRDFRA
1420 005366 000701          BR          ; @SRCHR,WRDFRA
1421 005370 012767 000165 177646          RDTFX:  MOV      @RDATA!FWD!IE!BITS!BIT4!DO,WRDFRB ; @RDATA!FWD!IE!BITS!BIT4!DO,WRDFRB
1422 005376 000755          BR          ; @RDATA!FWD!IE!BITS!BIT4!DO,WRDFRB
1423          ; EMT SUB TO SELECT SEQUENTIAL DECTAPE UNIT.
1424 005400 005267 000010          SQDRV:  INC      SQDRVA
1425 005404 042767 177770 000002          BIC      #177770,SQDRVA
1426 005412 104060          SELDRV          ; SELECT UNIT SET IN SQDRVA IF AVAILABLE.
1427 005414 000000          SQDRVA: OPEN          ; UNIT NUMBER GOES HERE.
1428 005416 000770          BR          SQDRV    ; UNIT NOT AVAILABLE RETURN.
1429 005420 000002          RTI            ; UNIT SELECTED. EXIT.
1430          ; EMT SUB TO SELECT SPECIFIED DECTAPE UNIT IF AVAILABLE.
1431 005422 104006          SELDRR: SAVOSS
1432 005424 012500          MOV      (5)+,RO      ; GET UNIT NUMBER.
1433 005426 136067 005454 173434          BITB     UNTAB(0),UNITS ; SEE IF UNIT AVAILABLE.
1434 005434 001405          BEQ      SELDRA      ; BR IF UNIT NOT AVAILABLE.
1435 005436 010067 173424          MOV      RO,UNITN    ; AVAILABLE. SELECT UNIT.
1436 005442 110067 173417          MOV      RO,UNIT+1
1437 005446 005725          TST      (5)+        ; SET UP SELECTED EXIT.
1438 005450 104007          SELDRA: RSTOSS
1439 005452 000002          RTI
1440 005454 001 002 004 UNTAB: .BYTE BIT0,BIT1,BIT2,BIT3,BIT4,BIT5,BIT6,BIT7

```



1441	005457	010	020	040			
1442	005462	100	200				
1443	005464	022445	124		EMO:	.ASCII	'%T'
1444	005467	040	020040	020040	ATNUMB:	.ASCII	' PC '
1445	005474	041520	040				
1446	005477	040	020040	020040	APC:	.ASCII	' ICNT '
1447	005504	020040	041511	052116			
1448	005512	040					
1449	005513	040	020040	020040	AICNT:	.ASCII	' . '
1450	005520	056					
1451	005521	040	047125	052111		.ASCII	' UNIT '
1452	005526	040					
1453	005527	040	000		AUNIT:	.ASCIZ	' '
1454	005531	045	044445	053116	AINCRT:	.ASCIZ	'%INVALID TEST.'
1455	005536	046101	042111	052040			
1456	005544	051505	027124	000			
1457	005551	045	052045	031503	PGTIT:	.ASCIZ	'%TC3 - TC11 TEST 3%'
1458	005556	026440	052040	030503			
1459	005564	020061	042524	052123			
1460	005572	031440	022445	000			
1461	005577	045	052123	047101	INST1:	.ASCII	'%STANDARD TAPES ON UNITS.'
1462	005604	040504	042122	052040			
1463	005612	050101	051505	047440			
1464	005620	020116	047125	052111			
1465	005626	027123					
1466	005630	051045	046505	052117		.ASCII	'%REMOTE, WRITE ENABLE.'
1467	005636	026105	053440	044522			
1468	005644	042524	042440	040516			
1469	005652	046102	027105				
1470	005656	053445	046101	051514		.ASCII	'%WALLSW: OFF, WRTMSW: OFF.'
1471	005664	035127	047440	043106			
1472	005672	020054	051127	046524			
1473	005700	053523	020072	043117			
1474	005706	027106					
1475	005710	051445	046105	041505		.ASCII	'%SELECT UNITS WITH SR7 - SRO. '
1476	005716	020124	047125	052111			
1477	005724	020123	044527	044124			
1478	005732	051440	033522	026440			
1479	005740	051440	030122	020056			
1480	005746	051120	051505	020123		.ASCIZ	'PRESS CONT%'
1481	005754	047503	052116	000045			
1482	005762	051445	052105	051440	ASETSR:	.ASCIZ	'%SET SR OPTIONS. NORMAL SR=0'
1483	005770	020122	050117	044524			
1484	005776	047117	027123	047040			
1485	006004	051117	040515	020114			
1486	006012	051123	030075	000			
1487	006017	007			APGEND:	.BYTE	007
1488	006020	025045	000			.ASCIZ	'%*'
1489	006023	040	041524	041527	CTCWC:	.ASCII	' TCWC '
1490	006030	040					
1491	006031	040	020040	020040	ATCWC:	.ASCIZ	' . '
1492	006036	020040	000				
1493	006041	040	041524	040502	CTCBA:	.ASCII	' TCBA '
1494	006046	040					
1495	006047	040	020040	020040	ATCBA:	.ASCIZ	' . '
1496	006054	020040	000				



1497	006057	040	041524	052104	CTCDT: .ASCII	' TCDT '
1498	006064	040				
1499	006065	040	020040	020040	ATCDT: .ASCIZ	' '
1500	006072	020040	000			
1501	006075	040	041524	052104	SBTCDT: .ASCII	' TCDSB '
1502	006102	041123	040			
1503	006105	040	020040	020040	ASBTCD: .ASCIZ	' '
1504	006112	020040	000			
1505	006115	040	041524	046503	STAT: .ASCII	' TCCM '
1506	006122	040				
1507	006123	040	020040	020040	ATCCM: .ASCII	' TCST '
1508	006130	020040	052040	051503		
1509	006136	020124				
1510	006140	020040	020040	020040	ATCST: .ASCIZ	' '
1511	006146	000040				
1512	006150	047045	020117	052104	INTFAI: .ASCIZ	'%NO DT INTRPT '
1513	006156	044440	052116	050122		
1514	006164	020124	000			
1515	006167	045	052104	042440	DTERR: .ASCIZ	'%DT ERR '
1516	006174	051122	000040			
1517	006200	041040	045514	050522	BLKSB: .ASCII	' BLKRQ '
1518	006206	040				
1519	006207	040	020040	020040	ABLKRO: .ASCIZ	' '
1520	006214	020040	000			
1521	006217	045	041524	041527	WCNOTO: .ASCIZ	'%TCWC NOT 0 '
1522	006224	047040	052117	030040		
1523	006232	000040				
1524	006234	052045	041103	020101	INCTCB: .ASCIZ	'%TCBA WRONG '
1525	006242	051127	047117	020107		
1526	006250	000				
1527	006251	040	041524	040502	TCBASB: .ASCII	' TCBA '
1528	006256	040				
1529	006257	040	020040	020040	ATCBAS: .ASCIZ	' '
1530	006264	020040	000040			
1531	006270	042045	052101	020101	DATERR: .ASCII	'%DATA ERR WORD '
1532	006276	051105	020122	053440		
1533	006304	051117	020104			
1534	006310	020040	020040	020056	AWDCNT: .ASCII	' S/B '
1535	006316	051440	041057	040		
1536	006323	040	020040	020040	ADATSB: .ASCII	' WAS '
1537	006330	020040	053440	051501		
1538	006336	040				
1539	006337	040	020040	020040	ADATWS: .ASCIZ	' '
1540	006344	000040				
1541	006346	051045	054504	042457	STCMMSG: .ASCIZ	'%RDY/ERR NOT 0 AFTER DO '
1542	006354	051122	047040	052117		
1543	006362	030040	040440	052106		
1544	006370	051105	042040	020117		
1545	006376	000				
1546	006377	045	050125	020123	EMSGO: .ASCIZ	'%UPS DIDNT SET '
1547	006404	044504	047104	020124		
1548	006412	042523	020124	000		
1549	006417	045	051120	046505	EMSGOA: .ASCIZ	'%PREMATURE UPS '
1550	006424	052101	051125	020105		
1551	006432	050125	020123	000		
1552	006437	045	050125	020123	EMSG1: .ASCIZ	'%UPS DIDNT CLEAR '



1553	006444	044504	047104	020124	
1554	006452	046103	040505	020122	
1555	006460	000			
1556	006461	045	047122	046525	EMSG2: .ASCIZ '%RNUM FAILURE '
1557	006466	043040	044501	052514	
1558	006474	042522	000040		
1559	006500	042445	042116	027132	EMSG3: .ASCIZ '%ENDZ. NO INTRPT '
1560	006506	047040	020117	047111	
1561	006514	051124	052120	000040	
1562	006522	042445	042116	020132	EMSG4: .ASCIZ '%ENDZ DIDNT SET '
1563	006530	044504	047104	020124	
1564	006536	042523	020124	000	
1565	006543	045	047105	055104	EMSG5: .ASCIZ '%ENDZ. UPS NOT 0 '
1566	006550	020056	050125	020123	
1567	006556	047516	020124	020060	
1568	006564	000			
1569	006565	045	051120	046505	EMSG6: .ASCIZ '%PREMATURE ENDZ. LAST BLK READ '
1570	006572	052101	051125	020105	
1571	006600	047105	055104	020056	
1572	006606	040514	052123	041040	
1573	006614	045514	051040	040505	
1574	006622	020104	000		
1575	006625	045	047111	020124	EMSG7: .ASCIZ '%INT AFTER BLK 110! NOT DUE TO ENDZ '
1576	006632	043101	042524	020122	
1577	006640	046102	020113	030461	
1578	006646	030460	047040	052117	
1579	006654	042040	042525	052040	
1580	006662	020117	047105	055104	
1581	006670	000040			
1582	006672	041045	045514	047040	SRCHER: .ASCIZ '%BLK NOT FOUND. '
1583	006700	052117	043040	052517	
1584	006706	042116	020056	000	
1585	006713	040	050106	020103	FPCMSG: .ASCII ' FPC '
1586	006720	020040	020040	020040	AFPC: .ASCIZ '%'
1587	006726	000045			
1588	006730	022445	000		CRLF: .ASCIZ '%%'
1589	006733	045	047516	052440	NOUNIT: .ASCIZ '%NO UNITS AVAILABLE.'
1590	006740	044516	051524	040440	
1591	006746	040526	046111	041101	
1592	006754	042514	000056		
1593	006760	053445	046111	020114	GOOD: .ASCIZ '%WILL TEST UNITS: '
1594	006766	042524	052123	052440	
1595	006774	044516	051524	020072	
1596	007002	000			
1597	007003	040	000054		GTAPES: .ASCIZ ' , '
1598	007006	060	061	062	GTAB: .BYTE '0','1','2','3','4','5','6','7
1599	007011	063	064	065	
1600	007014	066	067		
1601					.EVEN
1602					



```

1603
1604
1605 007016 000000
1606 007020 007150
1607 007022 000012
1608 007024 007026
1609
1610
1611
1612
1613 007026 012767 000002 000022
1614 007034 004767 000014
1615 007040 012767 004002 000010
1616 007046 004767 000002
1617 007052 104000
1618 007054 104017
1619 007056 000000
1620 007060 104000
1621 007062 104016
1622 007064 104023
1623 007066 104000
1624 007070 000411
1625 007072 032777 000200 171704
1626 007100 001005
1627 007102 104012
1628 007104 006377
1629 007106 006115
1630 007110 177777
1631 007112 104000
1632 007114 104017
1633 007116 000010
1634 007120 104000
1635 007122 104020
1636 007124 032777 000200 171652
1637 007132 001405
1638 007134 104012
1639 007136 006437
1640 007140 006115
1641 007142 177777
1642 007144 104000
1643 007146 000207
1644
1645 007150 000001
1646 007152 007314
1647 007154 000012
1648 007156 007160
1649
1650
1651
1652
1653 007160 012767 000103 000034
1654 007166 004767 000014
1655 007172 012767 004103 000022
1656 007200 004767 000002
1657 007204 104000
1658 007206 104015

```

