

TU58

TU58 PERF EXERCISER
CZTUUB0

AH E649B MC

COPYRIGHT 1979

FICHE 1 OF 1

NOV 1979

000000

MADE IN USA

.REM 8

IDENTIFICATION

PRODUCT CODE: AC-E648B-MC
PRODUCT NAME: CZTUUB0 TU58 PERF EXER
PRODUCT DATE: JULY 1979
MAINTAINER: DIAGNOSTIC ENGINEERING GROUP
AUTHOR: R. J. ROSS

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 ERRORS 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	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

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 TUSB 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.

THE TESTS ARE PERFORMED USING THE SPECIFIED ALGORITHM ON ALL 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.

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.

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

ERROR DESCRIPTIONS:

AN EXPLANATION OF THE EXTENDED ERROR INFORMATION FOLLOWS. SEE ALSO THE SECTION IN THIS LISTING SUBTITLED "ERROR MESSAGE DESCRIPTIONS".

BLOCK #: THE RECORD NUMBER (1 PER 512. BYTES) IN LAST COMMAND PACK.

COMMAND: THE MOST RECENT COMMAND PACKET OF ODE.

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' ERROR 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.

1.2 SYSTEM REQUIREMENTS

1.2.1 HARDWARE

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

TU58 CONTROLLER AND DRIVE(S). DL, DLV, OR PDT COMPATIBLE INTER-
FACE; AND REVISION 'I' TU58 MICROCODE (OR LATER) ASSUMED.

1.2.2 SOFTWARE

THE PROGRAM IS REVISION C DIAGNOSTIC SUPERVISOR COMPATIBLE.
CONSULT XXDP+ USERS MANUAL FOR OPERATING INSTRUCTIONS.

1.3 RELATED DOCUMENTS AND STANDARDS

XXDP+ USERS MANUAL CHQS

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

APPROPRIATE INTERFACE DIAGNOSTICS MAY BE RUN TO ISOLATE INTERFACE
ERRORS.

1.5 ASSUMPTIONS

SYSTEM HARDWARE OTHER THAN TU58(S) IS OPERATIONAL.

2.0 OPERATING INSTRUCTIONS

2.1 HOW TO RUN THIS DIAGNOSTIC

THE DIAGNOSTIC MAY BE INVOLVED WITH A 'START' RESPONSE TO THE SUPERVISOR PROMPT. 'STA'(CR) IS SUFFICIENT. IF THE DEVICE IS NOT AT THE STANDARD ADDRESS AND VECTOR (176500, 300), THEN ANSWER "CHANGE HW?" WITH 'YES' INITIALLY TO SET UP HARDWARE CONFIGURATION TABLES FOR EACH UNIT. THAT INFORMATION IS:

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

VECTOR ADDR. - ADDRESS OF INTERRUPT VECTOR LOCATION.

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

TEST DRO - YES OR NO

TEST DR1 - YES OR NO

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

THE STANDARD ADDRESS AND VECTOR LOCATIONS FOR THE PDT 11/130 ARE 177170 AND 260 RESPECTIVELY.

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 IN SECTION 6.0.

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 A EOP -- SELECTS WHETHER OR NOT TO PRINT INFORMATION AT END OF PASS OR *C. THESE STATISTICS MAY ALSO BE RETRIEVED 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), WRD OR SFT ERROR MESSAGES BECOME DVC FTL ERRORS AFTER 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 MODIFIED THRESHOLD TAPE

```
368 .TITLE PROGRAM HEADER AND TABLES
369 .SBTIL PROGRAM HEADER
395
397 .ENABL ABS,AMA
398 = 2000
400 .NLIS BEX
401 BGNMOD
402
403 :
404 : THE PROGRAM HEADER IS THE INTERFACE BETWEEN
405 : THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
406 :
407
408 002000 POINTER BGNRPT,BGNSW,BGNSFT,BGNAU,BGNDU,BGNSETUP
409
417
418 002000 HEADER (ZTUU,B,0,3600.,1
419
420 002122 DESCRIP <TU58 PERF EXER>
```

```
423  
424  
425  
426  
427  
428 002142  
429 002142 000000  
430 002144 177777  
431 002146 177777  
432 002150  
;***  
;THE PROTECT TABLE IS USED BY THE MONITOR TO WARN THE OPERATOR WHEN HE  
;TRIES TO TEST THE LOAD DEVICE.  
;--  
BGNPROT  
      .WORD 0           ;DEVICE CSR  
      .WORD -1          ;NO MASS BUS  
      .WORD -1          ;NO DRIVE  
ENDPROT
```

