

TU58

TU58 PERF EXERCISER
CZTUUA0

AH-E649A-MC

COPYRIGHT 1979
FICHE 1 OF 1

SEP 1979

digital
MADE IN USA

.REM 8

IDENTIFICATION

PRODUCT CODE: AC-E648A-MC
PRODUCT NAME: CZTUUA0 TU58 PERF EXER
PRODUCT DATE: 17-APR-79
MAINTAINER: PERIPHERAL DIAGNOSTIC GROUP

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERROR THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1979 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL
DEC

PDP
DECUS

UNIBUS
DECTAPE

MASSBUS

TABLE OF CONTENTS

- 1.0 GENERAL INFORMATION
 - 1.1 PROGRAM ABSTRACT
 - 1.2 SYSTEM REQUIREMENTS
 - 1.3 RELATED DOCUMENTS AND STANDARDS
 - 1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
 - 1.5 ASSUMPTIONS
- 2.0 OPERATING INSTRUCTIONS
 - 2.1 HOW TO RUN THIS DIAGNOSTIC
- 3.0 ERROR INFORMATION
- 4.0 PERFORMANCE AND PROGRESS REPORTS
- 5.0 DEVICE INFORMATION TABLES
- 6.0 TEST SUMMARIES

1.0 GENERAL INFORMATION

THIS DIAGNOSTIC EXERCISES FROM 1 TO 8 TU58 CONTROLLER BOARDS, EACH OF WHICH MAY SUPPORT 1 OR 2 DRIVES. THE PROGRAM IMPLEMENTS THE 'MAINTENANCE MODE' SWITCH WITHIN ALL PACKET COMMANDS, THUS RETRIEVING MAXIMUM INFORMATION FROM THE DEVICE UPON CERTAIN DEVICE RECOGNIZED ERRORS.

STATISTICAL SUMMARIES ARE PROVIDED FOR ALL UNITS TESTED. RETRIES ARE PERFORMED ON DATA-RELATED ERROR CONDITIONS.

USE OF LOOP ON ERROR FLAG (:LOE) IS IMPLEMENTED BUT NOT RECOMMENDED FOR USE, SINCE THE LOOPS ARE QUITE LENGTHLY DUE TO COMMUNICATIONS PROTOCOL OVERHEAD.

1.1 PROGRAM ABSTRACT

IN ORDER TO EXERCISE MULTIPLE UNITS IN AN EFFICIENT MANNER, A SCHEDULING ALGORITHM BUILDS, THEN SENDS THE NEXT COMMUNICATION PACKET (COMMAND OR DATA) FORMULATED BY EXECUTING MACRO CODE WITHIN THE TEST ALGORITHMS. THE USE OF MACROS TO IMPLEMENT THE COMMUNICATIONS PROTOCOL SIMPLIFIES CONTEXT SWITCHING FROM UNIT TO UNIT BY NOT REQUIRING 8 SEPARATE DEVICE STACKS IN ADDITION TO THE SYSTEM STACK. COMPLETE INDEPENDENCE OF UNITS AND TESTS WAS ALSO ACHIEVED: FOR INSTANCE, UNIT 0 COULD BE PERFORMING TEST 6, WHILE UNIT 7 WAS PUT ON LINE AND STARTED ON TEST 1, OR UNIT 5 ABORTED AND WAS TAKEN OFF LINE, ETC... FOLLOWING CONVERSION TO RUN UNDER THE DIAGNOSTIC SUPERVISOR MONITOR, THIS CAPABILITY WAS DEFEATED. ALL UNITS NOW PERFORM THE TEST ALGORITHM SPECIFIED ON DRIVE 0'S, THEN REPEAT THE TEST AFTER SWITCHING DRIVES, IF ANY DRIVE '1'S' WERE SELECTED.

FOLLOWING THE TRANSMISSION OF 1 PACKET TO EACH DEVICE (WITH XOFF PRECEDING) THE UNITS ARE POLLED, AND THEIR ENTIRE RESPONSES EVALUATED ROUND ROBIN. IF ANY ERROR INITIATES A RETRY, THE SCHEDULING PROCESS IS MODIFIED TO COMMUNICATE WITH ONLY 1 UNIT UNTIL

COMPLETION OF THE RETRY PROCEDURE. THEN, A RETRY BY ANOTHER UNIT MAY PROCEED, OR THE SYSTEM CONTINUES NORMALLY.

UPON OCCURANCE OF A DEVICE FATAL ERROR, THAT UNIT IS DESCHEDULED (ABORTED) ALLOWING THE REMAINING (IF ANY) TO PROCEED WITH TESTING.

ERROR DESCRIPTIONS:

BLOCK #:	THE RECORD NUMBER (1 PER 512. BYTES) IN LAST COMMAND PACK.
COMMAND:	THE MOST RECENT COMMAND PACKET OP CODE.
EXPCID:	THE DATA PATTERN USED ON WRITE COMMAND AND FOR DATA COMPARE AFTER READ OP.
SUCCESS:	THE SUCCESS CODE RECEIVED IN END PACKET.
PAK SENT:	TYPE OF PACKET JUST SENT (0 FOR DATA; 1 FOR COMMAND)
FLAG RCVD:	FLAG BYTE OF PACKET CURRENTLY BEING CHECKED, OR 1ST BYTE OF RESPONSE.

SINCE IN MAINTENANCE MODE TU58 WILL SEND A BAD DATA PACK WITH A 'DATA CHECK' SUCCESS STATUS IN THE FOLLOWING END PACK, THE HOST WILL, UPON CHECKING THOSE DATA PACK(S), DETERMINE 'BAD DATA' IN PACKET ERROR FIRST, THEN INTERPRET THE SUCCESS CODE TO DIFFERENTIATE A COMMUNICATIONS GLITCH (GOOD SUCCESS) VS. TU DATA CHECK CODE. THIS WOULD SEEM TO RESULT IN TWO 'ERROR' MESSAGES FOR ONE ERROR CONDITION, BUT ONLY THE SECOND ERROR MESSAGE WILL CONTAIN PERTINENT (NOT ZERO) ERROR NUMBER.

THROUGHOUT THE PROGRAM, R5 POINTS TO ONE OF 8 POSSIBLE DATA STRUCTURES CONTAINING STATUS, TEST PARAMETERS, AND STATISTICAL INFORMATION FOR THE CURRENT UNIT. 'START' CLEARS STATISTICS. 'RESTART' AND 'CONTINUE' DO NOT.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE

PDP-11/LSI-11 CPU WITH AT LEAST 16K WORDS OF MEMORY AND CONSOLE DEVICE.

1.2.2 SOFTWARE

THE PROGRAM IS REVISION C DIAGNOSTIC SUPERVISOR COMPATIBLE EXCEPT 'LOOP ON ERROR' CAPABILITY IS NOT RECOMMENDED. CONSULT XXDP+

USERS MANUAL FOR OPERATING INSTRUCTIONS.

1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USERS MANUAL CHQUS

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

NONE

1.5 ASSUMPTIONS

SYSTEM HARDWARE OTHER THAN TU58(S) IS OPERATIONAL.

2.0 OPERATING INSTRUCTIONS

2.1 HOW TO RUN THIS DIAGNOSTIC

ANSWER 'CHANGE HW?' WITH YES INITIALLY TO SET UP HARDWARE CONFIGURATION TABLES. THAT INFORMATION IS:

TU58 CSR - ADDRESS OF RCSR OF DLV-11 OR OTHER INTERFACE BOARD.

VECTOR ADDR. - ADDRESS OF INTERRUPT VECTOR LOCATION.

PDT (PARALLFL) INTERFACE -- IS THE TU58 IN A PDT 11/130,
OR SYSTEM WHOSE BUFFERS ARE:
RCSR
RCDB (AND XMDB)
XMSR

TEST DR0 - YES OR NO

TEST DR1 - YES OR NO

SUBSEQUENT RESPONSES TO 'CHANGE HW?' MAY THEN BE 'NO'.

THE SOFTWARE QUESTIONS ARE AS FOLLOWS:

NUMBER OF BLOCKS: TEST 4-7 -- ONE MAY SELECT A MINIMUM OF 8, TO
A MAXIMUM OF 512 BLOCKS TO WRITE,
READ; WRITE VERIFY; AND READ REDUCED,
AS EXPLAINED ELSEWHERE IN THIS DOCUMENT.

ADD DR # TO DATA PATTERN -- FOR THOSE SAME READ AND WRITE TESTS
4-7, THE DRIVE NUMBER (0 OR 1) MAY

BE ADDED TO DATA WRITTEN ON TAPE TO
INSURE DRIVE SELECT BIT OPERATION.

STATISTICS PRINTED AT EOP -- SELECTS WHETHER OR NOT TO PRINT
INFORMATION AT END OF PASS OR ^C.
THESE STATISTICS MAY ALSO BE RE-
TRIEVED WITH THE 'PRI' COMMAND.

COMPARE DATA ON READ -- SELECTS WHETHER OR NOT TO DO A
DATA COMPARE ON DATA PACKETS RE-
CEIVED.

PRINT PACKET ON ERROR -- PRINTS 132. BYTE DATA PACKET ON A COMPARE
ERROR, IF SELECTED.

ERRORS-DVC FATAL IF 'EVL' SET -- IF USER SETS EVL FLAG (EVALUATE)
MODE), HRD OR SFT ERROR MESSAGES
BECOME DVC FTL ERRORS AFTER WHEN
THE NUMBER SPECIFIED IS EXCEEDED.

3.0 ERROR INFORMATION

ERROR INFORMATION IS PROVIDED ON OCCURRENCE OF ERRORS AS OUTLINED IN
SECTION 1.1.

4.0 PERFORMANCE AND PROGRESS REPORTS

STATISTICS ARE AVAILABLE PER SECTION 1.1 AT END OF PASS, CONTROL-C, OR
UPON ENTERING A 'PRI' COMMAND. THEY CONSIST OF # BLOCKS WRITTEN AND READ, # OF
DATA ERRORS, HARD OR SOFT.

5.0 DEVICE INFORMATION TABLES

CONSULT SECTION SUBTITLED 'DATA BLOCK FORMAT' FURTHER ON IN THIS LISTING.

6.0 TEST SUMMARIES

INIT: INIT IS SENT TO DEVICE IF:

- OR
1. INIT CODE IN SUPERVISOR IS EXECUTED
 2. INIT IS REQUESTED BY DEVICE AS A RESULT OF
ERROR.

TEST 1: INITIATES FIRMWARE DIAGNOSTICS AT DEVICE LEVEL (SELF TEST)

TEST 2: SEEK TEST. SEEKS BOT ON BOTH TRACKS, THEN
VERIFIES 60 IPS OPERATION TO SEEK EOT ON
ON BOTH TRACKS, ENDING THEN AT BOT.

TEST 3: PERFORMS WRITE, THEN READ OF ADJACENT BLOCKS AT BOT
WITH VARYING DATA, THEN SEEKS HALF WAY INTO REMAINING
TAPE AND REPEATS THE ABOVE UNTIL EOT.

TESTS 4-7: READS OR WRITES BLOCK # AS DATA INTO SUCCESSIVE BLOCKS
ON TAPE, THE LENGTH OF WHICH IS DETERMINED BY SOFTWARE
QUESTION #1: DEFAULT IS SHORT TAPE (8.) MINIMUM,
(8.) RESULTS IN TRANSFER OF 8. (OR 4 PER TRACK) 512. BYTE
BLOCKS OF DATA PER READ (OR WRITE) OPERATION. THE
ALGORITHM SWITCHES TRACKS REGARDLESS OF THE NUMBER
BLOCKS SELECTED. DRIVE NUMBER IS ADDED TO RECORD
AS DEFAULT, SO FOR TAPE INTERCHANGE
TESTING, ANSWER (N) TO SOFTWARE (SW) QUESTION #2.

NOTE: THE AMOUNT OF TIME SPENT IN TESTS 4-7 IS QUITE
LONG IF THE FULL TAPE (512.) IS SELECTED.

TEST 4: WRITE TAPE

TEST 5: READ TAPE

TEST 6: WRITE VERIFY TAPE

TEST 7: READ REDUCED TAPE

```
355 .TITLE PROGRAM HEADER AND TABLES
356 .SBTTL PROGRAM HEADER
382
384 .ENABL ABS,AMA
385 002000 = 2000
387
388 002000 BGNMOD
389
390
391 :++
392 : THE PROGRAM HEADER IS THE INTERFACE BETWEEN
393 : THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
394 :--
395 002000 POINTER BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU,
396
404
405 002000 HEADER CZTUU,A,0,3600.,1
002000
002000 103
002001 132
002002 124
002003 125
002004 125
002005 000
002006 000
002007 000
002010
002010 101
002011
002011 060
002012
002012 000000
002014
002014 007020
002016
002016 035752
002020
002020 036130
002022
002022 002172
002024
002024 002204
002026
002026 036522
002030
002030 000000
002032
002032 000000
002034
002034 000001
002036
002036 000000
002040
002040 002152
002042
002042 000000
002044
```

```
L$NAME::
        .ASCII /C/
        .ASCII /Z/
        .ASCII /T/
        .ASCII /U/
        .ASCII /U/
        .BYTE 0
        .BYTE 0
        .BYTE 0
L$REV::
        .ASCII /A/
L$DEPO::
        .ASCII /O/
L$UNIT::
        .WORD 0
L$TIML::
        .WORD 3600.
L$HPCP::
        .WORD L$HARD
L$SPCP::
        .WORD L$SOFT
L$HPTP::
        .WORD L$HW
L$SPTP::
        .WORD L$SW
L$LADP::
        .WORD L$LAST
L$STA::
        .WORD 0
L$CO::
        .WORD 0
L$DTYP::
        .WORD 1
L$APT::
        .WORD 0
L$DTP::
        .WORD L$DISPATCH
L$PRIO::
        .WORD 0
L$ENVI::
```

PROGRAM HEADER AND TABLES
PROGRAM HEADER

MACRO M1110 12-JUN-79 12:23 PAGE 3-1¹

SEQ 0008

002044 000000
002046
002046 000000
002050
002050 003
002051 003
002052
002052 000000
002054 000000
002056
002056 000000
002060
002060 002220
002062
002062 002612
002064
002064 000000
002066
002066 000000
002070
002070 004722
002072
002072 004576
002074
002074 000000
002076
002076 002122
002100
002100 104035
002102
002102 000000
002104
002104 003626
002106
002106 004564
002110
002110 004402
002112
002112 002142
002114
002114 000000
002116
002116 000000
002120
002120 000000

L\$EXP1:: .WORD 0
L\$MREV:: .WORD 0
 .BYTE C\$REVISION
 .BYTE C\$EDIT
L\$EF:: .WORD 0
 .WORD 0
L\$SPC:: .WORD 0
L\$DEVP:: .WORD L\$DVTYP
L\$REPP:: .WORD L\$RPT
L\$EXP4:: .WORD 0
L\$EXP5:: .WORD 0
L\$AUT:: .WORD L\$AU
L\$DUT:: .WORD L\$DU
L\$LUN:: .WORD 0
L\$DESP:: .WORD L\$DESC
L\$LOAD:: EMT E\$LOAD
L\$ETP:: .WORD 0
L\$ICP:: .WORD L\$INIT
L\$CCP:: .WORD L\$CLEAN
L\$ACP:: .WORD L\$AUTO
L\$PRT:: .WORD L\$PROT
L\$TEST:: .WORD 0
L\$DLY:: .WORD 0
L\$HIME:: .WORD 0

L\$DESC:: .ASCIZ /TU58 PERF EXER/
 .EVEN

406
407

002122
002122
002122 124 125 065
DESCRIP <TU58 PERF EXER>

409 002142
002142
410 002142 000000
411 002144 177777
412 002146 177777
413 002150

BGNPROT

.WORD 0
.WORD -1
.WORD -1

;DEVICE CSR
;NO MASS BUS
;NO DRIVE

L\$PROT::

ENDPROT

414
420
421
422
423
424
425
426
427

.SBTTL DISPATCH TABLE

;++
; THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
; IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
;--

428 002150
002150 000007
002152
002152 010500
002154 010670
002156 011130
002160 012522
002162 013472
002164 014252
002166 015222

DISPATCH 7

.WORD 7
L\$DISPATCH::
.WORD T1
.WORD T2
.WORD T3
.WORD T4
.WORD T5
.WORD T6
.WORD T7

429

437
 438
 439
 440
 441
 442
 443
 444
 445
 446
 447
 448
 449
 450
 451
 457
 458

002170
 002170 000004
 002172
 002172
 002172 176500
 002174 000300
 002176 000003
 002200 000000
 002202
 002202

.SBTTL DEFAULT HARDWARE P-TABLE

;++
 ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
 ; THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
 ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES.
 :--

BGNHW DFPTBL

.WORD L10001-L\$HW/2
 L\$HW::
 DFPTBL::

.WORD 176500
 .WORD 300
 .WORD 3
 .WORD 0

;CSR ADDRESS
 ;VECTOR ADDR.
 ;TEST DRIVE ZERO AND ONE
 ;NOT PDT TYPE INTERFACE

ENDHW

L10001:

```

460      .SBTTL  SOFTWARE P-TABLE
461
462      ;++
463      ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
464      ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
465      ;--
466
467      002202      BGNSW   SFPTBL
         002202      000006
         002204
         002204
                                L$SW::      .WORD  L10002-L$SW/2
                                SFPTBL::
468
469      002204      000010
470      002206      000001
471      002210      000001
472      002212      000001
473      002214      000001
474      002216      000001
475
482
483      002220
         002220
                                L10002:
484
485      002220
                                ENDSW
                                ENDMOD

```

```

LENGTH: .WORD 8.
STAEOP: .WORD 1
PRBUF: .WORD 1
CMPDAT: .WORD 1
DRVCHK: .WORD 1
EVLTHR: .WORD 1

;TAPE LENGTH
;PRINT STATISTICS AT EOP
;PRINT DATA BUF ON COMP. ERROR
;COMPARE DATA
;ADD DR # TO DATA
;THRESHOLD FOR EVL TEST

```

498
499
527
537
538 002220
539
540
541
542
543
544
545 002220

.TITLE GLOBAL AREAS
.SBTTL GLOBAL EQUATES SECTION

BGNMOD

;++
: THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
: ARE USED IN MORE THAN ONE TEST.
:--

EQUALS

:
: BIT DIFINITIONS
:

100000	BIT15== 100000
040000	BIT14== 40000
020000	BIT13== 20000
010000	BIT12== 10000
004000	BIT11== 4000
002000	BIT10== 2000
001000	BIT09== 1000
000400	BIT08== 400
000200	BIT07== 200
000100	BIT06== 100
000040	BIT05== 40
000020	BIT04== 20
000010	BIT03== 10
000004	BIT02== 4
000002	BIT01== 2
000001	BIT00== 1

001000	BIT9== BIT09
000400	BIT8== BIT08
000200	BIT7== BIT07
000100	BIT6== BIT06
000040	BIT5== BIT05
000020	BIT4== BIT04
000010	BIT3== BIT03
000004	BIT2== BIT02
000002	BIT1== BIT01
000001	BIT0== BIT00

:
: EVENT FLAG DEFINITIONS
: EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION
:

000040	EF.START== 32.	: START COMMAND WAS ISSUED
000037	EF.RESTART== 31.	: RESTART COMMAND WAS ISSUED
000036	EF.CONTINUE== 30.	: CONTINUE COMMAND WAS ISSUED
000035	EF.NEW== 29.	: A NEW PASS HAS BEEN STARTED
000034	EF.PWR== 28.	: A POWER-FAIL/POWER-UP OCCURRED

:
: PRIORITY LEVEL DEFINITIONS
:

000340	PRI07== 340
000300	PRI06== 300

000240	PRI05== 240
000200	PRI04== 200
000140	PRI03== 140
000100	PRI02== 100
000040	PRI01== 40
000000	PRI00== 0
	;
	OPERATOR FLAG BITS
	;
000004	EVL== 4
000010	LOT== 10
000020	ADR== 20
000040	IDU== 40
000100	ISR== 100
000200	UAM== 200
000400	BOE== 400
001000	PNT== 1000
002000	PRI== 2000
004000	IXE== 4000
010000	IBE== 10000
020000	IER== 20000
040000	LOE== 40000
100000	HOE== 100000

546

GLOBAL AREAS MACRO M1110 12-JUN-79 12:23 PAGE 8
GLOBAL DATA SECTION

8 2

SEQ 0014

559
560
561
568
569

.SBTTL GLOBAL DATA SECTION

578 .SBTTL GLOBAL TEXT SECTION
579
580
581 :
582 : NAMES OF DEVICES SUPPORTED BY PROGRAM
583 :
002220 DEVTYP <TU58 CONTROLLER>
002220
002220 124 125 065

LSDVTYP::
.ASCIZ /TU58 CONTROLLER/
.EVEN

584
596
597
615
622
623

```

625 .SBTTL GLOBAL SUBROUTINES SECTION
626 .MACRO RET
627
628 .ENDM
629
630 .MACRO PUSH ,REG
631
632 .ENDM
633 .MACRO POP,REG
634
635 .ENDM
636
637
638
639 :++
640 : THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES THAT
641 : ARE USED TO LINK THE DIAGNOSTIC TO THE SUPERVISOR (TSTID).
642 :--
643
644 :++
645 : SWAPDR
646 : SUBROUTINE TO DETERMINE IF TO TEST OTHER DRIVE (FOR ALL UNITS)
647
648
649
650
651
652
653
654 : INPUTS: NONE
655
656 : OUTPUTS: DR(R5) UPDATED TO TEST SAME OR OTHER DRIVE
657 : CARRY SET IF SECOND PASS NECESSARY
658
659
660 :--
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716 002240 005002 SWAPDR:: CLR R2 ;FOR # DRIVE 1'S.
717 002242 012737 004724 002340 MOV #BLKTBL,SWPTR
718 002250 017705 000064 1$: MOV @SWPTR,R5
719 002254 032715 100000 BIT #BIT15,R5 ;ABORTED?
720 002260 001013 BNE 3$ ;YES
721 002262 032765 000001 000060 BIT #BIT0,DR(R5) ;DID DR. 0?
722 002270 001007 BNE 3$ ;NO, DIDN'T; NO ZERO
723 002272 032765 001000 000060 BIT #BIT9,DR(R5) ;YES; 1 SELECTED?
724 002300 001403 BEQ 3$ ;NO
725 002302 105265 000060 INCB DR(R5) ;YES, SWAP
726 002306 005202 INCB R2 ;ONE MORE TO TEST
727 002310 023727 002340 004742 3$: CMP SWPTR,#LSTDEV ;LAST DEVICE?
728 002316 103004 BHIS 4$ ;YES
729 002320 062737 000002 002340 ADD #2,SWPTR ;NO-POINT NEXT
730 002326 000750 BF 1$ ;DO
731
732 002330 005702 4$: TST R2 ;((CLEAR CARRY),MORE TO DO?
733 002332 001401 BEQ 5$ ;NO
734 002334 000261 SEC ;YES
735 002336 5$: RET ;RETURN
736
737 002340 000000 SWPTR: .WORD
738
739 :++
740 : SETDR - SUBROUTINE TO GET DRIVE FOR 1ST PASS FOR EACH TEST

```

```

741
742      : INPUTS:      NONE
743      :
744      : OUTPUTS:     DR(R5) IS SET TO TEST DRIVE 0 OR DRIVE 1
745      :--
746
747
748 002342 012737 004724 002416 SETDR:: MOV      #BLKTBL,SETPTR
749 002350 017705 000042 1$:      MOV      @SETPTR,R5
750 002354 105065 000060      CLRB      DR(R5)      ;PRESET
751 002360 032765 000400 000060      BIT      #BIT8,DR(R5) ;DO DR0?
752 002366 001002      BNE      2$      ;YES
753 002370 105265 000060      INCB      DR(R5)      ;NO-USE DRIVE 1
754 002374 023727 002416 004742 2$:      CMP      SETPTR,#LSTDEV
755 002402 103004      BHIS      3$
756 002404 062737 000002 002416      ADD      #2,SETPTR
757 002412 000756      BR       1$
758 002414      3$:      RET
759 002416 000000      SETPTR: .WORD
760
761      :++
762      : CLRALL - CLEARS INPUT BUFFER FOR RESPONSE FROM UNIT.
763      : INPUTS:  NONE
764      : OUTPUTS: ALL UNITS BUFFERS CLEARED.
765      :--
766
767 002420 012737 004724 002512 CLRALL:: MOV      #BLKTBL,CLRPTR ;INIT
768 002426 017705 000060 1$:      MOV      @CLRPTR,R5 ;GET DATA BLOCKS
769 002432 004737 002460      CALL      CLRBUF ;CLEAR IT'S RECEIVE BUFFER
770 002436 023727 002512 004742      CMP      CLRPTR,#LSTDEV ;LAST DEV?
771 002444 103004      BHIS      2$      ;YES
772 002446 062737 000002 002512      ADD      #2,CLRPTR ;-->NEXT
773 002454 000764      BR       1$      ;CONTINUE
774 002456      2$:      RET
775
776
777      :++
778      : CLRBUF - CLEARS 1 UNIT'S INPUT BUFFER.
779      : INPUTS:  RCVBUF(R5) IS BUFFER START
780      : OUTPUTS: CLEARED AREA.
781      :--
782
783 002460      CLRBUF:: PUSH      R0
784 002462      PUSH      R4
785 002464 016500 000102      MOV      RCVBUF(R5),R0 ;GET ADDRESS OF BUFFER
786 002470 012704 001036      MOV      #RCBFSZ,R4 ;SIZE IN BYTES
787 002474 005020 1$:      CLR      (R0)+ ;CLEAR IT
788 002476 162704 000002      SUB      #2,R4 ;2 BYTES LESS
789 002502 001374      BNE      1$      ;MORE
790 002504      POP      R4
791 002506      POP      R0
792 002510      RET
793 002512 000000      CLRPTR: .WORD

```

```

795
796
797
798
799
800
801
802 002514 005037 010140
803 002520 012737 004724 010142
804 002526 017705 005410
805 002532 013765 010144 000020
806 002540 023727 010142 004742
807 002546 103004
808 002550 062737 000002 010142
809 002556 000763
810 002560
811
812
813
814
815
816
817
818 002562 004737 010162
819
820 002566 005737 010140
821 002572 001006
822 002574 004737 016354
823
824 002600
      002600 104422
825
826 002602 004737 017530
827 002606 000765
828 002610
829
830 002612
831

```

```

;--
; **
; SETUP - CALLED WITHIN EACH TEST TO INSERT BEGINNING ADDRESS OF THE
; TEST INTO ALL UNITS TEST PC'S.
; INPUTS: TSTTOP LOADED WITH TEST ALGORITHMS STARTING ADDR.
; OUTPUTS: TSTPC(R5) FOR ALL UNITS
;--
; **
SETUP:: CLR      DONE          ;NOT DONE YET
        MOV      #BLKTBL, IDPTR ;TABLE TOP ADDR
1$:      MOV      @IDPTR, R5     ;DEVICE'S DATA BLOCK
        MOV      TSTTOP, TSTPC(R5) ;INSERT PC FOR TOP OF TEST
        CMP      IDPTR, #LSTDEV ;ALL UNITS SET?
        BHS      2$            ;YES
        ADD      #2, IDPTR      ;NO, GET NEXT POINTER
        BR       1$            ;SET HIM UP
2$:      RET                    ;DONE
;--
; **
; RUN - IMPLEMENTS THE CALLS TO SEND PACKETS, RECEIVE PACKETS, THEN
; CHECK ANSWERS DURING TEST RUN TIME.
; INPUTS: NONE
; OUTPUTS: NONE
;--
; **
RUN::   CALL      NXTST          ;MAKE AND SEND NEXT PACK TO ALL
        TST       DONE          ;UNABORTED UNITS
        BNE      2$            ;COMPLETE?
        CALL      GETANS        ;YES
        ;NO, GET ALL RESPONSES
        BREAK      ;SUPERVISOR CHECK
        TRAP      C$BRK
2$:      CALL      CHKANS        ;CHECK ALL RESPONSES
        BR       RUN           ;CONTINUE TILL DONE
        RET
ENDMOD

```

844
845
873
874 002612
875

.TITLE MISCELLANEOUS SECTIONS
.SBTTL REPORT CODING SECTION

BGNMOD

876
877
878
879
880

THE REPORT CODING SECTION CONTAINS THE
'PRINTS' CALLS THAT GENERATE STATISTICAL REPORTS.

881 002612
002612
882 002612
883 002614
884 002616
885 002620
886 002622
887 002624
888

EUNRPT

LSRPT::

PUSH R0
PUSH R1
PUSH R2
PUSH R3
PUSH R4
PUSH R5

889 002626

BREAK

002626 104422
890 002630 012737 004724 003240

MOV #BLKTBL,RPTR ;GET 1ST DEVICE BLOCK
PRINTS #STATHD ;HEADER

TRAP C\$BRK

891 002636

MOV #STATHD,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C\$PNTS
ADD #4,SP

002636 012746 003242
002642 012746 000001
002646 010600
002650 104416
002652 062706 000004

892 002656

BREAK ;^C CHECK

002656 104422

TRAP C\$BRK

893 002660

PRINTS #STHD2 ;2ND HEADER

002660 012746 003516
002664 012746 000001
002670 010600
002672 104416
002674 062706 000004

MOV #STHD2,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C\$PNTS
ADD #4,SP

894 002700

1\$: BREAK ;^C CHECK

002700 104422

TRAP C\$BRK

895 002702

MOV @RPTR,R5 ;GET DEVICE BLOCK
BIT #BIT11,@R5 ;UNIT NOT TESTED?
BNE 2\$;TRUE, DON'T PRINT STATISTICS
;OK TO PRINT

896 002706 032715 004000

897 002712 001131

898

899 002714 011537 003236

MOV @R5,RLUN ;SAVE STATUS WORD
BIC #177770,RLUN ;MASK UNIT NUM.
MOVB SOFTR(R5),R1 ;SOFTREAD
BIC #177400,R1 ;SIGN-UNEXTEND
MOVB SOFTW(R5),R2 ;SOFT WRITE

900 002720 042737 177770

003236

901 002726 116501 000122

902 002732 042701 177400

903 002736 116502 000124

904 002742 042702 177400

905 002746 116503 000136

906 002752 042703 177400

907 002756 116504 000140

908 002762 042704 177400

909 002766

BIC #177400,R2
MOVB HARDR(R5),R3 ;HARD READ
BIC #177400,R3
MOVB HARDW(R5),R4 ;HARD WRITE
BIC #177400,R4
PRINTS #FMO,RLUN ;SUMMARY/UNIT #

MOV RLUN,-(SP)
MOV #FMO,-(SP)
MOV #2,-(SP)
MOV SP,R0

002766 013746 003236

002772 012746 003354

002776 012746 000002

003002 010600

MISCELLANEOUS SECTIONS
REPORT CODING SECTION

MACRO M1110 12-JUN-79 12:23 PAGE 12-1

H 2

SEQ 0020

910	003004	104416	000006						TRAP	C\$PNTS
	003006	062706							ADD	#6,SP
	003012			PRINTS	#FM,#0,WRTN0(R5),RDNO(R5),<B,BDATA(R5)>,R1,R2,R3,R4				MOV	R4,-(SP)
	003012	010446							MOV	R3,-(SP)
	003014	010346							MOV	R2,-(SP)
	003016	010246							MOV	R1,-(SP)
	003020	010146							CLR	-(SP)
	003022	005046							BISB	BDATA(R5),(SP)
	003024	156516	000134						MOV	RDNO(R5),-(SP)
	003030	016546	000114						MOV	WRTN0(R5),-(SP)
	003034	016546	000110						MOV	#0,-(SP)
	003040	012746	000000						MOV	#FM,-(SP)
	003044	012746	003372						MOV	#11,-(SP)
	003050	012746	000011						MOV	SP,R0
	003054	010600							TRAP	C\$PNTS
	003056	104416							ADD	#24,SP
911	003060	062706	000024							
911	003064	116501	000123	MOVB	SOFT+1(R5),R1	; SAME				
912	003070	042701	177400	BIC	#177400,R1	; AS				
913	003074	116502	000125	MOVB	SOFTW+1(R5),R2	; ABOVE				
914	003100	042702	177400	BIC	#177400,R2	; THIS				
915	003104	116503	000137	MOVB	HARDR+1(R5),R3	; TIME				
916	003110	042703	177400	BIC	#177400,R3	; FOR				
917	003114	116504	000141	MOVB	HARDW+1(R5),R4	; DRIVE				
918	003120	042704	177400	BIC	#177400,R4	; ONE				
919										
920	003124			PRINTS	#FM,#1,WRTN1(R5),RDN1(R5),<B,BDATA+1(R5)>,R1,R2,R3,R4					
	003124	010446							MOV	R4,-(SP)
	003126	010346							MOV	R3,-(SP)
	003130	010246							MOV	R2,-(SP)
	003132	010146							MOV	R1,-(SP)
	003134	005046							CLR	-(SP)
	003136	156516	000135						BISB	BDATA+1(R5),(SP)
	003142	016546	000116						MOV	RDN1(R5),-(SP)
	003146	016546	000112						MOV	WRTN1(R5),-(SP)
	003152	012746	000001						MOV	#1,-(SP)
	003156	012746	003372						MOV	#FM,-(SP)
	003162	012746	000011						MOV	#11,-(SP)
	003166	010600							MOV	SP,R0
	003170	104416							TRAP	C\$PNTS
	003172	062706	000024						ADD	#24,SP
921	003176	023727	003240	004742	2\$:	CMP	RPTR,#LSTDEV	; ALL UNITS DONE?		
922	003204	103005				BHIS	3\$	; YES		
923	003206	062737	000002	003240		ADD	#2,RPTR	; NO-DO		
924										
925	003214	000137	002700			JMP	1\$	; MORE UNITS		
926										
927	003220				3\$:	POP	R5			
928	003222					POP	R4			
929	003224					POP	R3			
930	003226					POP	R2			
931	003230					POP	R1			
932	003232					POP	R0			
933	003234					ENDRPT				
	003234									
	003234	104425								
934	003236	000000				RLW:	.WORD			
									L10003:	TRAP C\$RPT