```

*****
TO: 0 ;ROUTINE NUMBER 0 *
T1 ;ADDRESS OF NEXT ROUTINE *
10. ;TEST ITERATION COUNT *
AA ;SCOPE ENTRY POINT *
.LIST
*****
:TEST THAT ISSUING RNUM-FWD-NODO, AND RNUM-REV-NODO CAUSE UPS TO SET
:CHECK THAT ISSUING SST COMMAND CAUSES UPS TO CLEAR.
AA: MOV #RNUM!FWD,ABA ;SET UP FORWARD RNUM COMMAND.
JSR PC,AB
MOV #RNUM!REV,ABA ;SET UP REVERSE RNUM COMMAND.
JSR PC,AB
AB: SCOPE ;SCOPE.
SETCOM ;ISSUE COMMAND.
ABA: OPEN ;COMMAND GOES HERE.
SCOPE ;SETCOM ERROR.
ABAA: DELAY ;WAIT.
CKERRZ ;CHECK FOR ERRORS.
SCOPE ;ERROR RETURN.
BR ABC ;ENDZ RETURN.
BIT #BIT7,ATCST ;UPS SET?
BNE ABC ;BR IF SET.
ERRORN ;ERROR. UPS FAILED TO SET AFTER
EMSGO ;ISSUING RNUM-WD-NO- DO, OR AFTER
STAT ;RNUM!REV!NO-DO. CHECK TYPED STATUS
-1
ABC: SCOPE ;SCOPE.
SETCOM ;ISSUE SST
SST
SCOPE ;SETCOM ERROR.
STATUS ;SAVE TCCM AND TCST
BIT #BIT7,ATCST ;UPS CLEAR?
BEQ AC ;BR IF CLEAR.
ERRORN ;ERROR. UPS FAILED TO CLEAR.
EMSG1 ;AFTER ISSUING SST-DO
STAT
-1
AC: SCOPE ;SCOPE
RTS PC ;EXIT.
*****
T1: 1 ;ROUTINE NUMBER 1 *
T2 ;ADDRESS OF NEXT ROUTINE *
10. ;TEST ITERATION COUNT *
BA ;SCOPE ENTRY POINT *
.LIST
*****
:TEST THAT WHEN RNUM COMMAND INTERRUPTS VALID DATA HAS BEEN
:TRANSFERRED TO THE DATA REGISTER.
BA: MOV #RNUM!FWD!IE!DO,BBA ;SET UP FORWARD RNUM.
JSR PC,BB
MOV #RNUM!REV!IE!DO,BBA ;SET UP REV RNUM.
JSR PC,BB
BB: SCOPE
SVECTR ;SET INTERRUPT VECTOR TO

```



1659	007210	007234				BC			;BC.
1660	007212	012777	177777	171574		MOV	#-1,@TCDT		;SET TCDT TO ALL 1'S
1661	007220	104017				SETCOM			;ISSUE RNUM!FWD!IE!DO
1662	007222	000000			BBA:	OPEN			;RNUM!FWD!IE!DO, OR RNUM!REV!IE!DO
1663	007224	104000				SCOPE			;SETCOM ERROR.
1664	007226	104016				DELAY			;WAIT.
1665	007230	104027				NOINT			;RNUM!FWD!IE!DO OR RNUM!REV!IE!DO FAILED
1666									;TO INTERRUPT. IF IN PRINTED TCCM
1667									;THE READY BIT IS SET, ERROR DUE
1668									;TO INTERRUPT FAILURE.
1669	007232	104000				SCOPE			;SCOPE.
1670	007234	012716	007242		BC:	MOV	#BD,(6)		;HERE WHEN INTERRUPT OCCURS.
1671	007240	000002				RTI			;EXIT TO BD.
1672	007242	022626			BD:	POPSP2			;RESTORE STACK.
1673	007244	104023				CKERRZ			;ERROR?
1674	007246	104000				SCOPE			;HERE IF ERROR.
1675	007250	000417				BR	BF		;ENDZ RETURN.
1676	007252	022777	177777	171534		CMP	#-1,@TCDT		;IS TCDT STILL A -1?
1677	007260	001007				BNE	BE		;NO. OK
1678	007262	104024			BCA:	SAVDTR			;SAVE TCDT.
1679	007264	104012				ERRORN			;TCDT STILL -1. ERROR RNUM FAILED
1680	007266	006461				MSG2			;TO READ BLOCK # INTO TCDT, OR
1681	007270	006115				STAT			;DATA IN TCDT IS NOT BETWEEN
1682	007272	006057				TCDT			;0 AND 1101. (CHECK TYPED TCDT).
1683	007274	177777				-1			
1684	007276	104000				SCOPE			;SCOPE.
1685	007300	027727	171510	001101	BE:	CMP	@TCDT,#1101		;BLOCK BETWEEN 0 AND 1101?
1686	007306	101365				BHI	BCA		;BR IF NOT (ERROR).
1687	007310	104021			BF:	STOPDT			;ALL STOP.
1688	007312	000207				RTS	PC		;EXIT.
1689						*****			
1690	007314	000002			T2:	2			;ROUTINE NUMBER 2 *
1691	007316	007552				T3			;ADDRESS OF NEXT ROUTINE *
1692	007320	000012				10.			;TEST ITERATION COUNT *
1693	007322	007324				CA			;SCOPE ENTRY POINT *
1694						.LIST			
1695						*****			
1696						;TEST ABILITY TO DETECT REV ENDZ, BUT NOT BEFORE BLOCK # 0 HAS BEEN READ.			
1697	007324	104017			CA:	SETCOM			;ISSUE RNUM!FWD!DO.
1698	007326	000003				RNUM!FWD!DO			
1699	007330	104000				SCOPE			;SETCOM ERROR.
1700	007332	104016			CAB:	DELAY			;WAIT.
1701	007334	104023				CKERRZ			;CHECK FOR ERROR
1702	007336	104000				SCOPE			;HERE IF ERROR
1703	007340	000240				NOP			;ENDZ RETURN,
1704	007342	005777	171446			TST	@TCDT		;BLOCK 0?
1705	007346	001766				BEQ	CA		;YES. GO PAST BLOCK 0
1706	007350	104015				SVECTR			;NO. SET INTERRUPT VECTOR TO CD
1707	007352	007376				CD			
1708	007354	104017				SETCOM			;ISSUE RNUM!REV!IE!DO.
1709	007356	004103				RNUM!REV!IE!DO			
1710	007360	104000				SCOPE			;SETCOM ERROR.
1711	007362	000402				BR	CCB		
1712	007364	005277	171416		CCA:	INC	@TCCM		;ISSUE DO.
1713	007370	104016			CCB:	DELAY			;TIME OUT INTERRUPT.
1714	007372	104027				NOINT			;INTERRUPT FAILURE.



```

1715 007374 104000          SCOPE
1716 007376 012716 007404  CD:  MOV      #CE,(6)      ;HERE WHEN INTERRUPT OCCURS.
1717 007402 000002          RTI              ;EXIT TO CE.
1718 007404 022626          CE:  POPSP2     ;RESTORE STACK.
1719 007406 005777 171374  TST      @TCCM  ;ERR BIT SET?
1720 007412 100003          BPL      CEA    ;BR IF NOT.
1721 007414 104023          CKERRZ        ;CHECK FOR ERROR
1722 007416 104000          SCOPE          ;HERE IF ERROR
1723 007420 104053          PRMEND       ;PREMATURE REV ENDZ.
1724 007422 017767 171366 171432  CEA:  MOV      @TCDT,BLKRQ ;SAVE BLOCK NUMBER.
1725 007430 005777 171360  TST      @TCDT  ;BLOCK 0?
1726 007434 001353          BNE      CCA   ;BR IF NOT. (KEEP LOOKING FOR BLOCK 0).
1727 007436 104015          SVECTR       ;SET INTERRUPT VECTOR TO CH
1728 007440 007522          CH
1729 007442 104016          DELAY        ;WAIT FOR POSSIBLE ENDZ INTERRUPT.
1730 007444 104020          STATUS      ;SAVE TCCM AND TCST
1731 007446 005777 171332  TST      @TCST  ;ENDZ BIT SET?
1732 007452 100405          BMI      CF   ;BR IF ENDZ IS SET.
1733 007454 104012          ERRORN      ;REV END ZONE FAILED TO SET
1734 007456 006522          MSG4
1735 007460 006115          STAT
1736 007462 177777          -1
1737 007464 104000          SCOPE
1738 007466 032777 000200 171310  CF:  BIT      #UPS,@TCST ;UPS CLEAR?
1739 007474 001405          BEQ      CG   ;BR IF UPS CLEAR
1740 007476 104012          CFA:  ERRORN    ;GETTING TO REV ENDZ FAILED TO
1741 007500 006543          MSG5        ;CLEAR UPS.
1742 007502 006115          STAT
1743 007504 177777          -1
1744 007506 104000          SCOPE
1745 007510 104012          CG:  ERRORN    ;SCOPE
1746 007512 006500          MSG3        ;ENDZ BIT FAILED TO CAUSE
1747 007514 006115          STAT       ;INTERRUPT. (CHECK TYPED TCCM
1748 007516 177777          -1         ;TO SEE IF ERROR BIT SET).
1749 007520 104000          SCOPE
1750 007522 012716 007530  CH:  MOV      #CI,(6)  ;SCOPE.
1751 007526 000002          RTI          ;HERE IF INTERRUPT OCCURS.
1752 007530 022626          CI:  POPSP2     ;EXIT TO CI.
1753 007532 104023          CKERRZ       ;RESTORE STACK.
1754 007534 104000          SCOPE        ;CHECK FOR ERROR OR ENDZ
1755 007536 000240          NOP         ;HERE IF ERROR
1756 007540 032777 000200 171236  BIT      #UPS,@TCST ;HERE IF ENDZ. UPS CLEAR?
1757 007546 001353          BNE      CFA  ;BR IF UPS NOT CLEAR.
1758 007550 104000          SCOPE       ;SCOPE.
1759
1760
1761 007552 000003          t3:  3          ;ROUTINE NUMBER 3 *
1762 007554 007706          T4         ;ADDRESS OF NEXT ROUTINE *
1763 007556 000001          1         ;TEST ITERATION COUNT *
1764 007560 007562          DA        ;SCOPE ENTRY POINT *
1765
1766
1767
1768 007562 104025          .LIST
1769 007564 104015          ;*****
1770 007566 007616          DA:  REWIND    ;SEQUENTIALLY SEARCH ALL BLOCK NUMBERS IN FORWARD DIRECTION.
          SVECTR    ;REWIND TO REV ENDZ.
          DB      ;SET INTERRUPT VECTOR TO DB

```