440
441
442
443
444
445
446
447
448 002150
449

.SBTTL DISPATCH TABLE

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

DISPATCH 7

```
458 .SBTTL DEFAULT HARDWARE P-TABLE
459
460 ;**
461 ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
462 ; THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
463 ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES.
464 ;--
465
466 00217C BGNHW DFPTBL
467
468 002172 176500 .WORD 176500 ;CSR ADDRESS
469 002174 000300 .WORD 300 ;VECTOR ADDR.
470 002176 000003 .WORD 3 ;TEST DRIVE ZERO AND ONE
471 002200 000000 .WORD 0 ;NOT PDT TYPE INTERFACE
472
473
474
475
476
477
478
479 002202 ENDMW
```

```

482          .SBTIL  SOFTWARE P-TABLE
483
484          ;++
485          ; THE SOFTWARE P-TABLE CONTAINS THE VALUES OF THE PROGRAM
486          ; PARAMETERS THAT CAN BE CHANGED BY THE OPERATOR.
487          ;--
488
489 002202          BGNSW  SFP1BL
490
491 002204 00 010    LENGTH: .WORD 8.          ;TAPE LENGTH
492 002206 000001    STAEOP: .WORD 1          ;PRINT STATISTICS AT EOP
493 002210 000001    PRBUF: .WORD 1          ;PRINT DATA BUF ON COMP. ERROR
494 002212 000001    CMPDAT: .WORD 1          ;COMPARE DATA
495 002214 000001    DRVCHK: .WORD 1          ;ADD DP # TO DATA
496 002216 C00001    EVLTHR: .WORD 1          ;THRESHOLD FOR EVL TEST
497
504
505 002220          ENDSW
506
507 002220          FNDMOD

```

```
520          .TITLE GLOBAL AREAS
521          .SBTTL GLOBAL EQUATES SECTION
549
559
560 002220          BGNMOD
561
562          :++
563          : THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
564          : ARE USED IN MORE THAN ONE TEST.
565          :--
566
567 002220          EQUALS
                    :
                    : BIT DEFINITIONS
                    :
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== 34
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	ALR==	20
000040	ICJ==	40
000100	ISR==	100
000200	UAM==	200
000400	BOE==	400
001000	PNT==	1000
002000	PRJ==	2000
004000	IXE==	4000
010000	IBE==	10000
020000	IER==	20000
040000	LOE==	40000
100000	HOE==	100000

```

582          .SBTTL  ERROR CODE EQUATES
583
584          ;THE ERROR CODE OFFSET VALUES :
585          ;USED BY ROUTINE 'LOG' TO INDEX (BY R5) INTO DEVICE'S DATA BLOCK AND
586          ;INCREMENT STATISTICS.
587
588          000002      SFTRD   ==      2
589          000004      SFTWR   ==      4
590          000006      RCINIT  ==      6
591          000012      OVRN    ==     10.
592          000014      BDCOM   ==     12.
593          000016      HRDRD   ==     14.
594          000020      HRDWR   ==     16.
595          000022      BDCHK   ==     18.
596          000024      SKERR   ==     20.
597          000026      WRLOCK  ==     22.
598          000030      NOMOT   ==     24.
599          000032      CNINIT  ==     26.
600          000034      PARTL   ==     28.
601          000036      NOUNIT  ==     30.
602          000040      CMNDER  ==     32.
603          000042      RECERR  ==     34.
604          000044      SLFER   ==     36.
605          000046      SUCOTL  ==     38.
606          000050      TORCVB  ==     40.
607          000052      OTL     ==     42.
608          000054      NCART   ==     44.
609          000056      TOSNDB  ==     46.
610
611          ;          IN ADDITION, SYSTEM SETUP OR RUNTIME ERRORS ARE:
612
613          ;          100.  -      ALL UNITS ABORTED
614
615          ;          101.  -      MORE THAN 8. UNITS (16 DRIVES) REQUESTED
616
617          ;          102.  -      NEITHER DRIVE SELECTED FOR THIS CONTROLLER
618
619          ;          ALL THE ABOVE ARE CLASSIFIED AS SYSTEM FATAL
620

```