MISCELLANEOUS SECTIONS
REPORT CODING SECTION

MACRO M1110 12-JUN-79 12:23 PAGE 12-2

I 2

SEQ 0021

```

935 003240 000000          RPTR:  .WORD
936
937 003242      045      116      045  STATHD: .ASCII  /%N%A      DR BLKS WR  BLKS RD  BDPAK  /
938 003310      104      103      110  .ASCIZ  @DCHK/RD DCHK/WR  DCHK/RD DCHK/WR%N@
939          .EVEN
940 003354      045      101      125  FMO:  .ASCIZ  /%AUNIT %D1%N/
941          .EVEN
942
943 003372      045      101      040  FM:  .ASCII  /%A      %D1%A %D5%A.  %D5%A.  %D3%A.  /
944 003446      045      104      063  .ASCIZ  /%D3%A.  %D3%A.  %D3%A.  %D3%A.%N/
945          .EVEN
946 003516      045      101      040  STHD2: .ASCII  /%A
947 003563      122      105      103  .ASCIZ  /RECOV  RECOV  UNRECOV UNRECOV%N/  /
948          .EVEN
949 003626          ENDMOD
950

```

```

952          .SBTTL  INITIALIZE SECTION
953
954          :++
955          : THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
956          : AT THE BEGINNING OF EACH PASS.
957          :--
958
959 003626          BGNINIT
003626
960
961 003626 000240          INIT:  NOP
962 003630 105037 004374          CLR      STRT          ;FOR STATS CLEAR
963 003634          REDEF  #EF.START      ;START COMMAND?
003634 012700 000040          MOV      #EF.START,R0
003640 104447          TRAP      CSREFG
964 003642          BNCOMPLETE INIT2          ;NO
003642 103002          BCC      INIT2
965 003644 005237 004374          INC      STRT          ;YES, SET START FLAG
966 003650 012737 004724 010130  INIT2:  MOV      #BLKTBL,DEVPTR ;SET ALL UNITS ABORTED:
967 003656 005004          CLR      R4          ;UNIT NUMBER
968 003660 017705 004244          1$:  MOV      @DEVPTR,R5      ;GET POINTER
969 003664 010415          MOV      R4,@R5          ;INSERT UNIT #
970 003666 052715 120000          BIS      #BIT15:BIT13,@R5      ;SET ABORTED, HALTED
971 003672 052715 004000          BIS      #BIT11,@R5          ;SET UNIT NOT TESTED
972 003676 006304          ASL      R4          ;*2 FOR LOOK-UP
973 003700 016465 024310 000102          MOV      BUFTBL(R4),RCVBUF(R5) ;SETUP POINTER TO UNIT'S BUFFER
974 003706 006204          ASR      R4          ;CORRECT BACK TO UNIT #
975 003710 023727 010130 004742          CMP      DEVPTR,#LSTDEV      ;LAST DEVICE DONE?
976 003716 103005          BHS      CHECK          ;YES
977 003720 062737 000002 010130          ADD      #2,DEVPTR          ;NO-GET
978 003726 005204          INC      R4          ;NEXT DEVICE AND
979 003730 000753          BR      1$          ;SERVICE
980
981 003732 022737 000010 002012  CHECK:  CMP      #8,LSUNIT          ;MAKE SURE NOT
982 003740 103005          BHS      GETHRD          ;TOO MANY UNITS
983 003742          ERRSF  0,TOMANY          ;TOMANY-REQUEST ^C
003742 104454          TRAP      CSERSF
003744 000000          .WORD      0
003746 004312          .WORD      TOMANY
003750 000000          .WORD      0
984 003752          DOCLN          ;EXIT
003752 104444          TRAP      C$DCLN
985
986 003754 012737 004724 010130  GETHRD:  MOV      #BLKTBL,DEVPTR ;INIT TABLE POINTER
987 003762 005004          CLR      R4          ;CLEAR DEVICE COUNTER
988 003764 017705 004140          1$:  MOV      @DEVPTR,R5      ;GET STATUS WORD
989 003770 010437 002074          MOV      R4,LSLUN          ;UNIT NUM. IN CASE ERROR
990 003774          GPHARD  R4,R2          ;GET HARD INFO
003774 010400          MOV      R4,R0
003776 104442          TRAP      C$GPHRD
004000 010002          MOV      R0,R2
991 004002          BNCOMPLETE 3$
004002 103105          BCC      3$
992 004004 042715 004000          BIC      #BIT11,@R5          ;UNIT IS TESTED!
993 004010 012203          MOV      (R2)+,R3          ;R3=CSR
994 004012 012265 000204          MOV      (R2)+,TUVECT(R5)      ;GET VECTOR ADDRESS
995 004016 112265 000061          MOV      (R2)+,DR+1(R5)      ;SAVE UNIT SUMMARY

```

MISCELLANEOUS SECTIONS
INITIALIZE SECTION

MACRO M1110 12-JUN-79 12:23 PAGE 13-1

K 2

SEQ 0023

996	004022	005202			INC	R2		;GET TO WORD BOUND		
997	004024	012237	004376		MOV	(R2)+,PDTFLG		;AND GET PDT FLAG		
998	004030	052715	040000		BIS	#BIT14,@R5		;SET SEND BREAK FLAG		
999	004034	032765	000400	000060	BIT	#BIT8,DR(R5)		;DRIVE 0?		
1000	004042	001011			BNE	13\$		;YES		
1001	004044	032765	001000	000060	BIT	#BIT9,DR(R5)		;DRIVE 1?		
1002	004052	001005			BNE	13\$		;OK		
1003	004054				ERRSF	0,NODRVS		;NEITHER?!		
	004054	104454							TRAP	C\$ERSF
	004056	000000							.WORD	0
	004060	004342							.WORD	NODRVS
	004062	000000							.WORD	0
1004	004064									
	004064	104444			DOCLN			;EXIT		
1005									TRAP	C\$DCLN
1006	004066	105737	004374		13\$: TSTB	STRT		;START COMMAND?		
1007	004072	001412			BEQ	14\$		;NO, DONT CLEAR		
1008								;YES-CLEAR STATS		
1009	004074	012702	000202		MOV	#BLKEND,R2		;R2-->END OF STATS		
1010	004100	012701	000110		MOV	#WRTNO,R1		;FORM ADDRESS OF START:		
1011	004104	060501			ADD	R5,R1		;R1-->START OF STATS.		
1012	004106	162702	000110		SUB	#WRTNO,R2		;FORM # TO CLEAR		
1013										
1014	004112	105021			2\$: CLRB	(R1)+		;CLEAR 'EM		
1015	004114	005302			DEC	R2		;MORE?		
1016	004116	001375			BNE	2\$		;YES		
1017	004120	042715	120000		14\$: BIC	#BIT15!BIT13,@R5		;SET NOT ABORTED NOT HALTED		
1018	004124	010365	000022		MOV	R3,RCSR(R5)		;GET DEVICE REGISTERS:		
1019	004130	062703	000002		ADD	#2,R3				
1020	004134	010365	000024		MOV	R3,RCD8(R5)				
1021	004140	062703	000002		ADD	#2,R3				
1022	004144	010365	000026		MOV	R3,XMSR(R5)				
1023	004150	062703	000002		ADD	#2,R3				
1024	004154	105737	004376		TSTB	PDTFLG		;UNIT A PDT?		
1025	004160	001402			BEQ	4\$		;NO		
1026	004162	162703	000004		SUB	#4,R3		;YES...RCDB=XMDB		
1027	004166	010365	000030		4\$: MOV	R3,XMDB(R5)				
1028	004172	005065	000072		CLR	PATTEN(R5)		;ZERO DATA PATTERN		
1029	004176	005065	000002		CLR	RETRY(R5)		;NO RETRIES		
1030	004202	005065	000064		CLR	REC(R5)		;NO RECORD		
1031	004206	005065	000076		CLR	SUCCS(R5)		;NO SUCCESS		
1032	004212	005065	000074		CLR	DLV(R5)		;NO DLV ERROR		
1033	004216	062737	000002	010130	3\$: ADD	#2,DEVPTR		;-->NEXT DEVICE		
1034	004224	005204			INC	R4		;INCREMENT UNIT NUMBER		
1035	004226	020437	002012		CMP	R4,LSUNIT		;MORE UNITS?		
1036	004232	001254			BNE	1\$		;YES, GP HARD THE NEXT		
1037										
1038	004234	005037	010124		CLR	SYSTAT		;SYSTEM STATUS WORD		
1039	004240				RFLAGS	FLGLOC		;GET USER FLAGS		
	004240	104421							TRAP	C\$RFLA
	004242	010037	004400						MOV	K0,FLGLOC
1040	004246	005037	010150		5\$: CLR	BLKER		;NO ERROR		
1041	004252	013737	002204	010126	SETLEN: MOV	LENGTH,TAPLEN		;GET # OF RECORDS		
1042	004260	006237	010126		ASR	TAPLEN		;GET # BLOCKS PER TRACK		
1043	004264	012737	000200	010152	MOV	#200,SECREC		;PRESET SECOND START AT 200		
1044	004272	022737	000200	010126	CMP	#200,TAPLEN		;# BLKS > 128.?		
1045	004300	101003			BHI	3\$		;NO-SWITCH TRACKS 2ND PASS		

1046 004302 012737 000400 010152 MOV #400,SECREC ;YES-START AT 400

1056

1068

1069 004310

004310

004310 104411

3\$: ENDINIT

L10004: TRAP C\$INIT

1070

1071

1072 004312 124 117 117 TOMANY: .ASCIZ /TOO MANY UNITS MAX.=8 /

1073

1074 004342 123 105 114 NODRVS: .ASCIZ /SELECT AT LEAST 1 DRIVE /

1075

1076 004374 000000

STRT:: .WORD

1077 004376 000000

PDTFLG:: .WORD

;TU58 IS IN PDT

1078 004400 000000

FLGLOC:: .WORD

;USER FLAGS

1079

1080

```

1082 004402          BGNAUTO
1083 004402 000240          NOP          ;AUTO DROP ROUTINE
1084 004404          SETVEC #4,#TRPHND,#PRI07 ;GET BUS TRAP VEC.
1085 004404 012746 000340          MOV      #PRI07,-(SP)
1086 004410 012746 004512          MOV      #TRPHND,-(SP)
1087 004414 012746 000004          MOV      #4,-(SP)
1088 004420 012746 000003          MOV      #3,-(SP)
1089 004424 104437          TRAP      C$SVEC
1090 004426 062706 000010          ADD      #10,SP
1091 004432 012737 004724 004510 1$: MOV      #BLKTBL,TRPPTR ;GET TOP OF DATA BLOCK TABLE
1092 004440 017705 000044          MOV      @TRPPTR,R5 ;GET DATA BLOCK
1093 004444 032715 104000          BIT      #BIT15!BIT11,@R5 ;NOT TESTED OR ABORTED?
1094 004450 100403          BMI      2$ ;YES
1095 004452 005775 000022          TST      @RCSR(R5) ;NO-VALID ADDRESS?
1096 004456 000240          NOP          ;YES... (TRAP IF NOT)
1097 004460 023727 004510 004742 2$: CMP      TRPPTR,#LSTDEV ;MORE TO TRY?
1098 004466 103004          BHS      3$ ;NO
1099 004470 062737 000002 004510          ADD      #2,TRPPTR ;ON TO NEXT
1100 004476 000760          BR      1$ ;GET IT
1101 004500          3$: CLRVEC #4 ;RESTORE
1102 004500 012700 000004          MOV      #4,R0
1103 004504 104436          TRAP      C$CVEC
1104 004506          ENDAUTO
1105 004506 104461          L10005: TRAP      C$AUTO
1106 004510 000000          TRPPTR: .WORD
1107          ;ILLEGAL ADDRESS TRAP HANDLER:
1108          TRPHND: PRINTF #M$AUTO ;SAY 'AUTO DROPPED'
1109 004512          MOV      #M$AUTO,-(SP)
1110 004512 012746 004544          MOV      #1,-(SP)
1111 004516 012746 000001          MOV      SP,R0
1112 004522 010600          TRAP      C$PNTF
1113 004524 104417          ADD      #4,SP
1114 004526 062706 000004          MOV      @R5,R0 ;GET UNIT #
1115 004532 011500          BIC      #177770,R0 ;MASK IT OFF
1116 004534 042700 177770          DODU      R0 ;DROP HIM
1117 004540          TRAP      C$DODU
1118 004540 104451          RTI
1119 004542 000002          RTI
1120 004544 045 101 101 M$AUTO: .ASCIIZ /%AAUTO DROP: %N/

```

1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1129
1141
1142

004564
004564
004564 005737 002206
004570 001401
004572 104424

004574
004574
004574 104412

.SBTTL CLEANUP CODING SECTION

;;
; THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
; AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
;--

BGNCLN

TST STAEOP
BEQ 1\$
DORPT

;STATS AT EOP?
;NO
;YES

L\$CLEAN::

TRAP C\$DRPT

1\$: ENDCLN

L10006:

TRAP C\$CLEAN

```

1144          .SBTTL DROP UNIT SECTION
1145
1146          ;++
1147          ; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
1148          ; TO NO LONGER BE TESTED.
1149          ;--
1150
1151 004576          BGNDU
1152 004576
1153 004576          ;RO=UNIT NUMBER
1154 004600          ;SAVE IT
1155 004602 004737 004642          ;SAVE PRESENT UNIT POINTER
1156 004606 052715 120000          ;GET POINTER TO UNIT
1157 004612          ;SET ABORTED, HALTED
1158 004614          ;RESTORE PRESENT UNIT POINTER
1159 004616          ;RETRIEVE UNIT NUMBER
1160          ;
1161          ;
1162          ;
1163          ;
1164          ;
1165          ;
1166          ;
1167          ;
1168          ;
1169          ;
1170          ;
1171          ;
1172          ;
1173          ;
1174          ;
1175          ;
1176          ;
1177          ;
1178          ;
1179          ;
1180          ;
1181          ;
1182          ;
1183          ;
1184          ;
1185          ;
1186          ;
1187          ;
1188          ;
1189          ;
1190          ;
1191          ;

```

1193
1194
1195
1196
1197
1198
1199
1200

.SBTTL ADD UNIT SECTION

+++
: THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
: TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
: TO THE TEST CYCLE.
:--

1201 004722
004722

BGNAU

L\$AU::

1202
1208
1209
1221
1222
1223

1224 004722
004722
004722

ENDAU

L10010: TRAP C\$AU

1225 104452

1228
1272
1273 004724
1285

.TITLE HARDWARE TESTS
BGNMOD
.NLIST ME,BEX

```

1287      .SBTTL  SYSTEM MACRO DEFINITIONS
1288
1289
1290      .MACRO  SWAPIN
1291
1292          MOV    6.(R5),R0
1293          MOV    8.(R5),R1
1294          MOV    10.(R5),R2
1295          MOV    12.(R5),R3
1296          MOV    14.(R5),R4
1297
1298      .ENDM
1299
1300      .MACRO  SWAPOW
1301
1302          MOV    R0,6.(R5)
1303          MOV    R1,8.(R5)
1304          MOV    R2,10.(R5)
1305          MOV    R3,12.(R5)
1306          MOV    R4,14.(R5)
1307
1308      .ENDM
1309
1310      .MACRO  TUWRIT PTRN,REC,BCNT,DR,VER,?A,?B,?C,?D,?E,?F,?G,?H,?T
1311      T:
1312          MOV    #STRBUF,R0      ;MAKE PACKET
1313          MOVB   #RSCMD,@R0      ;COMMAND
1314          MOVB   #R$MSIZ,1(R0)   ;PACKET SIZE
1315          MOVB   #R$SWR,2(R0)    ;OP CODE
1316          MOVB   VER,3.(R0)      ;VERIFY 1 OR 0
1317          MOVB   DR,4.(R0)       ;DRIVE #
1318          MOVB   #020,5.(R0)     ;MAINTENANCE
1319          CLR    6.(R0)          ;NO SEQUENCE #
1320          MOV    BCNT,8.(R0)      ;TOTAL COUNT
1321          MOV    REC,10.(R0)      ;WHICH RECORD
1322          MOV    #R$MSIZ,R1      ;PACKET SIZE PLUS
1323          TST    (R1)+           ;FLAG AND COUNT CORRECTION
1324          MOV    #R$SNSZ,SND CNT(R5) ;PACKET SIZE TO SEND
1325          CALL   CHKSUM          ;R0 --> BUFF
1326          MOV    R1,(R0)         ;PUT CHKSUM IN PACKET
1327          ;SET UP EXPECTATIONS
1328          MOV    #R$CONT,X$FLG(R5) ;FOR RESPONSE
1329          MOV    #1,X$CNT(R5)     ;FROM TU
1330          MOV    #1,X$PKNM(R5)   ;# PACKETS EXPECTED
1331          CALL   RSVP           ;SEND AND RETURN TO SCHEDULER
1332          BIC    #BIT12,@R5      ;FOR LAST TIME
1333          MOV    BCNT,R2         ;GET # OF DATA BYTES
1334          MOV    #STRBUF,R0      ;POINT TO TOP
1335          CMP    R2,#128.        ;START DATA PACKET(S)
1336          BHI    B              ;BCNT > 128.
1337          MOV    R2,R1           ;OTHERWISE OK
1338          BIS    #BIT12,@R5      ;LAST TIME
1339          BR     C              ;USE REMAINING COUNT
1340          B:      MOV    #128.,R1 ;USE 128. BYTES
1341          C:      MOVB   R1,1(R0) ;COPY COUNT TO BUFFER
1342          MOV    R1,R3           ;FOR COUNTER TO LOAD BUFFER
1343          MOVB   #R$DATA,@R0     ;FLAG FIRST
1344          TST    (R0)+           ;SKIP COUNT
1345          D:      MOVB   PTRN,(R0)+ ;R3=CNT    R0 --> WHERE
1346          DEC    R3              ;MORE?
1347          BHI    D              ;YES

```

```

1344      MOV      $STRBUF,R0      ;-->TOP
1345      MOVB     1(R0),R1        ;GET COUNT
1346      BIC      #177400,R1      ;ZERO SIGN EXTEND
1347      MOV      R1,SND CNT(R5)   ;HOW MANY TO SEND PLUS
1348      ADD      #4,SND CNT(R5)   ;FLAG,COUNT,CHKSUM
1349      ADD      #2,R1            ;COMPENSATE FOR FLAG + COUNT
1350      CALL     CHKSUM           ;FOR CHECKSUM CALC.
1351      MOVB     R1,(R0)+         ;CHKSUM INTO PACKET
1352      SWAB     R1              ;EVEN ON A BOUNDARY
1353      MOVB     R1,(R0)+         ;BYTE BOUNDARY
1354      BIT      #BIT12,@R5      ;LAST DATA PACKET?
1355      BEQ      E               ;NO
1356      MOV      #RSEND,X$FLG(R5) ;YES-EXPECT END
1357      MOV      #R$NDSZ,X$CNT(R5) ;OF THIS SIZE
1358      MOV      #1,X$PKNM(R5)    ;AND 1 PACKET
1359      BR       F               ;SEND
1360      E:      MOV      #R$CONT,X$FLG(R5) ;EXPECT 'CONTINUE'
1361      MOV      #1,X$CNT(R5)     ;AND 1 BYTE
1362      MOV      #1,X$PKNM(R5)    ;AND 1 PACKET
1363      F:      CALL     RSVP      ;SEND PACKET
1364      ;AND RETURN TO SCHEDULER
1365      BIT      #BIT10,@R5      ;RETRY?
1366      BNE     G               ;YES
1367      SUB      #128.,R2        ;NO, MORE DATA TO SEND?
1368      BHI     A               ;YES
1369      BR       H               ;NO
1370      G:      TURTRY   REC,BCNT,DR ;RETRY HERE
1371      BIT      #BIT10,@R5      ;RETRY AGAIN?
1372      BNE     G               ;YES
1373      H:      NOP              ;DONE
1374      .ENDM
1375
1376      .MACRO   TUSEEK  REC,DR
1377
1378      MOV      $STRBUF,R0      ;-->XMIT BUFFER
1379      MOVB     #R$CMND,@R0      ;FORM MESSAGE PACK
1380      MOVB     #R$MSIZ,1(R0)    ;THIS BIG
1381      MOVB     #R$SSEK,2(R0)    ;OP CODE IS SEEK
1382      MOV      REC,10.(R0)      ;TO HERE
1383      MOVB     DR,4.(R0)        ;AND WHICH DRIVE
1384      CLRB     3.(R0)          ;NO MODIFIER
1385      CLRB     5.(R0)          ;NO SWITCHES
1386      CLR      6.(R0)          ;NO SEQUENCE #
1387      CLR      8.(R0)          ;NO BYTE COUNT
1388      MOV      #R$MSIZ,R1      ;GET COUNT
1389      TST      (R1)+           ;PLUS FLAG + BCNT
1390      ;FOR CHECKSUM CALC
1391      CALL     CHKSUM           ;RO-->TOP R1=# OF BYTES
1392      MOV      R1,(R0)         ;INSERT INTO PACKET
1393
1394      MOV      #R$SNSZ,SND CNT(R5) ;HOW MANY TO SEND
1395      MOVB     #R$CMND,X$FLG(R5) ;EXPECT END PACK
1396      MOV      #R$NDSZ,X$CNT(R5) ;COUNT WITH THIS
1397      MOV      #1.,X$PKNM(R5)    ;EXPECT ONLY 1 PACKET
1398
1399      CALL     RSVP            ;SEND
1400      ;AND RETURN TO SCHEDULER

```



```

1458      MOV      #R$MSIZ,R1      ;PREPARE
1459      TST      (R1)+            ;SIZE FOR CHECKSUM
1460      MOV      #R$SNSZ,SND CNT(R5) ;SIZE TO SEND
1461
1462      CALL     CHKSUM            ;FORM CHECKSUM
1463      MOV      R1,(R0)          ;INSERT CHECKSUM
1464
1465      MOV      BCNT,R1           ;SET EXPECTATIONS:
1466      ;CALC # OF DATA PACKETS TO EXPECT
1467      MOV      #X$FLG,R3        ;OFFSET
1468      ADD      R5,R3             ;ABS. ADDR. OF X$FLG
1469      CLR      R2               ;PRESET
1470      INC      R2               ;# PACKETS EXPECTED
1471      MOV      #R$DATA,(R3)+     ;LOAD X$FLG
1472      MOV      #132.,(R3)+      ;AND EXPECTED
1473      SUB      #128.,R1         ;NEG RESULT LAST TIME
1474      BLOS     C               ;LAST TIME
1475      BR       A               ;MORE TO DO
1476      ;C:
1477      INC      R2               ;ADD ONE FOR END PACK TO TOTAL
1478      MOV      R2,X$PKNM(R5)    ;SAVE # PACKETS TO EXPECT
1479      MOV      #R$END,(R3)+     ;EXPECT AN END
1480      MOV      #R$NDSZ,(R3)     ;THIS BIG-14. BYTES
1481
1482      CALL     RSVP             ;SEND
1483      ;AND RETURN TO SCHEDULER
1484      D:      BIT      #BIT10,@R5 ;RETRY?
1485      BEQ      B               ;NO.
1486      TURTRY   REC,BCNT,DR      ;YES
1487      BR       D               ;ANOTHER RETRY?
1488      B:      NOP
1489
1490      .ENDM
1491      .MACRO TUSELF
1492
1493      MOV      #STRBUF,R0        ;FORM COMMAND PACKET
1494      MOVB     #R$CMND,@R0
1495      MOVB     #R$MSIZ,1(R0)
1496      MOVB     #R$SSLF,2(R0)
1497      CLRB     3(R0)
1498      CLR      4(R0)
1499      CLR      6(R0)
1500      CLR      8.(R0)
1501      CLR      10.(R0)
1502      MOV      #R$MSIZ,R1        ;FORM SIZES:
1503      TST      (R1)+            ;CHECKSUM
1504      MOV      #R$SNSZ,SND CNT(R5) ;TO SEND
1505      CALL     CHKSUM            ;FORM CHECKSUM
1506
1507      MOV      R1,(R0)          ;INTO PACKET
1508      MOV      #R$END,X$FLG(R5) ;EXPECT END,
1509      MOV      #R$NDSZ,X$CNT(R5) ;THIS BIG
1510      MOV      #1,X$PKNM(R5)    ;AND 1 PACKET
1511      ;SEND
1512      CALL     RSVP             ;RETURN TO SCHEDULER
1513
1514      .ENDM

```

```
1515      .MACRO TSTID  ADDRES,?A
1516
1517      .NLIST
1518      .LIST ME
1519      .LIST
1520
1521      MOV      ADDRES,TSTTOP  ;SAVE ADDR
1522      CALL     SETUP        ;INIT UNITS
1523      CALL     SETDR        ;GET 1ST DRVS.
1524      CALL     RUN          ;DO TEST
1525      CALL     SWAPDR       ;GET NEXT DR.
1526      BCC      A            ;BR NO 2ND DRVS
1527      CALL     SETUP        ;REINIT UNITS
1528      CALL     RUN          ;REPEAT TEST
1529
1530      .NLIST
1531      .LIST ME
1532      .LIST
1533      .ENDM
1534
1535      A:
```

1534		.SBTTL DATA BLOCK FORMAT	
1535		-----	
1536		:R5 --> TOP OF 1 OF THE 8 DATA BLOCKS (1 PER UNIT) DURING EXECUTION	
1537		:@R5 IS THE STATUS WORD CONTAINING:	
1538			:BIT15 = ABORTED
1539			:BIT14 = SEND 'BREAK'
1540			:BIT13 = HALTED
1541			:BIT12 = TEMP STOR WRITE MACRO
1542			:BIT11 = UNIT NOT BEING TESTED
1543			:BIT10 = RETRYING
1544			:BIT9 = TUSB CHKSUM ERROR
1545			:BIT8 = RD/WR OPERATION
1546			:BIT7 = NORMAL/REDUCED THRESHOLD (MACROS)
1547			:BIT6 = HOST DATA COMPARE ERROR
1548			:BIT5 = WR VERIFY OPERATION
1549			:BIT4 = TYPE OF PAK SENT ODATA 1CMD
1550			:BIT3 = NOT USED
1551			:BIT0,1,2=UNIT NO.
1552			
1553			
1554			
1555	000000	STATUS =	0. :DEVICE STATE
1556	000002	RETRY =	2. :# OF RETRIES
1557	000004	ABNDX =	4. :ERROR NUMBER FOR LOG
1558		:R0 =	6. :STORAGE FOR REGISTERS USED IN TEST BODY
1559		:R1 =	8. :STORED WITH SWAPOW
1560		:R2 =	10. :RETRIEVED WITH SWAPIN
1561		:R3 =	12. :
1562		:R4 =	14. :
1563	000020	TSTPC =	16. :POINTER TO NEXT EXECUTABLE TEST INST.
1564	000022	RCSR =	18. :DLV RCV STATUS ADDRESS
1565	000024	RCDB =	20. :DLV RCV DATA ADDRESS
1566	000026	XMSR =	22. :DLV SND STATUS ADDRESS
1567	000030	XMDB =	24. :DLV SND DATA ADDRESS
1568	000032	X\$PKNM =	26. :THE NUMBER OF PACKETS TO RECEIVE
1569	000034	X\$FLG =	28. :THE EXPECTED FLAG OF 1ST PACKET
1570	000036	X\$CNT =	30. :THE EXPECTED COUNT OF 1ST PACKET
1571			:FOR MULTIPLE PACKET RECIEVES (MAX.4)
1572		: .BLKW 8.	:CONSECUTIVE X\$FLGS AND X\$CNTS
1573			
1574	000060	DR =	48. :DR=0 OR 1; BIT8,9 DRIVE SELECTED BY OPERATOR
1575	000062	TRK =	50. :COUNTER FOR TRACK NUMBER
1576	000064	REC =	52. :RECORD (BLOCK #)
1577			
1578	000066	TMP =	54. :TEST MACRO REGISTER
1579	000070	SND CNT =	56. :THE # OF BYTES FOR SENDING PACKET
1580	000072	PATTEN =	58. :DATA PATTERN-LOWER BYTE USED
1581	000074	DLV =	60. :CONTENTS OF RCDB ON DLV ERROR
1582	000076	SUCCS =	62. :SUCCESS CODE OF LAST END PACKET
1583	000100	CMDSNT =	64. :TYPE OF COMMAND CURRENT IN EVEN BYTE; BIT15=VERIFY OP.
1584			
1585	000102	RCVBUF =	66. :POINTER TO 542. BYTE BUFFER (4 DATA PAKS + END PAK)
1586	000104	PKPTR =	68. :POINTER TO TOP OF PACKET
1587	000106	X\$PTR =	70. :POINTER TO CURRENTLY USED X\$FLG OR X\$CNT
1588	000110	WRTNO =	72. :THE # OF 512. BYTE BLOCKS WRITTEN DRO
1589	000112	WRTN1 =	74. :THE # OF 512. BYTE BLOCKS WRITTEN DRI
1590	000114	RDNO =	76. :THE # OF 512. BYTE BLOCKS READ DRO

```

1591      000116      RDN1      =      78.      ;THE # OF 512. BYTE BLOCKS READ DR1
1592
1593      ;AND ERROR LOG      +-----+
1594      ;SPLIT INTO BYTES:      ! DR1 ! DR0 !
1595      ;THE 1ST SECTION OF NEVER FATAL:      +-----+
1596
1597      ;-----
1598
1599      ;OFFSET IN DATA BLOCK      ;ERROR TYPE      ;ERRCODE;MSG CODE;SUC. CODE
1600      ;-----
1601
1602      000120      LGOFST      =      80.      ;*RESERVED*
1603      000122      SOFTR      =      82.      ;SOFT READ      ;SFTRD      ;MSSFRTD      ;ESCKSM
1604      000124      SOFTW      =      84.      ;SOFT WRITE      ;SFTWR      ;MSSFWR      ;ESSKSM
1605      ;      .WORD      ;RECIEVED INIT      ;RCINIT      ;MSRNIT      ;*****
1606      ;      .WORD      ;*RESERVED*
1607
1608      ;THEN THOSE CODES WHICH HAVE N TRIES BEFORE ABORT
1609
1610      000132      T4TRY      =      90.      ;DLV ERROR      ;OVRN      ;MSOVRN      ;*****
1611      000134      BDATA      =      92.      ;BAD DATA      ;BDCOM      ;MSDATA      ;*****
1612      000136      HARDR      =      94.      ;HARD READ      ;HRDRD      ;MSHDRD      ;ESCKSM
1613      000140      HARDW      =      96.      ;HARD WRITE      ;HRDWR      ;MSHDWR      ;ESCKSM
1614      ;      .WORD      98.      ;CHKSM AT HOST      ;BDCHK      ;MSHCHK      ;*****
1615      ;      .WORD      100.      ;SEEK ERROR TOTAL      ;SKERR      ;MSSKER      ;*****
1616      000146      T1TRY      =      102.      ;WRITE PROTECT      ;WRLOCK      ;MSWPRO      ;ESWLOC
1617      ;      .WORD      104.      ;NO MOTOR      ;NOMOT      ;MSNOMO      ;ESNOMO
1618      ;      .WORD      ;CANT INIT      ;CNINIT      ;MSNIT      ;*****
1619      ;      .WORD      ;PARTIAL OP      ;PARTL      ;MSPART      ;ESPART
1620      ;      .WORD      ;NO UNIT      ;NOUNIT      ;MSUNIT      ;ESNONX
1621      ;      .WORD      ;COMMAND ERROR      ;CMNDER      ;MSCMD      ;ESCMD
1622      ;      .WORD      ;BAD RECORD NO.      ;RECERR      ;MSREC      ;ESREC
1623      ;      .WORD      ;SELF TEST ERROR      ;SLFER      ;MSSELF      ;*****
1624
1625      ;      .WORD      ;WRONG SUC.CODE      ;SUCOTL      ;MSWRSP      ;*****
1626      ;      .WORD      ;NO RESPONSE      ;TORCVB      ;MSNRSP      ;*****
1627      ;      .WORD      ;WEIRD FLAG      ;OTL      ;MSQRSP      ;*****
1628      ;      .WORD      ;NO CARTRIDGE      ;NOCART      ;MSNOTP      ;ESNCRT
1629      ;      .WORD      ;TIME OUT SEND      ;TOSNDB      ;MSTOSN      ;*****
1630
1631
1632      000202      BLKEND      =      130.      ;OFFSET OF END OF STATISTICS (RESERVED)
1633      ;WORD      ;** RESERVED **
1634      000204      TUVECT      =      132.      ;VECTOR ADDRESS
1635      ;WORD      ;** RESERVED **
1636      000210      BLKSIZ      =      136.      ;** RESERVED **
1637      ;-----

```

1639

.SBTTL DEVICE DATA BLOCK ALLOCATION

1640

1641

1642

;TABLE OF DEVICE DATA BLOCK ADDRESSES

1643

1644

1645 004724 004744

BLKTBL:

.WORD DEV0

1646 004726 005154

.WORD DEV1

1647 004730 005364

.WORD DEV2

1648 004732 005574

.WORD DEV3

1649 004734 006004

.WORD DEV4

1650 004736 006214

.WORD DEV5

1651 004740 006424

.WORD DEV6

1652 004742 006634

.WORD DEV7

1653 004744

LSTDEV:

DEV0:

.BLKB BLKS1Z

1654 005154

DEV1:

.BLKB BLKS1Z

1655 005364

DEV2:

.BLKB BLKS1Z

1656 005574

DEV3:

.BLKB BLKS1Z

1657 006004

DEV4:

.BLKB BLKS1Z

1658 006214

DEV5:

.BLKB BLKS1Z

1659 006424

DEV6:

.BLKB BLKS1Z

1660 006634

DEV7:

.BLKB BLKS1Z

```
1662      .SBTTL  ERROR CODE EQUATES
1663      :-----
1664
1665      ;THE ERROR CODE OFFSET VALUES :
1666      ;USED BY 'LOG' TO INDEX BY R5 AND INCREMENT STATISTICS
1667
1668      000002      SFTRD   =      2
1669      000004      SFTWR   =      4
1670      000006      RCINIT  =      6
1671      000012      OVRN    =     10.
1672      000014      BDCOM   =     12.
1673      000016      HRDRD   =     14.
1674      000020      HRDWR   =     16.
1675      000022      BDCHK   =     18.
1676      000024      SKERR   =     20.
1677      000026      WRLOCK  =     22.
1678      000030      NOMOT   =     24.
1679      000032      CNINIT  =     26.
1680      000034      PARTL   =     28.
1681      000036      NOUNIT  =     30.
1682      000040      CMNDER  =     32.
1683      000042      RECERR  =     34.
1684      000044      SLFER   =     36.
1685      000046      SUCQTL  =     38.
1686      000050      TORCVB  =     40.
1687      000052      OTL     =     42.
1688      000054      NCART   =     44.
1689      000056      TOSNDB  =     46.
```

```

1691          .SBTTL  ERROR MESSAGE DESCRIPTIONS
1692
1693          ;THE TABLE OF REASONS (ADDRESSES) ABORT OCCURRED.  ABNDX (R5) CONTAINS
1694          ;THE OFFSET OF THE REASON.  IT'S ABSOLUTE ADDRESS IS RSNTAB + ABNDX(R5).
1695
1696 007044 007140  RSNTAB: MSNLOG
1697 007046 007672      MSSFRD
1698 007050 007732      MSSFWR
1699 007052 007354      MSRNIT
1700 007054 007140      MSNLOG
1701 007056 010076      MSOVRN
1702 007060 007222      MSCOM
1703 007062 007772      MSHDRD
1704 007064 010034      MSHDWR
1705 007066 007374      MSHCHK
1706 007070 007124      MSSKER
1707 007072 007332      MSWPRO
1708 007074 007264      MSNOMO
1709 007076 007436      MSNIT
1710 007100 007452      MSPART
1711 007102 007474      MSUNIT
1712 007104 007522      MSCMD
1713 007106 007536      MSREC
1714 007110 007202      MSSELF
1715 007112 007556      MSWRSP
1716 007114 007602      MSNRSP
1717 007116 007616      MSQRSP
1718 007120 007302      MSNOTP
1719 007122 007650      MSTOSN
1720
1721          ;HERE ARE THE MESSAGES PROPER:
1722
1723 007124      123      105      105  MSSKER: .ASCIZ  /SEEK ERROR/          ;DEVICE COULD NOT READ HEADER
1724          .EVEN
1725 007140      123      131      123  MSNLOG: .ASCIZ  /SYSTEM ERROR/      ;DIAGNOSTIC HUNG. BETTER RE-BOOT
1726          .EVEN
1727 007156      102      101      104  MSBDA: .ASCIZ  /BAD DATA IN PACKET/  ;HOST DATA CHECK FOUND ERROR, DEVICE MAY
1728          .EVEN          ;HAVE READ CORRECTLY.
1729 007202      123      105      114  MSSELF: .ASCIZ  /SELF TEST ERROR/    ;MICRO DIAGNOSTIC FAILED, BUT DEVICE COULD STILL
1730          .EVEN          ;SEND.
1731 007222      102      101      104  MSCOM: .ASCIZ  /BAD DATA W-O DATA CHECK ERR AT TU/ ;PREVIOUS DATA CHECK
1732          .EVEN          ;ERROR NOT DUE TO DEVICE READ OP.
1733 007264      115      117      124  MSNOMO: .ASCIZ  /MOTOR STOPPED/      ;DEVICE COULD NOT GET SIGNAL
1734          .EVEN          ;FROM TAPE OR MOTOR HUNG
1735 007302      103      101      122  MSNOTP: .ASCIZ  /CARTRIDGE NOT IN PLACE/ ;NO MEDIA OR BAD SWITCH
1736          .EVEN
1737 007332      127      122      111  MSWPRO: .ASCIZ  /WRITE PROTECTION/    ;CARTRIDGE WRITE PROTECT TAB MISSING OR
1738          .EVEN          ;SWITCH BAD
1739 007354      122      105      103  MSRNIT: .ASCIZ  /RECIEVING INIT/      ;DEVICE SENT INIT REQUEST
1740          .EVEN
1741 007374      110      117      123  MSHCHK: .ASCIZ  /HOST FOUND PACKET CHECKSUM ERROR/ ;DEVICE SENT PACK WITH
1742          .EVEN          ;BAD CHECKSUM
1743 007436      103      101      116  MSNIT: .ASCIZ  /CAN'T INIT/          ;DEVICE SENT BYTE OTHER THAN "CONT"
1744          .EVEN          ;DURING INITIALIZATION
1745 007452      120      101      122  MSPART: .ASCIZ  /PARTIAL OPERATION/    ;END OF MEDIUM ENCOUNTERED
1746          .EVEN
1747 007474      042      116      117  MSUNIT: .ASCIZ  /'NON-EXISTENT' DRIVE/ ;DEVICE RECV'D TOO LARGE DRIVE NUMBER

```

HARDWARE TESTS MACRO M1110 12-JUN-79 12:23 PAGE 23-1
 ERROR MESSAGE DESCRIPTIONS

SEQ 0040

1748					.EVEN		
1749	007522	102	101	104	MSCMD:	.ASCIZ /BAD COMMAND/	;DEVICE COULD NOT UNDERSTAND HOST
1750					.EVEN		
1751	007536	102	101	104	MSREC:	.ASCIZ /BAD RECORD NO./	;DEVICE RECV'D TOO LARGE A RECORD NUMBER
1752					.EVEN		
1753	007556	127	122	117	MSWRSP:	.ASCIZ /WRONG SUCCESS CODE/	;HOST COULD NOT DECIPHER CODE IN END PACK
1754					.EVEN		
1755	007602	116	117	040	MSNRSP:	.ASCIZ /NO RESPONSE/	;TIME OUT WAITING FOR BYTE IN BUF ON DLV.
1756					.EVEN		
1757	007616	111	116	104	MSQRSP:	.ASCIZ \INDECIPHERABLE FLAG BYTE\	;HOST COULD NOT UNDERSTAND 1ST BYTE OF
1758					.EVEN		;RESPONSE FROM TU AS PROPER PROTOCOL
1759	007650	124	111	115	MSOSN:	.ASCIZ /TIME OUT ON SEND/	;DLV READY NEVER WENT HIGH
1760					.EVEN		
1761	007672	122	105	103	MSFRD:	.ASCIZ /RECOV. DATA CHECK ERR ON RD OP/	;TU58 RESPONDED WITH 'DATA-CHECK'
1762					.EVEN		;ERROR ON READ OP. ;HOST RETRY(S) SUCCESSFUL
1763	007732	122	105	103	MSFWR:	.ASCIZ /RECOV. DATA CHECK ERR ON WR OP/	;SAME BUT WR OR WR VERIFY OPERATION
1764					.EVEN		
1765	007772	125	116	122	MSHARD:	.ASCIZ /UNRECOV. DATA CHECK ERR ON RD OP/	;TU58 RESPONDED WITH 'DATA-CHECK'
1766					.EVEN		;ERROR ON READ OP. ;COULD NOT RECOVER
1767	010034	125	116	122	MSHDWR:	.ASCIZ /UNRECOV. DATA CHECK ERR ON WR OP/	;SAME BUT WR OPERATION
1768					.EVEN		
1769	010076	104	114	126	MSOVRN:	.ASCIZ /DLV ERROR IN RECEIVE/	;DLV ERROR
1770					.EVEN		


```

1806      ;RADIAL SERIAL CODES:
1807
1808      ;THE FLAG BYTE CODES ARE:
1809      000002      R$CMND = 2
1810      000020      R$CNT = 20
1811      000020      R$XON = 20
1812      000023      R$XOFF = 23
1813      000004      R$INIT = 4
1814      000001      R$DATA = 1
1815      000002      R$END = R$CMND
1816      -----
1817      ;END PACK SIZE:
1818      000016      R$NDSZ = 14.
1819      ;MESSAGE PACK SIZE:
1820      000012      R$MSIZ = 12      ;10. BYTES FOR BYTE COUNT INSIDE CMND PACK
1821      ;DATA PACK SIZE:
1822      000204      R$DASZ = 132.
1823      ;DATA + END PACK SIZE:
1824      000222      R$DNSZ = R$DASZ+R$NDSZ
1825
1826      000016      R$SNSZ = R$MSIZ + 4      ;SIZE FOR SENDING COMMAND PACK
1827      001036      RCBFSZ== 4*R$DASZ+R$NDSZ      ;4 DATA PAKS AND END PACK
1828      ;IS SIZE OF RCV BUFFERS
1829      -----
1830      ;
1831      ;      THE OP CODES ARE:
1832
1833
1834      000100      R$$END = 100
1835      000003      R$$WR = 3
1836      000002      R$$RD = 2
1837      000005      R$$SEK = 5
1838      000000      R$$NOP = 0
1839      000001      R$$NIT = 1
1840      000007      R$$SLF = 7
1841      -----
1842      ;THE SUCCESS CODES ARE:
1843
1844      177720      E$ABO =-48.      ;BAD COMMAND FROM HOST
1845      177767      E$NCRT =-9.      ;NO CARTRIDGE
1846      177770      E$NONX =-8.      ;NO DRIVE
1847      000000      E$OK =0      ;OP COMPLETE SUCCESS
1848      177776      E$PART =-2      ;PARTIAL OP
1849      177740      E$SK =-32.      ;SEEK ERROR
1850      000001      E$TRY =1      ;RETRY
1851      177765      E$WLOC =-11.      ;WRITE PROTECTED
1852      177737      E$NOMO =-33.      ;MOTOR STOPPED
1853      177720      E$CMD =-48.      ;COMMAND ERROR
1854      177711      E$REC =-55.      ;BAD RECORD NUMBER.
1855      177757      E$CKS =-17.      ;TU CHKSUM
1856      177777      E$SLF =-1.      ;SELF TEST ERROR
1857      177757      E$CKSM=E$CKS
1858      177757      E$WR=E$CKS
1859      177757      E$RD=E$CKS
1860      -----

```

```

1862          .SBTTL  NXTST / THE SCHEDULER
1863
1864          :++
1865          : NXTST - USING EACH UN-ABORTED UNIT'S TEST PROGRAM COUNTER
1866          : (TSTPC(R5)), EXECUTES THE TEST CODE THAT COMPRISES MAKING A
1867          : PACKET AND SENDING IT. ACTION IS ROUND ROBIN. CHECKS FIRST
1868          : FOR ANY UNIT RETRYING AND IF SO SERVICES ONLY THAT UNIT THIS
1869          : PASS. INITIS NON-RETRYING UNITS IF NECESSARY.
1870          : INPUTS: (IMPLIED) DATA BLOCKS.
1871          : OUTPUTS: ERRSF IF ALL UNITS ARE ABORTED.
1872          :--
1873
1874 010162 012737 004724 010130 NXTST: MOV    #BLKTBL,DEVPTN ;UNIT 0 TO START
1875 010170 017705 177734 1$:      MOV    @DEVPTN,R5      ;GET DATA BLOCK
1876 010174 032715 002000          BIT    #BIT10,@R5      ;RETRYING?
1877 010200 001422          BEQ     2$      ;NOT THIS GUY
1878 010202 005715          TST     @R5      ;YES, ABORTED THO?
1879 010204 100420          BMI     2$      ;YES ON TO NEXT UNIT
1880 010206 052737 000002 010124      BIS    #BIT1,SYSTAT ;NOT ABORTED-SET RETRY STATUS
1881 010214          SWAPIN          ;GET DEVICE REGISTERS
1882 010240 004775 000020          JSR     PC,@TSTPC(R5) ;DO TEST FOR
1883 010244 000475          BR      NXTRET ;THIS UNIT ONLY-EXIT
1884 010246 023727 010130 004742 2$:  CMP     DEVPTN,#LSTDEV ;TRY NEXT UNIT?
1885 010254 103004          BHIS    NXTST2 ;NO
1886 010256 062737 000002 010130      ADD     #2,,DEVPTN ;YES,->NEXT
1887 010264 000741          BR      1$      ;GET BLOCK
1888
1889 010266 005037 010442          NXTST2: CLR     ABONM ;HERE=NO RETRIES TO DO, NO UNIT ABORTED YET
1890 010272 012737 004724 010130      MOV    #BLKTBL,DEVPTN ;-->UNIT 0 STORAGE BLOCK
1891 010300 017705 177624      PERDEV: MOV    @DEVPTN,R5 ;R5-->NEXT DEVICE STORAGE BLOCK
1892
1893 010304 005715          3$:      TST     @R5      ;ABORTED?
1894 010306 100426          BMI     4$      ;YES
1895 010310 032715 040000          BIT     #BIT14,@R5 ;SEND BREAK?
1896 010314 001407          BEQ     6$      ;NO
1897 010316 004737 022640          CALL    DOBRK ;YES
1898 010322 032715 040000          BIT     #BIT14,@R5 ;SUCCESSFUL INIT?
1899 010326 001016          BNE     4$      ;NO ON TO NEXT UNIT
1900 010330 005715          TST     @R5      ;ABORTED?
1901 010332 100414          BMI     4$      ;YES-ON TO NEXT UNIT
1902 010334          6$:      SWAPIN          ;NO,GET DEVICE REGISTERS R0-R4 CONTAINING TEST PARAMETERS
1903 010360 004775 000020          JSR     PC,@TSTPC(R5) ;INITIATE 1 PACKET TRANSMISSION AND RETURN
1904 010364 005715          4$:      TST     @R5      ;ABORTED?
1905 010366 100002          BPL     8$      ;NO-ON TO NEXT UNIT
1906 010370 005237 010442          INC     ABONM ;YES...ONE MORE TALLIED
1907 010374 023727 010130 004742 8$:  CMP     DEVPTN,#LSTDEV ;ALL TU'S TRIED?
1908 010402 103004          BHIS    5$      ;YES
1909 010404 062737 000002 010130      ADD     #2,,DEVPTN ;NO THE ADDRESS+2=NEXT ADDRESS
1910 010412 000732          BR      PERDEV ;DO NEXT UNIT
1911 010414 022737 000010 010442 5$:  CMP     #8,,ABONM ;ALL ABORTED?
1912 010422 001006          BNE     NXTRET ;NO
1913 010424          ERRSF    0,,NOMOR ;YES!
1914 010434          11$:  BREAK ;SUPERVISOR BREAK

```

```

TRAP  C$ERSF
.WORD 0
.WORD NOMOR
.WORD 0

```

1915 010434 104422
1916 010436 000776
1917 010440
1918 010442 000000
1919 010444 101
1920

NXTRET: BR 11\$
RET

TRAP C\$BRK

ABONM: .WORD ; THE NUMBER OF ABORTED UNITS
NOMOR: .ASCIZ /ALL UNITS ABORTED! CNTRL-C/
.EVEN

```
1922          .SBTTL TEST 1 / DEVICE SELF-DIAGNOSTIC EXECUTION
1923
1924 010500          BGNTST
1925 010500          TSTID  #TST1
010500 012737 010542 010144
010506 004737 002514
010512 004737 002342
010516 004737 002562
010522 004737 002240
010526 103004
010530 004737 002514
010534 004737 002562
010540
1926 010540          ENDTST
010540
010540 104401
1927
1928 010542          TST1:  TUSELF
1929 010662 005237 010140  INC  DONE
1930 010666          RET
```

MOV #TST1,TSTTOP ;SAVE ADDR
CALL SETUP ;INIT UNITS
CALL SETDR ;GET 1ST DRVS.
CALL RUN ;DO TEST
CALL SWAPDR ;GET NEXT DR.
BCC 64\$;BR NO 2ND DRVS
CALL SETUP ;REINIT UNITS
CALL RUN ;REPEAT TEST
;DONE

64\$: L10011: TRAP C\$ETST

```

1932 .SBTTL TEST 2 /SEEK EOT,BOT
1933
1934 010670 BGNTST
1935 010670 TSTID #TST2
010670 012737 010732 010144 MOV #TST2,TSTTOP ;SAVE ADDR
010676 004737 002514 CALL SETUP ;INIT UNITS
010702 004737 002342 CALL SETDR ;GET 1ST DRVS.
010706 004737 002562 CALL RUN ;DO TEST
010712 004737 002240 CALL SWAPDR ;GET NEXT DR.
010716 103004 BCC 64$ ;BR NO 2ND DRVS
010720 004737 002514 CALL SETUP ;REINIT UNITS
010724 004737 002562 CALL RUN ;REPEAT TEST
010730 ;DONE
1936 010730 ENDTST 64$:
010730
010730 104401 L10012:
010732 005004 TRAP C$ETST
1937 010732 016465 011114 000064 TST2: CLR R4
1938 010734 016465 011114 000064 1$: MOV RECDAT(R4),REC(R5)
1939
1940 010742 TUSEEK REC(R5),DR(R5)
1941
1942 011072 062704 000002 ADD #2,R4
1943 011076 026427 011114 177777 CMP RECDAT(R4),#-1.
1944 011104 001313 BNE 1$
1945 011106 005237 010140 INC DONE
1946 011112 RET
1947
1948 011114 000000 RECDAT: 0. ;BOT
1949 011116 000200 200 ;BOT OTHER TRACK
1950 011120 000177 177 ;EOT
1951 011122 000377 377 ;EOT OTHER TRACK
1952 011124 000400 400 ;BOT AGAIN
1953 011126 177777 -1.

```

```

1955          .SBTTL TEST 3 / HIGH ACTIVITY WRITE/READ
1956
1957 011130      BGNSTST
1958 011130      TSTID  #TST3
011130      012737 011172 010144      MOV  #TST3,TSTTOP      ;SAVE ADDR
011136      004737 002514      CALL  SETUP      ;INIT UNITS
011142      004737 002342      CALL  SETDR      ;GET 1ST DRVS.
011146      004737 002562      CALL  RUN      ;DO TEST
011152      004737 002240      CALL  SWAPDR      ;GET NEXT DR.
011156      103004      BCC  64$      ;BR NO 2ND DRVS
011160      004737 002514      CALL  SETUP      ;REINIT UNITS
011164      004737 002562      CALL  RUN      ;REPEAT TEST
011170      64$:      ;DONE
1959 011170      ENDTST
011170      L10013:      TRAP  C$ETST
011170      104401
1960 011172      012765 000100 000066 TST3: MOV  #100,TMP(R5)      ;INIT
1961 011200      005004      CLR  R4      ;FOR INDEX TO DATA
1962 011202      005065 000064      CLR  REC(R5)
1963 011206      016465 012510 000072 1$: MOV  TST3PT(R4),PATTEN(R5) ;GET DATA
1964 011214      066565 000060 000072      ADD  DR(R5),PATTEN(R5) ;ADD DRIVE I.D.
1965 011222      TUWRIT PATTEN(R5),REC(R5),#512.,DR(R5),#0
1966 012012      TUREAD REC(R5),#512.,DR(R5),#0
1967
1968 012412      062704 000002      ADD  #2,R4      ;NEXT INDEX
1969 012416      005764 012510      TST  TST3PT(R4)      ;END?
1970 012422      001402      BEQ  2$      ;YES
1971 012424      000137 011206      JMP  1$
1972 012430      005004      2$: CLR  R4      ;FIRST DATA
1973 012432      062765 000200 000064      ADD  #200,REC(R5) ;ADJACENT RECORD
1974 012440      032765 001000 000064      BIT  #1000,REC(R5) ;TOO FAR?
1975 012446      001002      BNC  3$      ;YES
1976 012450      000137 011206      JMP  1$
1977 012454      162765 001000 000064 3$: SUB  #1000,REC(R5) ;BACK UP
1978 012462      066565 000066 000064      ADD  TMP(R5),REC(R5) ;HALF INTO REST OF TAPE
1979 012470      006265 000066      ASR  TMP(R5) ;HALF OF HALF
1980 012474      103402      BCS  4$      ;DONE?
1981 012476      000137 011206      JMP  1$      ;NO
1982 012502      005237 010140      4$: INC  DONE
1983 012506      RET
1984 012510      000000      1ST3PT: .WORD 000000
1985 012512      125252      .WORD 125252
1986 012514      177777      .WORD 177777
1987 012516      052525      .WORD 052525
1988 012520      000000      .WORD 000000
1989
1990

```

```

1992
1993
1994
1995 012522
      012522
1996 012522
      012522 012737 012564 010144
      012530 004737 002514
      012534 004737 002342
      012540 004737 002562
      012544 004737 002240
      012550 103004
      012552 004737 002514
      012556 004737 002562
      012562
1997 012562
      012562
      012562 104401
1998 012564 005065 000064
1999 012570 013765 010126 000066
2000 012576 005065 000062
2001 012602 016565 000064 000072
2002 012610 005737 002214
2003 012614 001403
2004 012616 066565 000060 000072
2005 012624
2006 013414 005365 000066
2007 013420 001404
2008 013422 005265 000064
2009 013426 000137 012602
2010 013432 005765 000062
2011 013436 001012
2012 013440 005265 000062
2013 013444 013765 010152 000064
2014 013452 013765 010126 000066
2015 013460 000137 012602
2016 013464 005237 010140
2017 013470

      .SBTTL TEST 4 / WRITE SELECTED NUMBER OF BLOCKS
      BGNTST
      TSTID #TST4
      MOV #TST4,TSTTOP ;SAVE ADDR
      CALL SETUP ;INIT UNITS
      CALL SETDR ;GET 1ST DRVS.
      CALL RUN ;DO TEST
      CALL SWAPDR ;GET NEXT DR.
      BCC 64$ ;BR NO 2ND DRVS
      CALL SETUP ;REINIT UNITS
      CALL RUN ;REPEAT TEST
      64$: ;DONE
      ENDTST
      L10014: TRAP C$ETST

TST4: CLR REC(R5) ;START AT REC 0
      MOV TAPLEN,TMP(R5) ;# OF BLOCKS
      CLR TRK(R5) ;1 OF 2 PASS COUNTER
1$: MOV REC(R5),PATTEN(R5) ;DATA IS RECORD NO.
      TST DRVCHK ;ADD DR #?
      BEQ 10$ ;NO
      ADD DR(R5),PATTEN(R5) ;ADD DRIVE ID
      TUWRIT PATTEN(R5),REC(R5),#512,DR(R5),#0
10$: DEC TMP(R5) ;DONE THIS TRACK?
      BEQ 2$ ;YES-GET OTHER TRACK
      INC REC(R5) ;NO-NEXT RECORD
      JMP 1$ ;EXECUTE
2$: TST TRK(R5) ;DONE 2 TRACKS?
      BNE TST4EX ;YES-EXIT
      INC TRK(R5) ;NO-SET FLAG FOR NEXT PASS
      MOV SECREC,REC(R5) ;GET NEW STARTING BLOCK #
      MOV TAPLEN,TMP(R5) ;RESET # OF BLOCKS
      JMP 1$ ;AND EXECUTE
TST4EX: INC DONE ;DONE
      RET ;RETURN

```

```

2019 .SBTTL TEST 5 / READ SELECTED NUMBER OF BLOCKS
2020
2021 013472 BGNTST
2022 013472 TSTID #TST5
013472 012737 013534 010144
013500 004737 002514
013504 004737 002342
013510 004737 002562
013514 004737 002240
013520 103004
013522 004737 002514
013526 004737 002562
013532
2023 013532 ENDTST
013532
013532 104401
2024 013534 005065 000064 TST5: CLR REC(R5) ;START AT REC 0
2025 013540 013765 010126 000066 MOV TAPLEN,TMP(R5) ;# OF BLOCKS
2026 013546 005065 000062 CLR TRK(R5) ;1 OR 2 PASS COUNTER
2027 013552 016565 000064 000072 1$: MOV REC(R5),PATTEN(R5) ;DATA IS RECORD NO.
2028 013560 005737 002214 TST DRVCHK ;ADD DR #?
2029 013564 001403 BEQ 10$ ;NO
2030 013566 066565 000060 000072 ADD DR(R5),PATTEN(R5) ;ADD IN DRIVE ID
2031 013574 10$: TUREAD REC(R5),#512.,DR(R5),#0
2032 014174 005365 000066 DEC TMP(R5) ;DONE THIS TRACK?
2033 014200 001404 BEQ 2$ ;YES-GET OTHER TRACK
2034 014202 005265 000064 INC REC(R5) ;NO-NEXT RECORD
2035 014206 000137 013552 JMP 1$ ;EXECUTE
2036 014212 005765 000062 2$: TST TRK(R5) ;DONE 2 TRACKS?
2037 014216 001012 BNE TST5EX ;YES-EXIT
2038 014220 005265 000062 INC TRK(R5) ;NO-SET FLAG FOR NEXT PASS
2039 014224 013765 010152 000064 MOV SECREC,REC(R5) ;GET NEW STARTING BLOCK #
2040 014232 013765 010126 000066 MOV TAPLEN,TMP(R5) ;RESET # OF BLOCKS
2041 014240 000137 013552 JMP 1$ ;AND EXECUTE
2042 014244 005237 010140 TST5EX: INC DONE ;DONE
2043 014250 RET ;RETURN
2044

```

T5::

```

MOV #TST5,TSTTOP ;SAVE ADDR
CALL SETUP ;INIT UNITS
CALL SETDR ;GET 1ST DRVS.
CALL RUN ;DO TEST
CALL SWAPDR ;GET NEXT DR.
BCC 64$ ;BR NO 2ND DRVS
CALL SETUP ;REINIT UNITS
CALL RUN ;REPEAT TEST

```

64\$:

L10015:

TRAP C\$ETST

```

2046          .SBTTL TEST 6 / WRITE-VERIFY SELECTED NUMBER OF BLOCKS
2047
2048 014252      BGNST
2049 014252      TSTID  #TST6
014252      012737 014314 010144      MOV  #TST6,TSTTOP      ;SAVE ADDR
014260      004737 002514      CALL  SETUP      ;INIT UNITS
014264      004737 002342      CALL  SETDR      ;GET 1ST DRVS.
014270      004737 002562      CALL  RUN      ;DO TEST
014274      004737 002240      CALL  SWAPDR      ;GET NEXT DR.
014300      103004      BCC  64$      ;BR NO 2ND DRVS
014302      004737 002514      CALL  SETUP      ;REINIT UNITS
014306      004737 002562      CALL  RUN      ;REPEAT TEST
014312      64$:      ;DONE
2050 014312      ENDIST
014312      L10016:      TRAP  C$ETST
014312      104401
2051 014314      005065 000064      TST6:  CLR  REC(R5)      ;START AT REC 0
2052 014320      013765 010126 000066      MOV  TAPLEN,TMP(R5)      ;# OF BLOCKS
2053 014326      005065 000062      CLR  TRK(R5)      ;1 OR 2 PASS COUNTER
2054 014332      016565 000064 000072 1$:  MOV  REC(R5),PATTEN(R5) ;DATA IS RECORD NO.
2055 014340      005737 002214      TST  DRVCHK      ;ADD DR #?
2056 014344      001403      BEQ  10$      ;NO
2057 014346      066565 000060 000072      ADD  DR(R5),PATTEN(R5) ;ADD DRIVE ID
2058 014354      10$:  TUWRIT PATTEN(R5),REC(R5),#512,DR(R5),#1
2059 015144      005365 000066      DEC  TMP(R5)      ;DONE THIS TRACK?
2060 015150      001404      BEQ  2$      ;YES-GET OTHER TRACK
2061 015152      005265 000064      INC  REC(R5)      ;NO-NEXT RECORD
2062 015156      000137 014332      JMP  1$      ;EXECUTE
2063 015162      005765 000062      2$:  TST  TRK(R5)      ;DONE 2 TRAXKS?
2064 015166      001012      BNE  TST6EX      ;YES-EXIT
2065 015170      005265 000062      INC  TRK(R5)      ;NO-SET FLAG FOR NEXT PASS
2066 015174      013765 010152 000064      MOV  SECREC,REC(R5) ;GET NEW STARTING BLOCK #
2067 015202      013765 010126 000066      MOV  TAPLEN,TMP(R5) ;RESET # OF BLOCKS
2068 015210      000137 014332      JMP  1$      ;AND EXECUTE
2069 015214      005237 010140      TST6EX: INC  DONE      ;DONE
2070 015220      RET      ;RETURN
2071

```

HARDWARE TESTS MACRO M1110 12-JUN-79 12:23 PAGE 33
 TEST 7 / READ-REDUCED THRESHOLD SELECTED NUMBER OF BLOCKS

SEQ 0051

```

2073          .SBTTL TEST 7 / READ-REDUCED THRESHOLD SELECTED NUMBER OF BLOCKS
2074
2075 015222          BGNIST
2076 015222          TSTID  #TST7
015222          MOV      #TST7,TSTTOP      ;SAVE ADDR
015230          CALL     SETUP             ;INIT UNITS
015234          CALL     SETDR             ;GET 1ST DRVS.
015240          CALL     RUN               ;DO TEST
015244          CALL     SWAPDR            ;GET NEXT DR.
015250          BCC      64$              ;BR NO 2ND DRVS
015252          CALL     SETUP             ;REINIT UNITS
015256          CALL     RUN               ;REPEAT TEST
015262          64$:
2077 015262          ENDTST
015262          L10017: TRAP C$ETST
2078 015264 005065 000064 TST7: CLR REC(R5) ;START AT REC 0
2079 015270 013765 010126 000060 MOV TAPLEN,TMP(R5) ;# OF BLOCKS
2080 015276 005065 000062 CLR TRK(R5) ;1 OR 2 PASS COUNTER
2081 015302 016565 000064 000072 1$: MOV REC(R5),PATTEN(R5) ;DATA IS RECORD NO.
2082 015310 005737 002214 TST DRVCHK ;ADD DR #?
2083 015314 001403 BEQ 10$ ;NO
2084 015316 066565 000060 000072 ADD DR(R5),PATTEN(R5) ;ADD DRIVE ID
2085 015324 TUREAD REC(R5),#512.,DR(R5),#1
2086 015724 005365 000066 DEC TMP(R5) ;DONE THIS TRACK?
2087 015730 001404 BEQ 2$ ;YES-GET OTHER TRACK
2088 015732 005265 000064 INC REC(R5) ;NO-NEXT RECORD
2089 015736 000137 015302 JMP 1$ ;EXECUTE
2090 015742 005765 000062 2$: TST TRK(R5) ;DONE 2 TRACKS?
2091 015746 001012 BNE TST7EX ;YES-EXIT
2092 015750 005265 000062 INC TRK(R5) ;NO-SET FLAG FOR NEXT PASS
2093 015754 013765 010152 000064 MOV SECREC,REC(R5) ;GET NEW STARTING BLOCK #
2094 015762 013765 010126 000066 MOV TAPLEN,TMP(R5) ;RESET # OF BLOCKS
2095 015770 000137 015302 JMP 1$ ;AND EXECUTE
2096 015774 005237 010140 TST7EX: INC DONE ;DONE
2097 016000 RET ;RETURN

```

```

2100 .SBTTL RSVP / XOFF AND SEND A PACKET TO ALL DEVICES
2101
2102
2103 :++
2104 : RSVP - SAVES TEST CODE PROGRAM COUNTER IN TSTPC(R5) AND UNIT'S REGIS-
2105 : TERS. POINTS TO 'XOFF' THAT PRECEEDS PACKET IN XMIT BUFFER
2106 : AND SENDS PACKET WITH XOFF. RETURNS TO SCHEDULER (NXTST) SO
2107 : THAT OTHER UNITS PACKETS MAY BE FORMED, TO GET ALL UNITS WORKING
2108 : AT ONCE.
2109 : INPUTS: (SP) CONTAINS UNITS PC TO SAVE SINCE RSVP WAS CALLED. THE
2110 : NUMBER PACKETS EXPECTED (X$PKNM), AND THE EXPECTED FLAGS AND
2111 : BYTE COUNTS OF EACH (X$FLG, X$CNT...) ARE LOADED BY TEST CODE
2112 : (MACROS).
2113 : OUTPUTS: CMDSNT - UPDATED WITH PACKET OP CODE
2114 : BLKER - RECORD NUMBER STATISTICS UPDATED IF NOT RETRYING
2115 : AND COMMAND PACKET SENT.
2116 : SUCCS - PRESET CLEAR
2117 : STATUS WORD @R5 - BIT9 - DATA CHECK ERROR - CLEARED
2118 : BIT5 - 'VERIFY' OPERATION
2119 : BIT4 - 0 = DATA PACK 1 = CMND
2120 : BIT8 - RD/WR OPERATION
2121 : X$PTR - POINTS TO EXPECTED FLAG
2122 : UPPER BYTE OF X$PKNM IS REPLICATED.
2123 : PACKET POINTER (PKPTR) POINTS TO TOP OF UNITS RECEIVE BUFFER
2124 : AREA (RCVBUF).
2125 :--
2126 RSVP: NOP ;FINISH TEST
2127 016002 000240 MOV (SP)+,TSTPC(R5) ;SAVE WHERE YOU WERE IN TEST BODY AND
2128 016004 012665 000020 SWAPOW ;SAVE TEST REGISTERS
2129
2130 :CORRECT FOR RETURN TO SCHEDULER
2131 016034 012700 024331 XFNSND: MOV #STRBUF-1,R0 ;POINT TO XOFF
2132 016040 005265 000070 INC SDCNT(R5) ;ONE MORE TO SEND, TOO.
2133 016044 000402 BR SND ;SEND XOFF+PACKET
2134 016046 012700 024332 NOXOFF: MOV #STRBUF,R0 ;FOR NORMAL PACKET SEND
2135 016052 004737 016304 SND: CALL SDBYT ;SEND BYTE
2136 016056 005715 TST @R5 ;R5--> TO STATUS BLK
2137 016060 100510 BMI 6$ ;ABORTED? YES...QUIT
2138 016062 005365 000070 DEC SDCNT(R5) ;NO, SEND MORE
2139 016066 001371 BNE SND ;IF MORE TO SEND
2140 016070 012700 024332 MOV #STRBUF,R0 ;-->BUFFER
2141 016074 016537 000064 010150 MOV REC(R5),BLKER ;PREPARE FOR RECEIVE
2142 016102 156565 000032 000033 BISB X$PKNM(R5),X$PKNM+1(R5) ;REPLICATE LO. BYTE TO HI FOR GTPAKS, CHKANS
2143 016110 005065 000076 CLR SUCCS(R5) ;NO SUCCESS YET
2144 016114 042715 001000 BIC #BIT9,@R5 ;NO DATA CHK ERROR YET
2145 016120 016565 000102 000104 MOV RCVBUF(R5),PKPTR(R5) ;TOP OF RCV BUFFER GOES THE 1ST PACKET
2146 016126 012704 000034 MOV #X$FLG,R4 ;FORM
2147 016132 060504 ADD R5,R4 ;ADDRESS
2148 016134 010465 000106 MOV R4,X$PTR(R5) ;OF 1ST X$FLG
2149
2150 016140 042715 000020 BIC #BIT4,@R5 ;PRESET AS DATA PAK
2151 016144 121027 000002 CMPB @R0,#R$CMND ;WAS IT COMMAND PAK?
2152 016150 001054 BNE 6$ ;NO...
2153 016152 116065 000002 000100 MOVB 2(R0),CMDSNT(R5) ;YES-SAVE COMMAND
2154 016160 052715 000020 BIS #BIT4,@R5 ;ITS CMND PAK
2155
2156 016164 032715 002000 BIT #BIT10,@R5 ;RETRYING?