```

1771 007570 005067 171266 CLR BLKRQ ;1ST EXPECTED BLOCK # TO BLKRQ.
1772 007574 012767 001102 171242 MOV #578.,CTRA ;# OF BLOCKS TO CHECK TO CTRA
1773 007602 104017 SETCOM ;ISSUE RNUM!FWD!IE!DO
1774 007604 000103 RNUM!FWD!IE!DO
1775 007606 104000 SCOPE ;SETCOM ERROR.
1776 007610 104016 DAA: DELAY ;TIMEOUT INTERRUPT
1777 007612 104027 NOINT ;NO DECTAPE INTERRUPT.
1778 007614 104000 SCOPE
1779 007616 012716 007624 DB: MOV #DC,(6) ;HERE WHEN INTERRUPT OCCURS. SET
1780 007622 000002 RTI ;INIT. EXIT TO DC AND EXIT.
1781 007624 022626 DC: POPSP2 ;RESTORE STACK.
1782 007626 104023 CKERRZ ;CHECK FOR ERROR OR ENDZ
1783 007630 104000 SCOPE ;HERE IF ERROR.
1784 007632 104053 PRMEND ;PREMATURE FWD ENDZ.
1785 007634 104024 SAVDTR ;SAVE TCDT
1786 007636 026777 171220 171150 CMP BLKRQ,TCDT ;TCDT=EXPECTED BLOCK?
1787 007644 001407 BEQ DD ;BR IF YES.
1788 007646 104012 ERRORN ;BLOCK # IN TCDT NOT EQUAL TO
1789 007650 006461 EMSG2 ;EXPECTED BLOCK NUMBER
1790 007652 006200 BLKSB
1791 007654 006057 CTCDT
1792 007656 006115 STAT
1793 007660 177777 -1
1794 007662 104000 SCOPE ;SCOPE
1795 007664 005267 171172 DD: INC BLKRQ
1796 007670 005367 171150 DEC CTRA ;578 BLOCK # TESTED?
1797 007674 001403 BEQ DF ;BR IF YES.
1798 007676 105277 171104 INCB TCDCM ;ISSUE DO TO CONTINUE RNUM!FWD!IE
1799 007702 000742 BR DAA
1800 007704 104000 DF: SCOPE ;SCOPE.
1801
1802 *****
1803 007706 000004 †4: 4 ;ROUTINE NUMBER 4 *
1804 007710 010116 T5 ;ADDRESS OF NEXT ROUTINE *
1805 007712 000012 10. ;TEST ITERATION COUNT *
1806 007714 007716 EA ;SCOPE ENTRY POINT *
1807 .LIST
1808 *****
1809 ;TEST ABILITY TO DETECT FWD ENDZ, BUT NOT BEFORE BLOCK 1101 HAS BEEN READ.
1810 007716 104025 EA: REWIND ;GO TO REV ENDZ.
1811 007720 000404 BR EAA
1812 007722 104017 SETCOM ;ISSUE RNUM!REV!DO
1813 007724 004003 RNUM!REV!DO
1814 007726 104000 SCOPE ;SETCOM ERROR.
1815 007730 104016 DELAY ;WAIT.
1816 007732 012777 177777 171054 EAA: MOV #-1,TCDT ;-1 TO TCDT.
1817 007740 104015 SVECTR ;OK RETURN
1818 007742 007760 EC
1819 007744 104017 SETCOM ;ISSUE RNUM!FWD!IE!DO
1820 007746 000103 RNUM!FWD!IE!DO
1821 007750 104000 SCOPE ;SETCOM ERROR.
1822 007752 104016 EB: DELAY ;FAILURE TO INTERRUPT.
1823 007754 104027 NOINT
1824 007756 104000 SCOPE
1825 007760 012716 007766 EC: MOV #ED,(6)
1826 007764 000002 RTI ;EXIT INTERRUPT

```



1827	007766	022626			ED:	POPSP2				: RESTORE STACK.
1828	007770	104023				CKERRZ				: CHECK FOR ERRORS.
1829	007772	104000				SCOPE				: ERROR RETURN
1830	007774	104053				PRMEND				: PREMATURE FWD ENDZ.
1831	007776	017767	171012	171056		MOV	@TCDT, BLKRQ			: SAVE BLOCK NUMBER.
1832	010004	022777	001101	171002		CMP	#1101, @TCDT			: BLOCK 1101 (577(10))
1833	010012	001403				BEQ	EE			: BR IF BLOCK 1101 FOUND.
1834	010014	105277	170766			INCB	@TCCM			: NOT YET. CONTINUE SEARCH.
1835	010020	000754				BR	EB			
1836	010022	104015			EE:	SVECTR				: SET INTERRUPT VECTOR TO EH.
1837	010024	010064				EH				
1838	010026	104016				DELAY				: TIMEOUT INTERRUPT
1839	010030	104020				STATUS				: SAVE TCCM AND TCST
1840	010032	005777	170746			TST	@TCST			: ENDZ SET?
1841	010036	100405				BMI	EF			: BR IF SET.
1842	010040	104012				ERRORN				: FWD ENDZ FAILED TO SET ENDZ BIT.
1843	010042	006522				MSG4				
1844	010044	006115				STAT				
1845	010046	177777				-1				
1846	010050	104000				SCOPE				: SCOPE.
1847	010052	104012			EF:	ERRORN				: ENDZ BIT FAILED TO CAUSE
1848	010054	006500				MSG3				: INTERRUPT. CHECK TYPED TCCM
1849	010056	006115				STAT				: TO SEE IF ERROR BIT SET.
1850	010060	177777				-1				
1851	010062	104000				SCOPE				: SCOPE.
1852	010064	012716	010072		EH:	MOV	#EI, (6)			: HERE WHEN INTERRUPT OCCURS. SET
1853	010070	000002				RTI				: INT. EXIT POINTER TO EI AND EXIT.
1854	010072	022626			EI:	POPSP2				: RESTORE STACK.
1855	010074	104023				CKERRZ				: CHECK FOR ERRORS
1856	010076	104000				SCOPE				: ERROR RETURN
1857	010100	000404				BR	EK			: ENDZ RETURN
1858	010102	104012				ERRORN				: ERROR. INTERRUPT AFTER BLOCK 1101 NOT
1859	010104	006625				MSG7				: DUE TO ENDZ.
1860	010106	006115				STAT				
1861	010110	177777				-1				
1862	010112	104021			EK:	STOPDT				: ALL STOP
1863	010114	104000				SCOPE				: SCOPE
1864						*****				
1865	010116	000005			T5:	5				: ROUTINE NUMBER 5
1866	010120	010254				T6				: ADDRESS OF NEXT ROUTINE
1867	010122	000001				1				: TEST ITERATION COUNT
1868	010124	010126				EEA				: SCOPE ENTRY POINT
1869						.LIST				
1870						*****				
1871					EEA:	: SEQUENTIALLY SEARCH ALL BLOCK NUMBERS IN REVERSE DIRECTION.				
1872	010126	104026				WIND				: GO TO FWD ENDZ.
1873	010130	104015				SVECTR				: SET INTERRUPT VECTOR TO EEB.
1874	010132	010164				EEB				
1875	010134	012767	001101	170720		MOV	#577., BLKRQ			: START WITH BLK# 577.
1876	010142	012767	001102	170674		MOV	#578., CTRA			: # OF BLOCKS TO CHECK TO CTRA.
1877	010150	104017				SETCOM				: ISSUE RNUM!REV!IE!DO.
1878	010152	004103				RNUM!REV!IE!DO				
1879	010154	104000			EEEA:	SCOPE				: SETCOM ERROR.
1880	010156	104016				DELAY				: TIME OUT INTERRUPT.
1881	010160	104027				NOINT				: FAILURE TO INTERRUPT.
1882	010162	104000				SCOPE				



```

1883 010164 012716 010172 EEB: MOV #EEC,(6) ;HERE WHEN INTERRUPT OCCURS.
1884 010170 000002 RTI ;EXIT TO EEC.
1885 010172 022626 EEC: POPSP2 ;RESTORE STACK.
1886 010174 104023 CKERRZ ;ERROR OR ENDZ?
1887 010176 104000 SCOPE ;ERROR RETURN.
1888 010200 104053 PRMEND ;PREMATURE REV ENDZ.
1889 010202 104024 SAVDTR ;SAVE TCDT.
1890 010204 026777 170652 170602 CMP BLKRQ,@TCDT ;EXPECTED AND ACTUAL BLK#'S SAME?
1891 010212 001407 BEQ EED ;BR IF SAME.
1892 010214 104012 ERRORN ;BLOCK# IN TCDT NOT EQUAL TO
1893 010216 006461 MSG2 ;EXPECTED BLOCK#.
1894 010220 006200 BLKSB
1895 010222 006057 CTCDT
1896 010224 006115 STAT
1897 010226 177777 -1
1898 010230 104000 SCOPE
1899 010232 005367 170624 EED: DEC BLKRQ ;DECREMENT EXPECTED BLK#.
1900 010236 005367 170602 DEC CTRA ;CHECKED 578 BLOCKS?
1901 010242 001403 BEQ EEF ;BR IF YES.
1902 010244 005277 170536 INC @TCCM ;ISSUE DO TO CONTINUE SEARCH.
1903 010250 000742 BR EEAA
1904 010252 104000 EEF: SCOPE
1905 *****
1906 010254 000006 T6: 6 ;ROUTINE NUMBER 6 *
1907 010256 010426 T7 ;ADDRESS OF NEXT ROUTINE *
1908 010260 000001 I ;TEST ITERATION COUNT *
1909 010262 010264 FA ;SCOPE ENTRY POINT *
1910 .LIST
1911 *****
1912 ;SEARCH FORWARD. READ EVERY ELEVENTH BLOCK # AND CHECK THAT IT MATCHES
1913 ;WITH EXPECTED BLOCK.
1914 010264 104025 FA: REWIND ;REWIND TO REV ENDZ.
1915 010266 012767 000062 170550 MOV #50.,CTRA
1916 010274 012767 000013 170544 MOV #11.,CTRB
1917 010302 012767 000012 170552 MOV #10.,BLKRQ
1918 010310 104015 SVECTR ;SET INTERRUPT VECTOR TO FB.
1919 010312 010330 FB
1920 010314 104017 SETCOM ;ISSUE RNUM!FWD!IE!DO
1921 010316 000103 RNUM!FWD!IE!DO
1922 010320 104000 SCOPE ;SETCOM ERROR.
1923 010322 104016 FAA: DELAY ;TIME OUT INTERRUPT
1924 010324 104027 NOINT ;FAILURE TO INTERRUPT
1925 010326 104000 SCOPE ;SCOPE.
1926 010330 012716 010336 FB: MOV #FC,(6) ;HERE WHEN INTERRUPT OCCURS. SET
1927 010334 000002 RTI ;EXIT TO FC AND EXIT.
1928 010336 022626 FC: POPSP2 ;RESTORE STACK.
1929 010340 104023 CKERRZ ;CHECK FOR ERRORS
1930 010342 104000 SCOPE ;ERROR RETURN.
1931 010344 104053 PRMEND ;PREMATURE FWD ENDZ.
1932 010346 005367 170474 DEC CTRB ;ELEVENTH BLOCK?
1933 010352 001403 BEQ FD ;BR IF YES.
1934 010354 005277 170426 FCA: INC @TCCM ;NO. CONTINUE
1935 010360 000760 BR FAA
1936 010362 104024 FD: SAVDTR ;SAVE TCDT.
1937 010364 026777 170472 170422 CMP BLKRQ,@TCDT ;BLKRQ=TCDT?
1938 010372 001402 BEQ FE ;BR IF SAME.

```