```

623      .SBTTL GENERAL EQUATES
624      :RADIAL SERIAL CODES:
625      :-----
626      :THE FLAG BYTE CODES ARE:
627      000002      RSCMND == 2      ;"COMMAND" PACKET
628      000020      RSCONT == 20     ;"CONTINUE" SINGLE BYTE
629      000020      RSXON  == 20     ;"XON" SINGLE BYTE
630      000023      RSXOFF == 23     ;"XOFF" SINGLE BYTE
631      000004      RSINIT == 4      ;"INIT" SINGLE BYTE
632      000001      RSDATA == 1      ;"DATA" PACKET
633      000002      RSEND  == RSCMND ;"END" PACKET FLAG IS "COMMAND"
634      :-----
635      :END PACK SIZE:
636      000016      RSNSZ  == 14.     ;TOTAL BYTES IN COMMAND PACKET
637      :MESSAGE PACK SIZE:
638      000012      RMSIZ  == 12     ;10. BYTES FOR BYTE COUNT INSIDE CMND PACK
639      :DATA PACK SIZE:
640      000204      RSDSZ  == 132.   ;TOTAL BYTES IN DATA PACKET
641      :DATA + END PACK SIZE:
642      000222      RSDNSZ == RSDSZ+RSNSZ
643
644      000016      RSSNSZ == RMSIZ + 4 ;SIZE FOR SENDING COMMAND PACK
645      001036      RCBFSZ == 4*RSDSZ+RSNSZ ;4 DATA PACKS AND END PACK
646
647      :-----
648      : THE 1ST CODES ARE:
649
650
651      000100      RSSEND == 100     ;END PACK DESCRIPTOR
652      000003      RSSWR  == 3      ;WRITE
653      000002      RSSRD  == 2      ;READ
654      000005      RSSSEK == 5      ;SEEK
655      000000      RSSNOP == 0      ;NO-OPERATION
656      000001      RSSNIT == 1      ;INITIALIZE
657      000007      RSSSLF == 7      ;SELF TEST
658      :-----
659      :THE SUCCESS CODES ARE:
660
661      177720      ESABO  ==-48.     ;BAD COMMAND FROM HOST
662      177767      ESNCR  ==-9.     ;NO CARTRIDGE
663      177770      ESNONX ==-8.     ;NO DRIVE
664      000000      ESOA   ==0       ;OP COMPLETE SUCCESS
665      177776      ESPART ==-2      ;PARTIAL OP
666      177740      ESSK   ==-32.    ;SEEK ERROR
667      000001      ESTRY  ==1       ;RETRY OCCURRED
668      177765      ESWLOC ==-11.    ;WRITE PROTECTED
669      177737      ESNOMO ==-33.    ;MOTOR STOPPED
670      177720      ESCMD  ==-48.    ;COMMAND ERROR
671      177711      ESREC  ==-55.    ;BAD RECORD NUMBER.
672      177757      ESCKS  ==-17.    ;TU CHKSUM ERROR
673      177777      ESSLF  ==-1.     ;SELF TEST ERROR
674      177757      ESCKSM=ESCKS
675      177757      ESWR=ESCKS
676      177757      ESRD=ESCKS
677      :-----

```

680
681
682
683
684
685 002220 002314
686 002222 003046
687 002224 003106
688 002226 002530
689 002230 002314
690 002232 003252
691 002234 002376
692 002236 003146
693 002240 003210
694 002242 002550
695 002244 002300
696 002246 002506
697 002250 002440
698 002252 002612
699 002254 002626
700 002256 002650
701 002260 002676
702 002262 002712
703 002264 002356
704 002266 002732
705 002270 002756
706 002272 002772
707 002274 002456
708 002276 003024

.SBTTL ERROR MESSAGE DESCRIPTIONS

;THE TABLE OF ERROR MESSAGES (ADDRESSES). ABNDX(R5) CONTAINS THE OFFSET
;OF THE REASON. IT'S ABSOLUTE ADDRESS IS RSNTAB + ABNDX(R5).

RSNTAB: MSNLOG
MSSFRD
MSSFWR
MSRNI
MSNLOG
MSOVRN
MSCOM
MSHORD
MSHWR
MSCHK
MSSKER
MSWPRO
MSNOMO
MSNIT
MSPART
MSUNIT
MSCMD
MSREC
MSSELF
MSWRSP
MSNRSP
MSQRSP
MSNOTP
MSTOSN

```

711 ;HERE ARE THE MESSAGES PROPER:
712
713 002300 123 105 105 MSSKER:: .ASCIZ /SEEK ERROR/ ;DEVICE COULD NOT READ HEADER
714 .EVEN
715 002314 123 131 123 MSNIOG:: .ASCIZ /SYSTEM ERROR/ ;DIAGNOSTIC HUNG. BETTER RE-BOOT
716 .EVEN
717 002332 102 101 104 MSBDA:: .ASCIZ /BAD DATA IN PACKET/ ;HOST DATA CHECK FOUND ERROR, DEVICE MAY
718 .EVEN ;HAVE READ CORRECTLY.
719 002356 123 105 114 MSSELF:: .ASCIZ /SELF TEST ERROR/ ;MICRO DIAGNOSTIC FAILED, BUT DEVICE COULD STILL
720 .EVEN ;SEND AN END PACKET.
721 002376 102 101 104 MSCOM:: .ASCIZ /BAD DATA W-O DATA CHECK ERR AT TU/ ;PREVIOUS DATA CHECK
722 .EVEN ;ERROR NOT DUE TO DEVICE READ OPERATION
723 002440 115 117 124 MSNOMO:: .ASCIZ /MOTOR STOPPED/ ;DEVICE COULD NOT GET ANY MEANINGFUL SIGNAL
724 .EVEN ;FROM TAPE.
725 002456 103 101 122 MSNOTP:: .ASCIZ /CARTRIDGE NOT IN PLACE/ ;NO MEDIA OR BAD SWITCH
726 .EVEN
727 002506 127 122 111 MSWPRO:: .ASCIZ /WRITE PROTECTION/ ;CARTRIDGE WRITE PROTECT TAB MISSING OR
728 .EVEN ;SWITCH BAD
729 002530 122 105 103 MSRNIT:: .ASCIZ /RECIEVING INIT/ ;DEVICE SENT INIT REQUEST
730 .EVEN
731 002550 110 117 123 MSHCHK:: .ASCIZ /HOST FOUND PACKET CHECKSUM ERROR/ ;DEVICE SENT PACK WITH
732 .EVEN ;BAD CHECKSUM
733 002612 103 101 116 MSNIT:: .ASCIZ /CAN'T INIT/ ;DEVICE SENT BYTE OTHER THAN "CONTINUE"
734 .EVEN ;DURING INITIALIZATION
735 002626 120 101 122 MSPART:: .ASCIZ /PARTIAL OPERATION/ ;END OF MEDIUM ENCOUNTERED
736 .EVEN
737 002650 042 116 117 MSUNIT:: .ASCIZ /"NON-EXISTENT" DRIVE/ ;DEVICE RECV'D TOO LARGE DRIVE NUMBER
738 .EVEN
739 002676 102 101 104 MSCMD:: .ASCIZ /BAD COMMAND/ ;DEVICE COULD NOT UNDERSTAND HOST
740 .EVEN
741 002712 102 101 104 MSREC:: .ASCIZ /BAD RECORD NO./ ;DEVICE RECV'D TOO LARGE A RECORD NUMBER
742 .EVEN
743 002732 127 122 117 MSWRSP:: .ASCIZ /WRONG SUCCESS CODE/ ;HOST COULD NOT DECIPHER CODE IN END PACK
744 .EVEN
745 002756 116 117 040 MSNRSP:: .ASCIZ /NO RESPONSE/ ;TIME OUT WAITING FOR BYTE IN RCV BUF ON INTERFACE.
746 .EVEN
747 002772 111 116 104 MSQRSP:: .ASCIZ \INDECIPHERABLE FLAG BYTE\ ;HOST COULD NOT UNDERSTAND 1ST BYTE OF
748 .EVEN ;RESPONSE FROM TU AS PROPER PROTOCOL
749 003024 124 111 115 MSTOSN:: .ASCIZ /TIME OUT ON SEND/ ;DLV "READY" NEVER WENT HIGH
750 .EVEN
751 003046 122 105 103 MSSFRD:: .ASCIZ /RECOV. DATA CHECK ERR ON RD OP/ ;TU58 RESPONDED WITH "DATA-CHECK"
752 .EVEN ;ERROR ON READ OP. ;HOST RETRY(S) SUCCESSFUL
753 003106 122 105 103 MSSFWR:: .ASCIZ /RECOV. DATA CHECK ERR ON WR OP/ ;SAME BUT WR OR WR VERIFY OPERATION
754 .EVEN
755 003146 125 116 122 MSHDRD:: .ASCIZ /UNRECOV. DATA CHECK ERR ON RD OP/ ;TU58 RESPONDED WITH "DATA-CHECK"
756 .EVEN ;ERROR ON READ OP. ;RETRIES UNSUCCESSFUL
757 003210 125 116 122 MSHDWR:: .ASCIZ /UNRECOV. DATA CHECK ERR ON WR OP/ ;SAME BUT WR OPERATION
758 .EVEN
759 003252 104 114 126 MSOVRN:: .ASCIZ /DLV ERROR IN RECEIVE/ ;DLV ERROR (THE CONTENTS PRINTED OUT)
760 .EVEN
  
```