```

2157	016170	001044			BNE	6\$		;YES-DON'T UPDATE ANY STATS OR CONDITION
2158	016172	126027	000002	000002	CMPB	2(R0),#R\$\$RD		;NO,A READ?
2159	016200	001012			BNE	4\$		;NO
2160	016202	042715	000400		BIC	#BIT8,@R5		;(FOR HARD/SOFT LOGGING) RD/WR FLAG=0
2161	016206	004737	022470		CALL	WHCHDR		;GET DRIVE
2162	016212	103403			BCS	8\$		
2163	016214	005265	000114		INC	RDN0(R5)		;DRIVE 0
2164	016220	000402			BR	4\$		
2165	016222	005265	000116	8\$:	INC	RDN1(R5)		;DRIVE 1
2166								
2167	016226	126027	000002	000003	4\$:	CMPB	2(R0),#R\$\$WR	;A WRITE?
2168	016234	001022			BNE	6\$		;NO
2169	016236	052715	000400		BIS	#BIT8,@R5		;YES, RD/WR FLAG=1
2170	016242	105760	000003		TSTB	3(R0)		;VERIFY TOO?
2171	016246	001403			BEQ	21\$		;NO
2172	016250	052715	000040		BIS	#BIT5,@R5		;YES-SET VERIFY FLAG
2173	016254	000402			BR	22\$		
2174	016256	042715	000040	21\$:	BIC	#BIT5,@R5		;(NO)-RESET VERIFY FLAG
2175	016262	004737	022470	22\$:	CALL	WHCHDR		;GET DRIVE NO
2176	016266	103403			BCS	5\$		;CARRY=DR1
2177	016270	005265	000110		INC	WRTN0(R5)		;# BLKS WRITTEN DR0
2178	016274	000402			BR	6\$		;EXIT
2179								
2180	016276	005265	000112	5\$:	INC	WRTN1(R5)		;# BLKS WRITTEN DRV1
2181	016302			6\$:	RET			;RETURN
2182								
2183	016304			SNDBYT:	PUSH	R1		;ENTER R0-->BYTE
2184	016306	013701	010156	4\$:	MOV	C\$NRDY,R1		;GET TIMEOUT CONSTANT FOR NOT READY ERROR
2185	016312	105775	000026	1\$:	TSTB	@XMSR(R5)		;READY TO SEND?
2186	016316	100412			BMI	2\$		;YES
2187	016320				PUSH	R0		;NO, SAVE R0
2188	016322				BREAK			;MONITOR BREAK
	016322	104422						
2189	016324				POP	R0		;RESTORE
2190								
2191	016326	005301			DEC	R1		;ABORTED?
2192	016330	001370			BNE	1\$		;NO
2193	016332	012704	000056		MOV	#TOSNDB,R4		;YES,SET CODE FOR TIMEOUT ERROR
2194	016336	004737	021464		CALL	LOG		;LOG IT
2195	016342	000402			BR	3\$		;QUIT
2196	016344	112075	000030	2\$:	MOVB	(R0)+,@XMDB(R5)		;SEND IT
2197	016350			3\$:	POP	R1		;RESTORE
2198	016352				RET			;DONE

TRAP C\$BRK

```
2200 .SBTTL GETANS / GETS RESPONSES ROUND ROBIN USING 'XON'
2201
2202
2203 :++
2204 : GETANS - IF A UNIT IS RETRYING CLEAR HIS RECEIVE BUFFER (CLRBUF) AND GET
2205 : HIS RESPONSE (GTPKS1), ELSE, CLEAR ALL BUFFERS (CLRALL) AND
2206 : GET ALL RESPONSES (GTPKS8).
2207 : INPUTS: NONE
2208 : OUTPUTS: SERVST = -1 IF NO RETRIES.
2209 :--
2210 016354 000240 GETANS: NOP ;1 UNIT IF RETRY; ELSE ALL
2211 016356 032737 000002 010124 BIT #BIT1,SYSTAT ;RETRY?
2212 016364 001010 BNE 1$ ;YES
2213 016366 012737 177777 017274 MOV #-1,SERVST ;PRESET NO UNITS SERVICED
2214 016374 004737 002420 CALL CLRALL ;CLEAR ALL INPUT BUFFERS
2215 016400 004737 016632 CALL GTPKS8 ;GET ALL REPLYs
2216 016404 000404 BR 2$ ;EXIT
2217 016406 004737 002460 1$: CALL CLRBUF ;RETRY-CLEAR 1 UNIT ONLY
2218 2219 016412 004737 016422 CALL GTPKS1 ;RS->UNIT BY NXTST
2220 016416 2221 RET ;GET 1 REPLY
2221 ;DONE
2222 016420 000000 GETPTR: .WORD
```

HARDWARE TESTS MACRO M1110 12-JUN-79 12:23 PAGE 37
GTPKS1 / GET RETRY RESPONSE-1 UNIT

SEQ 0055

```

2224 .SBTTL GTPKS1 / GET RETRY RESPONSE-1 UNIT
2225
2226 :++
2227 : GTPKS1 - SENDS XON TO UNIT, GETS FLAG BYTE (IF ANY), CHECKS IF IT IS
2228 :          WHAT WAS EXPECTED. IF IT IS, USE EXPECTED BYTE COUNT. IF
2229 :          NOT, CHECK IF PREMATURE-END PACK OR (SINCE MAINTENANCE MODE)
2230 :          IF IT'S A PREMATURE DATA PACK. ADJUST COUNT, GET REST OF
2231 :          PACKET, AND REPEAT ABOVE UNTIL NO MORE PACKETS.
2232 : INPUTS: NONE PASSED.
2233 : OUTPUTS: SYSTAT UPPER BYTE = FLAG BYTE RECEIVED
2234 :--
2235
2236 016422 000240 GTPKS1: NOP ;R5->THE UNIT
2237 016424 012703 000034 MOV #X$FLG,R3 ;THE OFFSET VALUE OF FLAG
2238 016430 060503 ADD R5,R3 ;FORM THE ABSOLUTE ADDRESS
2239 016432 010301 MOV R3,R1 ;R3-->ADDR. OF EXPECTED FLAG
2240 016434 062701 000002 ADD #2.,R1 ;R1-->ADDR. OF EXPECTED COUNT
2241 016440 012700 016630 MOV #EXON,R0 ;R0=ADDRESS
2242 016444 004737 016304 CALL SNDBYT ;XON THE DEVICE
2243
2244 016450 016500 000102 MOV RCVBUF(R5),R0 ;*** TIME CRITICAL
2245 016454 116502 000033 MOVX #X$PKM+1(R5),R2 ;***--> TO THE BUFFER
2246 016460 032702 177400 BIT #177400,R2 ;***GET THE # OF PACKETS TO RECEIVE
2247 016464 011137 010134 1$: MOV @R1,RCBCNT ;***SIGN UN-EXTEND
2248 016470 011337 010132 MOV @R3,RCFLG ;***HOW MANY BYTES IT SHOULD BE
2249 016474 004737 017300 CALL GTBYTE ;***WHAT THE FIRST BYTE SHOULD BE
2250 016500 032715 100000 BIT #BIT15,@R5 ;***GET THE ALL IMPORTANT FLAG
2251 016504 001050 BNE 4$ ;TIMEOUT?
2252 016506 005300 DEC R0 ;YES
2253 016510 111037 010125 MOVX @R0,SYSTAT+1 ;-> BYTE RECIEVED
2254 016514 121037 010132 CMPB @R0,RCFLG ;SAVE IT AS FLAG BYTE
2255 016520 001420 BEQ 2$ ;1ST BYTE WHAT WAS EXPECTED?
2256 016522 121027 000002 CMPB @R0,#R$END ;YES
2257 016526 001006 BNE 14$ ;NO, WAS IT END PAK?
2258 016530 012737 000016 010134 MOV #R$NDSZ,RCBCNT ;NO
2259 016536 012702 000001 MOV #1,R2 ;YES, USE END SIZE FOR COUNT
2260 016542 000407 BR 2$ ;AND ASSUME IT'S LAST PACKET!
2261 016544 121027 000001 14$: CMPB @R0,#R$DATA ;CONTINUE RECEIVE
2262 016550 001026 BNE 4$ ;WAS IT DATA?
2263 016552 012737 000204 010134 MOV #R$DASZ,RCBCNT ;NO,CHKANS MAY FIND INIT...
2264 016560 005202 INC R2 ;YES, SET FOR DATA PAK SIZE
2265
2266 016562 005200 2$: INC R0 ;ONE MORE PACK THAN EXPECTED (END PAK)
2267 016564 005337 010134 5$: DEC RCBCNT ;RESTORE TO -> NEXT BYTE
2268 016570 001411 BEQ 3$ ;THAT'S ONE LESS BYTE TO GO
2269 016572 004737 017300 CALL GTBYTE ;DONE
2270 016576 005765 000074 TST DLV(R5) ;GET REST OF PACKET
2271 016602 001011 BNE 4$ ;ERROR
2272 016604 032715 100000 BIT #BIT15,@R5 ;YES-ALL OVER
2273 016610 001006 BNE 4$ ;UNLESS ABORTED
2274 016612 000764 BR 5$ ;THEN QUIT
2275
2276 016614 005302 3$: DEC R2 ;CONTINUE RECEIVE
2277 016616 001403 BEQ 4$ ;ONE LESS PACKET TO GO
2278
2279 016620 022121 CMP (R1)+,(R1)+ ;MORE PACKETS IN TRANSACTION?
2280 016622 022323 CMP (R3)+,(R3)+ ;YES
;POINT TO NEW EXPECTED COUNT
;AND FLAG,

```

2281	016624	000717			
2282	016626		4\$:	BR	1\$
2283				RET	
2284	016630	020			;AND RECEIVE,
2285	016631	023			;RETURN
			EXON:	.BYTE	R\$XON
			EXOFF:	.BYTE	R\$XOFF

```

2287 .SBTTL GTPKS8 / GET RESPONSES (NO RETRIES)
2288
2289 :++
2290 : GTPKS8 - SET ALL ABORTED UNITS SERVICED (SERVST: BIT POSITION). UNTIL
2291 : ALL UNITS SERVICED (SERVST=0), IF NO MORE PACKETS, SET UNIT
2292 : SERVICED, ELSE, GET A FLAG BYTE FROM UNIT, DECREMENTING THE
2293 : NUMBER OF PACKETS LEFT. CHECK TO SEE IF EXPECTED FLAG,
2294 : ADJUST COUNT IF NOT, GET REST OF PACKET. IF WAS DATA PAK,
2295 : SEND 'XOFF' TO ENHANCE THROUGHPUT AND GO ON TO NEXT UNIT
2296 : (IF ANY).
2297 : INPUTS: IMPLIED IN DATA BLOCK POINTED BY R5. NONE PASSED.
2298 : OUTPUTS: NONE
2299 :--
2300
2301 016632 000240 GTPKS8: NOP ;GET ALL UNITS RESPONSES XOFF IF DATA PAK (THROUGHPUT)
2302 016634 012737 004724 017276 MOV #BLKTBL,GTPTR ;->1ST
2303 016642 017705 000430 GTAGIN: MOV @GTPTR,R5 ;GET DATA BLOCK
2304 016646 032715 100000 BIT #BIT15,@R5 ;ABORTED?
2305 016652 001403 BEQ 2$ ;NO
2306 016654 004737 017210 CALL SETSRV ;YES-SET 'SERVICED' AND
2307 016660 000534 BR GTDOWN ;ON TO NEXT UNIT
2308 016662 105765 000033 2$: TSTB X$PKNM+1(R5) ;NO, ANY PACKETS LEFT?
2309 016666 001003 BNE 3$ ;YES
2310 016670 004737 017210 CALL SETSRV ;NO-HE'S DONE
2311 016674 000526 BR GTDOWN ;SO ON TO NEXT UNIT
2312 016676 105365 000033 3$: DECB X$PKNM+1(R5) ;NOW ITS ONE LESS PACKET
2313 016702 017537 000106 010132 MOV @X$PTR(R5),RCFLG ;GET EXPECTED FLAG
2314 016710 062765 000002 000106 ADD #2,X$PTR(R5) ;--> COUNT
2315 016716 017537 000106 010134 MOV @X$PTR(R5),RCBCNT ;AND EXPECTED COUNT
2316 016724 012700 016630 MOV #EXON,R0 ;-> XON
2317 ;***TIME CRITICAL
2318 016730 004737 016304 CALL SNDBYT ;***SEND IT
2319 016734 016500 000104 MOV PKPTR(R5),R0 ;***->WHERE 1ST BYTE GOES
2320 016740 004737 017300 CALL GTBYTE ;***GET IT
2321 016744 032715 100000 BIT #BIT15,@R5 ;ABORTED?
2322 016750 001403 BEQ 4$ ;NO-CONTINUE
2323 016752 105065 000033 CLRB X$PKNM+1(R5) ;YES-NO MORE PACKETS EXPECTED
2324 016756 000475 BR GTDOWN ;ON TO NEXT
2325 016760 005300 4$: DEC R0 ;-->BYTE JUST RECEIVED
2326 016762 111037 010125 MOVB @R0,SYSTAT+1 ;SAVE IT
2327 016766 121037 010132 CMPB @R0,RCFLG ;IS IT WHAT EXPECTED?
2328 016772 001436 BEQ GTOK ;YES
2329 016774 105065 000033 UNXPCT: CLRB X$PKNM+1(R5) ;NO, MUST BE LAST REPLY
2330 017000 121027 000002 CMPB @R0,#R$END ;MAYBE AN END PAK?
2331 017004 001004 BNE 4$ ;NO
2332 017006 012737 000016 010134 MOV #R$NDSZ,RCBCNT ;YES, USE PROPER COUNT
2333 017014 000406 BR GTUM ;AND GET IT
2334 017016 121027 000001 4$: CMPB @R0,#R$DATA ;IS IT DATA?
2335 017022 001053 BNE GTDOWN ;NO, ALL OVER, CHKANS WILL INIT UNIT
2336 017024 012737 000222 010134 MOV #R$DNSZ,RCBCNT ;YES, USE COUNT OF DATA + END PAK SURE TO FOLLOW
2337 017032 005200 GTUM: INC R0 ;WHERE TO STUFF THE REST
2338 017034 005337 010134 5$: DEC RCBCNT ;ONE DOWN
2339 017040 001444 BEQ GTDOWN ;NONE TO GO
2340 017042 004737 017300 CALL GTBYTE ;MORE TO GO
2341 017046 032715 100000 BIT #BIT15,@R5 ;TIMEOUT?
2342 017052 001037 BNE GTDOWN ;YES
2343 017054 005765 000074 TST DLV(R5) ;BUT DLV ERROR?

```

```

2344 017060 001765          BEQ      5$      ;NO
2345 017062 105065 000033   CLRB     X$PKNM+1(R5) ;YES-LAST TIME
2346 017066 000431          BR       GTDOWN ;ON TO NEXT
2347
2348 017070 005200          GTOK: INC      R0      ;NEXT PLACE IN BUFFER
2349 017072 005337 010134   1$: DEC      RCBCNT ;MORE BYTES?
2350 017076 001413          BEQ      2$      ;NO-ALL DONE
2351 017100 004737 017300   CALL     GTBYTE ;YES-GET IT
2352 017104 032715 100000   BIT      #BIT15,@R5 ;TIMEOUT?
2353 017110 001020          BNE      GTDOWN ;YES
2354 017112 005765 000074   TST      DLV(R5) ;ERROR?
2355 017116 001765          BEQ      1$      ;NO
2356 017120 105065 000033   CLRB     X$PKNM+1(R5) ;LAST TIME
2357 017124 000412          BR       GTDOWN ;EXIT
2358 017126 122775 000001 000104 2$: CMPB   #R$DATA,@PKPTR(R5) ;WAS DATA?
2359 017134 001006          BNE      GTDOWN ;NO, ALL DONE
2360 017136 010065 000104   MOV      R0,PKPTR(R5) ;START OF NEXT PACK NEXT TIME
2361 017142 012700 016631   MOV      #EXOFF,R0 ;XOFF AND SEND TO
2362 017146 004737 016304   CALL     SNDBYT ;ENHANCE THROUGHPUT
2363 017152 062765 000002 000106 GTDOWN: ADD  #2,X$PTR(R5) ;NEXT X$FLG FOR NEXT TRY
2364 017160 023727 017276 004742   CMP      GTPTR,#LSTDEV ;DONE ONE CYCLE ALL UNITS?
2365 017166 103004          BHIS     1$      ;YES
2366 017170 062737 000002 017276   ADD      #2,GTPTR ;NEXT UNIT
2367 017176 000621          BR       GTAGIN ;CONTINUE RECEIVE
2368 017200 105737 017274   1$: TSTB   SERVST ;DONE SERVICING ALL PAKS
2369
2370 017204 001212          BNE      GTPKS8 ;FROM ALL UNITS?
2371 017206          RET      ;NO, KEEP TRYING
2372
2373 017210          SETSRV: PUSH     R5      ;SET UNIT SERVICED
2374 017212          PUSH     R0
2375 017214 011505          MOV      @R5,R5 ;GET STAT WD
2376 017216 042705 177770   BIC      #177770,R5 ;MASK UNIT #
2377 017222 012700 017254   MOV      #SRVTBL,R0 ;->TOP OF BIT TABLE
2378 017226 005705          1$: TST      R5 ;RIGHT ONE?
2379 017230 001404          BEQ      2$      ;YES
2380 017232 062700 000002   ADD      #2,R0 ;NO, ->NEXT
2381 017236 005305          DEC      R5 ;1 LESS
2382 017240 000772          BR       1$      ;CONTINUE
2383 017242 041037 017274   2$: BIC      @R0,SERVST ;MOW IT DOWN
2384 017246          POP      R0
2385 017250          POP      R5
2386 017252          RET      ;RETURN
2387
2388 017254 000001          SRVTBL: .WORD   BIT0 ;BIT POSITION LOOKUP TABLE
2389 017256 000002          .WORD   BIT1
2390 017260 000004          .WORD   BIT2
2391 017262 000010          .WORD   BIT3
2392 017264 000020          .WORD   BIT4
2393 017266 000040          .WORD   BIT5
2394 017270 000100          .WORD   BIT6
2395 017272 000200          .WORD   BIT7
2396
2397 017274 000000          SERVST: .WORD
2398 017276 000000          GTPTR:  .WORD

```

```

2400 017300 005037 017524      GTBYTE: CLR      GBTMP      ;TIMEOUT REGISTER
2401
2402 017304 013704 010160      MOV      C$RCVB,R4      ;TIMEOUT ERROR CONSTANT (MULTIPLIER)
2403 017310 105775 000022      1$:  TSTB     @RCSR(R5)      ;READY?
2404 017314 100013              BPL      3$          ;NO
2405 017316 017565 000024 000074  MOV     @RCD8(R5),DLV(R5) ;GET ERROR + BYTE
2406 017324 116520 000074      MOV     DLV(R5),(R0)+ ;COPY BYTE TO BUFFER
2407 017330 005765 000074      TST     DLV(R5)      ;ERROR?
2408 017334 100472              BMI     4$          ;YES-EXIT
2409 017336 005065 000074      CLR     DLV(R5)      ;NO-RESET
2410 017342 000467              BR      4$          ;AND EXIT
2411
2412 017344 005337 017524      3$:  DEC     GBTMP      ;DEC T.O. CONSTANT
2413 017350 001357              BNE     1$          ;STILL VALID
2414
2415      ;CODE TO SEE ^C DURING LONG SEEK OR REWIND
2416
2417 017352 010037 017526      MOV     R0,GBTMP2      ;R0 MUST BE PRESERVED!
2418 017356 012700 016631      MOV     #EXOFF,R0      ;QUIET THE DEVICE
2419 017362 004737 016304      CALL    SNDBYT        ;BY SENDING XOFF
2420 017366 105775 000022      6$:  TSTB     @RCSR(R5)      ;CHARACTER SLOP OVER?
2421 017372 100415              BMI     5$          ;YES
2422 017374 005337 017524      DEC     GBTMP      ;NO-WAIT A WHILE
2423 017400 105737 017524      TSTB     GBTMP      ;DONE WAITING?
2424 017404 001370              BNE     6$          ;NO
2425 017406              BREAK      ;YES-NO SLOP OVER
2426 017410 012700 016630      MOV     #EXON,R0       ;START DEVICE TALKING
2427 017414 004737 016304      CALL    SNDBYT        ;AGAIN
2428 017420 013700 017526      MOV     GBTMP2,R0      ;RESTORE R0
2429 017424 000426              BR      7$          ;END KLUGE
2430 017426 013700 017526      5$:  MOV     GBTMP2,R0      ;RESTORE R0
2431 017432 017565 000024 000074  MOV     @RCD8(R5),DLV(R5) ;GET ERROR + BYTE
2432 017440 116520 000074      MOV     DLV(R5),(R0)+ ;COPY BYTE TO BUFFER
2433 017444 005765 000074      TST     DLV(R5)      ;ERROR?
2434 017450 100403              BMI     17$         ;YES-EXIT
2435 017452 005065 000074      CLR     DLV(R5)      ;NO-CLEAR
2436 017456 000400              BR      17$         ;EXIT
2437 017460 010037 017526      17$:  MOV     R0,GBTMP2      ;AGAIN SAVE R0
2438 017464 012700 016630      MOV     #EXON,R0      ;RESTORE TO TALKING STATE
2439 017470 004737 016304      CALL    SNDBYT
2440 017474 013700 017526      MOV     GBTMP2,R0      ;RESTORE R0
2441 017500 000410              BR      4$          ;DONE
2442
2443 017502 005037 017524      7$:  CLR     GBTMP      ;TIMEOUT?
2444 017506 005304              DEC     R4          ;NO
2445 017510 001277              BNE     1$          ;YES
2446 017512 012704 000050      MOV     #TORCVB,R4    ;
2447 017516 004737 021464      CALL    LOG           ;
2448 017522              RET      ;RETURN
2449 017524 000000      GBTMP:  .WORD 0
2450 017526 000000      GBTMP2: .WORD 0

```

HARDWARE TESTS MACRO M1110 12-JUN-79 12:23 PAGE 41
CHKANS / CHECK DEVICE(S) RESPONSE

SEQ 0060

```

2453      .SBTTL  CHKANS / CHECK DEVICE(S) RESPONSE
2454
2455      ;**
2456      ; CHKANS - AS IN 'GETANS', IF RETRYING DO ONLY 1 UNIT ELSE DO ALL NON-
2457      ;          ABORTED UNITS.
2458      ; INPUTS: IMPLIED SYSTAT BIT1 (RETRYING)
2459      ; OUTPUTS: NONE PASSED.
2460      ;--
2461
2462 017530 000240      CHKANS: NOP                      ;IF RETRY THEN CHECK ONE
2463                                     ;ELSE CHECK ALL
2464 017532 032737 000002 010124      BIT      #BIT1,SYSTAT ;RETRYING?
2465 017540 001403      BEQ      CHK8                     ;NO DO NORMAL
2466 017542 004737 017620      CALL     CHKPKS            ;YES DO BAZARRE WITH
2467                                     ;R5 -> UNIT
2468 017546 000422      BR        CHKANR                  ;ALL DONE
2469
2470 017550 012737 004724 017616      CHK8:  MOV      #BLKTBL,CHKPTR ;YOU KNOW ... TOP OF TABLE
2471 017556 017705 000034      2$:      MOV      @CHKPTR,R5      ;GET UNIT'S BLOCK ADDRESS
2472 017562 032715 100000      BIT      #BIT15,@R5            ;ABORTED?
2473 017566 001002      BNE      3$                        ;YES
2474 017570 004737 017620      CALL     CHKPKS            ;NO, DO THIS GUY
2475 017574 023727 017616 004742      3$:      CMP      CHKPTR,#LSTDEV ;ALL DONE?
2476 017602 103004      BHS      CHKANR                  ;YES
2477 017604 062737 000002 017616      ADD      #2,CHKPTR      ;NO,-->NEXT DEVICE
2478 017612 000761      BR        2$                      ;DO DA
2479
2480 017614      CHKANR: RET
2481
2482 017616 000000      CHKPTR: .WORD

```

```

2485      .SBTTL  CHKPKS / DECIPHERS RESPONSE OF UNIT POINTED TO BY R5 /
2486
2487      :++
2488      :  CHKPKS - FOR UNIT R5 AND FOR ALL PACKETS, CHECK TO SEE IF PACKET IS DATA OR
2489      :              END PACK, CHECK CHECKSUMS, COMPARE DATA IF DATA PACK, CHECK
2490      :              SUCCESS CODE IF END.  IF UNKNOWN PACKET TYPE, CHECK FOR INTERFACE
2491      :              ERROR.  IF 'CONTINUE' FALL THROUGH.  IF 'INIT' SET 'SEND
2492      :              BREAK' FLAG.  CALL 'LOG' WITH R4=ERROR NUMBER IF ERROR.
2493      :  INPUTS:  IMPLIED BY PREVIOUS ROUTINES.
2494      :  OUTPUTS:  ERRORS - DLV ERROR
2495      :                  - UNKNOWN FLAG BYTE ERROR
2496      :                  - CHECKSUM ERROR
2497      :                  - DATA COMPARE ERROR
2498      :--
2499
2500 017620 000240      CHKPKS: NOP      ;CHECK WHAT WAS RECIEVED
2501 017622 016500 000102      MOV      RCVBUF(R5),R0      ;GET BUFFER ADDR.
2502 017626 016502 000032      MOV      X$PKNM(R5),R2      ;AND # OF PACKETS EXPECTED
2503 017632 012703 000034      MOV      #X$FLG,R3      ;THE OFFSET VALUE
2504 017636 060503      ADD      R5,R3      ;R3-->THIS UNIT X$FLG AGAIN
2505 017640 010301      MOV      R3,R1      ;COPY TO R1
2506 017642 062701 000002      ADD      #2,R1      ;R1-->X$BCNT FOR 1ST PACKET
2507 017646 010065 000104      1$:  MOV      R0,PKPTR(R5)      ;POINT TO PACKET
2508 017652 111037 010125      MOV      @R0,SYSTAT+1      ;SAVE RCV'D BYTE
2509 017656 011137 010134      MOV      @R1,RCBCNT      ;GET COUNT
2510 017662 011337 010132      MOV      @R3,RCFLG      ;AND FLAG
2511 017666 121013      CMPB      @R0,@R3      ;1ST BYTE=EXPECTED?
2512 017670 001050      BNE      5$      ;UH OH...
2513 017672 121027 000020      CMPB      @R0,#R$CONT      ;OK, IS IT 1 BYTE?
2514 017676 001516      BEQ      7$      ;YES...ONTO NEXT PACK
2515      ;NO, SO > 1 BYTE (NEVER EXPECT INIT!)
2516 017700 013704 010134      MOV      RCBCNT,R4      ;EXPECTED, SO COUNT MUST BE RIGHT
2517 017704 005744      TST      -(R4)      ;ADJUST FROM RECEIVE COUNT TO COUNT FOR CHECKSUM
2518 017706 004737 022600      CALL      CKCKSM      ;CHECK CHECKSUM
2519 017712 103005      BCC      2$      ;NO CARRY...NO INCORRECT
2520 017714 012704 000022      MOV      #BDCHK,R4      ;ERROR
2521 017720 004737 021464      CALL      LOG      ;LOG IT
2522 017724 000503      ER      7$      ;ON TO NEXT PACK
2523 017726 122710 000002      2$:  CMPB      #R$END,(R0)      ;END PAK?
2524 017732 001005      BNE      3$      ;NO
2525 017734 004737 020154      CALL      CHKEND      ;YES-CHECK
2526 017740 012702 000001      MOV      #1,R2      ;LAST PACKET
2527 017744 000473      BR      7$      ;AND FALL THROUGH
2528 017746 122710 000001      3$:  CMPB      #R$DATA,@R0      ;DATA PAK?
2529 017752 001003      BNE      4$      ;NO
2530 017754 004737 023400      CALL      COMPAR      ;YES-CHECK DATA
2531 017760 000465      BR      7$      ;ALL DONE?
2532
2533 017762 052715 040000      4$:  BIS      #BIT14,@R5      ;SET 'DOBREAK' FLAG
2534 017766 012704 000052      MOV      #OTL,R4      ;OUT TO LUNCH
2535 017772 005765 000074      TST      DLV(R5)      ;AH,BUT DLV ERROR?
2536 017776 001402      BEQ      20$      ;NO
2537 020000 012704 000012      MOV      #OVRN,R4      ;YES-USE CORRECT ERROR #
2538 020004 004737 021464      20$:  CALL      LOG      ;TALLY
2539 020010 000460      BR      8$      ;DONE
2540
2541      ;HERE CHECKS UNEXPECTED RESPONSE

```

```

2542
2543 020012 122710 000004      5$:  CMPB  #R$INIT,@R0      ;INIT?
2544 020016 001007              BNE    6$              ;NO
2545 020020 052715 040000      BIS    #BIT14,@R5      ;YES-SET 'DOBREAK' FLAG
2546 020024 012704 000006      MOV    #RCINIT,R4      ; WE GOT AN INIT
2547 020030 004737 021464      CALL   LOG              ;TALLY IT
2548 020034 000446              BR     8$              ;DONE
2549 020036 122710 000001      6$:  CMPB  #R$DATA,@R0      ;DATA PAK?
2550 020042 001013              BNE    9$              ;NO
2551 020044 012704 000204      MOV    #R$DASZ,R4      ;YES, USE DATA SIZE
2552 020050 005744              TST    -(R4)           ;ADJUST FOR CHKSUM
2553 020052 004737 022600      CALL   CKCKSM          ;AND CHECK
2554 020056 103421              BCS    10$             ;GOOF
2555 020060 004737 023400      CALL   COMPAR          ;OK, HOW'S THE DATA?
2556                                ;EXPECTED END, GOT
2557                                ;DATA + END.
2558 020064 062700 000204      ADD    #R$DASZ,R0      ;POINT TO END PACK
2559 020070 000666              BR     1$              ;CHECK IT, USE SAME X$FLG
2560
2561 020072 122710 000002      9$:  CMPB  #R$END,(R0)      ;END?
2562 020076 001331              BNE    4$              ;NO-OUT TO LUNCH
2563
2564 020100 012704 000016      MOV    #R$SNSZ,R4      ;YES, TOTAL SIZE MINUS
2565 020104 005744              TST    -(R4)           ;TWO (THE CHKSUM)
2566 020106 004737 022600      CALL   CKCKSM          ;CHECK IT
2567 020112 103403              BCS    10$             ;OOPS
2568 020114 004737 020154      CALL   CHKEND          ;OK,NOW TEST SUC. CODE
2569
2570 020120 000414              BR     8$              ;ALL DONE
2571
2572 020122 012704 000022      10$: MOV    #BDCHK,R4      ;CHECKSUM ERROR
2573 020126 004737 021464      CALL   LOG              ;EXIT
2574 020132 000407              BR     8$
2575
2576 020134 005302              7$:  DEC    R2              ;NO. OF PACKETS LEFT TO CHECK
2577 020136 001405              BEQ    8$              ;ALL DONE
2578 020140 063700 010134      ADD    RCBcnt,R0        ;POINT TO NEXT PACKET
2579 020144 022121              CMP    (R1)+,(R1)+      ;POINT TO NEXT EXPECTED COUNT
2580 020146 022323              CMP    (R3)+,(R3)+      ;AND EXPECTED FLAG
2581 020150 000636              BR     1$              ;TRY ANOTHER,THEY'RE SMALL
2582
2583 020152              8$:  RET                      ;RETURN
2584