```

1939 010374 104034 WRNGBK ;THE BLOCK NOT THE EXPECTED BLOCK.
1940 010376 104000 SCOPE ;SCOPE
1941 010400 062767 000013 170454 FE: ADD #11.,BLKRQ ;SET UP BCKRQ TO NEXT EXPECTED BLOCK.
1942 010406 012767 000013 170432 MOV #11.,CTRB ;RESET CTRB TO 11.
1943 010414 005367 170424 DEC CTRA ;DONE 50 TIMES?
1944 010420 001355 BNE FCA ;BR IF NOT.
1945 010422 104021 STOPDT ;DONE TO. ALL STOP.
1946 010424 104000 SCOPE ;SCOPE
1947 *****
1948 010426 000007 t7: 7 ;ROUTINE NUMBER 7 *
1949 010430 010600 T10 ;ADDRESS OF NEXT ROUTINE *
1950 010432 000001 1 ;TEST ITERATION COUNT *
1951 010434 010436 GA ;SCOPE ENTRY POINT *
1952 .LIST
1953 *****
1954 ;SEARCH REVERSE. READ EVERY 11TH BLOCK # AND CHECK THAT IT MATCHES
1955 ;WITH EXPECTED BLOCK.
1956 010436 104026 GA: WIND ;GO TO FWD ENDZ.
1957 010440 012767 000062 170376 MOV #50.,CTRA
1958 010446 012767 000013 170372 MOV #11.,CTRB
1959 010454 012767 001067 170400 MOV #567.,BLKRQ ;1ST EXPECTED BLK # TO BLKRQ
1960 010462 104015 SVECTR ;SET INTERRUPT VECTOR TO GB.
1961 010464 010502 GB
1962 010466 104017 SETCOM ;ISSUE RNUM!REV!IE!DO
1963 010470 004103 RNUM!REV!IE!DO
1964 010472 104000 SCOPE ;SETCOM ERROR.
1965 010474 104016 GAA: DELAY ;TIME OUT INTERRUPT
1966 010476 104027 NOINT ;FAILURE TO INTERRUPT
1967 010500 104000 SCOPE ;SCOPE
1968 010502 012716 010510 GB: MOV #GC,(6) ;HERE WHEN INTERRUPT OCCURS. SET
1969 010506 000002 RTI ;EXIT TO GC AND EXIT
1970 010510 022626 GC: POPSP2 ;RESTORE STACK.
1971 010512 104023 CKERRZ ;CHECK FOR ERROR OR ENDZ.
1972 010514 104000 SCOPE ;HERE IF ERROR
1973 010516 104053 PRMEND ;PREMATURE REV ENDZ.
1974 010520 005367 170322 DEC CTRB ;ELEVENTH BLOCK?
1975 010524 001403 BEQ GD ;BR IF YES.
1976 010526 005277 170254 GCA: INC @TCCM ;NO. CONTINUE
1977 010532 000760 BR GAA
1978 010534 104024 GD: SAVDTR ;SAVE TCDT.
1979 010536 026777 170320 170250 CMP BLKRQ,@TCDT ;TCDT SAME AS EXPECTED?
1980 010544 001402 BEQ GE ;BR IF YES.
1981 010546 104034 WRNGBK ;NO BLK# IN TCDT NOT SAME AS
1982 010550 104000 SCOPE ;EXPECTED. SCOPE.
1983 010552 162767 000013 170302 GE: SUB #11.,BLKRQ ;SET UP BLKRQ TO NEXT EXPECTED BLOCK.
1984 010560 012767 000013 170260 MOV #11.,CTRB ;RESET CTRB TO 11.
1985 010566 005367 170252 DEC CTRA ;DONE 50 TIMES
1986 010572 001355 BNE GCA ;BR IF NOT
1987 010574 104021 STOPDT ;DONE 50. ALL STOP.
1988 010576 104000 SCOPE ;SCOPE
1989 *****
1990 010600 000010 t10: 10 ;ROUTINE NUMBER 10 *
1991 010602 010732 T11 ;ADDRESS OF NEXT ROUTINE *
1992 010604 000001 1 ;TEST ITERATION COUNT *
1993 010606 010610 HA ;SCOPE ENTRY POINT *
1994 .LIST

```



```

1995
1996
1997
1998 010610 104025
1999 010612 012767 000024 170224
2000 010620 012767 000000 170234
2001 010626 104015
2002 010630 010646
2003 010632 104017
2004 010634 000103
2005 010636 104000
2006 010640 104016
2007 010642 104027
2008 010644 104000
2009 010646 012716 010654
2010 010652 000002
2011 010654 022626
2012 010656 104023
2013 010660 104000
2014 010662 104053
2015 010664 104021
2016 010666 104024
2017 010670 027767 170120 170164
2018 010676 101402
2019 010700 104034
2020 010702 104000
2021 010704 017767 170104 170150
2022 010712 062767 000005 170142
2023 010720 104016
2024 010722 005367 170116
2025 010726 001341
2026 010730 104000
2027
2028 010732 000011
2029 010734 011064
2030 010736 000001
2031 010740 010742
2032
2033
2034
2035
2036 010742 104026
2037 010744 012767 000024 170072
2038 010752 012767 001101 170102
2039 010760 104015
2040 010762 011000
2041 010764 104017
2042 010766 004103
2043 010770 104000
2044 010772 104016
2045 010774 104027
2046 010776 104000
2047 011000 012716 011006
2048 011004 000002
2049 011006 022626
2050 011010 104023

```

```

;*****
;FORWARD STOP-START TEST. FROM STAND STILL THE FIRST BLOCK # READ
;MUST NOT BE MORE THAN 5 BLOCKS AWAY FROM PREVIOUSLY READ BLOCK.
HA: REWIND ;REWIND TO REV ENDZ.
MOV #20,CTRA ;TIMES TO DO TEST TO CTRA.
MOV #0,BLKRG ;SET EXPECTED BLOCK TO 0
SVECTR ;SET INTERRUPT VECTOR TO HB.
HB
HAA: SETCOM ;ISSUE RNUM!FWD!IE!DO.
RNUM!FWD!IE!DO
SCOPE ;SETCOM ERROR.
DELAY ;TIME OUT INTERRUPT
NOINT ;FAILURE TO INTERRUPT
SCOPE ;SCOPE
HB: MOV #HC,(6) ;HERE WHEN INTERRUPT OCCURS EXIT
RTI ;TO HC.
HC: POPSP2 ;RESTORE STACK.
CKERRZ ;CHECK FOR ERROR OR ENDZ.
SCOPE
PRMEND ;PREMATURE FWD ENDZ.
STOPDT ;ALL STOP
SAVDTR ;SAVE TCDT.
CMP @TCDT,BLKRG ;COMPARE EXPECTED AND CURRENT BLOCKS.
BLOS HD ;BR IF TCDT LOWER OR SAME.
WRNGBK ;BLOCK# READ MORE THAN 5 BLOCKS AWAY
SCOPE ;FROM PREVIOUS BLOCK READ.
HD: MOV @TCDT,BLKRG ;SAVE CURRENT BLOCK#.
ADD #5,BLKRG ;SET UP FOR NEXT EXPECTED BLOCK
DELAY ;WAIT. INSURE TAPE IS STOPPED.
DEC CTRA ;DONE TIMES REQUIRED?
BNE HAA ;BR IF NOT.
SCOPE ;DONE. SCOPE.
;*****
T11: 11 ;ROUTINE NUMBER 11 *
12 ;ADDRESS OF NEXT ROUTINE *
1. ;TEST ITERATION COUNT *
IA ;SCOPE ENTRY POINT *
.LIST
;*****
;REVERSE STOP-START TEST. FROM STAND STILL THE FIRST BLOCK # READ
;MUST NOT BE MORE THAN 5 BLOCKS AWAY FROM PREVIOUSLY READ BLOCK.
IA: WIND ;GO TO FWD ENDZ.
MOV #20,CTRA ;DO TEST 20 TIMES
MOV #1101,BLKRG ;1ST EXPECTED BLOCK # TO BLKRG
SVECTR ;SET INTERRUPT VECTOR TO IB
IB
IAA: SETCOM ;ISSUE RNUM!REV!IE!DO.
RNUM!REV!IE!DO
SCOPE ;SETCOM ERROR.
DELAY ;TIME OUT INTERRUPT
NOINT ;FAILURE TO INTERRUPT
SCOPE ;SCOPE
IB: MOV #IC,(6) ;HERE WHEN INTERRUPT OCCURS.
RTI ;EXIT INTERRUPT TO IC.
IC: POPSP2 ;RESTORE STACK.
CKERRZ ;CHECK FOR ERROR OR ENDZ

```



```

2051 011012 104000          SCOPE          ;ERROR OR ENDZ RETURN
2052 011014 104053          PRMEND        ;PREMATURE REV ENDZ.
2053 011016 104021          STOPDT       ;ALL STOP
2054 011020 104024          SAVDTR       ;SAVE TCDT.
2055 011022 026777 170034 167764  CMP          BLKRQ, @TCDT ;TCDT SAME AS EXPECTED BLOCK #?
2056 011030 101402          BLOS         ID        ;BR IF TCDT EQUAL OR GREATER.
2057 011032 104034          WRNGBK      ;BLOCK# READ MORE THAN 5 BLOCKS AWAY
2058 011034 104000          SCOPE        ;FROM PREVIOUS BLOCK READ.
2059 011036 017767 167752 170016 ID:  MOV          @TCDT, BLKRQ ;SAVE CURRENT BLOCK#.
2060 011044 162767 000005 170010  SUB          #5, BLKRQ  ;SET UP FOR NEXT EXPECTED BLOCK
2061 011052 104016          DELAY       ;WAIT (INSURE TAPE IS STOPPED).
2062 011054 005367 167764          DEC         CTRA      ;DONE TIMES REQUIRED?
2063 011060 001341          BNE         IAA       ;BR IF NOT.
2064 011062 104000          SCOPE        ;DONE. SCOPE.
2065 *****
2066 011064 000012          T12:        12        ;ROUTINE NUMBER 12 *
2067 011066 011162          T13        13        ;ADDRESS OF NEXT ROUTINE *
2068 011070 000001          I          1        ;TEST ITERATION COUNT *
2069 011072 011074          JA         ;SCOPE ENTRY POINT *
2070 .LIST
2071 *****
2072 ;SEARCH FORWARD AND REVERSE ALL BLOCKS USING SRCHF AND SRCHR SUBS.
2073 011074 012767 001102 167742 JA:  MOV          #578., CTRA ;SET UP TO SEARCH 578 BLOCKS
2074 011102 012767 177777 167752  MOV          #-1, BLKRQ ;STARTING WITH BLOCK 0.
2075 011110 005267 167746          JB:  INC         BLKRQ
2076 011114 104032          SRCHF       ;FIND FWD BLOCK # SET IN BLKRQ.
2077 011116 104000          SCOPE        ;ERROR IN SRCHF.
2078 011120 005367 167720          DEC         CTRA      ;FOUND 578 BLOCKS?
2079 011124 001371          BNE         JB        ;BR IF NOT.
2080 011126 012767 001102 167710  MOV          #578., CTRA ;SET UP TO SEARCH 578 BLOCKS
2081 011134 012767 001102 167720  MOV          #578., BLKRQ ;STARTING WITH BLOCK 577.
2082 011142 005367 167714          JC:  DEC         BLKRQ
2083 011146 104033          SRCHR       ;FIND REVERSE BLOCK SET IN BLKRQ.
2084 011150 104000          SCOPE        ;ERROR IN SRCHR.
2085 011152 005367 167666          DEC         CTRA      ;578 BLOCKS FOUND?
2086 011156 001371          BNE         JC        ;BR IF NOT.
2087 011160 104000          SCOPE        ;DONE. SCOPE.
2088 *****
2089 011162 000013          T13:        13        ;ROUTINE NUMBER 13 *
2090 011164 011240          T14        14        ;ADDRESS OF NEXT ROUTINE *
2091 011166 000024          20.        ;TEST ITERATION COUNT *
2092 011170 011174          KA         ;SCOPE ENTRY POINT *
2093 .LIST
2094 *****
2095 ;SINGLE BLOCK TRANSFER TEST. WDATA FWD, RDATA FWD. DATA IS ALL 0'S.
2096 011172 104030          RNDBLK      ;RANDOM BLK# TO BLKRQ.
2097 011174 104044          KA:  CLEAR     ;CLEAR 256 WORD WRITE BUFFER TO 0'S.
2098 011176 013350          WBUF
2099 011200 000400          256.
2100 011202 104045          FILL        ;CLEAR 256 WORD READ BUFFER TO 1'S.
2101 011204 015420          RBUF
2102 011206 177777          -1
2103 011210 000400          256.
2104 011212 104037          WDATAF     ;CALL WDATAF SUB TO WRITE FWD 256. WORDS
2105 011214 000400          256.        ;STARTING AT ADDR WBUF
2106 011216 013350          WBUF

```