```

763 .SBTTL MISC STORAGE AND EQUATES
764
765
766 003300 000000 SYSTAT:: .WORD ;SYSTEM STATUS WORD
767 ;BIT7-BIT15 = 1ST BYTE OF PACK RCVD
768 ;BIT05 = NOT USED
769 ;BIT04 = NOT USED
770 ;BIT03 = NOT USED
771 ;BIT02 = NOT USED
772 ;BIT01 = RETRY OCCURRING
773 ;BIT00 = CHKSUM IS ODD (TEMP STORAGE)
774
775 003302 000000 TAPLEN:: .WORD ;# RECORDS
776 003304 000000 DEVPTR:: .WORD ;->NEXT UNIT ADDRESS
777 003306 000000 RCFLG:: .WORD ;TEMP STORE FOR GTBYTE
778 003310 000000 RCBcnt:: .WORD ;TEMP STORAGE FOR GTBYTE
779 003312 000004 FTLNM: .WORD 4 ;NUMBER OF TRIES BEFORE ABORT
780 003314 000000 DONE:: .WORD ;1 IF TEST ALGORITHM COMPLETE
781 003316 000000 IDPTR:: .WORD ;PTR IN TESTID MACRO
782 003320 000000 TSTTOP: .WORD ;POINTER TO CURRENT TEST
783 003322 000010 MXRTY:: .WORD 8. ;NUMBER OF RETRIES, BEFORE "HARD" ERROR
784 003324 000000 BLKER:: .WORD ;RECORD # FOR ERROR REPORT
785 003326 000000 SECRC:: .WORD ;RECORD # TO START AFTER SWITCHING TRACKS
786 003330 000000 PRNSIZ:: .WORD ;SIZE OF PACK FOR "PRNPAK"
787 003332 000000 ALLGON:: .WORD ;SECONDARY DON'T PRINT STATISTICS FLAG.
788
789
790
791
792 ; TIME OUT CONSTANTS:
793
794 003334 177777 (SNRDY:: -1 ;TIME OUT ON SEND DELAY (DLV NOT READY)
795 003336 000226 (SRCVB:: 150. ;45 SEC. MULTIPLIER REWIND TIME
796 ;CAN BE OPTIMISED PER CPU
```