```

```

2587      .SBTTL  CHKEND / CHECK SUCCESS AND DETERMINE RETRY STATUS /
2588
2589      :++
2590      :  CHKEND - IF RETRYING; CHECK SUCCESS CODE AND IF 0, PRINT RECOVERED, LOG
2591      :             SOFT ERROR, END RETRY STATUS.  IF NOT 0 AND WAS STILL 'DATA
2592      :             CHECK' ERROR - DETERMINE WHETHER TO CONTINUE ANOTHER RETRY OR
2593      :             LOG 'UNRECOVERABLE' ERROR.
2594
2595      :             IF NOT RETRYING; CHECK IF DATA CHECK ERROR SUCCESS.  IF SO,
2596      :             START RETRY ELSE EXIT.
2597      :  INPUTS:  IMPLIED BY PREVIOUS CODE.  NONE PASSED.
2598      :  OUTPUTS: RETRY (SYSTAT BIT 1), (BIT10 @R5) SET IF RETRYING.
2599      :             - DATA COMARE ERROR (BIT6 @R5) CLEARED.
2600      :--
2601
2602      020154      CHKEND: PUSH      R0          ;R0 --> END PAK
2603      020156      PUSH      R4
2604      020160      032737 000002 010124 1$:  BIT      #BIT1,SYSTAT ;RETRYING?
2605      020166      001052      BNE      CHKREE      ;YES-BRANCH
2606      020170      004737 021154      CALL     CHKSUC      ;NO,GET SUCCESS CODE
2607      :             ;LOG ERROR...
2608      020174      032715 100000      BIT      #BIT15,@R5      ;ABORTED?
2609      020200      001402      BEQ      3$          ;NO,CONTINUE
2610      020202      000137 020660      JMP      CHKRET      ;YES,EXIT
2611      020206      105765 000077      3$:  TSTB     SUCCS+1(R5) ;NO; HOW'D WE DO?
2612      020212      001013      BNE      CHKERR      ;NOT SO GOOD.
2613      020214      032715 000100      BIT      #BIT6,@R5      ;OK, HOST FIND DATA PAK ERROR?
2614      020220      001002      BNE      2$          ;YES
2615      020222      000137 020660      JMP      CHKRET      ;NO
2616      020226      012704 000014      2$:  MOV      #BDCOM,R4      ;YES; JUST BAD DATA-NO DATACHK ERR
2617      020232      004737 021464      CALL     LOG          ;BAD DATA IN PACKET
2618      020236      000137 020660      JMP      CHKRET      ;QUIT
2619      020242      032715 001000      CHKERR: BIT      #BIT9,@R5      ;BAD SUCCESS; TU DATA CHK ERROR?
2620      020246      001002      BNE      1$          ;YES
2621      020250      000137 020660      JMP      CHKRET      ;NO. ALL DONE.
2622      020254      052715 002000      1$:  BIS      #BIT10,@R5      ;YES-START RETRY
2623      020260      012765 000001 000002      MOV      #1,RETRY(R5) ;CALL IT 1ST
2624      020266      PRINTX   #RTRYN,RETRY(R5) ;** PRINT **
2625      020266      016546 000002      MOV      RETRY(R5),-(SP)
2626      020272      012746 021040      MOV      #RTRYN, -(SP)
2627      020276      012746 000002      MOV      #2, -(SP)
2628      020302      010600      MOV      SP,R0
2629      020304      104415      TRAP     C$PNTX
2630      020306      062706 000006      ADD      #6,SP
2631      020312      000562      BR      CHKRET      ;ALL DONE
2632      020314      004737 021154      CHKREE: CALL     CHKSUC      ;RETRYING,GET SUCCESS
2633      020320      105765 000077      TSTB     SUCCS+1(R5)      ; SUCCESSFUL YET?
2634      020324      001054      BNE      UNSUC      ;NO, CHECK COUNT
2635      020326      PRINTX   #RECOV,RETRY(R5)
2636      020326      016546 000002      MOV      RETRY(R5),-(SP)
2637      020332      012746 020700      MOV      #RECOV, -(SP)
2638      020336      012746 000002      MOV      #2, -(SP)
2639      020342      010600      MOV      SP,R0
2640      020344      104415      TRAP     C$PNTX
2641      020346      062706 000006      ADD      #6,SP
2642      020352      105715      TSTB     (R5)          ;DETERMINE THRESHOLD
2643      020354      100411      BMI      2$          ;IT'S MODIFIED

```

HARDWARE TESTS MACRO M1110 12-JUN-79 12:23 PAGE 45-1
CHKEND / CHECK SUCCESS AND DETERMINE RETRY STATUS /

SEQ 0064

```

2632 020356          PRINTX #THRSLO          ;NORMAL
      020356 012746 020760
      020362 012746 000001
      020366 010600
      020370 104415
      020372 062706 000004
2633 020376          BR 3$
2634 020400          PRINTX #THRSHI          ;ENHANCED
      020400 012746 021006
      020404 012746 000001
      020410 010600
      020412 104415
      020414 062706 000004
2635 020420 032715 000400          3$: BIT #BIT8,@R5          ;WRITE OR READ OPERATION?
2636 020424 001003          BNE 4$          ;WRITE
2637 020426 012704 000002          MOV #SFTRD,R4          ;READ
2638 020432 000402          BR 5$
2639 020434 012704 000004          4$: MOV #SFTWR,R4          ;WRITE
2640 020440 004737 021464          5$: CALL LOG
2641 020444 005065 000002          CLR RETRY(R5)          ;RESTORE TO HUNKY-DORY STATE
2642 020450 042715 002200          BIC #BIT10!BIT7,@R5          ;NO RETRY, NORM THRESHOLD
2643 020454 000501          BR CHKRET          ;QUIT
2644
2645 020456 000240          UNSUC: NOP          ;RETRYING; SEE IF HARD YET
2646 020460 032715 001000          BIT #BIT9,@R5          ;TU DATA CHECK ERROR?
2647 020464 001015          BNE 2$          ;YES
2648 020466          PRINTB #RETErr          ;NO-'OTHER-ERROR' ERROR
      020466 012746 021102
      020472 012746 000001
      020474 010600
      020500 104414
      020502 062706 000004
2649 020506 005065 000002          CLR RETRY(R5)          ;NO RETRIES
2650 020512 042715 002200          BIC #BIT10!BIT7,@R5          ;NO RETRY, NORM THRESHOLD
2651 020516 000460          BR CHKRET          ;EXIT
2652 020520 023765 010146 000002 2$: CMP MXRTY,RETRY(R5)          ;YES. DID WE GRADUATE TO HARD?
2653 020526 001425          BEQ HRD1          ;YES
2654 020530 005265 000002          INC RETRY(R5)          ;NO. JUST ANOTHER
2655 020534          PRINTX #RTRYN,RETRY(R5)          ;PRINT OUT
      020534 016546 000002
      020540 012746 021040
      020544 012746 000002
      020550 010600
      020552 104415
      020554 062706 000006
2656 020560 032715 000200          BIT #BIT7,@R5          ;WAS NORMAL THRESHOLD?
2657 020564 001403          BEQ 1$          ;YES-REDUCE GAIN
2658 020566 042715 000200          BIC #BIT7,@R5          ;NO-NORM
2659 020572 000432          BR CHKRET
2660 020574 052715 000200          1$: BIS #BIT7,@R5          ;REDUCED
2661 020600 000427          BR CHKRET          ;DONE
2662 020602 000240          HRD1: NOP          ;HERE IS HARD ERROR.
2663 020604          PRINTX #UNREC
      020604 012746 021060
      020610 012746 000001
      020614 010600
      020616 104415

      MOV #THRSLO,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
      TRAP C$PNTX
      ADD #4,SP

      MOV #THRSHI,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
      TRAP C$PNTX
      ADD #4,SP

      MOV #RETErr,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
      TRAP C$PNTB
      ADD #4,SP

      MOV RETRY(R5),-(SP)
      MOV #RTRYN,-(SP)
      MOV #2,-(SP)
      MOV SP,R0
      TRAP C$PNTX
      ADD #6,SP

      MOV #UNREC,-(SP)
      MOV #1,-(SP)
      MOV SP,R0
      TRAP C$PNTX

```

HARDWARE TESTS MACRO M1110 12-JUN-79 12:23 PAGE 45-2
CHKEND / CHECK SUCCESS AND DETERMINE RETRY STATUS /

SEQ 0065

```

2664 020620 062706 000004
2665 020624 032715 000400          BIT    #BIT8,@R5      ;RD OR WR?
2666 020630 001003          BNE     4$              ;WRITE
2667 020632 012704 000016          MOV     #HRDRD,R4      ;READ
2668 020636 000402          BR      5$              ;LOG IT
2669 020640 012704 000020          4$: MOV     #HRDWR,R4      ;WRITE
2670 020644 004737 021464          5$: CALL    LOG              ;LOG IT
2671 020650 005065 000002          CLR     RETRY(R5)      ;BACK TO NORMAL
2672 020654 042715 002200          BIC     #BIT10!BIT7,@R5 ;NO RETRY, NOT REDUCED
2673 020660 042737 000002 010124 CHKRET: BIC     #BIT1,SYSTAT ;NO SYSTEM RETRY NEXT PASS
2674 020666 042715 000100          BIC     #BIT6,@R5      ;NO MORE HOST DATA CHECK ERROR
2675 020672          POP     R4
2676 020674          POP     R0
2677 020676          RET
2678
2679
2680 020700      045      101      122 RECCV: .ASCIZ  /%ARECOVERED FROM DATA CHECK ERROR RETRY # %D1%N/
2681          .EVEN
2682 020760      045      101      040 THRSLO: .ASCIZ  /%A NORMAL THRESHOLD%N/
2683          .EVEN
2684 021006      045      101      040 THRSHI: .ASCIZ  /%A MODIFIED THRESHOLD %N/
2685          .EVEN
2686 021040      045      101      122 RTRYN:  .ASCIZ  /%ARETRY # %D1%N/
2687          .EVEN
2688 021060      045      101      125 UNREC:  .ASCIZ  /%AUNRECOVERABLE%N/
2689          .EVEN
2690 021102      045      101      117 RETERR: .ASCIZ  /%AOTHER ERROR DURING RETRY : EXIT RETRY%N/
2691          .EVEN
2692

```

```

2694 .SBTTL CHKSUC / INTERPRET SUCCESS CODE /
2695
2696 :++
2697 : CHKSUC - COPY SUCCESS CODE (BYTE) TO SUCCS+1(R5). INTERPRET SUCCESS
2698 : AND IF NOT 0, LOG APPROPRIATE ERROR.
2699 : INPUTS: R0 POINTS TO FND PACKET.
2700 : OUTPUTS: R4 IS ERROR NUMBER IF ERROR. SUCCS(R5) UPDATED.
2701 :--
2702
2703 CHKSUC: NOP
2704 021154 000240 MOV 2(R0),SUCCS(R5) ;R0-->END PACKET
2705 021156 016065 000002 000076 CMPB #E$OK,3(R0) ;GET SUCCESS BYTE
2706 021164 122760 000000 000003 BEQ 12$ ;COMPLETE SUCCESS-EXIT
2707
2708 021174 122760 000001 000003 CMPB #E$TRY,3(R0) ;OK BUT RETRIES?
2709 021202 001012 BNE 20$ ;NO
2710 021204 126527 000100 000002 CMPB CMDSNT(R5),#R$$RD;A READ?
2711 021212 001001 BNE 22$ ;NO
2712
2713 021214 000516 BR 10$ ;NO RETRIES IN MAINTENANCE!
2714 021216 126527 000100 000003 22$: CMPB CMDSNT(R5),#R$$WR ;A WRITE?
2715 021224 001001 BNE 20$ ;NO
2716 021226 000511 BR 10$ ;LOG IT
2717 021230 122760 177737 000003 20$: CMPB #E$NOMO,3(R0) ;NO MOTOR?
2718 021236 001003 BNE 1$ ;NO
2719 021240 012704 000030 MOV #NOMOT,R4 ;YES-
2720 021244 000504 BR 11$ ;LOG
2721
2722 021246 122760 177757 000003 1$: CMPB #E$CKS,3(R0) ;'DATA CHECK'' ERROR?
2723 021254 001003 BNE 2$ ;NO
2724 021256 052715 001000 BIS #BIT9,@R5 ;SET DATA-CHK-ERROR FLAG
2725 021262 000477 BR 12$ ;DONT LOG
2726
2727 021264 126527 000100 000007 2$: CMPB CMDSNT(R5),#R$$SLF ;SELF TEST?
2728 021272 001006 BNE 3$ ;NOPE
2729 021274 105760 000003 TSTB 3(R0) ;YES, NEG. IF ERROR
2730 021300 100070 BPL 12$ ;OK
2731
2732 021302 012704 000044 MOV #SLFER,R4 ;YES-ERROR
2733 021306 000463 BR 11$ ;LOG IT
2734
2735 021310 122760 177740 000003 3$: CMPB #E$SK,3(R0) ;SEEK ERROR?
2736 021316 001003 BNE 4$ ;NO
2737 021320 012704 000024 MOV #SKERR,R4 ;YES-
2738 021324 000454 BR 11$ ;LOG
2739
2740 021326 122760 177767 000003 4$: CMPB #E$NCRT,3(R0) ;NO CART?
2741 021334 001003 BNE 5$ ;NO
2742 021336 012704 000054 MOV #NCART,R4 ;YES-
2743 021342 000445 BR 11$ ;LOG
2744
2745 021344 122760 177720 000003 5$: CMPB #E$CMD,3(R0) ;NO UNDERSTAND HOST?
2746 021352 001003 BNE 6$ ;NO
2747 021354 012704 000040 MOV #CMNDR,R4 ;YES-
2748 021360 000436 BR 11$ ;LOG
2749
2750 021362 122760 177770 000003 6$: CMPB #E$NONX,3(R0) ;NON EXISTENT UNIT?

```

2751	021370	001003			BNE	7\$	:NO
2752	021372	012704	000036		MOV	#NOUNIT,R4	:YES-
2753	021376	000427			BR	11\$	:LOG
2754							
2755	021400	122760	177765	000003	7\$: CMPB	#ESWLOC,3(R0)	:WRITE LOCKED?
2756	021406	001003			BNE	8\$	:NO
2757	021410	012704	000026		MOV	#WRLOCK,R4	:YES-
2758	021414	000420			BR	11\$	:LOG
2759							
2760	021416	122760	177776	000003	8\$: CMPB	#ESPART,3(R0)	:PARTIAL OP?
2761	021424	001003			BNE	9\$	:NO
2762	021426	012704	000034		MOV	#PARTL,R4	:YES-
2763	021432	000411			BR	11\$	:LOG
2764							
2765	021434	122760	177711	000003	9\$: CMPB	#ESREC,3(R0)	:WRONG RECORD?
2766	021442	001003			BNE	10\$	:NO
2767	021444	012704	000042		MOV	#RECERR,R4	:YES-
2768	021450	000402			BR	11\$	:LOG
2769							
2770	021452	012704	000046	10\$:	MOV	#SUCOTL,R4	:UNDEFINED
2771	021456	004737	021464	11\$:	CALL	LOG	:LOG ERROR
2772	021462			12\$:	RET		:RETURN

```

2774 .SBTTL LOG / TO LOG ERROR IN CORRECT PLACE
2775
2776
2777 :++
2778 : LOG - DETERMINE IF ERROR IS FATAL, NON-FATAL OR FATAL AFTER N TRIES
2779 : BY INDEX (ERROR #) INTO DEVICE DATA BLOCK. ADD THE DRIVE # TO
2780 : INDICATE UPPER OR LOWER BYTE AND INCREMENT THAT ERROR UNLESS
2781 : THAT BYTE WOULD OVERFLOW. DETERMINE IF EVL FLAG SET, AND IF SO,
2782 : CHECK THRESHOLD (EVLTHR) AND PRINT APPROPRIATE ERROR MESSAGE
2783 : DESCRIPTION. ABORT THE UNIT IF INDICATED THROUGH DODROP CODE.
2784 : INPUTS: R4 = ERROR CODE
2785 : OUTPUTS: ABNDX(R5) = ERROR CODE. DLV(R5) = 0
2786 :--
2787 LOG: 021464 PUSH R0
2788 021466 PUSH R1
2789 021470 PUSH R3
2790 021472 PUSH R4
2791
2792 021474 011537 002074 MOV @R5,L$UN ;GET UNIT NUMBER
2793 021500 042737 177770 002074 BIC #177770,L$UN ;MASK IT OFF
2794 021506 010465 000004 MOV R4,ABNDX(R5) ;SAVE INDEX IN CASE OF ABORT MESSAGE
2795 021512 012703 000120 MOV #LGOFST,R3 ;OFFSET TO LOW ORDER BYTE (DRIVE0)
2796 021516 060403 ADD R4,R3 ;FORM INDEX OF PARAM. TO UPDATE
2797 021520 060503 ADD R5,R3 ;FORM ABSOLUTE ADDR. THIS UNIT
2798 021522 004737 022470 CALL WHCHDR ;SEE WHICH DRIVE T'WAS
2799 021526 103001 BCC 2$ ;WAS DRIVE 0
2800 021530 005203 INC R3 ;DRIVE 1; POINT TO UPPER BYTE
2801 021532 122713 000377 2$: CMPB #255.,@R3 ;POTENTIAL OVERFLOW POSSIBLE?
2802 021536 001005 BNE LOGOK ;NO
2803 021540 LOGO: ERRDF 0.,OVRFLO,ERRDES ;YES
2804 021540 104455 TRAP C$ERDF
2805 021542 000000 .WORD 0
2806 021544 022364 .WORD OVRFLO
2807 021546 022020 .WORD ERRDES
2808 021550 000512
2809 021552 105213 LOGOK: BR ABO ;ABORT UNIT
2810 021554 111304 INCB @R3 ;INCREMENT THE ERROR
2811 021556 016503 000004 MOV @R3,R4 ;TEMP'LY SAVE IT
2812 021562 012701 007044 MOV ABNDX(R5),R3 ;GET INDEX AGAIN
2813 021566 066501 000004 MOV #RSNTAB,R1 ;FORM ADRS OF MSG
2814 021572 042701 000001 ADD ABNDX(R5),R1 ;LIKE THIS
2815 021576 032737 000004 004400 BIC #BIT0,R1 ;INSURE WORD BOUNDARY
2816 021604 001414 BIT #EVL,FLGLOC ;EVL SELECTED?
2817 021606 123704 002216 BEQ LOGOK2 ;NO-CONT
2818 021612 101011 CMPB EVLTHR,R4 ;YES,OVER THRESHOLD?
2819 021614 010337 021626 BHI LOGOK2 ;NO
2820 021620 011137 021630 MOV R3,DFTL1+2 ;YES,LOAD ERROR #
2821 021624 104455 DFTL1: MOV @R1,DFTL1+4 ;AND MESSAGE ADDR
2822 021626 000000 ERRDF 0,DFTL1,ERRDES ;ERROR
2823 021630 021624 TRAP C$ERDF
2824 021632 022020 .WORD 0
2825 021634 000460 .WORD DFTL1
2826 021636 120327 000014 LOGOK2: BR ABO ;DROP IT
2827 021642 103011 CMPB R3,#BDCOM ;'NEVER FATAL' TYPE?
2828 021644 010337 021656 BHS NTSFT ;NO
2829 021650 011137 021660 MOV R3,LOG1+2 ;YES, ERROR CODE
2830 MOV @R1,LOG1+4 ;DESCRIPTION

```

Address	Offset	Value	Label	Instruction	Comment	Trap	Control
2823	021654	104457	LOG1:	ERRSOFT 0,LOG1,ERRDES			
	021654	000000				TRAP	C\$ERSOFT
	021656	000000				.WORD	0
	021660	021654				.WORD	LOG1
	021662	022020				.WORD	ERRDES
2824	021664	000450		BR LOGO	;EXIT		
2825							
2826	021666	120327	000026	NTSFT: CMPB R3,#WRLOCK	;ONE TRY?		
2827	021672	103411		BLO MABEE	;NO, MAYBE A MULTIPLE		
2828	021674	010337	021706	MOV R3,LOG2+2.	;YES		
2829	021700	011137	021710	MOV @R1,LOG2+4			
2830	021704			LOG2: ERRHRD 0,LOG2,ERRDES	;PRINT HARD MESSAGE		
	021704	104456				TRAP	C\$ERHRD
	021706	000000				.WORD	0
	021710	021704				.WORD	LOG2
	021712	022020				.WORD	ERRDES
2831	021714	000430		BR ABO	;DROP UNIT		
2832							
2833	021716	042704	177400	MABEE: BIC #177400,R4	;NEGATE SIGN EXTEND		
2834	021722	163704	010136	1\$: SUB FTLNM,R4	;SEE IF MULTIPLE OF		
2835	021726	001413		BEQ HRD	;FTLNM-YES!		
2836	021730	103401		BLO SFT	;NO		
2837	021732	000773		BR 1\$	;NOT THERE YET		
2838							
2839	021734	010337	021746	SFT: MOV R3,LOG3+2	;ERROR CODE		
2840	021740	011137	021750	MOV @R1,LOG3+4	;DESCRIPTION		
2841	021744			LOG3: ERRSOFT 0,LOG3,ERRDES			
	021744	104457				TRAP	C\$ERSOFT
	021746	000000				.WORD	0
	021750	021744				.WORD	LOG3
	021752	022020				.WORD	ERRDES
2842	021754	000414		BR LOGO	;EXIT		
2843	021756	010337	021770	HRD: MOV R3,LOG3B+2	;HARD ERROR CODE		
2844	021762	011137	021772	MOV @R1,LOG3B+4	;DESCRIPTION		
2845	021766			LOG3B: ERRHRD 0,LOG3B,ERRDES			
	021766	104456				TRAP	C\$ERHRD
	021770	000000				.WORD	0
	021772	021766				.WORD	LOG3B
	021774	022020				.WORD	ERRDES
2846							
2847	021776	011500		ABO: MOV @R5,R0	;GET UNIT NUMBER		
2848	022000	042700	177770	BIC #177770,R0	;UN-SIGN EXTEND		
2849	022004			DODU R0	;USE LOGICAL # TO DROP		
	022004	104451				TRAP	C\$DODU
2850	022006			LOGO: POP R4	;RESTORE		
2851	022010			POP R3			
2852	022012			POP R1			
2853	022014			POP R0			
2854	022016			RET	;RETURN		
2855							

```

2857
2858
2859
2860
2861
2862 022020          BGNMSG  ERRDES          ;ERROR DESCRIPTION
      022020
2863 022020          PUSH    R0
2864 022022          PUSH    R2
2865 022024 005002    CLR      R2          ;PRESET TO DATA TYPE
2866 022026 032715 000020  BIT    #BIT4,@R5 ;WHAT PACK TYPE?
2867 022032 001401    BEQ      2$          ;DATA
2868 022034 005202    INC      R2          ;COMMAND
2869 022036          PRINTB  #UNIT,<B,DR(R5)>,R2,<B,SYSTAT+1>
      022036 005046
      022040 153716 010125
      022044 010246
      022046 005046
      022050 156516 000060
      022054 012746 022212
      022060 012746 000004
      022064 010600
      022066 104414
      022070 062706 000012
2870 022074 016500 000064
2871 022100 016502 000072
2872 022104          MOV      REC(R5),R0    ;RECORD NUMBER
      022104 005046          MOV      PATTEN(R5),R2 ;DATA EXPECTED
      022106 156516 000077          PRINTB #RECID,R0,<B,CMDSENT(R5)>,<B,R2>,<B,SUCCS+1(R5)>
      022112 005046
      022114 150216
      022116 005046
      022120 156516 000100
      022124 010046
      022126 012746 022272
      022132 012746 000005
      022136 010600
      022140 104414
      022142 062706 000014
2873 022146 005765 000074
2874 022152 001414
2875 022154          TST      DLV(R5)        ;DLV ERROR?
      022154 016546 000074          BEQ      3$          ;NO
      022160 012746 022446          PRINTB #RECID2,DLV(R5) ;YES-PRINT
      022164 012746 000002
      022170 010600
      022172 104414
      022174 062706 000006
2876 022200 005065 000074
2877 022204          CLR      DLV(R5)        ;RESET
2878 022206          POP      R2            ;RESTORE
2879 022210          POP      R0
      022210          ENDMSG          ;EXIT
      022210 104423
2880 022212 045 101 104 UNIT:: .ASCIZ /%ADRIVE# %01%A PAK SENT %01%A FLAG RCVD %03%N/
2881
2882 022272 045 101 102 RECID:: .ASCIZ /%ABLOCK# %04%A COMMAND %02%A EXPCTD %03%A SUCCESS %03%N/

```

L10020: TRAP C\$MSG

2883						.EVEN	
2884	022364	103	101	116	OVRFLO:	.ASCIZ	/CAN'T UPDATE ERROR OR STATISTIC:OVERFLOW PENDING/
2885						.EVEN	
2886	022446	045	101	040	RECID2:	.ASCIZ	/A RCDB WAS %06%N/
2887						.EVEN	

```
2889          .SBTTL WHCHDR /SEE WHICH DRIVE IS ACTIVE
2890
2891          ;+
2892          ; INPUTS: NONE
2893          ; OUTPUTS: CARRY=DRIVE (1 OR 0)
2894          ; -
2895
2896
2897 022470 000241          WHCHDR: CLC                      ;CLEAR CARRY
2898
2899 022472 105765 000060          TSTB    DR(R5)          ;DR 0?
2900 022476 001401          BEQ      2$                    ;YES
2901 022500 000261          SEC                      ;NO
2902
2903 022502          2$:    RET                      ;RETURN
2904
```

```

2906 .SBTTL CHKSUM / FORM THE PACKET CHECKSUM
2907
2908
2909 ;+
2910 ; INPUTS: R0 -> (POINTS TO) TOP OF PACKET
2911 ;          R1 = # OF BYTES
2912 ; OUTPUTS: R0 -> WHERE TO PUT CHECKSUM
2913 ;          R1 = CHECKSUM
2914 ; -
2915
2916 022504 CHKSUM: PUSH R3
2917 022506          PUSH R2
2918 022510 042737 000001 010124          BIC #BIT0,SYSTAT ;'CHECKSUM IS ODD' BIT
2919 022516 032701 000001          BIT #BIT0,R1 ;AN ODD # OF BYTES?
2920 022522 001403          BEQ 1$ ;NO
2921 022524 052737 000001 010124          BIS #BIT0,SYSTAT ;YES
2922
2923 022532 006001          1$: ROR R1 ;/2 FOR WORDS
2924
2925 022534 005003          2$: CLR R3 ;PREP CHECKSUM WORD
2926
2927 022536 062003          3$: ADD (R0)+,R3 ;FORM SUM
2928 022540 005503          ADC R3 ;WITH CARRY
2929 022542 005301          DEC R1 ;MORE WORDS?
2930 022544 001374          BNE 3$ ;YES
2931
2932 022546 032737 000001 010124          BIT #BIT0,SYSTAT ;WAS IT ODD
2933 022554 001405          BEQ 4$ ;NO
2934 022556 112002          MOVB (R0)+,R2 ;YES GET NEXT BYTE
2935 022560 042702 177400          BIC #177400,R2 ;UN-SIGN EXTEND
2936 022564 060203          ADD R2,R3 ;ADD IT IN
2937 022566 005503          ADC R3 ;AND CARRY JUST IN CASE
2938
2939 022570 010301          4$: MOV R3,R1 ;RETURN IT IN CORRECT PLACE
2940 022572          POP R2 ;RESTORE
2941 022574          POP R3
2942 022576          RET ;RETURN

```

```

2944 .SBTTL CKCKSM /MODULE TO CHECK THE CHKSUMS
2945
2946
2947 :+
2948 : INPUTS: R4 = THE PACKET BYTE COUNT
2949 :          R0 -> THE PACKET TOP
2950 : OUTPUTS: CARRY SET IF CHECKSUM CALC'D DOES NOT EQUAL CHECKSUM SENT
2951 :          R0 -> THE PACKET TOP
2952 :-
2953
2954 CKCKSM: PUSH R1
2955          PUSH R0          ;SAVE
2956          MOV R4,R1        ;COPY BYTE COUNT TO CORRECT
2957          CALL CHKSUM      ;REGISTER FOR CHKSUM AND
2958                               ;FORM CHECKSUM
2959
2960 ;HERE R0 --> XMITTED CHKSUM, R1=CHKSUM CALC'D
2961
2962          CMPB (R0)+,R1     ;LOWER ORDER CHECK
2963          BNE 2$            ;WRONG
2964
2965          SWAB R1           ;OK-PREP FOR
2966
2967          CMPB (R0)+,R1     ;HIGH ORDER CHECK
2968          BNE 2$            ;WRONG
2969          CLC              ;OK-CLEAR SAILING
2970
2971          BR 3$             ;EXIT
2972
2973 2$: SEC                   ;LET ERROR BE KNOWN
2974
2975
2976 3$: POP R0
2977      POP R1
2978      RET                  ;RETURN
2979

```

```

2981      .SBTTL DOBRK / MODULE TO INIT TU58 AND TEST INTERRUPTS
2982
2983      :++
2984      : DOBRK - SEND RADIAL SERIAL 'BREAK' TO DEVICE:
2985      : - SET 'BREAK' ON INTERFACE.
2986      : - SEND 8. NULLS
2987      : - CLEAR 'BREAK' ON INTERFACE
2988      : - SET VECTORS FOR RCV AND XMIT
2989      : - SEND 2 BYTES OF 'INIT'
2990      : - RECEIVE 'CONTINUE'
2991      : - IF RECEIVE GARBAGE OR TIMEOUT - ERROR
2992      : - CLEAR INTERRUPTS AND VECTORS
2993      : INPUTS: @R5 BIT14 WAS SET - (SEND BREAK)
2994      : OUTPUTS: @R5 BIT14 CLEAR IF SUCCESSFUL INIT.
2995      :          SYSTAT+1 = RECEIVED BYTE
2996      :          ERRORS R4 = ERROR CODE:
2997      :              - SEND NOT READY TIMEOUT (TOSNDB)
2998      :              - NO RESPONSE
2999      :              - DLV ERROR
3000      :              - WRONG RESPONSE
3001      :--
3002
3003 022640 105037 023373 DOBRK: CLRB INITWD+1 ;CLEAR BYTE RECEIVE ADDR
3004 022644 005037 023374 CLR BRKTO ;CLEAR TIME OUT CONSTANT
3005 022650 052775 000001 000026 BIS #BIT0,@XMSR(R5) ;SET 'BREAK'
3006 022656 012765 000001 000100 MOV #R$SNIT,CMD$NT(R5) ;SAY WE SENT 'INIT'
3007 022664 052715 000020 BIS #BIT4,@R5 ;PAK SENT TYPE =COMMAND, SORT OF
3008 022670 012704 000010 MOV #8.,R4 ;BREAK-IT'S-BACK COUNT=8
3009 022674 1$: BREAK ;SUPERVISOR TAKE FIVE
3010 022674 104422 TRAP C$BRK
3011 022676 105775 000026 TSTB @XMSR(R5) ;FOR ^C CHECK, ETC.
3012 022702 100410 BMI 4$ ;READY?
3013 022704 005337 023374 DEC BRKTO ;YES
3014 022710 001371 BNE 1$ ;NO, TIME OUT?
3015 022712 012704 000056 MOV #TOSNDB,R4 ;YES, SET ERROR CODE
3016 022716 004737 021464 CALL LOG ;LOG IT
3017 022722 000535 BR 3$ ;EXIT
3018 022724 113775 023370 000030 4$: MOVB BRKWD,@XMDB(R5) ;SEND NULL
3019 022732 005037 023374 CLR BRKTO ;RESET TIME OUT
3020 022736 005304 DEC R4 ;MORE NULLS TO SEND?
3021 022740 001355 BNE 1$ ;YES
3022 022742 005075 000026 CLR @XMSR(R5) ;NO, CLEAR 'BREAK'
3023 022746 017500 000024 MOV @RCDB(R5),R0 ;HEAVE 'GARBAGE' 1ST BYTE
3024 022752 SETPRI #PRI00 ;SET TO INTERRUPT FO SURE
3025 022752 012700 000000 MOV #PRI00,R0
3026 022756 104441 TRAP C$SPRI
3027 022760 SETVEC TUVECT(R5),#RCVINT,#PRI07 ;SET VECTO INFO
3028 022760 012746 000340 MOV #PRI07,-(SP)
3029 022764 012746 023300 MOV #RCVINT,-(SP)
3030 022770 016546 000204 MOV TUVECT(R5),-(SP)
3031 022774 012746 000003 MOV #3,-(SP)
3032 023000 104437 TRAP C$SVEC
3033 023002 062706 000010 ADD #10,SP
3034 023006 062765 000004 000204 ADD #4,TUVECT(R5) ;AND INC TO SND VECTOR
3035 023014 SETVEC TUVECT(R5),#SNDINT,#PRI07;AND SET IT
3036 023014 012746 000340 MOV #PRI07,-(SP)