```

2107 011220 104041          RDATAF          ;CALL RDATAF SUB TO READ FWD 256. WORDS
2108 011222 000400          256.          ;AND STORE AT ADDR STARTING AT RBUF
2109 011224 015420          RBUF
2110 011226 104047          DATCHK          ;CALL DATCHK SUB TO CHECK DATA STORED AT
2111 011230 013350          WBUF          ;WBUF AGAINST DATA STORED AT RBUF
2112 011232 015420          RBUF          ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
2113 011234 000400          256.          ;ERRORS.
2114 011236 104000          SCOPE
*****
2115 011240 000014          †14: 14          ;ROUTINE NUMBER 14 *
2116 011242 011304          T15          ;ADDRESS OF NEXT ROUTINE *
2117 011244 000024          20.          ;TEST ITERATION COUNT *
2118 011246 011266          LA          ;SCOPE ENTRY POINT *
      .LIST
*****
2119          ;SINGLE BLOCK TRANSFER TEST. WDATA FWD, RDATAFWD. BINARY COUNT PATTERN.
2120          RNDBLK          ;RANDOM BLK# TO BLKRG
2121          MOV          #WDATAF,SBINTX ;SET UP WRITE DATA FWD.
2122          MOV          #RDATAF,SBINTY ;SET UP READ DATA FWD.
2123 011250 104030          LA: JSR          PC,SBINT† ;GO DO SINGLE BLOCK TEST.
2124 011252 012767 104037 000214 DATCHK          ;CALL DATCHK SUB TO CHECK DATA STORED AT
2125 011260 012767 104041 000214 WBUF          ;WBUF AGAINST DATA STORED AT RBUF
2126 011266 004767 000166 RBUF          ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
2127 011272 104047          256.          ;ERRORS.
2128 011274 013350          SCOPE
*****
2129          †15: 15          ;ROUTINE NUMBER 15 *
2130 011276 015420          T16          ;ADDRESS OF NEXT ROUTINE *
2131 011300 000400          20.          ;TEST ITERATION COUNT *
2132 011302 104000          MA          ;SCOPE ENTRY POINT *
      .LIST
*****
2133          ;SINGLE BLOCK TRANSFER TEST. WDATA REV, RDATA REV. BINARY COUNT PATTERN.
2134          RNDBLK          ;RANDOM BLK# TO BLKRG
2135          MOV          #WDATAR,SBINTX ;SET UP WRITE DATA REV.
2136          MOV          #RDATAR,SBINTY ;SET UP READ DATA REV.
2137 011304 000015          MA: JSR          PC,SBINT† ;GO DO SINGLE BLOCK TEST.
2138 011306 011350          DATCHK          ;CALL DATCHK SUB TO CHECK DATA STORED AT
2139 011310 000024          WBUF          ;WBUF AGAINST DATA STORED AT RBUF
2140 011312 011332          RBUF          ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
2141          256.          ;ERRORS.
2142          SCOPE
*****
2143          †16: 16          ;ROUTINE NUMBER 16 *
2144 011314 104030          T17          ;ADDRESS OF NEXT ROUTINE *
2145 011316 012767 104040 000150          20.          ;TEST ITERATION COUNT *
2146 011324 012767 104042 000150          NA          ;SCOPE ENTRY POINT *
2147 011332 004767 000122          .LIST
*****
2148          ;SINGLE BLOCK TRANSFER TEST. WDATA FWD, RDATA REV. BINARY COUNT PATTERN.
2149          RNDBLK          ;RANDOM BLK# TO BLKRG
2150          MOV          #WDATAF,SBINTX ;SET UP WRITE DATA FWD.
2151          MOV          #RDATAR,SBINTY ;SET UP READ DATA REV.
2152 011336 104047          NA: JSR          PC,SBINT† ;GO DO SINGLE BLOCK TRANSFERS.
2153 011340 013350          DATCKI          ;CALL DATCKI TO CHECK DATA STORED AT
2154 011342 015420          WBUF          ;WBUF AGAINST DATA STORED AT RBUF+510.
2155 011344 000400          256.
2156 011346 104000          SCOPE
*****
2157          †16: 16          ;ROUTINE NUMBER 16 *
2158 011350 000016          T17          ;ADDRESS OF NEXT ROUTINE *
2159 011352 011414          20.          ;TEST ITERATION COUNT *
2160 011354 000024          NA          ;SCOPE ENTRY POINT *
2161 011356 011376          .LIST
*****
2162          ;SINGLE BLOCK TRANSFER TEST. WDATA FWD, RDATA REV. BINARY COUNT PATTERN.
2163          RNDBLK          ;RANDOM BLK# TO BLKRG
2164          MOV          #WDATAF,SBINTX ;SET UP WRITE DATA FWD.
2165          MOV          #RDATAR,SBINTY ;SET UP READ DATA REV.
2166 011360 104030          NA: JSR          PC,SBINT† ;GO DO SINGLE BLOCK TRANSFERS.
2167 011362 012767 104037 000104          DATCKI          ;CALL DATCKI TO CHECK DATA STORED AT
2168 011370 012767 104042 000104          WBUF          ;WBUF AGAINST DATA STORED AT RBUF+510.
2169 011376 004767 000056
2170 011402 104050
2171 011404 013350

```



2163 011406 016416  
 2164 011410 000400  
 2165 011412 104000  
 2166  
 2167 011414 000017  
 2168 011416 011512  
 2169 011420 000024  
 2170 011422 011442  
 2171  
 2172  
 2173  
 2174 011424 104030  
 2175 011426 012767 104040 000040  
 2176 011434 012767 104041 000040  
 2177 011442 004767 000012  
 2178 011446 104050  
 2179 011450 013350  
 2180 011452 016416  
 2181 011454 000400  
 2182 011456 104000  
 2183  
 2184 011460 104046  
 2185 011462 013350  
 2186 011464 000400  
 2187 011466 104044  
 2188 011470 015420  
 2189 011472 000400  
 2190 011474 000000  
 2191 011476 000400  
 2192 011500 013350  
 2193 011502 000000  
 2194 011504 000400  
 2195 011506 015420  
 2196 011510 000207  
 2197  
 2198 011512 000020  
 2199 011514 011560  
 2200 011516 000024  
 2201 011520 011542  
 2202  
 2203  
 2204  
 2205 011522 012767 104037 000240  
 2206 011530 012767 104041 000272  
 2207 011536 005067 000276  
 2208 011542 004767 000200  
 2209 011546 104047  
 2210 011550 013350  
 2211 011552 015420  
 2212 011554 001000  
 2213 011556 104000  
 2214  
 2215 011560 000021  
 2216 011562 011626  
 2217 011564 000024  
 2218 011566 011610

```

RBUF+510. ;CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
256. ;ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
SCOPE
:*****
†17: 17 ;ROUTINE NUMBER 17 *
T20 ;ADDRESS OF NEXT ROUTINE *
20. ;TEST ITERATION COUNT *
0A ;SCOPE ENTRY POINT *
.LIST
:*****
;SINGLE BLOCK TRANSFER TEST. WDATA REV, RDATA FWD. BINARY COUNT PATTERN.
RDNBLK ;RANDOM BLK# TO BLK#Q.
MOV #WDATAR,SBINTX ;SET UP WRITE DATA REV.
MOV #RDATAF,SBINTY ;SET UP READ DATA FWD.
OA: JSR PC,SBINT ;GO DO SINGLE BLOCK TRANSFERS.
DATCKI ;CALL DATCKI TO CHECK DATA STORED AT
WBUF ;WBUF AGAINST DATA STORED AT RBUF+510.
RBUF+510. ;CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
256. ;ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
SCOPE
SBINT: BINFIL ;FILL 256 WORD WRITE BUFFER WITH
WBUF ;BINARY COUNT PATTERN.
256.
CLEAR ;CLEAR 256 WORD READ BUFFER.
RBUF
256.
SBINTX: OPEN ;WDATAR OR WDATAF.
256.
WBUF ;ADDR TO START WRITE FROM.
SBINTY: OPEN ;RDATAF OR RDATAF.
256. ;READ 256 WORDS
RBUF ;INTO THIS ADDR AND UP.
RTS PC ;EXIT SBINT SUB.
:*****
†20: 20 ;ROUTINE NUMBER 20 *
T21 ;ADDRESS OF NEXT ROUTINE *
20. ;TEST ITERATION COUNT *
PA ;SCOPE ENTRY POINT *
.LIST
:*****
;TWO BLOCK TRANSFER TEST. WDATA FWD, RDATA FWD. BINARY COUNT PATTERN.
MOV #WDATAF,DBINTX ;SET UP WDATAF.
MOV #RDATAF,DBINTY ;SET UP RDATAF.
PA: CLR DBINTZ ;CLEAR DIFFERENT DIRECTION INDICATOR.
JSR PC,DBINT ;GO DO 2 BLOCK TRANSFERS.
DATCHK ;CALL DATCHK SUB TO CHECK DATA STORED AT
WBUF ;WBUF AGAINST DATA STORED AT RBUF
RBUF ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
512. ;ERRORS.
SCOPE
:*****
†21: 21 ;ROUTINE NUMBER 21 *
T22 ;ADDRESS OF NEXT ROUTINE *
20. ;TEST ITERATION COUNT *
QA ;SCOPE ENTRY POINT *

```



```

2219
2220
2221
2222 011570 012767 104040 000172
2223 011576 012767 104042 000224
2224 011604 005067 000230
2225 011610 004767 000132
2226 011614 104047
2227 011616 013350
2228 011620 015420
2229 011622 001000
2230 011624 104000
2231
2232 011626 000022
2233 011630 011676
2234 011632 000024
2235 011634 011660
2236
2237
2238
2239 011636 012767 104037 000124
2240 011644 012767 104042 000156
2241 011652 012767 177777 000160
2242 011660 004767 000062
2243 011664 104050
2244 011666 013350
2245 011670 017416
2246 011672 001000
2247 011674 104000
2248
2249 011676 000023
2250 011700 012042
2251 011702 000024
2252 011704 011730
2253
2254
2255
2256 011706 012767 104040 000054
2257 011714 012767 104041 000106
2258 011722 012767 177777 000110
2259 011730 004767 000012
2260 011734 104050
2261 011736 013350
2262 011740 017416
2263 011742 001000
2264 011744 104000
2265 011746 104046
2266 011750 013350
2267 011752 001000
2268 011754 104044
2269 011756 015420
2270 011760 001000
2271 011762 012767 000310 167072
2272 011770 000000
2273 011772 001000
2274 011774 013350

```