```

799      .SBTTL DATA BLOCK FORMAT
800      -----
801      ;R5 --> TOP OF 1 OF THE 8 DATA BLOCKS (1 PER UNIT) DURING EXECUTION
802      ;@R5 IS THE STATUS WORD CONTAINING:
803      ;BIT15 = ABORTED
804      ;BIT14 = SEND "BREAK"
805      ;BIT13 = HALTED
806      ;BIT12 = TEMP STOR WRITE MACRO
807      ;BIT11 = UNIT NOT BEING TESTED
808      ;BIT10 = RETRYING
809      ;BIT9 = TUS8 CHKSUM ERROR
810      ;BIT8 = RD/WR OPERATION
811      ;BIT7 = NORMAL/REDUCED THRESHOLD (MACROS)
812      ;BIT6 = HOST DATA COMPARE ERROR
813      ;BIT5 = WR ERIFY OPERATION
814      ;BIT4 = TYPE OF PAK SENT ODATA 1CMD
815      ;BIT3 = NOT USED
816      ;BIT0,1,2=UNIT NO.
817      ;DEVICE STATE
818      ;# OF RETRIES
819      ;ERROR NUMBER FOR LOG
820      ;STORAGE FOR REGISTERS USED IN TEST BODY
821      ;STORED WITH SWAPOW
822      ;RETRIEVED WITH SWAPIN
823      ;
824      ;
825      ;POINTER TO NEXT EXECUTABLE TEST INST.
826      ;DLV RCV STATUS ADDRESS
827      ;DLV RCV DATA ADDRESS
828      ;DLV SND STATUS ADDRESS
829      ;DLV SND DATA ADDRESS
830      ;THE NUMBER OF PACKETS TO RECEIVE
831      ;THE EXPECTED FLAG OF 1ST PACKET
832      ;THE EXPECTED COUNT OF 1ST PACKET
833      ;FOR MULTIPLE PACKET RECIEVES (MAX.4)
834      ;CONSECUTIVE XSFLGS AND XSCNTS
835      ;DR==0 OR 1; BIT8,9 DRIVE SELECTED BY OPERATOR
836      ;COUNTER FOR TRACK NUMBER
837      ;RECORD (BLOCK #)
838      ;
839      ;TEST MACRO REGISTER
840      ;THE # OF BYTES FOR SENDING PACKET
841      ;DATA PATTERN-LOWER BYTE USED
842      ;CONTENTS OF RCDB ON DLV ERROR
843      ;SUCCESS CODE OF LAST END PACKET
844      ;TYPE OF COMMAND CURRENT IN EVEN BYTE; BIT15==VERIFY OP.
845      ;
846      ;POINTER TO 542. BYTE BUFFER (4 DATA PAKS + END PAK)
847      ;POINTER TO TOP OF PACKET
848      ;POINTER TO CURRENTLY USED XSFLG OR XSCNT
849      ;THE # OF 512. BYTE BLOCKS WRITTEN DRO
850      ;THE # OF 512. BYTE BLOCKS WRITTEN DRI
851      ;THE # OF 512. BYTE BLOCKS READ DRO
852      ;THE # OF 512. BYTE BLOCKS READ DRI

817      000000 STATUS == 0.
818      000002 RETRY == 2.
819      000004 ABNDX == 4.
820      ;R0
821      ;R1
822      ;R2
823      ;R3
824      ;R4
825      000020 TSTPC == 16.
826      000022 RCSR == 18.
827      000024 RCDB == 20.
828      000026 XMSR == 22.
829      000030 XMDB == 24.
830      000032 XSPKMN == 26.
831      000034 XSFLG == 28.
832      000036 XSCNT == 30.
833      ;
834      ; BLKW 8.
835      000060 DR == 48.
836      000062 TRK == 50.
837      000064 REC == 52.
838      ;
839      000066 TMP == 54.
840      000070 SDCNT == 56.
841      000072 PATTEN == 58.
842      000074 DLV == 60.
843      000076 SUCCS == 62.
844      000100 CMDSNT == 64.
845      ;
846      000102 RCVBUF == 66.
847      000104 PKPTR == 68.
848      000106 XSPTN == 70.
849      000110 WRTNO == 72.
850      000112 WRTNI == 74.
851      000114 RDNO == 76.
852      000116 RDN1 == 78.

```

```

855      ;AND THE ERROR LOG...
856      ;SPLIT INTO A BYTE PER DRIVE:
857      ;
858
859      ;-----
860      ;OFFSET IN DATA BLOCK      ;ERROR TYPE      ;ERRCODE;MSG CODE;SUC. CODE
861      ;-----
862
863      000120      LGOFST ==      80.      ;**RESERVED**
864      000122      SOFTR ==      82.      ;SOFT READ      ;SFTRD ;MSSF RD ;ESCKSM
865      000124      SOFTW ==      84.      ;SOFT WRITE     ;SFTWR ;MSSFWR ;ESSKSM
866      ;          WORD      ;RECIEVED INIT ;RCINIT ;MSRNIT ;*****
867      ;          WORD      ;**RESERVED**
868
869      ;THEN THOSE CODES WHICH HAVE N TRIES BEFORE ABORT
870
871      000132      T4TRY ==      90.      ;DLV ERROR      ;OVRN ;MSOVRN ;*****
872      000134      BDATA ==      92.      ;BAD DATA      ;BDCOM ;MSDATA ;*****
873      000136      HARDR ==      94.      ;HARD READ      ;HRDRD ;MSHDRD ;ESCKSM
874      000140      HARDW ==      96.      ;HARD WRITE     ;HRDWR ;MSHDWR ;ESCKSM
875      ;          WORD      ;CHKSM AT HOST ;BDCHK ;MSHCHK ;*****
876      ;          WORD      ;SEEK ERROR TOTAL ;SKERR ;MSSKER ;*****
877      000146      T1TRY ==      102.     ;WRITE PROTECT ;WRLOCK ;MSWPRO ;ESWLOC
878      ;          WORD      ;NO MOTOR      ;NOMOT ;MSNOMO ;ESNOMO
879      ;          WORD      ;CANT INIT     ;CNINIT ;MSNIT ;*****
880      ;          WORD      ;PARTIAL OP    ;PARTL ;MSPART ;ESPART
881      ;          WORD      ;NO UNIT       ;NOUNIT ;MSUNIT ;ESNONY
882      ;          WORD      ;COMMAND ERROR ;CMNDER ;MSCMD ;ESCMD
883      ;          WORD      ;BAD RECORD NO.;RECERR ;MSREC ;ESREC
884      ;          WORD      ;SELF TEST ERROR;SLFER ;MSSELF ;*****
885      ;          WORD      ;WRONG SUC.CODE ;SUCOTL ;MSWRSP ;*****
886      ;          WORD      ;NO RESPONSE   ;TORCVB ;MSNRSP ;*****
887      ;          WORD      ;WEIRD FLAG    ;OTL ;MSQRSP ;*****
888      ;          WORD      ;NO CARTRIDGE ;NOCART ;MSNOTP ;ESNCRT
889      ;          WORD      ;TIME OUT SEND;TOSNDB ;MSTOSN ;*****
890
891
892      000202      BLKEND ==      130.     ;OFFSET OF END OF STATISTICS (RESERVED)
893      ;          WORD      ;** RESERVED **
894      000204      TUVECT ==      132.     ;VECTOR ADDRESS
895      ;          WORD      ;** RESERVED **
896      000210      BLKSIZ ==      136.     ;** RESERVED **
897      ;-----

```