```

PC	PC+4	PC+8	PC+12	PC+16	PC+20	PC+24	PC+28	PC+32	PC+36	PC+40	PC+44	PC+48	PC+52	PC+56	PC+60	PC+64	PC+68	PC+72	PC+76	PC+80	PC+84	PC+88	PC+92	PC+96	PC+100	PC+104	PC+108	PC+112	PC+116	PC+120	PC+124	PC+128	PC+132	PC+136	PC+140	PC+144	PC+148	PC+152	PC+156	PC+160	PC+164	PC+168	PC+172	PC+176	PC+180	PC+184	PC+188	PC+192	PC+196	PC+200	PC+204	PC+208	PC+212	PC+216	PC+220	PC+224	PC+228	PC+232	PC+236	PC+240	PC+244	PC+248	PC+252	PC+256	PC+260	PC+264	PC+268	PC+272	PC+276	PC+280	PC+284	PC+288	PC+292	PC+296	PC+300	PC+304	PC+308	PC+312	PC+316	PC+320	PC+324	PC+328	PC+332	PC+336	PC+340	PC+344	PC+348	PC+352	PC+356	PC+360	PC+364	PC+368	PC+372	PC+376	PC+380	PC+384	PC+388	PC+392	PC+396	PC+400	PC+404	PC+408	PC+412	PC+416	PC+420	PC+424	PC+428	PC+432	PC+436	PC+440	PC+444	PC+448	PC+452	PC+456	PC+460	PC+464	PC+468	PC+472	PC+476	PC+480	PC+484	PC+488	PC+492	PC+496	PC+500	PC+504	PC+508	PC+512	PC+516	PC+520	PC+524	PC+528	PC+532	PC+536	PC+540	PC+544	PC+548	PC+552	PC+556	PC+560	PC+564	PC+568	PC+572	PC+576	PC+580	PC+584	PC+588	PC+592	PC+596	PC+600	PC+604	PC+608	PC+612	PC+616	PC+620	PC+624	PC+628	PC+632	PC+636	PC+640	PC+644	PC+648	PC+652	PC+656	PC+660	PC+664	PC+668	PC+672	PC+676	PC+680	PC+684	PC+688	PC+692	PC+696	PC+700	PC+704	PC+708	PC+712	PC+716	PC+720	PC+724	PC+728	PC+732	PC+736	PC+740	PC+744	PC+748	PC+752	PC+756	PC+760	PC+764	PC+768	PC+772	PC+776	PC+780	PC+784	PC+788	PC+792	PC+796	PC+800	PC+804	PC+808	PC+812	PC+816	PC+820	PC+824	PC+828	PC+832	PC+836	PC+840	PC+844	PC+848	PC+852	PC+856	PC+860	PC+864	PC+868	PC+872	PC+876	PC+880	PC+884	PC+888	PC+892	PC+896	PC+900	PC+904	PC+908	PC+912	PC+916	PC+920	PC+924	PC+928	PC+932	PC+936	PC+940	PC+944	PC+948	PC+952	PC+956	PC+960	PC+964	PC+968	PC+972	PC+976	PC+980	PC+984	PC+988	PC+992	PC+996	PC+1000	PC+1004	PC+1008	PC+1012	PC+1016	PC+1020	PC+1024	PC+1028	PC+1032	PC+1036	PC+1040	PC+1044	PC+1048	PC+1052	PC+1056	PC+1060	PC+1064	PC+1068	PC+1072	PC+1076	PC+1080	PC+1084	PC+1088	PC+1092	PC+1096	PC+1100	PC+1104	PC+1108	PC+1112	PC+1116	PC+1120	PC+1124	PC+1128	PC+1132	PC+1136	PC+1140	PC+1144	PC+1148	PC+1152	PC+1156	PC+1160	PC+1164	PC+1168	PC+1172	PC+1176	PC+1180	PC+1184	PC+1188	PC+1192	PC+1196	PC+1200	PC+1204	PC+1208	PC+1212	PC+1216	PC+1220	PC+1224	PC+1228	PC+1232	PC+1236	PC+1240	PC+1244	PC+1248	PC+1252	PC+1256	PC+1260	PC+1264	PC+1268	PC+1272	PC+1276	PC+1280	PC+1284	PC+1288	PC+1292	PC+1296	PC+1300	PC+1304	PC+1308	PC+1312	PC+1316	PC+1320	PC+1324	PC+1328	PC+1332	PC+1336	PC+1340	PC+1344	PC+1348	PC+1352	PC+1356	PC+1360	PC+1364	PC+1368	PC+1372	PC+1376	PC+1380	PC+1384	PC+1388	PC+1392	PC+1396	PC+1400	PC+1404	PC+1408	PC+1412	PC+1416	PC+1420	PC+1424	PC+1428	PC+1432	PC+1436	PC+1440	PC+1444	PC+1448	PC+1452
----	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------	---------

```

3071      .SBTTL  INTERRUPT SERVICE ROUTINE
3072
3073 023264      BGNSRV  SNDINT          ;'SEND' INTERRUPT SERVICE:
      023264                                SNDINT::
3074
3075 023264 042775 000100 000026  SNDHND: BIC      #BIT6,@XMSR(R5) ;DISABLE INTERRUPT
3076 023272 112475 000030          MOV      (R4)+,@XMDB(R5);OUTPUT BYTE
3077 023276      ENDSRV
      023276                                L10021:
      023276 000002                                RTI
3078
3079 023300      BGNSRV  RCVINT          ;'RCV' INTERRUPT SERVICE:
      023300                                RCVINT::
3080
3081 023300 042775 000100 000022  RCVHND: BIC      #BIT6,@RCR(R5) ;DISABLE INTS
3082 023306 017565 000024 000074      MOV      @RCDB(R5),DLV(R5) ;SAVE BYTE
3083 023314 116524 000074          MOV      DLV(R5),(R4)+ ;BYTE TO BUFFER
3084 023320 005765 000074          TST      DLV(R5) ;ERROR?
3085 023324 100402          BMI      10$ ;YES
3086 023326 005065 000074          CLR      DLV(R5) ;NO CLEAR ERROR
3087 023332      10$:
3088 023332      ENDSRV
      023332                                L10022:
      023332 000002                                RTI
3089
3090 023334 000240      WAIT:  NOP          ;WAIT LOOP FOR
3091          ;INTERRUPT SERVICING
3092 023336 020437 023376      CMP      R4,BRKPTR ;IF=,THEN NO INTERRUPT
3093 023342 001011          BNE      1$ ;GOT ONE!
3094 023344          BREAK          ;SUPERVISOR BREAK
      023344 104422          TRAP      C$BRK
3095 023346          BREAK          ;KILL SOME TIME
      023346 104422          TRAP      C$BRK
3096 023350 005337 023374      DEC      BRKTO ;TIME OUT?
3097 023354 001367          BNE      WAIT ;NO...CONT.
3098 023356 012704 000050      MOV      #TORCVB,R4 ;YES LOAD ERROR #
3099 023362 004737 021464      CALL     LOG ;LOG IT
3100 023366      1$:      RET      ;RETURN
3101
3102 023370 000000      BRKWD:  .WORD  0 ;NULL
3103 023372      004      INITWD: .BYTE  R$INIT ;INIT COMMAND
3104 023373      000          .BYTE  0 ;R$CONT IS EXPECTED HERE
3105 023374 000000      BRKTO:  .WORD  0 ;TIME OUT
3106 023376 000000      BRKPTR: .WORD  0 ;POINTER TO INITWD
  
```

```

3108      .SBTTL  COMPAR/DATA COMPARISON MODULE
3109
3110      :++
3111      : COMPAR - IF 'COMPARE DATA' SELECTED, COMPARE EACH DATA BYTE OF PACKET
3112      : TO PATTEN(R5).  SAVE NUMBER OF BYTES NOT CORRECT.  IF NOT
3113      : 0, PRINT SOFT ERROR AND TOTAL # WRONG BYTES.  SET 'BAD_DATA_
3114      : IN PACKET' BIT (BIT6 @R5) FOR HIGHER LEVEL MODULES.
3115      : INPUTS:  - (CMPDAT) FLAG TO NOT COMPARE (=1)
3116      :           - PKPTR(R5) POINTS TO DATA PACK.
3117      : OUTPUTS: - BIT6 @R5 ADJUSTED.
3118      :--
3119
3120      COMPAR: PUSH    R0      ;COMPARE DATA IS DATA PACKET
3121               PUSH    R4      ;TO PATTERN WRITTEN
3122               PUSH    R1      ;USING BYTE COUNT IN PACKET
3123               CLR     BDBYTS   ;CLEAR TOTAL WRONG
3124               MOV     PKPTR(R5),R4 ;GET TOP OF PACKET
3125               TST     CMPDAT   ;COMPARE SELECTED?
3126               BEQ     4$      ;NO-EXIT
3127               INC     R4      ;YES, LOCATE COUNT
3128               MOVB    @R4,R1   ;GET IT
3129               BIC     #177400,R1 ;SIGN-UNEXTEND
3130               ;MUST TEST BYTE-WISE...
3131               INC     R4      ;-->FIRST DATA BYTE
3132               CMPB    PATTEN(R5),(R4)+ ;DATA-WHAT WAS EXPECTED?
3133               BEQ     2$      ;YES
3134               INC     BDBYTS   ;NO, INCREMENT TOTAL WRONG
3135               DEC     R1      ;MORE LEFT?
3136               BNE     1$      ;YES
3137               TST     BDBYTS   ;ANY WRONG?
3138               BEQ     4$      ;NO
3139               MOV     @R5,L$UN ;GET UNIT NUMBER
3140               BIC     #177770,L$UN ;MASK IT OFF
3141               ERRSOFT 0.,M$BDA,ERRDES ;YES-PRINT 'BAD DATA IN PACKET' ERROR
3142               TRAP    C$ERSOFT,
3143               .WORD   0
3144               .WORD   M$BDA
3145               .WORD   ERRDES
3146               PRINTB #DESC,BDBYTS
3147               MOV     BDBYTS,-(SP)
3148               MOV     #DESC,-(SP)
3149               MOV     #2,-(SP)
3150               MOV     SP,R0
3151               TRAP    C$PNTB
3152               ADD     #6,SP
3153
3154               BIS     #BIT6,@R5 ;LET 'EM KNOW UPSTAIRS-BAD DATA FLAG
3155               MOV     #132.,PRNSIZ ;SIZE IS ONE DATA PACK
3156               CALL    PRNPAK      ;AND PRINT THE PACKET
3157               ;RESTORE
3158               POP     R1
3159               POP     R4
3160               POP     R0
3161               RET
3162
3163      BDBYTS: .WORD
3164      DESC:  .ASCIZ  /%ATOTAL BAD BYTES- %D3%A.%N/
3165               .EVEN

```

```

3156          .SBTTL  PRNPAK/MODULE TO PRINT DATA PACKET
3157
3158          :++
3159          : PRNPAK - IF PRINT_DATA_PACK_CN_ERROR SELECTED: PRINT EACH BYTE OF PACKET
3160          :          TO BY PKPTR(R5).
3161          : INPUTS:  PRNSIZ - # OF BYTES IN PACKET.
3162          : OUTPUTS: NONE
3163          :--
3164
3165 023614 000240          PRNPAK: NOP                      :PRINTS 1 PACKET
3166                                     :PKPTR(R5)->TOP OF PACKET
3167                                     :PRNSIZ (PASSED)=BYTE COUNT
3168 023616          PUSH      R0
3169 023620          PUSH      R4
3170 023622 105737 002210  TSTB      PRBUF                  :PRINT PACKET SELECTED?
3171 023626 001451          BEQ      4$                      :NO
3172 023630 016504 000104  MOV      PKPTR(R5),R4             :YES-GET TOP OF PACK
3173 023634 012737 000020  MOV      #16.,LNCNT              :16 BYTES PER LINE
3174 023642 112437 023762  MOV      (R4)+,PRDAT             :AVOID SIGN EXTEND
3175 023646          PRINTF   #PRFORM,<B,PRDAT>             :PRINT BYTE
3176          023646 005046          CLR      -(SP)
3177          023650 153716 023762  BISB      PRDAT,(SP)
3178          023654 012746 023764  MOV      #PRFORM,-(SP)
3179          023660 012746 000002  MOV      #2,-(SP)
3180          023664 010600          MOV      SP,R0
3181          023666 104417          TRAP      C$PNTF
3182          023670 062706 000006  ADD      #6,SP
3183 023674 005337 010154  DEC      PRNSIZ                  :ONE LESS
3184 023700 001414          BEQ      3$                      :NO MORE
3185 023702 005337 023760  DEC      LNCNT                  :NEW LINE?
3186 023706 001355          BNE      2$                      :NOT YET
3187 023710          PRINTF   #CARLF                        :YES
3188          023710 012746 023774  MOV      #CARLF,-(SP)
3189          023714 012746 000001  MOV      #1,-(SP)
3190          023720 010600          MOV      SP,R0
3191          023722 104417          TRAP      C$PNTF
3192          023724 062706 000004  ADD      #4,SP
3193 023730 000741          BR       1$                      :NEXT LINE
3194 023732          PRINTF   #CARLF                        :FINISH UP
3195          023732 012746 023774  MOV      #CARLF,-(SP)
3196          023736 012746 000001  MOV      #1,-(SP)
3197          023742 010600          MOV      SP,R0
3198          023744 104417          TRAP      C$PNTF
3199          023746 062706 000004  ADD      #4,SP
3200 023752          4$:      POP      R4
3201 023754          POP      R0
3202 023756          RET                                :RETURN
3203
3204 023760 000000          LNCNT: .WORD
3205 023762 000000          PRDAT: .WORD
3206 023764 045 117 063  PRFORM: .ASCIIZ  /%03%A /
3207          .EVEN
3208 023774 045 116 000  CARLF:  .ASCIIZ  /%N/
3209          .EVEN

```

SEQ 0080

3195
3196
3197
3198

000144

```
.SBTTL  PATCH AREA
          .REPT    100.
          .WORD
          .ENDR
```

3200
3201
3202
3203
3204 024310 025370
3205 024312 026426
3206 024314 027464
3207 024316 030522
3208 024320 031560
3209 024322 032616
3210 024324 033654
3211 024326 034712
3212
3213
3214
3215
3216
3217 024330 023
3218 024331 023
3219
3220 024332
3221
3222
3223
3224 025370
3225 026426
3226 027464
3227 030522
3228 031560
3229 032616
3230 033654
3231 034712
3232
3233
3249

.SBTTL I/O BUFFER AREAS:

;WHO-GETS-WHAT-SPACE TABLE

BUFTBL: .WORD BUF0
 .WORD BUF1
 .WORD BUF2
 .WORD BUF3
 .WORD BUF4
 .WORD BUF5
 .WORD BUF6
 .WORD BUF7

; ONLY 1 TRANSMIT BUFFER NECESSARY:

.BYTE R\$XOFF
.BYTE R\$XOFF ;SEND XOFF BEFORE EVERY PACKET

\$TRBUF: .BLKB RCBFSZ

BUF0: .BLKB RCBFSZ
BUF1: .BLKB RCBFSZ
BUF2: .BLKB RCBFSZ
BUF3: .BLKB RCBFSZ
BUF4: .BLKB RCBFSZ
BUF5: .BLKB RCBFSZ
BUF6: .BLKB RCBFSZ
BUF7: .BLKB RCBFSZ

HARDWARE TESTS MACRO M1110 12-JUN-79 12:23 PAGE 58
I/O BUFFER AREAS:

E 7

SEQ 0082

3256 035750
3257

ENDMOD

3260
3271
3272
3300
3301 035750
3302
3303
3304
3305
3306
3307
3308
3309
3310
3311
3312 035750
035750 000021
035752
3313
3314
3315 035752
035752 000031
035754 036014
035756 160000
035760 177777
3316 035762
035762 001031
035764 036025
035766 000000
035770 000776
3317 035772
035772 003130
035774 036042
035776 000001
3318 036000
036000 002130
036002 036073
036004 000001
3319 036006
036006 002130
036010 036110
036012 000002
3320
3326
3327 036014
036014
3328
3329 036014 124 125 065
3330 036025 126 105 103
3331 036042 120 104 124
3332 036073 124 105 123
3333 036110 124 105 123
3334
3335
3336
3343

.TITLE PARAMETER CODING

.SBTTL HARDWARE PARAMETER CODING SECTION

BGNMOD

;++

: THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
: THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
: MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
: INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
: MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
: WITH THE OPERATOR.
:--

BGNHRD

.WORD L10023-L\$HARD/2
L\$HARD::

GPRMA MSG1,0,0,160000,177777,YES

.WORD T\$CODE
.WORD MSG1
.WORD T\$LLOLIM
.WORD T\$HILIM

GPRMA MSG1B,2,0,0,776,YES

.WORD T\$CODE
.WORD MSG1B
.WORD T\$LLOLIM
.WORD T\$HILIM

GPRML MSG1C,6,1,YES

.WORD T\$CODE
.WORD MSG1C
.WORD 1

GPRML MSG2,4,1,YES

.WORD T\$CODE
.WORD MSG2
.WORD 1

GPRML MSG3,4,2,YES

.WORD T\$CODE
.WORD MSG3
.WORD 2

ENDHRD

L10023: .EVEN

MSG1: .ASCII /TU58 CSR/
MSG1B: .ASCII /VECTOR ADDR./
MSG1C: .ASCII /PDT (PARALLEL) INTERFACE/
MSG2: .ASCII /TEST DRIVE 0/
MSG3: .ASCII /TEST DRIVE 1/
.EVEN

```

3345      .SBTTL  SOFTWARE PARAMETER CODING SECTION
3346
3347      ;**
3348      ; THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
3349      ; THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES.  THE
3350      ; MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
3351      ; INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES.  THE
3352      ; MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
3353      ; WITH THE OPERATOR.
3354      ;--
3355
3356 036126      BGNSFT
3357      036126      000026
3358      036130      GPRMD  MSG4,0,D,1777,8.,512.,YES
3359      036142      GPRML  MSG4B,10,1,YES
3360      036150      GPRML  MSG5,2,1,YES
3361      036156      GPRML  MSG6,6,1,YES
3362      036164      GPRML  MSG7,4,1,YES
3363      036172      GPRMD  MSG8,10.,D,377,1,254.,YES
3371
3372 036204      SFTOUT: ENDSFT
3373
3374 036204      116      125      115  MSG4:  .ASCIZ  'NUMBER OF BLOCKS:TEST 4-7 (8 TO 512)'
3375 036251      101      104      104  MSG4B: .ASCIZ  /ADD DR # TO DATA PATTERN:TEST 4-7/
3376 036313      123      124      101  MSG5:  .ASCIZ  /STATISTICS PRINTED AT EOP/
3377 036345      103      117      115  MSG6:  .ASCIZ  /COMPARE DATA ON READ/
3378 036372      120      122      111  MSG7:  .ASCIZ  /PRINT PACKET ON ERROR/
3379 036420      043      040      105  MSG8:  .ASCIZ  /# ERRORS = DVC FATAL IF 'EVL'SET/
3380
3381      000016      .EVEN
3381      .REPT  14.      ;LASTAD CORRECTION

```

```

      .WORD  L10024-L$SOFT/2
L$SOFT::
      .WORD  T$CODE
      .WORD  MSG4
      .WORD  1777
      .WORD  T$LOLIM
      .WORD  T$HILIM
      .WORD  T$CODE
      .WORD  MSG4B
      .WORD  1
      .WORD  T$CODE
      .WORD  MSG5
      .WORD  1
      .WORD  T$CODE
      .WORD  MSG6
      .WORD  1
      .WORD  T$CODE
      .WORD  MSG7
      .WORD  1
      .WORD  T$CODE
      .WORD  MSG8
      .WORD  377
      .WORD  T$LOLIM
      .WORD  T$HILIM
      .EVEN
L10024:

```

3382
3383
3390 036516

.WORD
.ENDR
LASTAD

036516 000000
036520 000000
036522
3391 036522
3392 000001

LAST::
ENDMOD
.END

.EVEN
.WORD 0
.WORD 0

PARAMETER CODING
SYMBOL TABLE

MACRO M1110 12-JUN-79 12:23 PAGE 60-2

I 7

SEQ 0086

ABNDX = 000004	CHKEND 020154	C\$PNTB= 000014	ESSK = 177740	G\$RADB= 000000
ABO 021776	CHKERR 020242	C\$PNTF= 000017	ESSLF = 177777	G\$RADD= 000040
ABOMSG 004674	CHKPKS 017620	C\$PNTS= 000016	ESTRY = 000001	G\$RADL= 000120
ABONM 010442	CHKPTR 017616	C\$PNTX= 000015	ESWLOC= 177765	G\$RADO= 000020
ADR = 000020 G	CHKREE 020314	C\$QIO = 000377	ESWR = 177757	G\$XFER= 000004
ASSEMB= 000010	CHKRET 020660	C\$RCVB 010160	FLGLOC 004400 G	G\$YES = 000010
BDATA = 000134	CHKSLC 021154	C\$RDBU= 000007	FM 003372	HARDR = 000136
BDBYTS 023556	CHKSLM 022504	C\$REFG= 000047	FMO 003354	HARDW = 000140
BDCHK = 000022	CHK8 017550	C\$RESE= 000033	FTLNM 010136	HELP = 000000
BDCOM = 000014	CKCKSM 022600	C\$REVI= 000003	F\$AU = 000015	HOE = 100000 G
BIT0 = 000001 G	CLRAIL 002420 G	C\$RFLA= 000021	F\$AUTO= 000020	HRD 021756
BIT00 = 000001 G	CLRBUF 002460 G	C\$SRPT = 000025	F\$BGN = 000040	HRDRD = 000016
BIT01 = 000002 G	CLRPTR 002512	C\$SEFG= 000046	F\$CLEA= 000007	HRDWF = 000020
BIT02 = 000004 G	CMDSNT= 000100	C\$SPRI= 000041	F\$DU = 000016	HRD1 020602
BIT03 = 000010 G	CMNDER= 000040	C\$SVEC= 000037	F\$END = 000041	IBE = 010000 G
BIT04 = 000020 G	CMPDAT 002212	C\$TPRI= 000013	F\$HARD= 000004	IDPTR 010142
BIT05 = 000040 G	CNINIT= 000032	DESC 023560	F\$HW = 000013	IDU = 000040 G
BIT06 = 000100 G	COMPAR 023400	DEVPTR 010130	F\$INIT= 000006	IER = 020000 G
BIT07 = 000200 G	C\$AU = 000052	DEV0 004744	F\$JMP = 000050	IN:T 003626
BIT08 = 000400 G	C\$AUTO= 000061	DEV1 005154	F\$MOD = 000000	INITWD 023372
BIT09 = 001000 G	C\$BRK = 000022	DEV2 005364	F\$MSG = 000011	INIT2 003650
BIT1 = 000002 G	C\$BSEG= 000004	DEV3 005574	F\$PROT= 000021	ISR = 000100 G
BIT10 = 002000 G	C\$BSUB= 000002	DEV4 006004	F\$PWR = 000017	IXE = 004000 G
BIT11 = 004000 G	C\$CEFG= 000045	DEV5 006214	F\$RPT = 000012	I\$AU = 000041
BIT12 = 010000 G	C\$CLCK= 000062	DEV6 006424	F\$SEG = 000003	I\$AUTO= 000041
BIT13 = 020000 G	C\$CLEA= 000012	DEV7 006634	F\$SOFT= 000005	I\$CLN = 000041
BIT14 = 040000 G	C\$CLOS= 000035	DFPTBL 002172 G	F\$SRV = 000010	I\$DU = 000041
BIT15 = 100000 G	C\$CLP1= 000006	DFTL1 021624	F\$SUB = 000002	I\$HRD = 000041
BIT2 = 000004 G	C\$CVEC= 000036	DIAGMC= 000000	F\$SW = 000014	I\$INIT= 000041
BIT3 = 000010 G	C\$DCLN= 000044	DLV = 000074	F\$TEST= 000001	I\$MOD = 000041
BIT4 = 000020 G	C\$DODU= 000051	DOBRK 022640	GBTMP 017524	I\$MSG = 000041
BIT5 = 000040 G	C\$DRPT= 000024	DONE 010140	GBTMP2 017526	I\$PROT= 000040
BIT6 = 000100 G	C\$DU = 000053	DR = 000060	GETANS 016354	I\$PTAB= 000041
BIT7 = 000200 G	C\$EDIT= 000003	DRVCHK 002214	GETHRD 003754	I\$PWR = 000041
BIT8 = 000400 G	C\$ERDF= 000055	EF.CON= 000036 G	GETPTR 016420	I\$RPT = 000041
BIT9 = 001000 G	C\$ERHR= 000056	EF.NEW= 000035 G	GETRS 004642	I\$SEG = 000041
BLKEND= 000202	C\$ERRO= 000060	EF.PWR= 000034 G	GTAGIN 016642	I\$SETU= 000041
BLKER 010150	C\$ERSF= 000054	EF.RES= 000037 G	GTBYTE 017300	I\$SFT = 000041
BLKSIZ= 000210	C\$ERSO= 000057	EF.STA= 000040 G	GTDOWN 017152	I\$SRV = 000041
BLKTBL 004724	C\$ESCA= 000010	ERRDES 022020 G	GTOK 017070	I\$SUB = 000041
BOE = 000400 G	C\$ESEG= 000005	EVL = 000004 G	GTPKS1 016422	I\$STST = 000041
BRKPTR 023376	C\$ESUB= 000003	EVLTHR 002216	GTPKS8 016632	J\$JMP = 000167
BRKTO 023374	C\$ETST= 000001	EXOFF 016631	GTPTR 017276	LENGTH 002204
BRKWD 023370	C\$EXIT= 000032	EXON 016630	GTUM 017032	LGOFST= 000120
BUFTBL 024310	C\$GETB= 000026	E\$ABO = 177720	G\$CNTD= 000200	LNCNT 023760
BUFO 025370	C\$GETW= 000027	E\$CKS = 177757	G\$DELM= 000372	LOE = 040000 G
BUF1 026426	C\$GMAN= 000043	E\$CKSM= 177757	G\$DISP= 000003	LOG 021464
BUF2 027464	C\$GPHR= 000042	E\$CMD = 177720	G\$EXCP= 000400	LOGO 022006
BUF3 030522	C\$GPLO= 000030	E\$END = 002100	G\$HILI= 000002	LOGOK 021552
BUF4 031560	C\$GPR1= 000040	E\$LOAD= 000035	G\$LOLI= 000001	LOGOK2 021636
BUF5 032616	C\$INIT= 000011	E\$NCRT= 177767	G\$NO = 000000	LOGO 021540
BUF6 033654	C\$INLP= 000020	E\$NOMO= 177737	G\$OFFS= 000400	LOG1 021654
BUF7 034712	C\$MANI= 000050	E\$NONX= 177770	G\$OFFSI= 000376	LOG2 021704
CARLF 023774	C\$MEM = 000031	E\$OK = 000000	G\$PRMA= 000001	LOG3 021744
CHECK 003732	C\$MSG = 000023	E\$PART= 177776	G\$PRMD= 000002	LOG3B 021766
CHKANR 017614	C\$NRDY 010156	E\$RD = 177757	G\$PRML= 000000	LQT = 000010 G
CHKANS 017530	C\$OPEN= 000034	E\$REC = 177711	G\$RADA= 000140	LSTDEV 004742

PARAMETER CODING
SYMBOL TABLE

MACRO M1110 12-JUN-79 12:23 PAGE 60-3

J 7

SEQ 0087

L\$ACP 002110 G	L10005 004506	NOMOT = 000030	RECOV 020700	SVCGBL = 000000
L\$APT 002036 G	L10006 004574	NOUNIT = 000036	RETERR 021102	SVCINS = 000001
L\$AU 004722 G	L10007 004640	NOXOFF 016046	RETRY = 000002	SVCSUB = 000001
L\$AUT 002070 G	L10010 004722	NTSFT 021666	RLUN 003236	SVCTAG = 000001
L\$AUTO 004402 G	L10011 010540	NXTRET 010440	RPTR 003240	SVCTST = 000001
L\$CCP 002106 G	L10012 010730	NXTST 010162	RSNTAB 007044	SWAPDR 002240 G
L\$CLEA 004564 G	L10013 011170	NXTST2 010266	RSVP 016002	SWPTR 002340
L\$CO 002032 G	L10014 012562	ONEFIL = 000001	RTRYN 021040	SYSTAT 010124
L\$DEPO 002011 G	L10015 013532	OTL = 000052	RUN 002562 G	\$LSYM = 010000
L\$DESC 002122 G	L10016 014312	OVRFLO 022364	R\$CMND = 000002	TAPLEN 010126
L\$DESP 002076 G	L10017 015262	OVRN = 000012	R\$CONT = 000020	THRSHI 021006
L\$DEVP 002060 G	L10020 022210	O\$APTS = 000000	R\$DASZ = 000204	THRSLO 020760
L\$DISP 002152 G	L10021 023276	O\$AU = 000001	R\$DATA = 000001	TMP = 000066
L\$DLY 002116 G	L10022 023332	O\$BGNR = 000001	R\$DNSZ = 000222	TOMANY 004312
L\$DTP 002040 G	L10023 036014	O\$BGNS = 000001	R\$END = 000002	TORCVB = 000050
L\$DTYP 002034 G	L10024 036204	O\$DU = 000001	R\$INIT = 000004	TOSNDB = 000056
L\$DU 004576 G	MABEE 021716	O\$ERRT = 000000	R\$MSIZ = 000012	TRK = 000062
L\$DUT 002072 G	MSG1 036014	O\$GNSW = 000001	P\$NDSZ = 000016	TRPHND 004512
L\$DVTY 002220 G	MSG1B 036025	O\$POIN = 000001	R\$SNSZ = 000016	TRPPTR 004510
L\$EF 002052 G	MSG1C 036042	O\$SETU = 000000	R\$XOFF = 000023	TSTPC = 000020
L\$ENVI 002044 G	MSG2 036073	PARTL = 000034	R\$XON = 000020	TSTTOP 010144
L\$ETP 002102 G	MSG3 036110	PATTEN = 000072	R\$END = 000100	TST1 010542
L\$EXP1 002046 G	MSG4 036204	PDTFLG 004376 G	R\$NIT = 000001	TST2 010732
L\$EXP4 002064 G	MSG4B 036251	PERDEV 010300	R\$NOP = 000000	TST3 011172
L\$EXP5 002066 G	MSG5 036313	PKPTR = 000104	R\$RD = 000002	TST3PT 012510
L\$HARD 035752 G	MSG6 036345	PNT = 001000 G	R\$SEK = 000005	TST4 012564
L\$HIME 002120 G	MSG7 036372	PRBUF 002210	R\$SLF = 000007	TST4EX 013464
L\$HPCP 002016 G	MSG8 036420	PRDAT 023762	R\$SWR = 000003	TST5 013534
L\$HPTP 002022 G	MXRTRY 010146	PRFORM 023764	SECREC 010152	TST5EX 014244
L\$HW 002172 G	M\$AUTO 004544	PRI = 002000 G	SERVST 017274	TST6 014314
L\$ICP 002104 G	M\$BDA 007156	PRI00 = 000000 G	SETDR 002342 G	TST6EX 015214
L\$INIT 003626 G	M\$CMD 007522	PRI01 = 000040 G	SETLEN 004252	TST7 015264
L\$LADP 002026 G	M\$COM 007222	PRI02 = 000100 G	SETPTR 002416	TST7EX 015774
L\$LAST 036522 G	M\$CHK 007374	PRI03 = 000140 G	SETSRV 017210	TUVECT = 000204
L\$LOAD 002100 G	M\$HDRD 007772	PRI04 = 000200 G	SETUP 002514 G	T\$ARGC = 000001
L\$LUN 002074 G	M\$HDWR 010034	PRI05 = 000240 G	SFPTBL 002204 G	T\$CODE = 005052
L\$MREV 002050 G	M\$INIT 007436	PRI06 = 000300 G	SFT 021734	T\$ERRN = 000000
L\$NAME 002000 G	M\$NLOG 007140	PRI07 = 000340 G	SFTOUT 036204	T\$EXCP = 000000
L\$PRIO 002042 G	M\$NOMO 007264	PRNPAK 023614	SFTRD = 000002	T\$GMAN = 000000
L\$PROT 002142 G	M\$NOTP 007302	PRNSIZ 010154	SFTWR = 000004	T\$HILI = 000376
L\$PRT 002112 G	M\$NRSP 007602	PTR 004672	SKERR = 000024	T\$LAST = 000001
L\$REPP 002062 G	M\$OVRN 010076	RCBCNT 010134	SLFER = 000044	T\$LOLI = 000001
L\$REV 002010 G	M\$PART 007452	RCBFSZ = 001036 G	SND 016052	T\$LSYM = 010000
L\$RPT 002612 G	M\$QRSP 007616	RCDB = 000024	SNDBYT 016304	T\$LTNO = 000007
L\$SOFT 036130 G	M\$REC 007536	RCFLG 010132	SNDCNT = 000070	T\$NEST = 177777
L\$SPC 002056 G	M\$RNIT 007354	RCINIT = 000006	SNDHND 023264	T\$NSO = 000000
L\$SPCP 002020 G	M\$SELF 007202	RCSR = 000022	SNDINT 023264 G	T\$NS1 = 000005
L\$SPTP 002024 G	M\$SFRD 007672	RCVBUF = 000102	SOFT = 000122	T\$PTNU = 000000
L\$STA 002030 G	M\$SFWR 007732	RCVHND 023300	SOFTW = 000124	T\$SAVL = 177777
L\$SW 002204 G	M\$SKER 007124	RCVINT 023300 G	SRVTBL 017254	T\$SEGL = 177777
L\$TEST 002114 G	M\$TOSN 007650	RDNO = 000114	STAEOP 002206	T\$SUBN = 000000
L\$TIML 002014 G	M\$UNIT 007474	RDN1 = 000116	STATHD 003242	T\$TAGL = 177777
L\$UNIT 002012 G	M\$WPRO 007332	REC = 000064	STATUS = 000000	T\$TAGN = 010025
L10001 002202	M\$WRSP 007556	RECDAT 011114	STHD2 003516	T\$TEMP = 000000
L10002 002220	NCART = 000054	RECERR = 000042	STRT 004374 G	T\$TEST = 000007
L10003 003234	NODRVS 004342	RECID 022272 G	SUCCS = 000076	T\$TSTM = 177777
L10004 004310	NOMOR 010444	RECID2 022446	SUCOTL = 000046	T\$TSTS = 000001

PARAMETER CODING
SYMBOL TABLE

MACRO M1110 12-JUN-79 12:23 PAGE 60-4

K 7

SEQ 0088

TSSAU = 010010	TSSRPT= 010003	T4 = 012522 G	UNXPCT 016774	XSALWA= 000000
TSSAUT= 010005	TSSSOF= 010024	T4TRY = 000132	WAIT 023334	XS CNT = 000036
TSSCLE= 010006	TSSSRV= 010022	T5 = 013472 G	WHCHDR 022470	XS FALS= 000040
TSSDU = 010007	TSSSW = 010002	T6 = 014252 G	WRLOCK= 000026	XS FLG = 000034
TSSHAR= 010023	TSSTES= 010017	T7 = 015222 G	WRTNO = 000110	XS OFFS= 000400
TSSHW = 010001	T1 = 010500 G	UAM = 000200 G	WRTN1 = 000112	XS PKNM= 000032
TSSINI= 010004	T1TRY = 000146	UNIT = 022212 G	XFNSND 016034	XS PTR = 000106
TSSMSG= 010020	T2 = 010670 G	UNREC 021060	XMDB = 000030	XS TRUE= 000020
TSSPRO= 010000	T3 = 011130 G	UNSUC 020456	XMSR = 000026	STRBUF 024332

. ABS. 036522 000
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 32864 WORDS (129 PAGES)

DYNAMIC MEMORY: 20308 WORDS (78 PAGES)