```

.LIST
;*****
;TWO BLOCK TRANSFER TEST. WDATA REV. RDATA REV. BINARY COUNT PATTERN.
MOV #WDATAR,DBINTX ;SET UP WDATAR.
MOV #RDATAF,DBINTY ;SET UP RDATAF.
CLR DBINTZ ;CLEAR DIFFERENT DIRECTION INDICATOR.
QA: JSR PC,DBINT ;GO DO 2 BLOCK TRANSFERS.
DATCHK ;CALL DATCHK SUB TO CHECK DATA STORED AT
WBUF ;WBUF AGAINST DATA STORED AT RBUF
RBUF ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
512. ;ERRORS.
SCOPE
;*****
†22: 22 ;ROUTINE NUMBER 22 *
T23 ;ADDRESS OF NEXT ROUTINE *
20. ;TEST ITERATION COUNT *
RA ;SCOPE ENTRY POINT *
.LIST
;*****
;TWO BLOCK TRANSFER TEST. WDATA FWD. RDATA REV. BINARY COUNT PATTERN.
MOV #WDATAF,DBINTX ;SET UP WDATAF.
MOV #RDATAF,DBINTY ;SET UP RDATAF.
MOV #-1,DBINTZ ;SET DIFFERENT DIRECTION INDICATOR.
RA: JSR PC,DBINT ;GO DO 2 BLOCK TRANSFERS.
DATCKI ;CALL DATCKI TO CHECK DATA STORED AT
WBUF ;WBUF AGAINST DATA STORED AT RBUF+1022.
RBUF+1022. ;CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
512. ;ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
SCOPE
;*****
†23: 23 ;ROUTINE NUMBER 23 *
T24 ;ADDRESS OF NEXT ROUTINE *
20. ;TEST ITERATION COUNT *
SA ;SCOPE ENTRY POINT *
.LIST
;*****
;TWO BLOCK TRANSFER TEST. WDATA REV. RDATA FWD. BINARY COUNT PATTERN.
MOV #WDATAR,DBINTX ;SET UP WDATAR.
MOV #RDATAF,DBINTY ;SET UP RDATAF.
MOV #-1,DBINTZ ;SET DIFFERENT DIRECTION INDICATOR.
SA: JSR PC,DBINT ;GO DO 2 BLOCK TRANSFERS.
DATCKI ;CALL DATCKI TO CHECK DATA STORED AT
WBUF ;WBUF AGAINST DATA STORED AT RBUF+1022.
RBUF+1022. ;CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
512. ;ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
SCOPE
DBINT: BINFIL ;FILL 512 WORD WRITE BUFFER WITH
WBUF ;BINARY COUNT PATTERN.
512.
CLEAR ;CLEAR 512 WORD READ BUFFER.
RBUF
512.
DBINTX: MOV #200.,BLKRQ ;200 TO BLOCK REQUIRED.
OPEN ;WDATAF OR WDATAR.
512. ;WRITE 512 WORDS.
WBUF ;ADDR TO START WRITE FROM.

```



```

2275 011776 005767 000036      TST      DBINTZ      ;DIFF DIR IND SET?
2276 012002 001412      BEQ      DBINTY      ;BR IF NOT.
2277 012004 062767 000001 167050    ADD      #1,BLKRQ    ;ASSUME DT GOING FWD. INC BLKRQ.
2278 012012 032777 004000 166766    BIT      #REV,DTCCM  ;DT GOING FWD?
2279 012020 001403      BEQ      DBINTY      ;BR IF YES.
2280 012022 162767 000002 167032    SUB      #2,BLKRQ    ;NO. -2 FROM BLKRQ.
2281 012030 000000      DBINTY: OPEN      ;RDATAF OR RDATAF.
2282 012032 001000      512.      ;READ 512 WORDS.
2283 012034 015420      RBUF      ;ADDR TO START READING INTO.
2284 012036 000207      RTS      PC      ;EXIT DBINT SUB.
2285 012040 000000      DBINTZ: OPEN      ;DIFFERENT DIRECTION INDICATOR.
2286      ;*****
2287 012042 000024      †24: 24      ;ROUTINE NUMBER 24      *
2288 012044 012124      T25      ;ADDRESS OF NEXT ROUTINE      *
2289 012046 000024      20.      ;TEST ITERATION COUNT      *
2290 012050 012062      TA      ;SCOPE ENTRY POINT      *
2291      .LIST
2292      ;*****
2293      ;PARTIAL BLOCK TRANSFER TEST. WDATA FWD 1.5 BLOCKS. RDATA FWD 2 BLOCKS.
2294      ;BINARY COUNT PATTERN. UNWRITTEN PORTION OF 2ND BLOCK SHOULD BE 0'S.
2295 012052 104051      INBIN      ;INIT BINARY COUNT.
2296 012054 104044      CLEAR      ;CLEAR 512 WORD WRITE BUFFER.
2297 012056 013350      WBUF      ;
2298 012060 001000      512.      ;
2299 012062 104046      TA: BINFIL      ;FILL 384 WORD WRITE BUFFER WITH
2300 012064 013350      WBUF      ;BINARY COUNT PATTERN.
2301 012066 000600      384.      ;
2302 012070 104044      CLEAR      ;CLEAR 512 WORD READ BUFFER.
2303 012072 015420      RBUF      ;
2304 012074 001000      512.      ;
2305 012076 104037      WDATAF      ;CALL WDATAF SUB TO WRITE FWD 384. WORDS
2306 012100 000600      384.      ;STARTING AT ADDR WBUF
2307 012102 013350      WBUF      ;
2308 012104 104041      RDATAF      ;CALL RDATAF SUB TO READ FWD 512. WORDS
2309 012106 001000      512.      ;AND STORE AT ADDR STARTING AT RBUF
2310 012110 015420      RBUF      ;
2311 012112 104047      DATCHK      ;CALL DATCHK SUB TO CHECK DATA STORED AT
2312 012114 013350      WBUF      ;WBUF AGAINST DATA STORED AT RBUF
2313 012116 015420      RBUF      ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
2314 012120 001000      512.      ;ERRORS.
2315 012122 104000      SCOPE
2316      ;*****
2317 012124 000025      †25: 25      ;ROUTINE NUMBER 25      *
2318 012126 012172      T26      ;ADDRESS OF NEXT ROUTINE      *
2319 012130 000024      20.      ;TEST ITERATION COUNT      *
2320 012132 012154      UA      ;SCOPE ENTRY POINT      *
2321      .LIST
2322      ;*****
2323      ;4 BLOCK TRANSFER TEST. WDATA FWD. RDATA FWD. WRITE AND READ BUFFERS ARE
2324      ;COMBINED AND USED AS 1024 WORD READ BUFFER. WRITE DATA IS TAKEN FROM THE
2325      ;PROGRAM CODE ITSELF STARTING AT SYMBOLIC ADDRESS EMO.
2326 012134 012767 104037 000232    MOV      #WDATAF,QBINTX ;SET UP WDATAF.
2327 012142 012767 104041 000264    MOV      #RDATAF,QBINTY ;SET UP RDATAF.
2328 012150 005067 000270      CLR      QBINTZ      ;CLEAR DIFFERENT DIRECTION INDICATOR.
2329 012154 004767 000200      JSR      PC,QBINT     ;GO DO 4 BLOCK TRANSFERS.
2330 012160 104047      DATCHK      ;CALL DATCHK SUB TO CHECK DATA STORED AT

```



2331 012162 005464  
 2332 012164 013350  
 2333 012166 002000  
 2334 012170 104000  
 2335  
 2336 012172 000026  
 2337 012174 012240  
 2338 012176 000024  
 2339 012200 012222  
 2340  
 2341  
 2342  
 2343  
 2344  
 2345 012202 012767 104040 000164  
 2346 012210 012767 104042 000216  
 2347 012216 005067 000222  
 2348 012222 004767 000132  
 2349 012226 104047  
 2350 012230 005464  
 2351 012232 013350  
 2352 012234 002000  
 2353 012236 104000  
 2354  
 2355 012240 000027  
 2356 012242 012310  
 2357 012244 000024  
 2358 012246 012272  
 2359  
 2360  
 2361  
 2362  
 2363  
 2364 012250 012767 104037 000116  
 2365 012256 012767 104042 000150  
 2366 012264 012767 177777 000152  
 2367 012272 004767 000062  
 2368 012276 104050  
 2369 012300 005464  
 2370 012302 017346  
 2371 012304 002000  
 2372 012306 104000  
 2373  
 2374 012310 000030  
 2375 012312 012446  
 2376 012314 000024  
 2377 012316 012342  
 2378  
 2379  
 2380  
 2381  
 2382  
 2383 012320 012767 104040 000046  
 2384 012326 012767 104041 000100  
 2385 012334 012767 177777 000102  
 2386 012342 004767 000012

```

EMO                                ;EMO AGAINST DATA STORED AT WBUF
WBUF                                ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
1024.                                ;ERRORS.
SCOPE
*****
T26: 26                                ;ROUTINE NUMBER 26 *
      T27                                ;ADDRESS OF NEXT ROUTINE *
      20.                                ;TEST ITERATION COUNT *
      VA                                ;SCOPE ENTRY POINT *
      .LIST
*****
;4 BLOCK TRANSFER TEST. WDATA REV. RDATA REV. WRITE AND READ BUFFERS ARE
;COMBINED AND USED AS 1024 WORD READ BUFFER. WRITE DATA IS TAKEN FROM THE
;PROGRAM CODE ITSELF STARTING AT SYMBOLIC ADDRESS EMO.
MOV      #WDATAR,QBINTX             ;SET UP WDATAR.
MOV      #RDATA,QBINTY             ;SET UP RDATA.
CLR      QBINTZ                     ;CLEAR DIFFERENT DIRECTION INDICATOR.
VA:      JSR      PC,QBINT           ;GO DO 4 BLOCK TRANSFERS.
          DATCHK                    ;CALL DATCHK SUB TO CHECK DATA STORED AT
          EMO                        ;EMO AGAINST DATA STORED AT WBUF
          WBUF                        ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
          1024.                        ;ERRORS.
          SCOPE
*****
T27: 27                                ;ROUTINE NUMBER 27 *
      T30                                ;ADDRESS OF NEXT ROUTINE *
      20.                                ;TEST ITERATION COUNT *
      BUA                                ;SCOPE ENTRY POINT *
      .LIST
*****
;4 BLOCK TRANSFER TEST. WDATAF, RDATAF. WRITE AND READ BUFFERS ARE
;COMBINED AND USED AS 1024 WORD READ BUFFER. WRITE DATA IS TAKEN FROM
;THE PROGRAM CODE ITSELF STARTING AT ADDR EMO.
MOV      #WDATAF,QBINTX             ;SET UP WDATAF.
MOV      #RDATAF,QBINTY            ;SET UP RDATAF.
MOV      #-1,QBINTZ                 ;SET DIFFERENT DIRECTION INDICATOR.
BUA:     JSR      PC,QBINT           ;GO DO 4 BLOCK TRANSFERS.
          DATCKI                    ;CALL DATCKI TO CHECK DATA STORED AT
          EMO                        ;EMO AGAINST DATA STORED AT WBUF+2046.
          WBUF+2046.                 ;CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
          1024.                        ;ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
          SCOPE
*****
T30: 30                                ;ROUTINE NUMBER 30 *
      T31                                ;ADDRESS OF NEXT ROUTINE *
      20.                                ;TEST ITERATION COUNT *
      BVA                                ;SCOPE ENTRY POINT *
      .LIST
*****
;4 BLOCK TRANSFER TEST. WDATAR, RDATAF. WRITE AND READ BUFFERS ARE
;COMBINED AND USED AS 1024 WORD READ BUFFER. WRITE DATA IS TAKEN FROM
;THE PROGRAM CODE ITSELF STARTING AT ADDR EMO.
MOV      #WDATAR,QBINTX             ;SET UP WDATAR.
MOV      #RDATAF,QBINTY            ;SET UP RDATAF.
MOV      #-1,QBINTZ                 ;SET DIFFERENT DIRECTION INDICATOR.
BVA:     JSR      PC,QBINT           ;GO DO 4 BLOCK TRANSFERS.

```