```
900 .SBTTL DEVICE DATA BLOCK ALLOCATION
901
902
903 ;TABLE OF DEVICE DATA BLOCK ADDRESSES
904
905
906 003340 003360 BLKTBL:: .WORD DEV0
907 003342 003570 .WORD DEV1
908 003344 004000 .WORD DEV2
909 003346 004210 .WORD DEV3
910 003350 004420 .WORD DEV4
911 003352 004630 .WORD DEV5
912 003354 005040 .WORD DEV6
913 003356 005250 LSTDEV:: .WORD DEV7
914
915
916 ;AND STORAGE FOR EACH:
917
918 003360 DEV0: .BLKB BLKS12
919 003570 DEV1: .BLKB BLKS12
920 004000 DEV2: .BLKB BLKS12
921 004210 DEV3: .BLKB BLKS12
922 004420 DEV4: .BLKB BLKS12
923 004630 DEV5: .BLKB BLKS12
924 005040 DEV6: .BLKB BLKS12
925 005250 DEV7: .BLKB BLKS12
```

941
942
943
944
945
946 005460
947
959
960
978

.SBTTL GLOBAL TEXT SECTION
:
: NAMES OF DEVICES SUPPORTED BY PROGRAM
:
DEVTP <IUS8 CONTROLLER>

```

987      .SBTTL  SYSTEM MACRO DEFINITIONS
988
989      .MACRO  PUSH      ,REG      MOV      REG,-(SP)
990
991      .ENDM
992
993      .MACRO  POP,REG      MOV      (SP)+,REG
994
995      .ENDM
996
997      :++
998      :THE MACRO 'SWAPIN' RETRIEVES THE TEST REGISTERS WHICH WERE SAVED
999      :IN THE DEVICE DATA BLOCK.
1000     :--
1001
1002     .MACRO  SWAPIN
1003
1004             MOV      6.(R5),R0
1005             MOV      8.(R5),R1
1006             MOV      10.(R5),R2
1007             MOV      12.(R5),R3
1008             MOV      14.(R5),R4
1009
1010     .ENDM
1011
1012     :++
1013     :THE MACRO 'SWAPOW' SAVES THE CURRENT STATE OF THE UNIT IN THE DRIVE
1014     :DATA BLOCK IN SO THAT THE SCHEDULER MAY 'SWAPIN' ANOTHER UNIT.
1015     :--
1016
1017     .MACRO  SWAPOW
1018
1019             MOV      R0,6.(R5)
1020             MOV      R1,8.(R5)
1021             MOV      R2,10.(R5)
1022             MOV      R3,12.(R5)
1023             MOV      R4,14.(R5)
1024
1025     .ENDM

```

1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080

;;
THE WRITE MACRO IMPLEMENTS THE COMPLETE PROTOCOL NECESSARY TO BUILD
A COMMAND PACKET AND SUBSEQUENT DATA PACKETS (UNTIL THE BYTE COUNT
(BCNT) IS SATISFIED).

SETS UP THE EXPECTED PROTOCOL RESPONSES: THE NUMBER OF PACKETS
(XSPKMN) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT). CALLS
'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
CHECKSUM.

INPUTS - DEVICE BLOCK @R5
 TRBUF - BUFFER ADDRESS
 UNIT'S TEST REGISTERS FROM 'SWAPIN'
OUTPUTS - SDCNT(R5) = # OF BYTES TO SEND
 XSPKMN = # OF PACKETS EXPECTED
 XSFLG = FLAG BYTE OF 1ST PACKET
 XSCNT = BYTE COUNT OF 1ST PACKET

 * SUBSEQUENT XSFLGS
 *
 * AND XSCNTS