ELAPSED TIME: 00:06:43

Z.BIN/FN:ABS:AMA,Z/NL:BEX/CR/-SP=LB1:[1,1]SVC/MLB,SY:[203,230]CZTUU

L 7

Z
 SYMBOL CROSS REFERENCE
 SYMBOL VALUE REFERENCES

CREATED BY MACRO ON 12-JUN-79 AT 12:26
 PAGE 1
 CREF V01

SEQ 0089

ABNDX	= 000004		#20-1557	*47-2794	47-2807	47-2809					
ABO	021776		47-2804	47-2818	47-2831	#47-2847					
ABOMSG	004674		16-1159	#16-1189							
ABONM	010442		*26-1889	*26-1906	26-1911	#26-1918					
ADR	= 000020	G	#7-545								
ASSEMB	= 000010		3-362	3-362							
BDATA	= 000134		12-910	12-920	#20-1611						
BDBYTS	= 023556		*54-3123	*54-3134	54-3137	54-3142	#54-3152				
BDCHK	= 000022		#22-1675	43-2520	43-2572						
BDCOM	= 000014		#22-1672	45-2616	47-2819						
BIT0	= 000001	G	#7-545	10-721	38-2388	47-2810	50-2918	50-2919	50-2921	50-2932	52-3005
BIT00	= 000001	G	#7-545	7-545							
BIT01	= 000002	G	#7-545	7-545							
BIT02	= 000004	G	#7-545	7-545							
BIT03	= 000010	G	#7-545	7-545							
BIT04	= 000020	G	#7-545	7-545							
BIT05	= 000040	G	#7-545	7-545							
BIT06	= 000100	G	#7-545	7-545							
BIT07	= 000200	G	#7-545	7-545							
BIT08	= 000400	G	#7-545	7-545							
BIT09	= 001000	G	#7-545	7-545							
BIT1	= 000002	G	#7-545	26-1880	36-2211	38-2389	41-2464	45-2604	45-2673		
BIT10	= 002000	G	#7-545	26-1876	29-1965	29-1965	29-1966	30-2005	30-2005	31-2031	32-2058
			32-2058	33-2085	35-2156	45-2622	45-2642	45-2650	45-2671		
BIT11	004000	G	#7-545	12-896	13-971	13-992	14-1087				
BIT12	- 010000	G	#7-545	29-1965	29-1965	29-1965	30-2005	30-2005	30-2005	32-2058	32-2058
			32-2058								
BIT13	- 020000	G	#7-545	13-970	13-1017	16-1156					
BIT14	040000	G	#7-545	13-998	26-1895	26-1898	43-2533	43-2545	52-3055		
BIT15	100000	G	#7-545	10-719	13-970	13-1017	14-1087	16-1156	37-2250	37-2272	38-2304
			38-2321	38-2341	38-2352	41-2472	45-2608				
BIT2	- 000004	G	#7-545	38-2390							
BIT3	- 000010	G	#7-545	38-2391							
BIT4	- 000020	G	#7-545	35-2150	35-2154	38-2392	48-2866	52-3007			
BIT5	- 000040	G	#7-545	35-2172	35-2174	38-2393					
BIT6	- 000100	G	#7-545	38-2394	45-2613	45-2674	52-3032	52-3040	52-3047	52-3063	52-3064
			53-3075	53-3081	54-3143						
BIT7	- 000200	G	#7-545	38-2395	45-2642	45-2650	45-2656	45-2658	45-2660	45-2671	
BIT8	000400	G	#7-545	10-751	13-999	35-2160	35-2169	45-2635	45-2664		
BIT9	001000	G	#7-545	10-723	13-1001	35-2144	45-2619	45-2646	46-2724		
BLKEND	000202		13-1009	#20-1632							
BLKER	010150		*13-1040	#24-1793	*35-2141						
BLKSIZ	- 000210		#20-1636	21-1653	21-1654	21-1655	21-1656	21-1657	21-1658	21-1659	21-1660
BLKTBL	004724		10-717	10-748	10-767	11-803	12-890	13-966	13-986	14-1085	16-1180
			#21-1645	26-1874	26-1890	38-2302	41-2470				
BOE	- 000400	G	#7-545								
BRKFTR	023376		*52-3031	*52-3039	*52-3046	53-3092	#53-3106				
BRKTO	023374		*52-3004	*52-3013	*52-3019	*52-3029	*52-3037	*53-3096	#53-3105		
BRKWD	023370		52-3018	#53-3102							
BUFTBL	024310		13-973	#57-3204							
BUFO	025370		57-3204	#57-3224							
BUF1	026426		57-3205	#57-3225							

Z
SYMBOL CROSS REFERENCE

PAGE 2

CREF V01

SEQ 0090

SYMBOL	VALUE	REFERENCES
BUF2	027464	57-3206 #57-3226
BUF3	030522	57-3207 #57-3227
BUF4	031560	57-3208 #57-3228
BUF5	032616	57-3209 #57-3229
BUF6	033654	57-3210 #57-3230
BUF7	034712	57-3211 #57-3231
CARLF	023774	55-3180 55-3182 #55-3191
CHECK	003732	13-976 #13-981
CHKANR	017614	41-2468 41-2476 #41-2480
CHKANS	017530	11-826 #41-2462
CHKEND	020154	43-2525 43-2568 #45-2602
CHKERR	020242	45-2612 #45-2619
CHKPKS	017620	41-2466 41-2474 #43-2500
CHKPTR	017616	*41-2470 41-2471 41-2475 *41-2477 #41-2482
CHKREE	020314	45-2605 #45-2626
CHKRET	020660	45-2610 45-2615 45-2618 45-2621 45-2625 45-2643 45-2651 45-2659 45-2661
CHKSUC	021154	#45-2673
CHKSUM	022504	45-2606 45-2626 #46-2703
		27-1928 28-1940 29-1965 29-1965 29-1966 29-1966 30-2005 30-2005
		30-2005 31-2031 31-2031 32-2058 32-2058 32-2058 33-2085 33-2085 #50-2916
		51-2957
CHK8	017550	41-2465 #41-2470
CKCKSM	022600	43-2518 43-2553 43-2566 #51-2954
CLRALL	002420	#10-767 36-2214
CLRBUF	002460	10-769 #10-783 36-2217
CLRPTR	002512	*10-767 10-768 10-770 *10-772 #10-793
CMDSENT	= 000100	#20-1583 *35-2153 46-2710 46-2714 46-2727 48-2872 *52-3006
CMNDER	= 000040	#22-1682 46-2747
CMPDAT	002212	#6-472 54-3125
CNINIT	= 000032	#22-1679 52-3059
COMPAR	023400	43-2530 43-2555 #54-3120
C\$AU	= 000052	#3-362 17-1224
C\$AUTO	= 000061	#3-362 14-1096
C\$BRK	= 000022	#3-362 11-824 12-889 12-892 12-894 26-1914 35-2188 39-2425 52-3009
		53-3094 53-3095
C\$BSEG	= 000004	#3-362
C\$BSUB	= 000002	#3-362
C\$CEFG	= 000045	#3-362
C\$CLCK	= 000062	#3-362
C\$CLEA	= 000012	#3-362 15-1142
C\$CLOS	= 000035	#3-362
C\$CLP1	= 000006	#3-362
C\$CVEC	= 000036	#3-362 14-1095 52-3065 52-3067
C\$DCLN	= 000044	#3-362 13-984 13-1004
C\$DODU	= 000051	#3-362 14-1107 47-2849
C\$DRPT	= 000024	#3-362 15-1121
C\$DU	= 000053	#3-362 16-1179
C\$EDIT	= 000003	#3-362 3-405
C\$ERDF	= 000055	#3-362 47-2803 47-2817
C\$ERHR	= 000056	#3-362 47-2830 47-2845
C\$ERRO	= 000060	#3-362
C\$ERSF	= 000054	#3-362 13-983 13-1003 26-1913

G
G

Z		CREATED BY		MACRO		ON 12-JUN-79 AT 12:26		PAGE 3		N 7																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																									
---	--	------------	--	-------	--	-----------------------	--	--------	--	-----	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Z SYMBOL		CREATED BY	MACRO ON 12-JUN-79 AT 12:26	PAGE 4	B 8						SEQ 0092
CROSS REFERENCE		VALUE	REFERENCES	CREF	V01						
DOBRK		022640	26-1897	#52-3003							
DONE		010140	*11-802	11-820	#24-1789	*27-1929	*28-1945	*29-1982	*30-2016	*31-2042	*32-2069
			*33-2096								
DR	=	000060	10-721	10-723	*10-725	*10-750	10-751	*10-753	*13-995	13-999	13-1001
			#20-1574	28-1940	29-1964	29-1965	29-1965	29-1966	29-1966	30-2004	30-2005
			30-2005	31-2030	31-2031	31-2031	32-2057	32-2058	32-2058	33-2084	33-2085
			33-2085	48-2869	49-2899						
DRVCHK		002214	#6-473	30-2002	31-2028	32-2055	33-2082				
EF.CON	=	000036	#7-545								
EF.NEW	=	000035	#7-545								
EF.PWR	=	000034	#7-545								
EF.RES	=	000037	#7-545								
EF.STA	=	000040	#7-545	13-963							
ERRDES		022020	47-2803	47-2817	47-2823	47-2830	47-2841	47-2845	#48-2862	54-3141	
EVL	=	000004	#7-545	47-2811							
EVLTHR		002216	#6-474	47-2813							
EXOFF		016631	#37-2285	38-2361	39-2418						
EXON		016630	37-2241	#37-2284	38-2316	39-2426	39-2438				
ESABO	=	177720	#25-1844								
ESCKS	=	177757	#25-1855	25-1857	25-1858	25-1859	46-2722				
ESCKSM	=	177757	#25-1857								
ESCMD	=	177720	#25-1853	46-2745							
ESEND	=	002100	#3-362								
ESLOAD	=	000035	#3-362	3-405							
ESNCRT	=	177767	#25-1845	46-2740							
ESNOMO	=	177737	#25-1852	46-2717							
ESNONX	=	177770	#25-1846	46-2750							
ESOK	=	000000	#25-1847	46-2705							
ESPART	=	177776	#25-1848	46-2760							
ESRD	=	177757	#25-1859								
ESREC	=	177711	#25-1854	46-2765							
ESSK	=	177740	#25-1849	46-2735							
ESSLF	=	177777	#25-1856								
ESTRY	=	000001	#25-1850	46-2708							
ESWLOC	=	177765	#25-1851	46-2755							
ESWR	=	177757	#25-1858								
FLGLOC		004400	*13-1039	#13-1078	47-2811						
FM		003372	12-910	12-920	#12-943						
FMO		003354	12-909	#12-940							
FTLNM		010136	#24-1788	47-2834							
FSAU	=	000015	#3-362	17-1201	17-1224						
FSAUTO	=	000020	#3-362	14-1082	14-1096						
F\$BGN	=	000040	#3-362	3-388	4-409	6-485	7-538	11-830	12-874	12-881	12-949
			13-959	14-1082	15-1118	16-1151	17-1201	18-1273	27-1924	27-1926	28-1934
			28-1936	29-1957	29-1959	30-1995	30-1997	31-2021	31-2023	32-2048	32-2050
			33-2075	33-2077	48-2862	53-3073	53-3079	58-3256	59-3301	59-3312	60-3356
			60-3391								
F\$CLEA	=	000007	#3-362	15-1118	15-1142						
F\$DU	=	000016	#3-362	16-1151	16-1179						
F\$END	=	000041	#3-362	3-362	3-362	3-362	3-362	3-362	3-362	3-362	3-362
			3-362	3-362	3-362	3-362	3-362	3-362	3-362	3-362	3-388
			6-485	7-538	11-830	12-874	12-933	12-949	13-1069	14-1096	15-1142

[illegible]

Z	CREATED BY MACRO ON 12-JUN-79 AT 12:26	PAGE 6	D 8
S	CROSS REFERENCE	CREF V01	SEQ 0094
SYMBOL	VALUE	REFERENCES	
GSRADA	= 000140	#3-362	
GSRADB	= 000000	#3-362	
GSRADD	= 000040	#3-362	60-3358 60-3364
GSRADL	= 000120	#3-362	59-3317 59-3318 59-3319 60-3359 60-3360 60-3361 60-3362
GSRADO	= 000020	#3-362	59-3315 59-3316
G\$XFER	= 000004	#3-362	
G\$YES	= 000010	#3-362	59-3315 59-3316 59-3317 59-3318 59-3319 60-3358 60-3359 60-3360
HARDR	= 000136	60-3361 60-3362 60-3364	
HARDW	= 000140	12-905 12-915 #20-1612	
HELP	= 000000	12-907 12-917 #20-1613	
		#3-347 3-357 3-379	3-397 4-415 4-430 5-452 6-476 #7-490
		7-528 7-547 8-562	8-570 9-585 9-590 9-598 9-605 9-610
		9-616 10-647 10-659	10-664 10-670 10-675 10-681 10-689 10-696
		10-703 10-710 #12-836	13-1047 13-1057 15-1123 15-1130 16-1161 16-1167
		17-1203 17-1210 #18-1231	18-1274 18-1280 57-3234 57-3239 57-3250 #59-3263
		59-3321 59-3337 60-3365	60-3384
HOE	= 100000 G	#7-545	
HRD	021756	47-2835 #47-2843	
HRDRD	= 000016	#22-1673 45-2666	
HRDWR	= 000020	#22-1674 45-2668	
HRD1	020602	45-2653 #45-2662	
IBE	= 010000 G	#7-545	
IDPTR	010142	*11-803 11-804 11-806	*11-808 #24-1790
IDU	= 000040 G	#7-545	
IER	- 020000 G	#7-545	
INIT	003626	#13-961	
INITWD	023372	*52-3003 52-3030 52-3038	52-3045 52-3052 52-3058 #53-3103
INIT2	003650	13-964 #13-966	
ISR	= 000100 G	#7-545	
IXE	= 004000 G	#7-545	
I\$AU	= 000041	#3-362 #17-1201 #17-1224	
I\$AUTO	= 000041	#3-362 #14-1082 #14-1096	
I\$CLN	= 000041	#3-362 #15-1118 #15-1142	
I\$DU	= 000041	#3-362 #16-1151 #16-1179	
I\$HRD	= 000041	#59-3312 #59-3327	
I\$INIT	= 000041	#3-362 #13-959 #13-1069	
I\$MOD	000041	#3-362 3-388 #3-388	6-485 #6-485 7-538 #7-538 11-830 #11-830
		12-874 #12-874 12-949	#12-949 18-1273 #18-1273 58-3256 #58-3256 59-3301
		#59-3301 60-3391 #60-3391	
I\$MSG	= 000041	#3-362 #48-2862 #48-2879	
I\$PROT	= 000040	#3-362 #4-409	
I\$PTAB	= 000041	#3-362	
I\$PWR	= 000041	#3-362	
I\$RPT	= 000041	#3-362	
I\$SEG	= 000041	#12-881 #12-933	
I\$SETU	= 000041	27-1924 28-1934	29-1957 30-1995 31-2021 32-2048 33-2075
I\$SFT	= 000041	#3-362	
I\$SRV	= 000041	#60-3356 #60-3372	
I\$SUB	= 000041	#3-362 #53-3073 #53-3077	#53-3079 #53-3088
I\$TST	= 000041	#3-362 27-1924 28-1934	29-1957 30-1995 31-2021 32-2048 33-2075
		#3-362 27-1924 #27-1924	27-1926 #27-1926 #27-1926 28-1934 #28-1934 28-1936
		#28-1936 #28-1936 29-1957	#29-1957 29-1959 #29-1959 #29-1959 30-1995 #30-1995
		30-1997 #30-1997	31-2021 #31-2021 31-2023 #31-2023 #31-2023 32-2048

Z SYMBOL		CREATED BY	MACRO	ON 12-JUN-79 AT 12:26	PAGE 7	E 8					SEQ 0095
SYMBOL		CROSS REFERENCE	VALUE	REFERENCES	CREF	V01					
J\$JMP	= 000167			#32-2048 32-2050 #32-2050 #32-2050 33-2075 #33-2075 33-2077 #33-2077 #33-2077							
LENGTH	002204			#3-362							
LGOFST	= 000120			#6-469 13-1041							
LNCNT	023760			#20-1602 47-2795							
LOE	= 040000	G		*55-3173 *55-3178 #55-3187							
LOG	021464			#7-545							
LOGO	022006			35-2194 39-2447 43-2521 43-2538 43-2547 43-2573 45-2617 45-2640 45-2669							
LOGOK	021552			46-2771 #47-2787 52-3016 52-3060 53-3099							
LOGOK2	021636			47-2824 47-2842 #47-2850							
LOGO	021540			47-2802 #47-2805							
LOG1	021654			47-2812 47-2814 #47-2819							
LOG2	021704			#47-2803							
LOG3	021744			*47-2821 *47-2822 #47-2823 47-2823							
LOG3B	021766			*47-2828 *47-2829 #47-2830 47-2830							
LOT	= 000010	G		*47-2839 *47-2840 #47-2841 47-2841							
LSTDEV	004742			*47-2843 *47-2844 #47-2845 47-2845							
L\$ACP	002110	G		#7-545							
L\$APT	002036	G		10-727 10-754 10-770 11-806 12-921 13-975 14-1091 #21-1652 26-1884							
L\$AU	004722	G		26-1907 38-2364 41-2475							
L\$AUT	002070	G		#3-405							
L\$AUTO	004402	G		#3-405 #17-1201							
L\$CCP	002106	G		#3-405							
L\$CLEA	004564	G		#3-405 #14-1082							
L\$CO	002032	G		#3-405							
L\$DEPO	002011	G		#3-405							
L\$DESC	002122	G		#3-405							
L\$DESP	002076	G		#3-405							
L\$DEVP	002060	G		#3-405							
L\$DISP	002152	G		#3-405							
L\$DLY	002116	G		#3-405							
L\$DTP	002040	G		#3-405							
L\$DTYP	002034	G		#3-405							
L\$DU	004576	G		#3-405							
L\$DUT	002072	G		#3-405							
L\$DVTY	002220	G		#3-405							
L\$EF	002052	G		#3-405							
L\$ENVI	002044	G		#3-405							
L\$ETP	002102	G		#3-405							
L\$EXP1	002046	G		#3-405							
L\$EXP4	002064	G		#3-405							
L\$EXP5	002066	G		#3-405							
L\$HARD</											

Z
SYMBOL CROSS REFERENCE

MACRO ON 12-JUN-79 AT 12:26

PAGE 8
CREF V01

F 8

SEQ 0096

SYMBOL	VALUE	REFERENCES
L\$LAST	036522 G	3-405 #60-3390
L\$LOAD	002100 G	#3-405
L\$LUN	002074 G	#3-405 *13-989 *47-2792 *47-2793 *54-3139 *54-3140
L\$MREV	002050 G	#3-405
L\$NAME	002000 G	#3-405
L\$PRIO	002042 G	#3-405
L\$PROT	002142 G	3-405 #4-409
L\$PRT	002112 G	#3-405
L\$REPP	002062 G	#3-405
L\$REV	002010 G	#3-405
L\$RPT	002612 G	3-405 #12-881
L\$SOFT	036130 G	3-405 60-3356 #60-3356
L\$SPC	002056 G	#3-405
L\$SPCP	002020 G	#3-405
L\$SPTP	002024 G	#3-405
L\$STA	002030 G	#3-405
L\$SW	002204 G	3-405 6-467 #6-467
L\$TEST	002114 G	#3-405
L\$TML	002014 G	#3-405
L\$UNIT	002012 G	#3-405 13-981 13-1035
L10001	002202	5-445 #5-458
L10002	002220	6-467 #6-483
L10003	003234	#12-933
L10004	004310	#13-1069
L10005	004506	#14-1096
L10006	004574	#15-1142
L10007	004640	#16-1179
L10010	004722	#17-1224
L10011	010540	#27-1926
L10012	010730	#28-1936
L10013	011170	#29-1959
L10014	012562	#30-1997
L10015	013532	#31-2023
L10016	014312	#32-2050
L10017	015262	#33-2077
L10020	022210	#48-2879
L10021	023276	#53-3077
L10022	023332	#53-3088
L10023	036014	59-3312 #59-3327
L10024	036204	60-3356 #60-3372
MABEE	021716	47-2827 #47-2833
MSG1	036014	59-3315 #59-3329
MSG1B	036025	59-3316 #59-3330
MSG1C	036042	59-3317 #59-3331
MSG2	036073	59-3318 #59-3332
MSG3	036110	59-3319 #59-3333
MSG4	036204	60-3358 #60-3374
MSG4B	036251	60-3359 #60-3375
MSG5	036313	60-3360 #60-3376
MSG6	036345	60-3361 #60-3377
MSG7	036372	60-3362 #60-3378
MSG8	036420	60-3364 #60-3379

SYMBOL CROSS REFERENCE		VALUE		REFERENCES		REF V01							
MXRTRY		010146		#24-1792	45-2652								
MSAUTO		004544		14-1104	#14-1109								
MSBDA		007156		#23-1727	54-3141								
MSCMD		007522		23-1712	#23-1749								
MSCOM		007222		23-1702	#23-1731								
MSHCHK		007374		23-1705	#23-1741								
MSHDDR		007772		23-1703	#23-1765								
MSHDWR		010034		23-1704	#23-1767								
MSNIT		007436		23-1709	#23-1743								
MSNLOG		007140		23-1696	23-1700	#23-1725							
MSNOMO		007264		23-1708	#23-1733								
MSNOTP		007302		23-1718	#23-1735								
MSNRSP		007602		23-1716	#23-1755								
MSOVRN		010076		23-1701	#23-1769								
MSPART		007452		23-1710	#23-1745								
MSQRSP		007616		23-1717	#23-1757								
MSREC		007536		23-1713	#23-1751								
MSRNIT		007354		23-1699	#23-1739								
MSSELF		007202		23-1714	#23-1729								
MSFDRD		007672		23-1697	#23-1761								
MSFWR		007732		23-1698	#23-1763								
MSKER		007124		23-1706	#23-1723								
MSOSN		007650		23-1719	#23-1759								
MSUNIT		007474		23-1711	#23-1747								
MSWPRO		007332		23-1707	#23-1737								
MSWRSP		007556		23-1715	#23-1753								
NCART	-	000054		#22-1688	46-2742								
NODRVS		004342		13-1003	#13-1074								
NOMOR		010444		26-1913	#26-1919								
NOMOT	-	000030		#22-1678	46-2719								
NOUNIT	=	000036		#22-1681	46-2752								
NOXOFF		016046		#35-2134									
NTSFT		021666		47-2820	#47-2826								
NXTRET		010440		26-1883	26-1912	#26-1916							
NXTST		010162		11-818	#26-1874								
NXTST2		010266		26-1885	#26-1889								
ONEFIL	=	000001		#2-4	2-8	2-343	3-344	3-383	6-486	7-487	7-500	11-832	
				12-833	12-846	17-1226	18-1227	18-1239	58-3258	59-3259	59-3273		
OTL	=	000052		#22-1687	43-2534								
OVRFLO		022364		47-2803	#48-2884								
OVRN	-	000012		#22-1671	43-2537								
OSAPTS	=	000000		#3-362	3-405								
OSAU	=	000001		#3-362	#3-395	3-405							
OSBGNR	=	000001		#3-362	#3-395	3-405							
OSBGNS	=	000001		#3-362	#3-395	3-405							
OSDU	=	000001		#3-362	#3-395	3-405							
OSERRT	=	000000		#3-362	3-405								
OSGNSW	=	000001		#3-362	#3-395	3-405							
OSPOIN	=	000001		#3-362	#3-395	#3-395	#3-395	#3-395	#3-395	3-395	3-405		
OSSETU	=	000000		#3-362	3-405	60-3390							
PARTL	=	000034		#22-1680	46-2762								
PATTEN	=	000072		*13-1028	#20-1580	*29-1963	*29-1964	29-1965	*30-2001	*30-2004	30-2005	*31-2027	

Z			CREATED BY MACRO ON 12-JUN-79 AT 12:26		PAGE 10		H 8					
SYMBOL	CROSS REFERENCE	VALUE	REFERENCES		CREF	V01						SEQ 0098
PDTFLG		004376 G	*31-2030	*32-2054	*32-2057	32-2058	*33-2081	*33-2084	48-2871	54-3132		
PERDEV		010300	*13-997	13-1024	#13-1077							
PKPTR	=	000104	#26-1891	26-1910								
PNT	=	001000 G	#20-1586	*35-2145	38-2319	38-2358	*38-2360	*43-2507	54-3124	55-3172		
PRBUF		002210	#7-545									
PRDAT		023762	#6-471	55-3170								
PRFORM		023764	*55-3174	55-3175	#55-3188							
PRI	=	002000 G	55-3175	#55-3189								
PRI00	=	000000 G	#7-545									
PRI01	=	000040 G	#7-545	52-3024								
PRI02	=	000100 G	#7-545									
PRI03	=	000140 G	#7-545									
PRI04	=	000200 G	#7-545									
PRI05	=	000240 G	#7-545									
PRI06	=	000300 G	#7-545									
PRI07	=	000340 G	#7-545	14-1084	52-3025	52-3027						
PRNPAK		023614	54-3145	#55-3165								
PRNSIZ		010154	#24-1795	*54-3144	*55-3176							
PTR		004672	*16-1180	16-1181	*16-1184	#16-1187						
RCBCNT		010134	#24-1787	*37-2247	*37-2258	*37-2263	*37-2267	*38-2315	*38-2332	*38-2336	*38-2338	
RCBFSZ		001036 G	*38-2349	*43-2509	43-2516	43-2578						
			10-786	#25-1827	57-3220	57-3224	57-3225	57-3226	57-3227	57-3228	57-3229	
			57-3230	57-3231								
RCDB	=	000024	*13-1020	#20-1565	39-2405	39-2431	52-3023	53-3082				
RCFLG		010132	#24-1786	*37-2248	37-2254	*38-2313	38-2327	*43-2510				
RCINIT	=	000006	#22-1670	43-2546								
RCSR	=	000022	*13-1018	14-1089	#20-1564	39-2403	39-2420	52-3047	52-3064	53-3081		
RCVBUF	=	000102	10-785	*13-973	#20-1585	35-2145	37-2244	43-2501				
RCVHND		023300	#53-3081									
RCVINT		023300 G	52-3025	#53-3079								
RDNO	=	000114	12-910	#20-1590	*35-2163							
RDNI		000116	12-920	#20-1591	*35-2165							
REC	=	000064	*13-1030	#20-1576	*28-1938	28-1940	*29-1962	29-1965	29-1965	29-1966	29-1966	
			*29-1973	29-1974	*29-1977	*29-1978	*30-1998	30-2001	30-2005	30-2005	*30-2008	
			*30-2013	*31-2024	31-2027	31-2031	31-2031	*31-2034	*31-2039	*32-2051	32-2054	
			32-2058	32-2058	*32-2061	*32-2066	*33-2078	33-2081	33-2085	33-2085	*33-2088	
			*33-2093	35-2141	48-2870							
RECDAT		011114	28-1938	28-1943	#28-1948							
RECERR	=	000042	#22-1683	46-2767								
RECID		022272 G	48-2872	#48-2882								
RECID2		022446	48-2875	#48-2886								
RECOV		020700	45-2629	#45-2680								
RETErr		021102	45-2648	#45-2690								
RETRY	=	000002	*13-1029	#20-1556	*45-2623	45-2624	45-2629	*45-2641	*45-2649	45-2652	*45-2654	
			45-2655	*45-2670								
RLUN		003236	*12-899	*12-900	12-909	#12-934						
RPTR		003240	*12-890	12-895	12-921	*12-923	#12-935					
RSNTAB		007044	#23-1696	47-2808								
RSVP		016002	27-1928	28-1940	29-1965	29-1965	29-1965	29-1966	29-1966	30-2005	30-2005	
			30-2005	31-2031	31-2031	32-2058	32-2058	32-2058	33-2085	33-2085	#35-2126	
RTRYN		021040	45-2624	45-2655	#45-2686							

2
 SYMBOL CROSS REFERENCE
 SYMBOL VALUE REFERENCES
 RUN 002562 G #11-818 11-827 27-1925 27-1925 28-1935 28-1935 29-1958 29-1958 30-1996
 RSCMND = 000002 #30-1996 31-2022 31-2022 32-2049 32-2049 33-2076 33-2076
 RSCONT = 000020 #25-1809 25-1815 27-1928 28-1940 28-1940 29-1965 29-1965 29-1966 29-1966
 RSDASZ = 000204 #30-2005 30-2005 31-2031 31-2031 32-2058 32-2058 33-2085 33-2085 35-2151
 RSDATA = 000001 #25-1810 29-1965 29-1965 30-2005 30-2005 32-2058 32-2058 43-2513 52-3052
 #25-1822 25-1824 25-1827 37-2263 43-2551 43-2558
 RSDNSZ = 000222 #25-1814 29-1965 29-1965 29-1966 29-1966 30-2005 30-2005 31-2031 31-2031
 RSEND = 000002 #32-2058 32-2058 33-2085 33-2085 37-2261 38-2334 38-2358 43-2528 43-2549
 #25-1824 38-2336
 R\$INIT = 000004 #25-1815 27-1928 29-1965 29-1965 29-1966 29-1966 30-2005 30-2005 31-2031
 R\$MSIZ = 000012 #31-2031 32-2058 32-2058 33-2085 33-2085 37-2256 38-2330 43-2523 43-2561
 #25-1813 43-2543 53-3103
 #25-1820 25-1826 27-1928 27-1928 28-1940 28-1940 29-1965 29-1965 29-1965
 29-1965 29-1966 29-1966 29-1966 30-2005 30-2005 30-2005 30-2005
 31-2031 31-2031 31-2031 31-2031 32-2058 32-2058 32-2058 32-2058 33-2085
 33-2085 33-2085
 R\$NDSZ = 000016 #25-1818 25-1824 25-1827 27-1928 28-1940 29-1965 29-1965 29-1966 29-1966
 30-2005 30-2005 31-2031 31-2031 32-2058 32-2058 33-2085 33-2085 37-2258
 38-2332
 R\$SNSZ = 000016 #25-1826 27-1928 28-1940 29-1965 29-1965 29-1966 29-1966 30-2005 30-2005
 31-2031 31-2031 32-2058 32-2058 33-2085 33-2085 43-2564
 R\$XOFF = 000023 #25-1812 37-2285 57-3217 57-3218
 R\$XON = 000020 #25-1811 37-2284
 R\$SEND = 000100 #25-1834
 R\$ENIT = 000001 #25-1839 52-3006
 R\$SNOP = 000000 #25-1838
 R\$SRD = 000002 #25-1836 29-1965 29-1966 29-1966 30-2005 31-2031 31-2031 32-2058 33-2085
 33-2085 35-2158 46-2710
 R\$SEK = 000005 #25-1837 28-1940
 R\$SLF = 000007 #25-1840 27-1928 46-2727
 R\$SWR = 000003 #25-1835 29-1965 30-2005 32-2058 35-2167 46-2714
 SECREC 010152 *13-1043 *13-1046 #24-1794 30-2013 31-2039 32-2066 33-2093
 SERVST 017274 *36-2213 38-2368 *38-2383 #38-2397
 SETDR 002342 G #10-748 27-1925 28-1935 29-1958 30-1996 31-2022 32-2049 33-2076
 SETLEN 004252 #13-1041
 SETPTR 002416 *10-748 10-749 10-754 *10-756 #10-759
 SETSRV 017210 38-2306 38-2310 #38-2373
 SETUP 002514 G #11-802 27-1925 27-1925 28-1935 28-1935 29-1958 29-1958 30-1996 30-1996
 31-2022 31-2022 32-2049 32-2049 33-2076 33-2076
 SFPTBL 002204 G #6-467
 SFT 021734 #47-2839
 SFTOUT 036204 #60-3372
 SFTRD = 000002 #22-1668 45-2637
 SFTWR = 000004 #22-1669 45-2639
 SKERR = 000024 #22-1676 46-2737
 SLFER = 000044 #22-1684 46-2732
 SND 016052 35-2133 #35-2135 35-2139
 SNDBYT 016304 35-2135 #35-2183 37-2242 38-2318 38-2362 39-2419 39-2427 39-2439
 SDCNT = 000070 #20-1579 *27-1928 *28-1940 *29-1965 *29-1965 *29-1965 *29-1965 *29-1966 *29-1966
 *30-2005 *30-2005 *30-2005 *30-2005 *31-2031 *31-2031 *32-2058 *32-2058 *32-2058
 *32-2058 *33-2085 *33-2085 *35-2132 *35-2138
 #53-3075
 SNDHND 023264

[illegible]

Z
SYMBOL CROSS REFERENCE
SYMBOL VALUE

CREATED BY MACRO ON 12-JUN-79 AT 12:26

PAGE 14
CREF V01

L 8

SEQ 0102

REFERENCES

SVCSUB = 000001
SVCTAG = 000001

SVCTST = 000001
SWAPDR 002240
SWPTR 002340
SYSTAT 010124

S\$LSYM = 010000

TAPLEN 010126

THRSHI 021006
THRSLO 020760
TMP = 000066

G

48-2875	48-2875	48-2875	48-2875	48-2875	48-2875	48-2875	48-2875	48-2875	48-2879
48-2879	48-2879	52-3009	52-3009	52-3009	52-3024	52-3024	52-3024	52-3024	52-3024
52-3024	52-3024	52-3025	52-3025	52-3025	52-3025	52-3025	52-3025	52-3025	52-3025
52-3025	52-3025	52-3025	52-3025	52-3025	52-3025	52-3025	52-3025	52-3025	52-3025
52-3025	52-3025	52-3027	52-3027	52-3027	52-3027	52-3027	52-3027	52-3027	52-3027
52-3027	52-3027	52-3027	52-3027	52-3027	52-3027	52-3027	52-3027	52-3027	52-3027
52-3027	52-3027	52-3065	52-3065	52-3065	52-3065	52-3065	52-3065	52-3065	52-3067
52-3067	52-3067	52-3067	52-3067	52-3067	53-3077	53-3077	53-3077	53-3077	53-3088
53-3088	53-3088	53-3094	53-3094	53-3094	53-3095	53-3095	53-3095	53-3095	54-3141
54-3141	54-3141	54-3141	54-3141	54-3141	54-3141	54-3141	54-3141	54-3141	54-3141
54-3141	54-3141	54-3142	54-3142	54-3142	54-3142	54-3142	54-3142	54-3142	54-3142
54-3142	54-3142	54-3142	54-3142	54-3142	54-3142	54-3142	54-3142	54-3142	54-3142
54-3142	54-3142	55-3175	55-3175	55-3175	55-3175	55-3175	55-3175	55-3175	55-3175
55-3175	55-3175	55-3175	55-3175	55-3175	55-3175	55-3175	55-3175	55-3175	55-3175
55-3175	55-3175	55-3175	55-3175	55-3175	55-3180	55-3180	55-3180	55-3180	55-3180
55-3180	55-3180	55-3180	55-3180	55-3180	55-3180	55-3180	55-3180	55-3180	55-3180
55-3180	55-3180	55-3182	55-3182	55-3182	55-3182	55-3182	55-3182	55-3182	55-3182
55-3182	55-3182	55-3182	55-3182	55-3182	55-3182	55-3182	55-3182	55-3182	59-3312
59-3312	59-3312	59-3315	59-3315	59-3315	59-3315	59-3315	59-3315	59-3315	59-3315
59-3315	59-3315	59-3315	59-3315	59-3315	59-3316	59-3316	59-3316	59-3316	59-3316
59-3316	59-3316	59-3316	59-3316	59-3316	59-3316	59-3316	59-3316	59-3316	59-3317
59-3317	59-3317	59-3317	59-3317	59-3317	59-3317	59-3317	59-3317	59-3317	59-3318
59-3318	59-3318	59-3318	59-3318	59-3318	59-3318	59-3318	59-3318	59-3318	59-3319
59-3319	59-3319	59-3319	59-3319	59-3319	59-3319	59-3319	59-3319	59-3319	59-3327
59-3327	59-3327	60-3356	60-3356	60-3356	60-3358	60-3358	60-3358	60-3358	60-3358
60-3358	60-3358	60-3358	60-3358	60-3358	60-3358	60-3358	60-3358	60-3358	60-3358
60-3358	60-3358	60-3359	60-3359	60-3359	60-3359	60-3359	60-3359	60-3359	60-3359
60-3359	60-3359	60-3360	60-3360	60-3360	60-3360	60-3360	60-3360	60-3360	60-3360
60-3360	60-3360	60-3361	60-3361	60-3361	60-3361	60-3361	60-3361	60-3361	60-3361
60-3361	60-3361	60-3362	60-3362	60-3362	60-3362	60-3362	60-3362	60-3362	60-3362
60-3362	60-3362	60-3364	60-3364	60-3364	60-3364	60-3364	60-3364	60-3364	60-3364
60-3364	60-3364	60-3364	60-3364	60-3364	60-3364	60-3364	60-3364	60-3364	60-3372
60-3372	60-3372	60-3390	60-3390	60-3390	60-3390	60-3390	60-3390	60-3390	60-3390
60-3390	60-3390								
#3-362	#3-370								
#3-362	#3-372	5-458	6-483	12-933	13-1069	14-1096	15-1142	16-1179	
17-1224	27-1926	28-1936	29-1959	30-1997	31-2023	32-2050	33-2077	48-2879	
53-3077	53-3088	59-3327	60-3372						
#3-362	#3-369	27-1924	28-1934	29-1957	30-1995	31-2021	32-2048	33-2075	
#10-716	27-1925	28-1935	29-1958	30-1996	31-2022	32-2049	33-2076		
*10-717	10-718	10-727	*10-729	#10-737					
*13-1038	#24-1775	*26-1880	36-2211	*37-2253	*38-2326	41-2464	*43-2508	45-2604	
*45-2673	48-2869	*50-2918	*50-2921	50-2932	*52-3058				
#3-362	#5-458	#6-483	#12-933	#13-1069	#14-1096	#15-1142	#16-1179	#17-1224	
#27-1926	#28-1936	#29-1959	#30-1997	#31-2023	#32-2050	#33-2077	#48-2879	#53-3077	
#53-3088	#59-3327	#60-3372							
*13-1041	*13-1042	13-1044	#24-1784	30-1999	30-2014	31-2025	31-2040	32-2052	
32-2067	33-2079	33-2094							
45-2634	#45-2684								
45-2632	#45-2682								
#20-1578	*29-1960	29-1978	*29-1979	*30-1999	*30-2006	*30-2014	*31-2025	*31-2032	
*31-2040	*32-2052	*32-2059	*32-2067	*33-2079	*33-2086	*33-2094			