```

2387 012346 104050          DATCKI          ;CALL DATCKI TO CHECK DATA STORED AT
2388 012350 005464          EMO            ;EMO AGAINST DATA STORED AT WBUF+2046.
2389 012352 017346          WBUF+2046.    ;CHECK # OF WORDS SPECIFIED. REPORT ERRORS.
2390 012354 002000          1024.         ;ACTUAL DATA IS CHECKED IN DESCENDING ORDER.
2391 012356 104000          SCOPE        ;SCOPE.
2392
2393 012360 104044          ;
2394 012362 013350          QBINT: CLEAR   ;CLEAR 1024 WORD READ BUFFER.
2395 012364 002000          WBUF          ;
2396 012366 012767 000310 166466 1024.
2397 012374 000000          QBINTX: MOV    #200.,BLKRQ ;200 TO BLOCK REQUIRED.
2398 012376 002000          OPEN        ;WDATAF OR WDATAR.
2399 012400 005464          1024.        ;WRITE 1024 WORDS.
2400 012402 005767 000036  TST          QBINTZ   ;ADDR TO START WRITE FROM.
2401 012406 001412          BEQ          QBINTY   ;DIFF DIR IND SET?
2402 012410 062767 000003 166444  ADD          #3,BLKRQ  ;BR IF NOT.
2403 012416 032777 004000 166362  BIT          #REV,DTCCM ;ASSUME DT GOING FWD. ADD 3 TO BLKRQ.
2404 012424 001403          BEQ          QBINTY   ;DT GOING FWD?
2405 012426 162767 000006 166426  SUB          #6,BLKRQ  ;BR IF YES.
2406 012434 000000          QBINTY: OPEN  ;RDATAF OR RDATAR.
2407 012436 002000          1024.        ;READ 1024 WORDS.
2408 012440 013350          WBUF          ;ADDR TO START READING INTO.
2409 012442 000207          RTS          PC      ;EXIT QBINT SUB.
2410 012444 000000          QBINTZ: OPEN  ;DIFFERENT DIRECTION INDICATOR.
2411
2412 012446 000031          T31: 31      ;ROUTINE NUMBER 31
2413 012450 012472          T32          ;ADDRESS OF NEXT ROUTINE
2414 012452 000001          1           ;TEST ITERATION COUNT
2415 012454 012464          WA         ;SCOPE ENTRY POINT
2416          .LIST
2417
2418          ;*****
2419          ;DOUBLE BLOCK TRANSFER TEST. ALL BLOCKS. DATA ALL 0'S.
2420 012456 012767 013072 000354  MOV          #ZCSB,SBKSBP
2421 012464 004767 000116  WA: JSR      PC,SBKSUB ;GO DO ALL BLOCKS TEST.
2422 012470 104000          SCOPE        ;SCOPE.
2423          ;*****
2424          T32: 32      ;ROUTINE NUMBER 32
2425 012472 000032          T33          ;ADDRESS OF NEXT ROUTINE
2426 012474 012516          1           ;TEST ITERATION COUNT
2427 012476 000001          XA         ;SCOPE ENTRY POINT
2428          .LIST
2429          ;*****
2430          ;DOUBLE BLOCK TRANSFER TEST. ALL BLOCKS. EACH BLOCK IS WRITTEN WITH
2431          ;ITS OWN BLOCK NUMBER.
2432 012502 012767 013042 000330  MOV          #WCSB,SBKSBP
2433 012510 004767 000072  XA: JSR      PC,SBKSUB ;GO DO ALL BLOCKS TEST.
2434 012514 104000          SCOPE        ;SCOPE.
2435          ;*****
2436          T33: 33      ;ROUTINE NUMBER 33
2437 012516 000033          T34          ;ADDRESS OF NEXT ROUTINE
2438 012520 012552          1           ;TEST ITERATION COUNT
2439 012522 000001          YA         ;SCOPE ENTRY POINT
2440          .LIST
2441          ;*****
2442          ;DOUBLE BLOCK TRANSFER TEST. ALL BLOCKS BINARY COUNT PATTERN.
2443 012526 012767 013062 000304  MOV          #YCSBB,SBKSBP

```



```

2443 012534 104051          INBIN          ;INIT BINARY COUNT PATTERN.
2444 012536 104046          BINFIL          ;FILL 512 WORD WRITE BUFFER WITH
2445 012540 013350          WBUF          ;BINARY COUNT PATTERN.
2446 012542 001000          512.
2447 012544 004767 000036 JSR          PC, SBKSUB ;GO DO ALL BLOCKS TEST.
2448 012550 104000          SCOPE          ;SCOPE.
2449          ;*****
2450 012552 000034          T34: 34          ;ROUTINE NUMBER 34 *
2451 012554 013112          T35          ;ADDRESS OF NEXT ROUTINE *
2452 012556 000005          5           ;TEST ITERATION COUNT *
2453 012560 012572          ZA          ;SCOPE ENTRY POINT *
2454          .LIST
2455          ;*****
2456          ;DOUBLE BLOCK TRANSFER TEST. ALL BLOCKS. RANDOM DATA.
2457 012562 012767 013062 000250 MOV          #YCSBB, SBKSBP
2458 012570 104055          INRND          ;INIT RANDOM NUMBER SUB.
2459 012572 104056          RDNFIL          ;FILL 512 WORD WRITE BUFFER WITH
2460 012574 013350          WBUF          ;RANDOM DATA.
2461 012576 001000          512.
2462 012600 004767 000002 JSR          PC, SBKSUB ;GO DO ALL BLOCKS TEST.
2463 012604 104000          SCOPE          ;SCOPE.
2464          ;
2465          ;SUBROUTINE TO WRITE-READ ALL BLOCKS. DOUBLE BLOCK TRANSFERS.
2466          ;ROUTINE WRITES 2 BLOCKS FORWARD, SKIPS 2 BLOCKS, WRITES ANOTHER 2 BLOCKS,
2467          ;ETC., SKIPPED BLOCKS ARE WRITTEN IN REVERSE. SAME PROCEDURE
2468          ;IS USED FOR READING DATA.
2469 012606 012767 000221 166230 SBKSUB: MOV          #145., CTRA ;SET UP TO WRITE 2 BLOCKS 145 TIMES.
2470 012614 012767 177774 166240 MOV          #-4, BLKRQ ;STARTING WITH BLOCK 0.
2471 012622 062767 000004 166232 SBKSUB: ADD          #4, BLKRQ
2472 012630 004777 000204 JSR          PC, SBKSBP ;GO SET UP WRITE BUFFER.
2473 012634 104037          WDATAF          ;CALL WDATAF SUB TO WRITE FWD 512. WORDS
2474 012636 001000          512.          ;STARTING AT ADDR WBUF
2475 012640 013350          WBUF
2476 012642 005367 166176 DEC          CTRA ;WRITTEN 145 TIMES?
2477 012646 001365          BNE          SBKSUB ;BR IF NOT.
2478 012650 012767 000220 166166 MOV          #144., CTRA ;SET UP TO WRITE 2 BLOCKS IN REV 144 TIMES
2479 012656 012767 001103 166176 MOV          #579., BLKRQ ;STARTING WITH BLOCK 575.
2480 012664 162767 000004 166170 SBKSUB: SUB          #4, BLKRQ
2481 012672 004777 000142 JSR          PC, SBKSBP ;GO SET UP WRITE BUFFER.
2482 012676 104040          WDATAR          ;CALL WDATAR SUB TO WRITE REV 512. WORDS
2483 012700 001000          512.          ;STARTING AT ADDR WBUF
2484 012702 013350          WBUF
2485 012704 005367 166134 DEC          CTRA ;WRITTEN 144 TIMES?
2486 012710 001365          BNE          SBKSUB ;BR IF NOT.
2487 012712 012767 000221 166124 MOV          #145., CTRA ;SET UP TO READ 2 BLOCKS FWD 145 TIMES
2488 012720 012767 177774 166134 MOV          #-4, BLKRQ ;STARTING WITH BLOCK 0.
2489 012726 062767 000004 166126 SBKSUB: ADD          #4, BLKRQ
2490 012734 004777 000100 JSR          PC, SBKSBP ;SET WRITE BUFFER WITH CHECK DATA.
2491 012740 104041          RDATAF          ;CALL RDATAF SUB TO READ FWD 512. WORDS
2492 012742 001000          512.          ;AND STORE AT ADDR STARTING AT RBUF
2493 012744 015420          RBUF
2494 012746 104047          DATCHK          ;CALL DATCHK SUB TO CHECK DATA STORED AT
2495 012750 013350          WBUF          ;WBUF AGAINST DATA STORED AT RBUF
2496 012752 015420          RBUF          ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
2497 012754 001000          512.          ;ERRORS.
2498 012756 005367 166062 DEC          CTRA ;READ 145 TIMES?

```