```

.MACRO TUWRIT PTRN,REC,BCNT,DR,VER,?A,?B,?C,?D,?E,?F,?G,?H,?I
:      MOV      #TRBUF,R0      ;MAKE COMMAND PACKET:
      MOVB     #RSCMND,@R0     ;COMMAND FLAG
      MOVB     #RSMSIZ,1(R0)   ;THIS SIZE
      MOVB     #RSSWR,2(R0)   ;INSERT OP CODE-WRITE
      MOVB     VER,3.(R0)     ;VERIFY (1 OR 0)
      MOVB     DR,4.(R0)      ;DRIVE #
      MOVB     #020,5.(R0)    ;MAINTENANCE MODE SWITCH
      CLR      6.(R0)         ;NO SEQUENCE #
      MOV      BCNT,8.(R0)    ;TOTAL COUNT TO WRITE
      MOV      REC,10.(R0)    ;AT RECORD N
      MOV      #RSMSIZ,R1     ;THE PACKET SIZE PLUS+2
      TST      (R1)+          ;(FLAG AND COUNT) INTO R1
      MOV      #RSSNSZ,SDCNT(R5) ;LOAD THE SIZE TO SEND
      CALL     CHKSUM         ;R0 --> R1=COUNT
      MOV      R1,(R0)        ;PUT CHKSUM IN PACKET
      ;SET UP EXPECTATIONS:
      MOV      #RSCONT,XSFLG(R5) ;THE FLAG
      MOV      #1,XSCNT(R5)    ;THE COUNT
      MOV      #1,XSPKMN(R5)  ;THE # PACKETS EXPECTED
      CALL     RSVP           ;SEND (AND RETURN TO SCHEDULER)
      BIC      #BIT12,@R5     ;FLAG FOR LAST PACKET
      MOV      BCNT,R2        ;GET # OF DATA BYTES
      A:      MOV      #TRBUF,R0 ;POINT TO TOP OF BUFFER AGAIN
      CMP      R2,#128.       ;START DATA PACKET(S)
      BHI      B              ;BCNT > 128.!
      MOV      R2,R1          ;BCNT<128.
      BIS      #BIT12,@R5     ;SO LAST PACKET NOW
      BR       C              ;USE REMAINING COUNT
      B:      MOV      #128.,R1 ;USE 128. BYTES
      C:      MOVB     R1,1(R0) ;COPY COUNT TO BUFFER
      MOV      R1,R3          ;R3=COUNTER TO LOAD BUFFER
      MOVB     #RSDATA,@R0    ;FLAG FIRST

```

```

1081      1ST      (R0)+      ;SKIP COUNT
1082      MOV     PTRN,(R0)+  ;INSERT DATA
1083      DEC     R3          ;MORE?
1084      BHI     D           ;YES
1085      MOV     #TRBUF,R0    ;-->TOP AGAIN
1086      MOV     1(R0),R1     ;GET COUNT
1087      BIC     #177400,R1   ;ZERO SIGN EXTEND
1088      MOV     R1,SND CNT(R5) ;HOW MANY TO SEND PLUS
1089      ADD     #4,SND CNT(R5) ;FLAG,COUNT,CHKSUM
1090      ADD     #2,R1        ;COMPENSATE FOR FLAG + COUNT
1091      CALL    CHKSUM       ;FOR CHECKSUM CALC.
1092      MOV     R1,(R0)+     ;CHKSUM INTO PACKET
1093      SWAB    R1           ;EVEN ON AN ODD
1094      MOV     R1,(R0)+     ;BYTE BOUNDARY
1095      BIT     #BIT12,@R5   ;LAST DATA PACKET?
1096      BEQ     E           ;NO
1097      MOV     #RSEND,XSFLG(R5) ;YES-EXPECT 'END'
1098      MOV     #RSNDSZ,XSCNT(R5) ;OF THIS SIZE
1099      MOV     #1,XSPK NM(R5) ;AND 1 PACKET
1100      BR      F           ;SEND
1101      E:      MOV     #RC CNT,XSFLG(R5) ; (NOT LAST), EXPECT 'CONTINUE'
1102      MOV     #1,XSCNT(R5)  ;AND 1 BYTE
1103      MOV     #1,XSPK NM(R5) ;AND 1 PACKET
1104      F:      ALL     RSVP  ;SEND PACKET
1105      ;AND RETURN TO SCHEDULER
1106      BIT     #BIT10,@R5   ;RETRY?
1107      BNE     G           ;YES
1108      SUB     #128.,R2     ;NO, MORE DATA TO SEND?
1109      BHI     A           ;YES
1110      BR      H           ;NO
1111      G:      TURTRY  REC,BCNT,DR ;RETRY HERE
1112      BIT     #BIT10,@R5   ;RETRY AGAIN?
1113      BNE     G           ;YES
1114      F:      NOP         ;DONE
1115
                                .ENDM

```


1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227

++
THE RETRY MACRO IMPLEMENTS THE COMPLETE PROTOCOL NECESSARY TO INITIATE
A RETRY (READ OPERATION) SEQUENCE.

SETS UP THE EXPECTED PROTOCOL RESPONSES: THE NUMBER OF PACKETS
(XSPKMN) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT). CALLS
'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
CHECKSUM.

INPUTS - DEVICE BLOCK @R5
TRBUF - BUFFER ADDRESS
UNITS TEST REGISTERS FROM SWAPIN

OUTPUTS - SDCNT(R5) = # OF BYTES TO SEND
XSPKMN = # OF PACKETS EXPECTED
XSFLG = FLAG BYTE OF 1ST PACKET
XSCNT = BYTE COUNT OF 1ST PACKET

. ***
. * SUBSEQUENT XSFLGS
. *
. * AND XSCNTS
. ***

MACRO TURTRY REC,BCNT,DR,?A,?B,?C,?D,?E

D:	MOV	#TRBUF,R0	:FORM CMND PACK:
	MOVB	#RSCMND,@R0	:MESSAGE PACK TYPE
	MOVB	#RSMSIZ,1(R0)	:THIS BIG
	MOVB	#RSSRD,2(R0)	:OP CODE-READ
	MOV	REC,10.(R0)	:THIS RECORD
	MOVB	DR,4.(R0)	:THIS DRIVE
	CLRB	3(R0)	:PRESET NORM THRESHOLD
	TSTB	@R5	:REDUCED?
	BPL	E	:NO
	INCB	3(R0)	:YES-CHANGE THRESHOLD
E:	MOV	BCNT,8.(R0)	:# BYTES DESIRED
	MOVB	#OZ0,5.(R0)	:MAINTENANCE MODE
	CLR	6.(R0)	:NO SEQUENCE #
	MOV	#RSMSIZ,R1	:SIZE OF PACKET
	TST	(R1)+	:PLUS FLAG+COUNT INTO R1
	MOV	#RSSNSZ,SDCNT(R5)	:SET UP SIZE TO SEND
	CALL	CHKSUM	:FORM CHECKSUM R1=COUNT
	MOV	R1,(R0)	:INSERT IN PACKET
	MOV	BCNT,R1	:SET EXPECTATIONS:
			:CALC # OF DATA PACKETS TO EXPECT
	MOV	#XSFLG,R3	:OFFSET OF FLAG
	ADD	R5,R3	:ABS. ADDR. OF XSFLG
	CLR	R2	:PRESET
A:	INC	R2	:# PACKETS EXPECTED
	MOV	#RSDATA,(R3)+	:LOAD XSFLG
	MOV	#132,(R3)+	:AND EXPECT COUNT
	SUB	#128,R1	:NEG RESULT LAST TIME
	BLOS	1	:LAST TIME!
	BR	A	:MORE TO DO

1228
1229
1230
1231
1232
1233
1234
1235