Z CREATED BY MACRO ON 12-JUN-79 AT 12:26

PAGE 15
CREF V01

SEQ 0103

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
TOMANY		004312	13-983 #13-1072
TORCVB	=	000050	#22-1686 39-2446 53-3098
TOSNDB	=	000056	#22-1689 35-2193 52-3015
TRK	=	000062	#20-1575 *30-2000 30-2010 *30-2012 *31-2026 31-2036 *31-2038 *32-2053 32-2063
			*32-2065 *33-2080 33-2090 *33-2092
TRPHND		004512	14-1084 #14-1104
TRPPT		004510	*14-1085 14-1086 14-1091 *14-1093 #14-1097
TSTPC	=	000020	*11-805 #20-1563 26-1882 26-1903 *35-2127
TSTTOP		010144	11-805 #24-1791 *27-1925 *28-1935 *29-1958
TST1		010542	27-1925 #27-1928
TST2		010732	28-1935 #28-1937
TST3		011172	29-1958 #29-1960
TST3PT		012510	29-1963 29-1969 #29-1984
TST4		012564	30-1996 #30-1998
TST4EX		013464	30-2011 #30-2016
TST5		013534	31-2022 #31-2024
TST5EX		014244	31-2037 #31-2042
TST6		014314	32-2049 #32-2051
TST6EX		015214	32-2064 #32-2069
TST7		015264	33-2076 #33-2078
TST7EX		015774	33-2091 #33-2096
TUVECT	=	000204	*13-994 #20-1634 52-3025 *52-3026 52-3027 *52-3028 52-3065 *52-3066 52-3067
			*52-3068
T\$ARGC	=	000001	#3-405 3-405 #3-405 3-405 3-405 #3-405 3-405 3-405 #3-405
			3-405 3-405 #3-405 3-405 3-405 #3-405 3-405 3-405 #3-405
			12-891 12-891 #12-893 12-893 12-893 #12-909 12-909 #12-909 12-909
			12-909 #12-910 12-910 #12-910 12-910 #12-910 12-910 #12-910 12-910
			#12-910 12-910 #12-910 12-910 #12-910 12-910 #12-910 12-910
			12-910 12-910 #12-920 12-920 #12-920 12-920 #12-920 12-920
			12-920 #12-920 12-920 #12-920 12-920 #12-920 12-920 #12-920 12-920
			#12-920 12-920 12-920 #14-1104 14-1104 14-1104 #16-1159 16-1159 #16-1159
			16-1159 16-1159 #45-2624 45-2624 #45-2624 45-2624 45-2624 #45-2629 45-2629
			#45-2629 45-2629 45-2629 #45-2632 45-2632 45-2632 #45-2634 45-2634 45-2634
			#45-2648 45-2648 45-2648 #45-2655 45-2655 #45-2655 45-2655 45-2655 #45-2663
			45-2663 45-2663 #48-2869 48-2869 #48-2869 48-2869 #48-2869 48-2869 #48-2869
			48-2869 48-2869 #48-2872 48-2872 #48-2872 48-2872 #48-2872 48-2872 #48-2872
			48-2872 #48-2872 48-2872 48-2872 #48-2875 48-2875 #48-2875 48-2875 48-2875
			#54-3142 54-3142 #54-3142 54-3142 54-3142 #55-3175 55-3175 #55-3175 55-3175
			55-3175 #55-3180 55-3180 55-3180 #55-3182 55-3182 55-3182 55-3182
T\$CODE	-	005052	#59-3315 59-3315 #59-3315 59-3315 #59-3315 59-3315 #59-3316 59-3316
			59-3316 #59-3316 59-3316 #59-3317 59-3317 #59-3317 59-3317 #59-3317
			#59-3318 59-3318 #59-3318 59-3318 #59-3318 59-3318 #59-3319 59-3319
			59-3319 #59-3319 59-3319 #60-3358 60-3358 #60-3358 60-3358 #60-3358
			#60-3359 60-3359 #60-3359 60-3359 #60-3359 60-3359 #60-3360 60-3360
			60-3360 #60-3360 60-3360 #60-3361 60-3361 #60-3361 60-3361 #60-3361
			#60-3362 60-3362 #60-3362 60-3362 #60-3362 60-3362 #60-3364 60-3364
			60-3364 #60-3364 60-3364 60-3364 #13-1003 13-1003 #26-1913 26-1913 #47-2803 47-2803
T\$ERRN		000000	#3-362 #13-983 13-983 #13-1003 13-1003 #47-2817 47-2817 #47-2823 47-2823 #47-2830 47-2830
			47-2845 #54-3141 54-3141 #47-2841 47-2841 #47-2841 47-2841 #47-2845 47-2845
T\$EXCP	=	000000	#59-3315 59-3315 #59-3316 59-3316 #60-3358 60-3358 #60-3364 60-3364
T\$GMAN		000000	#3-362

Z
SYMBOL CROSS REFERENCEPAGE 16
CREF V01

SEQ 0104

SYMBOL	VALUE	REFERENCES
T\$HILI	= 000376	#59-3315 59-3315 #59-3316 59-3316 #60-3358 60-3358 #60-3364 60-3364
T\$LAST	= 000001	#3-362 #60-3390
T\$LOLI	= 000001	#59-3315 59-3315 #59-3316 59-3316 #60-3358 60-3358 #60-3364 60-3364
T\$LSYM	= 010000	#3-362 3-362 5-458 6-483 12-933 13-1069 #60-3364 60-3364 16-1179
		17-1224 27-1926 28-1936 29-1959 30-1997 31-2023 32-2050 33-2077 48-2879
		53-3077 53-3088 59-3327 60-3372
T\$LTNO	= 000007	#60-3390
T\$NEST	= 177777	#3-362 3-388 #3-388 3-388 4-409 #4-409 4-409 4-413 4-413
		4-413 #4-413 5-445 #5-445 5-445 5-458 5-458 5-458 #5-458
		6-467 #6-467 6-467 6-483 6-483 6-483 #6-483 6-485 6-485
		6-485 #6-485 7-538 #7-538 7-538 11-830 11-830 11-830 #11-830
		12-874 #12-874 12-874 12-881 #12-881 12-881 12-933 12-933 12-933
		#12-933 12-949 12-949 12-949 #12-949 13-959 #13-959 13-959 13-1069
		13-1069 13-1069 #13-1069 14-1082 #14-1082 14-1082 14-1096 14-1096 14-1096
		#14-1096 15-1118 #15-1118 15-1118 15-1142 15-1142 15-1142 #15-1142 16-1151
		#16-1151 16-1151 16-1179 16-1179 16-1179 #16-1179 17-1201 #17-1201 17-1201
		17-1224 17-1224 17-1224 #17-1224 18-1273 #18-1273 18-1273 27-1924 #27-1924
		27-1924 27-1926 27-1926 27-1926 #27-1926 28-1934 #28-1934 28-1934 28-1936
		28-1936 28-1936 #28-1936 29-1957 #29-1957 29-1957 29-1959 29-1959 29-1959
		#29-1959 30-1995 #30-1995 30-1995 30-1997 30-1997 30-1997 #30-1997 31-2021
		#31-2021 31-2021 31-2023 31-2023 31-2023 #31-2023 32-2048 #32-2048 32-2048
		32-2050 32-2050 32-2050 #32-2050 33-2075 #33-2075 33-2075 33-2077 33-2077
		33-2077 #33-2077 48-2862 #48-2862 48-2862 48-2879 48-2879 #48-2879 48-2879
		53-3073 #53-3073 53-3073 53-3077 53-3077 #53-3077 53-3079 #53-3079 53-3079
		53-3079 53-3088 53-3088 53-3088 #53-3088 58-3256 58-3256 58-3256 #58-3256
		59-3301 #59-3301 59-3301 59-3312 #59-3312 59-3312 59-3327 59-3327
		#59-3327 60-3356 #60-3356 60-3356 60-3372 60-3372 60-3372 #60-3372 60-3391
		60-3391 60-3391 #60-3391
T\$NSO	= 000000	#3-388 6-485 #7-538 11-830 #12-874 12-949 #13-959 13-1069 #14-1082
		14-1096 #15-1118 15-1142 #16-1151 16-1179 #17-1201 17-1224 #18-1273 58-3256
T\$NS1	= 000005	#59-3301 60-3391
		#4-409 4-413 #5-445 5-458 #6-467 6-483 #12-881 12-933 #27-1924
		27-1926 #28-1934 28-1936 #29-1957 29-1959 #30-1995 30-1997 #31-2021 31-2023
		#32-2048 32-2050 #33-2075 33-2077 #48-2862 48-2879 #53-3073 53-3077 #53-3079
		53-3088 #59-3312 59-3327 #60-3356 60-3372
T\$PTNU	= 000000	#3-362
T\$SAVL	= 177777	#3-362
T\$SEGL	= 177777	#3-362
T\$SUBN	= 000000	#3-362 #27-1924 #28-1934 #29-1957 #30-1995 #31-2021 #32-2048 #33-2075
T\$TAGL	= 177777	#3-362
T\$TAGN	= 010025	#3-362 4-409 4-409 #4-409 5-445 5-445 #5-445 6-467 6-467
		#6-467 12-881 12-881 #12-881 13-959 13-959 #13-959 14-1082 14-1082
		#14-1082 15-1118 15-1118 #15-1118 16-1151 16-1151 #16-1151 17-1201 17-1201
		#17-1201 27-1924 27-1924 #27-1924 28-1934 28-1934 #28-1934 29-1957 29-1957
		#29-1957 30-1995 30-1995 #30-1995 31-2021 31-2021 #31-2021 32-2048 32-2048
		#32-2048 33-2075 33-2075 #33-2075 48-2862 48-2862 #48-2862 53-3073 53-3073
		#53-3073 53-3079 53-3079 #53-3079 59-3312 59-3312 #59-3312 60-3356 60-3356
		#60-3356
T\$TEMP	= 000000	#4-413 4-413 #4-428 4-428 4-428 #4-428 4-428 4-428 #4-428
		4-428 4-428 #4-428 4-428 4-428 #4-428 4-428 4-428 #4-428
		6-483 #6-485 6-485 #11-830 11-830 #12-933 12-933 #12-949 12-949

Z	CREATED BY	MACRO	ON 12-JUN-79 AT 12:26	PAGE 18	C 9										
SYMBOL	CROSS REFERENCE			CREF	V01	SEQ 0106									
SYMBOL	VALUE	REFERENCES													
JXPCT	016774	#38-2329													
WAIT	023334	52-3033	52-3041	52-3048	#53-3090	53-3097									
WHCHDR	022470	35-2161	35-2175	47-2798	#49-2897										
WRLOCK	= 000026	#22-1677	46-2757	47-2826											
WRTNO	= 000110	12-910	13-1010	13-1012	#20-1588	*35-2177									
WRTN1	= 000112	12-920	#20-1589	*35-2180											
XFNSND	016034	#35-2131													
XMDB	= 000030	*13-1027	#20-1567	35-2196	52-3018	53-3076									
XMSR	= 000026	*13-1022	#20-1566	35-2185	52-3005	52-3011	52-3022	52-3032	52-3040	52-3063					
		53-3075													
XSALWA	= 000000	#3-362													
XSCNT	= 000036	#20-1570	*27-1928	*28-1940	*29-1965	*29-1965	*29-1965	*30-2005	*30-2005	*30-2005					
		*32-2058	*32-2058	*32-2058											
XSFALS	= 000040	#3-362													
X\$FLG	= 000034	#20-1569	*27-1928	*28-1940	*29-1965	*29-1965	*29-1965	29-1965	29-1966	29-1966					
		*30-2005	*30-2005	*30-2005	30-2005	31-2031	31-2031	*32-2058	*32-2058	*32-2058					
		32-2058	33-2085	33-2085	35-2146	37-2237	43-2503								
X\$OFFS	= 000400	#3-362													
X\$PKNM	= 000032	#20-1568	*27-1928	*28-1940	*29-1965	*29-1965	*29-1965	*29-1965	*29-1966	*29-1966					
		*30-2005	*30-2005	*30-2005	*30-2005	*31-2031	*31-2031	*32-2058	*32-2058	*32-2058					
		*32-2058	*33-2085	*33-2085	35-2142	*35-2142	37-2245	38-2308	*38-2312	*38-2323					
		*38-2329	*38-2345	*38-2356	43-2502										
X\$PTR	= 000106	#20-1587	*35-2148	38-2313	*38-2314	38-2315	*38-2363								
X\$TRUE	= 000020	#3-362													
\$TRBUF	024332	27-1928	28-1940	29-1965	29-1965	29-1965	29-1965	29-1966	29-1966	30-2005					
		30-2005	30-2005	30-2005	31-2031	31-2031	32-2058	32-2058	32-2058	32-2058					
		33-2085	33-2085	35-2131	35-2134	35-2140	#57-3220								

MSGNGB

#60-3391	60-3391									
#3-388	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405
#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405
#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405
#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405
#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405
#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405
#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405
#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405	#3-405
#4-409	#4-428	#4-428	#5-445	#5-445	#5-445	#6-467	#6-467	#6-467	#7-538	
#9-583	#9-583	#12-874	#12-881	#12-881	#13-959	#13-959	#14-1082	#14-1082	#15-1118	
#15-1118	#16-1151	#16-1151	#17-1201	#17-1201	#18-1273	#48-2862	#48-2862	#53-3073	#53-3073	
#53-3079	#53-3079	#59-3301	#59-3312	#59-3312	#60-3356	#60-3356	#60-3390	#60-3390		
#3-405	3-405	3-405	3-405	3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-405	3-405	#3-405	3-405	#3-405	3-405	3-405	3-405	3-405	3-405	
#3-407	#3-407	3-407	3-407	#4-428	4-428	#4-428	4-428	#4-428	4-428	
#4-428	4-428	#4-428	4-428	#4-428	4-428	#4-428	4-428	#4-428	4-428	
#5-445	5-445	#6-467	6-467	#9-583	9-583	9-583	9-583	#11-824	11-824	
#12-889	12-889	#12-891	#12-891	12-891	#12-891	12-891	12-891	#12-891	12-891	
12-891	#12-892	12-892	#12-893	#12-893	12-893	#12-893	12-893	12-893	#12-893	
12-893	12-893	#12-894	12-894	#12-909	#12-909	12-909	#12-909	12-909	#12-909	
12-909	12-909	#12-909	12-909	12-909	#12-910	#12-910	12-910	#12-910	12-910	
#12-910	12-910	#12-910	12-910	#12-910	12-910	12-910	#12-910	12-910	#12-910	
12-910	#12-910	12-910	#12-910	12-910	#12-910	12-910	12-910	#12-910	12-910	
12-910	#12-920	#12-920	12-920	#12-920	12-920	#12-920	12-920	#12-920	12-920	
#12-920	12-920	12-920	#12-920	12-920	#12-920	12-920	#12-920	12-920	#12-920	
12-920	#12-920	12-920	12-920	#12-920	12-920	12-920	#12-920	12-920	#12-920	
13-963	#13-963	13-963	#13-964	13-964	#13-983	#13-983	13-983	#13-983	13-983	
#13-983	13-983	#13-983	13-983	#13-984	13-984	#13-990	13-990	#13-990	13-990	
#13-990	13-990	#13-991	13-991	#13-1003	#13-1003	13-1003	#13-1003	13-1003	#13-1003	
13-1003	#13-1003	13-1003	#13-1004	13-1004	#13-1039	13-1039	#13-1039	13-1039	#13-1039	
13-1069	#14-1084	#14-1084	14-1084	#14-1084	14-1084	#14-1084	14-1084	#14-1084	14-1084	
#14-1084	14-1084	14-1084	#14-1095	14-1095	#14-1095	14-1095	#14-1096	14-1096	#14-1104	
#14-1104	14-1104	#14-1104	14-1104	14-1104	#14-1104	14-1104	14-1104	#14-1107	#14-1107	
14-1107	#15-1121	15-1121	#15-1142	15-1142	#16-1159	#16-1159	16-1159	#16-1159	16-1159	
#16-1159	16-1159	16-1159	#16-1159	16-1159	#16-1179	#16-1179	16-1179	#17-1224	17-1224	
#26-1913	#26-1913	26-1913	#26-1913	26-1913	#26-1913	26-1913	#26-1913	26-1913	#26-1914	
26-1914	#27-1926	27-1926	#28-1936	28-1936	#29-1959	29-1959	#30-1997	30-1997	#31-2023	
31-2023	#32-2050	32-2050	#33-2077	33-2077	#35-2188	35-2188	#39-2425	39-2425	#45-2624	
#45-2624	45-2624	#45-2624	45-2624	#45-2624	45-2624	45-2624	#45-2624	45-2624	45-2624	
#45-2629	#45-2629	45-2629	#45-2629	45-2629	#45-2629	45-2629	45-2629	#45-2629	45-2629	
45-2629	#45-2632	#45-2632	45-2632	#45-2632	45-2632	45-2632	#45-2632	45-2632	45-2632	
#45-2634	#45-2634	45-2634	#45-2634	45-2634	45-2634	#45-2634	45-2634	45-2634	#45-2648	
#45-2648	45-2648	#45-2648	45-2648	45-2648	#45-2648	45-2648	45-2648	#45-2655	#45-2655	
45-2655	#45-2655	45-2655	#45-2655	45-2655	45-2655	#45-2655	45-2655	45-2655	#45-2663	

MSGNIN

2	CREATED BY MACRO ON 12-JUN-79 AT 12:26				PAGE 22						
MACRO CROSS REFERENCE				CREF V01		SEQ 0110					
MACRO NAME				REFERENCES							
	#45-2663	45-2663	#45-2663	45-2663	45-2663	#45-2663	45-2663	45-2663	#47-2803	#47-2803	
	47-2803	#47-2803	47-2803	#47-2803	47-2803	#47-2803	47-2803	#47-2817	#47-2817	47-2817	
	#47-2817	47-2817	#47-2817	47-2817	#47-2817	47-2817	#47-2823	#47-2823	47-2823	#47-2823	
	47-2823	#47-2823	47-2823	#47-2823	47-2823	#47-2830	#47-2830	47-2830	#47-2830	47-2830	
	#47-2830	47-2830	#47-2830	47-2830	#47-2841	#47-2841	47-2841	#47-2841	47-2841	#47-2841	
	47-2841	#47-2841	47-2841	#47-2845	#47-2845	47-2845	#47-2845	47-2845	#47-2845	47-2845	
	#47-2845	47-2845	#47-2849	#47-2849	47-2849	#48-2869	#48-2869	48-2869	48-2869	#48-2869	
	48-2869	#48-2869	48-2869	48-2869	#48-2869	48-2869	#48-2869	48-2869	48-2869	#48-2869	
	48-2869	48-2869	#48-2872	#48-2872	48-2872	#48-2872	48-2872	48-2872	48-2872	#48-2872	
	48-2872	48-2872	#48-2872	48-2872	#48-2872	48-2872	#48-2872	48-2872	48-2872	#48-2872	
	48-2872	48-2872	#48-2875	#48-2875	48-2875	#48-2875	48-2875	#48-2875	48-2875	48-2875	
	#48-2875	48-2875	48-2875	#48-2879	48-2879	#52-3009	52-3009	#52-3024	52-3024	#52-3024	
	52-3024	#52-3025	#52-3025	52-3025	#52-3025	52-3025	#52-3025	52-3025	#52-3025	52-3025	
	#52-3025	52-3025	52-3025	#52-3027	#52-3027	52-3027	#52-3027	52-3027	#52-3027	52-3027	
	#52-3027	52-3027	#52-3027	52-3027	52-3027	#52-3065	52-3065	#52-3065	52-3065	#52-3067	
	52-3067	#52-3067	52-3067	#53-3077	53-3077	#53-3088	53-3088	#53-3094	53-3094	#53-3095	
	53-3095	#54-3141	#54-3141	54-3141	#54-3141	54-3141	#54-3141	54-3141	#54-3141	54-3141	
	#54-3142	#54-3142	54-3142	#54-3142	54-3142	#54-3142	54-3142	54-3142	#54-3142	54-3142	
	54-3142	#55-3175	#55-3175	55-3175	55-3175	#55-3175	55-3175	#55-3175	55-3175	55-3175	
	#55-3175	55-3175	55-3175	#55-3180	#55-3180	55-3180	#55-3180	55-3180	55-3180	#55-3180	
	55-3180	55-3180	#55-3182	#55-3182	55-3182	#55-3182	55-3182	55-3182	#55-3182	55-3182	
	55-3182	#59-3312	59-3312	#59-3315	59-3315	59-3315	59-3315	59-3315	#59-3316	59-3316	
	59-3316	59-3316	59-3316	#59-3317	59-3317	59-3317	59-3317	#59-3318	59-3318	59-3318	
	59-3318	#59-3319	59-3319	59-3319	59-3319	#59-3327	59-3327	#60-3356	60-3356	#60-3358	
	60-3358	60-3358	60-3358	60-3358	60-3358	#60-3359	60-3359	60-3359	60-3359	#60-3360	
	60-3360	60-3360	60-3360	#60-3361	60-3361	60-3361	60-3361	#60-3362	60-3362	60-3362	
	60-3362	#60-3364	60-3364	60-3364	60-3364	60-3364	60-3364	#60-3372	60-3372	#60-3390	
	60-3390	#60-3390	60-3390	#60-3390	60-3390						
MSGNTA	#5-458	#5-458	#6-483	#6-483	#12-933	#12-933	#13-1069	#13-1069	#14-1096	#14-1096	
	#15-1142	#15-1142	#16-1179	#16-1179	#17-1224	#17-1224	#27-1926	#27-1926	#28-1936	#28-1936	
	#29-1959	#29-1959	#30-1997	#30-1997	#31-2023	#31-2023	#32-2050	#32-2050	#33-2077	#33-2077	
MSGNTE	#48-2879	#48-2879	#53-3077	#53-3077	#53-3088	#53-3088	#59-3327	#59-3327	#60-3372	#60-3372	
	#27-1924	#27-1924	#28-1934	#28-1934	#29-1957	#29-1957	#30-1995	#30-1995	#31-2021	#31-2021	
	#32-2048	#32-2048	#33-2075	#33-2075							
MSHAPT	#3-405	#3-405									
MSHNAP	#3-405	#3-405									
MSINCR	#3-388	#3-388	#4-409	#4-409	#4-409	#4-409	#5-445	#5-445	#5-445	#5-445	
	#6-467	#6-467	#6-467	#6-467	#7-538	#7-538	#11-824	#12-874	#12-874	#12-881	
	#12-881	#12-881	#12-881	#12-889	#12-891	#12-892	#12-893	#12-894	#12-909	#12-910	
	#12-920	#12-933	#13-959	#13-959	#13-959	#13-959	#13-963	#13-983	#13-984	#13-990	
	#13-1003	#13-1004	#13-1039	#13-1069	#14-1082	#14-1082	#14-1082	#14-1082	#14-1084	#14-1095	
	#14-1096	#14-1104	#14-1107	#15-1118	#15-1118	#15-1118	#15-1121	#15-1121	#15-1142	#16-1151	
	#16-1151	#16-1151	#16-1151	#16-1159	#16-1179	#17-1201	#17-1201	#17-1201	#17-1201	#17-1224	
	#18-1273	#18-1273	#26-1913	#26-1914	#27-1924	#27-1924	#27-1924	#27-1924	#27-1924	#27-1924	
	#27-1926	#28-1934	#28-1934	#28-1934	#28-1934	#28-1934	#28-1934	#28-1936	#29-1957	#29-1957	
	#29-1957	#29-1957	#29-1957	#29-1957	#29-1959	#30-1995	#30-1995	#30-1995	#30-1995	#30-1995	
	#30-1995	#30-1997	#31-2021	#31-2021	#31-2021	#31-2021	#31-2021	#31-2021	#31-2023	#32-2048	
	#32-2048	#32-2048	#32-2048	#32-2048	#32-2048	#32-2050	#33-2075	#33-2075	#33-2075	#33-2075	
	#33-2075	#33-2075	#33-2077	#35-2188	#39-2425	#45-2624	#45-2629	#45-2632	#45-2634	#45-2648	
	#45-2655	#45-2663	#47-2803	#47-2817	#47-2823	#47-2830	#47-2841	#47-2845	#47-2849	#48-2862	
	#48-2862	#48-2862	#48-2862	#48-2869	#48-2872	#48-2875	#48-2879	#52-3009	#52-3024	#52-3025	
	#52-3027	#52-3065	#52-3067	#53-3073	#53-3073	#53-3073	#53-3073	#53-3079	#53-3079	#53-3079	

7 MACRO CROSS REFERENCE MACRO NAME REFERENCES CREATED BY MACRO ON 12-JUN-79 AT 12:26 PAGE 24 REF V01 1 9 SEQ 0112										
MSRNRO MSSETS	#60-3358	#60-3358	#60-3359	#60-3359	#60-3360	#60-3360	#60-3361	#60-3361	#60-3362	#60-3362
	#60-3364	#60-3364								
	#13-990	#13-990	#13-1039	#13-1039						
	#3-388	#3-388	#4-409	#4-409	#5-445	#5-445	#6-467	#6-467	#7-538	#7-538
	#12-874	#12-874	#12-881	#12-881	#13-955	#13-959	#14-1082	#14-1082	#15-1118	#15-1118
	#16-1151	#16-1151	#17-1201	#17-1201	#18-1273	#18-1273	#27-1924	#27-1924	#28-1934	#28-1934
	#29-1957	#29-1957	#30-1995	#30-1995	#31-2021	#31-2021	#32-2048	#32-2048	#33-2075	#33-2075
	#48-2862	#48-2862	#53-3073	#53-3073	#53-3079	#53-3079	#59-3301	#59-3301	#59-3312	#59-3312
	#60-3356	#60-3356								
	#11-824	11-824	#12-889	12-889	#12-891	12-891	#12-892	12-892	#12-893	12-893
MS SVC	#12-894	12-894	#12-909	12-909	#12-910	12-910	#12-920	12-920	#12-933	12-933
	#13-963	13-963	13-983	#13-984	13-984	#13-990	13-990	13-1003	#13-1004	13-1004
	#13-1039	13-1039	#13-1069	13-1069	#14-1084	14-1084	#14-1095	14-1095	#14-1096	14-1096
	#14-1104	14-1104	#14-1107	14-1107	#15-1121	15-1121	#15-1142	15-1142	#16-1159	16-1159
	#16-1179	16-1179	#17-1224	17-1224	26-1913	#26-1914	26-1914	#27-1926	27-1926	#28-1936
	28-1936	#29-1959	29-1959	#30-1997	30-1997	#31-2023	31-2023	#32-2050	32-2050	#33-2077
	33-2077	#35-2188	35-2188	#39-2425	39-2425	#45-2624	45-2624	#45-2629	45-2629	#45-2632
	45-2632	#45-2634	45-2634	#45-2648	45-2648	#45-2655	45-2655	#45-2663	45-2663	47-2803
	47-2817	47-2823	47-2830	47-2841	47-2845	#47-2849	47-2849	#48-2869	48-2869	#48-2872
	48-2872	#48-2875	48-2875	#48-2879	48-2879	#52-3009	52-3009	#52-3024	52-3024	#52-3025
MST LAB	52-3025	#52-3027	52-3027	#52-3065	52-3065	#52-3067	52-3067	#53-3094	53-3094	#53-3095
	53-3095	54-3141	#54-3142	54-3142	#55-3175	55-3175	#55-3180	55-3180	#55-3182	55-3182
	#11-824	#12-889	#12-891	#12-892	#12-893	#12-894	#12-909	#12-910	#12-920	#12-933
	#13-963	#13-983	#13-984	#13-990	#13-1003	#13-1004	#13-1039	#13-1069	#14-1084	#14-1095
	#14-1096	#14-1104	#14-1107	#15-1121	#15-1142	#16-1159	#16-1179	#17-1224	#26-1913	#26-1914
	#27-1926	#28-1936	#29-1959	#30-1997	#31-2023	#32-2050	#33-2077	#35-2188	#39-2425	#45-2624
	#45-2629	#45-2632	#45-2634	#45-2648	#45-2655	#45-2663	#47-2803	#47-2817	#47-2823	#47-2830
	#47-2841	#47-2845	#47-2849	#48-2869	#48-2872	#48-2875	#48-2879	#52-3009	#52-3024	#52-3025
	#52-3027	#52-3065	#52-3067	#53-3094	#53-3095	#54-3141	#54-3142	#55-3175	#55-3180	#55-3182
	#11-824	11-824	#12-889	12-889	#12-891	12-891	#12-892	12-892	#12-893	12-893
MST STL	#12-894	12-894	#12-909	12-909	#12-910	12-910	#12-920	12-920	#12-933	12-933
	#13-963	13-963	#13-983	#13-983	13-983	#13-984	13-984	#13-990	13-990	#13-1003
	#13-1003	13-1003	#13-1004	13-1004	#13-1039	13-1039	#13-1069	13-1069	#14-1084	14-1084
	#14-1095	14-1095	#14-1096	14-1096	#14-1104	14-1104	#14-1107	14-1107	#15-1121	15-1121
	#15-1142	15-1142	#16-1159	16-1159	#16-1179	16-1179	#17-1224	17-1224	#26-1913	#26-1913
	26-1913	#26-1914	26-1914	#27-1926	27-1926	#28-1936	28-1936	#29-1959	29-1959	#30-1997
	30-1997	#31-2023	31-2023	#32-2050	32-2050	#33-2077	33-2077	#35-2188	35-2188	#39-2425
	39-2425	#45-2624	45-2624	#45-2629	45-2629	#45-2632	45-2632	#45-2634	45-2634	#45-2648
	45-2648	#45-2655	45-2655	#45-2663	45-2663	#47-2803	#47-2803	47-2803	#47-2817	#47-2817
	47-2817	#47-2823	#47-2823	47-2823	#47-2830	#47-2830	47-2830	#47-2841	#47-2841	47-2841
MSWORD	#47-2845	#47-2845	47-2845	#47-2849	47-2849	#48-2869	48-2869	#48-2872	48-2872	#48-2875
	48-2875	#48-2879	48-2879	#52-3009	52-3009	#52-3024	52-3024	#52-3025	52-3025	#52-3027
	52-3027	#52-3065	52-3065	#52-3067	52-3067	#53-3094	53-3094	#53-3095	53-3095	#54-3141
	#54-3141	54-3141	#54-3142	54-3142	#55-3175	55-3175	#55-3180	55-3180	#55-3182	55-3182
	#3-405	#3-405	#4-428	#4-428	#4-428	#4-428	#4-428	#4-428	#4-428	#4-428
	#4-428	#13-983	#13-983	#13-983	#13-983	#13-1003	#13-1003	#13-1003	#13-1003	#26-1913
	#26-1913	#26-1913	#26-1913	#47-2803	#47-2803	#47-2803	#47-2803	#47-2817	#47-2817	#47-2817
	#47-2817	#47-2823	#47-2823	#47-2823	#47-2823	#47-2830	#47-2830	#47-2830	#47-2830	#47-2841
	#47-2841	#47-2841	#47-2841	#47-2845	#47-2845	#47-2845	#47-2845	#54-3141	#54-3141	#54-3141
	#54-3141	#59-3315	#59-3315	#59-3316	#59-3316	#59-3317	#59-3317	#59-3318	#59-3318	#59-3319
	#59-3319	#60-3358	#60-3358	#60-3359	#60-3359	#60-3360	#60-3360	#60-3361	#60-3361	#60-3362
	#60-3362	#60-3364	#60-3364	#60-3390	#60-3390					