```

2499 012762 001361          BNE      SBKSBC      ;BR IF NOT.
2500 012764 012767 000220 166052      MOV      #144.,CTRA ;SFT UP TO READ 2 BLOCKS IN REV 144 TIMES
2501 012772 012767 001103 166062      MOV      #579.,BLKRQ ;STARTING WITH BLOCK 575.
2502 013000 162767 000004 166054      SBKSBD: SUB      #4,BLKRQ
2503 013006 004777 000026          JSR      PC,SBKSBP ;SET WRITE BUFFER WITH CHECK DATA.
2504 013012 104042          RDATAR          ;CALL RDATAR SUB TO READ REV 512. WORDS
2505 013014 001000          512.          ;STARTING AT ADDR RBUF
2506 013016 015420          RBUF
2507 013020 104047          DATCHK          ;CALL DATCHK SUB TO CHECK DATA STORED AT
2508 013022 013350          WBUF          ;WBUF AGAINST DATA STORED AT RBUF
2509 013024 015420          RBUF          ;CHECK NUMBER OF WORDS SPECIFIED. REPORT
2510 013026 001000          512.          ;ERRORS.
2511 013030 005367 166010      DEC      CTRA      ;READ 144 TIMES?
2512 013034 001361          BNE      SBKSBD      ;BR IF NOT.
2513 013036 000207          RTS      PC        ;DONE. EXIT.
2514 013040 000000          SBKSBP: OPEN      ;WRITE BUFFER SET UP SUB POINTER.
2515          ;SUB TO FILL 512 WORD WRITE BUFFER WITH # OF BLOCK TO BE WRITTEN, AND
2516          ;TO CLEAR 512 WORD READ BUFFER.
2517 013042 016767 166014 000004      WCSB:  MOV      BLKRQ,WCSBA ;BLK# TO WCSBA
2518 013050 104045          FILL          ;FILL WRITE BUFFER WITH DATA IN WCSBA.
2519 013052 013350          WBUF
2520 013054 000000          WCSBA: OPEN
2521 013056 001000          512.
2522 013060 000400          BR      YCSBB      ;GO CLEAR READ BUFFER.
2523          ;SUB TO CLEAR 512 WORD READ BUFFER.
2524 013062 104044          YCSBB: CLEAR      ;CLEAR 512 WORD READ BUFFER.
2525 013064 015420          RBUF
2526 013066 001000          512.
2527 013070 000207          RTS      PC        ;EXIT.
2528          ;SUB TO FILL WRITE BUFFER WITH 0'S AND READ BUFFER WITH 1'S
2529          ZCSB: CLEAR      ;CLEAR 512 WORD WRITE BUFFER.
2530 013074 013350          WBUF
2531 013076 001000          512.
2532 013100 104045          FILL          ;FILL READ BUFFER WITH 1'S.
2533 013102 015420          RBUF
2534 013104 177777          -1
2535 013106 001000          512.
2536 013110 000207          RTS      PC        ;EXIT.
2537          ;*****
2538 013112 000035      t35:  35          ;ROUTINE NUMBER 35
2539 013114 177777          TLAST          ;ADDRESS OF NEXT ROUTINE
2540 013116 000005          5              ;TEST ITERATION COUNT
2541 013120 013122          C1A           ;SCOPE ENTRY POINT
2542          .LIST
2543          ;*****
2544          ;4 BLOCK TRANSFER TEST. ALL BLOCKS. WRITE AND READ BUFFER ARE COMBINED
2545          ;AND USED AS 1024 WORD READ BUFFER. WRITE DATA IS TAKEN FROM THE PROGRAM
2546          ;CODE ITSELF STARTING AT ADDRESS EMO.
2547 013122 012767 000110 165714      C1A:  MOV      #72.,CTRA ;SET UP TO WRITE 4 BLOCKS FWD 72 TIMES.
2548 013130 012767 177770 165724      MOV      #-8.,BLKRQ ;STARTING WITH BLOCK 0.
2549 013136 062767 000010 165716      C1B:  ADD      #8.,BLKRQ
2550 013144 104037          WDATAF          ;CALL WDATAF SUB TO WRITE FWD 1024. WORDS
2551 013146 002000          1024.          ;STARTING AT ADDR EMO
2552 013150 005464          EMO
2553 013152 005367 165666      DEC      CTRA      ;WRITTEN 72 TIMES?
2554 013156 001367          BNE      C1B      ;BR IF NOT.

```



2555	013160	012767	000110	165656		MOV	#72.,CTRA		;SET UP TO WRITE 4 BLOCKS REV 72 TIMES
2556	013166	012767	001107	165666		MOV	#583.,BLKRG		;STARTING WITH BLOCK 575
2557	013174	162767	000010	165660	C1C:	SUB	#8.,BLKRG		
2558	013202	104040				WDATAR			;CALL WDATAR SUB TO WRITE REV 1024. WORDS
2559	013204	002000				1024.			;STARTING AT ADDR EMO
2560	013206	005464				EMO			
2561	013210	005367	165630			DEC	CTRA		;WRITTEN 72 TIMES?
2562	013214	001367				BNE	C1C		;BR IF NOT.
2563	013216	012767	000110	165620		MOV	#72.,CTRA		;SET UP TO READ 4 BLOCKS FWD 72 TIMES.
2564	013224	012767	177770	165630		MOV	#-8.,BLKRG		;STARTING WITH BLOCK 0.
2565	013232	062767	000010	165622	C1D:	ADD	#8.,BLKRG		
2566	013240	104044				CLEAR			;CLEAR 1024 WORD READ BUFFER.
2567	013242	013350				WBUF			
2568	013244	002000				1024.			
2569	013246	104041				RDATAF			;CALL RDATAF SUB TO READ FWD 1024. WORDS
2570	013250	002000				1024.			;AND STORE AT ADDR STARTING AT WBUF
2571	013252	013350				WBUF			
2572	013254	104047				DATCHK			;CALL DATCHK SUB TO CHECK DATA STORED AT
2573	013256	005464				EMO			;EMO AGAINST DATA STORED AT WBUF
2574	013260	013350				WBUF			;CHECK NUMBER OF WORDS SPECIFIED. REPORT
2575	013262	002000				1024.			;ERRORS.
2576	013264	005367	165554			DEC	CTRA		;READ 72 TIMES?
2577	013270	001360				BNE	C1D		;BR IF NOT.
2578	013272	012767	000110	165544		MOV	#72.,CTRA		;SET UP TO READ 4 BLOCKS REV 72 TIMES.
2579	013300	012767	001107	165554		MOV	#583.,BLKRG		;STARTING WITH BLOCK 575.
2580	013306	162767	000010	165546	C1E:	SUB	#8.,BLKRG		
2581	013314	104044				CLEAR			;CLEAR 1024 WORD READ BUFFER.
2582	013316	013350				WBUF			
2583	013320	002000				1024.			
2584	013322	104042				RDATAR			;CALL RDATAR SUB TO READ REV 1024. WORDS
2585	013324	002000				1024.			;STARTING AT ADDR WBUF
2586	013326	013350				WBUF			
2587	013330	104047				DATCHK			;CALL DATCHK SUB TO CHECK DATA STORED AT
2588	013332	005464				EMO			;EMO AGAINST DATA STORED AT WBUF
2589	013334	013350				WBUF			;CHECK NUMBER OF WORDS SPECIFIED. REPORT
2590	013336	002000				1024.			;ERRORS.
2591	013340	005367	165500			DEC	CTRA		;READ 72 TIMES?
2592	013344	001360				BNE	C1E		;BR IF NOT.
2593	013346	104000				SCOPE			;DONE. SCOPE.
2594	013350				WBUF:				
2595		015420							.=WBUF+1064.
2596	015420				RBUF:				.=RBUF+1040.
2597		017440							
2598		000001							.END























MA	011332	2136	2143#											
MACHER	000004	386#	697*											
MAINT =	020000	462#												
MANUAL =	100000	408#												
NA	011376	2153	2160#											
NOINT =	104027	573#	1327	1397	1665	1714	1777	1823	1881	1924	1956	2007	2045	
NOINTR	004046	572	1150#											
NOP =	000240	406#	744	770	855									
NOUNIT	006733	665	1589#											
NXTST	001040	508#	696*	711	736	748	750*							
OA	011442	2170	2177#											
OACNV =	104061	625#	858	865	869	877	881	885	889	893	897	901	1204	1208
		1276	1304											
OACNVA	003312	1009#	1018											
OACNVV	003276	624	1003#											
OACNVX =	003302	1004#	1009	1013*	1014*	1015*	1016*							
OPEN =	000000	407#	503	504	506	507	508	509	510	511	512	513	514	515
		516	517	518	519	520	521	522	523	641	677	909	944	988
		989	990	1133	1166	1182	1205	1209	1224	1229	1254	1379	1380	1389
		1393	1408	1427	1619	1662	2190	2193	2272	2281	2285	2397	2406	2410
		2514	2520											
PA	011542	2201	2208#											
PC =	X:000007	433#	743*	1614*	1616*	1643*	1654*	1656*	1688*	2126*	2143*	2160*	2177*	2196*
		2208*	2225*	2242*	2259*	2284*	2329*	2348*	2367*	2386*	2409*	2420*	2432*	2447*
		2462*	2472*	2481*	2492*	2503*	2513*	2527*	2536*					
PGTIT	005551	631	1457#											
POPSP =	005726	436#												
POPSP2 =	022626	437#	789	800	820	1331	1401	1672	1718	1752	1781	1827	1854	1885
		1928	1970	2011	2049									
PRENOZ	003744	612	1120#											
PRMEND =	104053	613#	1723	1784	1830	1888	1931	1973	2014	2052				
PRTY0 =	000000	445#												
PRTY1 =	000040	444#												
PRTY2 =	000100	443#												
PRTY3 =	000140	442#												
PRTY4 =	000200	441#												
PRTY5 =	000240	440#												
PRTY6 =	000300	439#	500											
PRTY7 =	000340	391	393	397	438#	698								
PSW =	177776	404#	698*	976*										
PTO	003226	985#	989#	992	997*									
PT1	003230	986#	990#	992*	993*	996*	997	999						
PUSH =	005746	434#												
PUSH2 =	024646	435#	782	814										
QA	011610	2218	2225#											
QBINT	012360	2329	2348	2367	2386	2393#								
QBINTX	012374	2326*	2345*	2364*	2383*	2397#								
QBINTY	012434	2327*	2346*	2365*	2384*	2401	2404	2406#						
QBINTZ	012444	2328*	2347*	2366*	2385*	2400	2410#							
RA	011660	2235	2242#											
RALL =	000006	478#												
RBUF	015420	2101	2109	2112	2129	2146	2163	2180	2188	2195	2211	2228	2245	2262
		2269	2283	2303	2310	2313	2493	2496	2506	2509	2525	2533	2596#	2597
RDATA =	000004	477#	1413	1418	1421									
RDATAF =	104041	593#	2107	2125	2176	2206	2257	2308	2327	2384	2491	2569		
RDATAR =	104042	595#	2142	2159	2223	2240	2346	2365	2504	2584				



























ADD	922 2549	926 2565	1008	1011	1042	1043	1050	1342	1388	1941	2022	2277	2402	2471	2489
ASL	761	1385													
BCS	1039														
BEQ	633 1222 1933 1339	656 1266 1938	668 1275 1975	691 1338 1980	710 1341 2276	723 1364 2279	731 1434 2401	741 1637 2404	746 1705	960 1739	1074 1787	1076 1797	1163 1833	1179 1891	1201 1901
BGT	1161	1169	1175	1187	1686										
BHI	639	1010	1013	1070	1088	1167	1183	1319	1365	1425					
BIC	670														
BICB	1071	1316	1367												
BIS	655	667	705	722	726	730	734	906	1073	1075	1162	1178	1317	1340	1346
BIT	1352	1363	1625	1636	1738	1756	2278	2403							
BITB	1433														
BLE	1350														
BLOS	1177	1185	1344	2018	2056										
BMI	658	1094	1101	1107	1140	1732	1841								
BNE	661 981 1369 2554	663 995 1626 2562	685 1018 1677 2577	706 1028 1726 2592	712 1047 1757	720 1053 1944	727 1226 1986	729 1236 2025	735 1246 2063	737 1257 2079	907 1318 2086	913 1347 2477	947 1353 2486	954 1355 2499	979 1361 2512
BPL	647	649	850	965	1136	1333	1403	1720							
BR	642 1086 1378 1935	645 1096 1412 1977	654 1113 1415 2522	666 1129 1417	671 1192 1420	678 1312 1422	686 1322 1428	716 1323 1624	747 1328 1675	768 1335 1711	771 1336 1799	854 1345 1811	962 1351 1835	971 1362 1857	1041 1366 1903
CLR	629	976	1037	1234	1771	2207	2224	2328	2347						
CLRB	1311														
CMP	711 1685	736 1786	912 1832	1160 1890	1168 1937	1174 1979	1176 2017	1184 2055	1186	1200	1274	1337	1343	1349	1676
CMPB	709	959													
COM	721	836	993	994											
DEC	660 1796 2511	684 1899 2553	728 1900 2561	946 1932 2576	978 1943 2591	980 1974	1017 1985	1027 2024	1046 2062	1052 2078	1221 2082	1225 2085	1235 2476	1245 2485	1256 2498
DECB	1368														
EMT	527 557 587 617	529 559 589 619	531 561 591 621	533 563 593 623	535 565 595 625	537 567 597 627	539 569 599	541 571 601	543 573 603	545 575 605	547 577 607	549 579 609	551 581 611	553 583 613	555 585 615
HALT	385	387	389	395	399	401	733	845	851						
INC	638	683	719	996	1040	1223	1324	1424	1712	1795	1902	1934	1976	2075	
INCB	1798	1834													
JMP	491	493	708	725											
JSR	704	743	956	961	968	970	1614	1616	1654	1656	2126	2143	2160	2177	2208
MOV	2225 628 732 766 791 811 837 941 1009 1116	2242 635 740 767 792 812 839 942 1023 1117	2259 637 748 770 793 813 843 945 1024 1128	2329 650 749 773 794 815 853 952 1025 1131	2348 651 750 774 795 816 855 975 1033 1170	2367 669 751 775 798 819 857 977 1034 1188	2386 674 752 776 801 821 863 984 1035 1191	2420 675 753 777 802 822 905 985 1036 1193	2432 679 754 778 803 823 911 986 1048 1195	2447 695 755 779 804 824 919 992 1049 1196	2462 696 758 780 805 828 923 997 1064 1197	2472 697 759 781 806 829 929 999 1065 1198	2481 698 760 784 807 830 930 1005 1069 1199	2490 699 762 785 808 834 936 1006 1072 1202	2503 719 765 790 810 835 937 1007 1115 1203







TC3 - TC11 TEST 3 MACY11 27(732) 10-SEP-76 15:51 PAGE 66  
DZTCCA.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

ERRORS DETECTED: 0  
DEFAULT GLOBALS GENERATED: 0

\*DZTCCA,DZTCCA.SEG/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DZTCCA.P11  
RUN-TIME: 9 17 5 SECONDS  
RUN-TIME RATIO: 78/33=2.3  
CORE USED: 12K (23 PAGES)