```

( :      INC      R2      :ADD ONE FOR END PACK
      MOV      R2,XSPKNM(R5) :SAVE # PACKETS TO EXPECT
      MOV      #RSEND,(R3)+ :EXPECT AN END
      MOV      #RSENDSZ,(R3) :THIS BIG-14. BYTES

      CALL     RSVP      :SEND
                          :AND RETURN TO SCHEDULER

```

. E N D M

```

1238      ***
1239      :THE READ MACRO IMPLMENTS THE COMPLETE PROTOCOL NECESSARY TO INITIATE
1240      :A READ SEQUENCE.
1241
1242      :SETS UP THE EXPECTED PROTOCOL RESPONSES: THE NUMBER OF PACKETS
1243      : (XSPKMN) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT). CALLS
1244      : 'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
1245      : CHECKSUM.
1246
1247      : INPUTS - DEVICE BLOCK @R5
1248      :          TRBUF - BUFFER ADDRESS
1249      :          UNITS TEST REGISTERS FROM SWAPIN
1250
1251      : OUTPUTS - SDCNT(R5) = # OF BYTES TO SEND
1252      :             XSPKMN = # OF PACKETS EXPECTED
1253      :             XSFLG = FLAG BYTE OF 1ST PACKET
1254      :             XSCNT = BYTE COUNT OF 1ST PACKET
1255      :             ***
1256      :             * SUBSEQUENT XSFLGS
1257      :             * >
1258      :             * AND XSCNTS
1259      :             ***
1260      :--
1261
1262      .MACRO TUREAD REC,BCNT,DR,VER,?A,?B,?C,?D
1263
1264
1265      MOV     #TRBUF,R0      ;FORM CMND PACK:
1266      MOVB    #RSCMND,@R0    ;MESSAGE PACK TYPE
1267      MOVB    #RSMSIZ,1(R0)  ;THIS BIG
1268      MOVB    #RSSRD,2(R0)   ;OP CODE IS READ
1269      MOV     REC,10.(R0)    ;THIS RECORD
1270      MOVB    DR,4.(R0)      ;THIS DRIVE
1271      MOVB    VER,3.(R0)     ;VERIFY
1272      MOV     BCNT,8.(R0)    ;TOTAL BYTES TO READ
1273      MOVB    #O20,5.(R0)   ;MAINTENANCE MODE
1274      CLR     6.(R0)         ;NO SEQUENCE #
1275      MOV     #RSMSIZ,R1     ;GET SIZE OF PACKET
1276      IST     (R1)+          ;+2 FOR CHECKSUM
1277      MOV     #RSSNSZ,SDCNT(R5) ;SIZE TO SEND
1278      CALL    CHKSUM         ;FORM CHECKSUM R1=COUNT
1279      MOV     R1,(R0)        ;INSERT CHECKSUM
1280
1281      MOV     BCNT,R1        ;SET EXPECTATIONS:
1282      :CALC # OF DATA PACKETS TO EXPECT:
1283      MOV     #XSFLG,R3      ;GET OFFSET
1284      ADD     R5,R3          ;ABS. ADDR. OF XSFLG
1285      CLR     R2             ;PRESET AS NONE
1286      INC     R2             ;# PACKETS EXPECTED
1287      MOV     #RSDATA,(R3)+  ;LOAD XSFLG
1288      MOV     #132.,(R3)+    ;AND EXPECTED COUNT
1289      SUB     #128.,R1       ;NEG RESULT LAST TIME
1290      BLOS    C              ;LAST TIME
1291      BR      A              ;MORE TO DO
1292      (:      INC     R2      ;ADD ONE FOR END PACK
1293      MOV     R2,XSPKMN(R5)  ;SAVE # PACKETS TO EXPECT
1294      MOV     #RSEND,(R)+    ;EXPECT AN END ALSO...

```

1295
1296
1297
1298
1299
1300
1301
1302
1303
1304

.ENDM

	MOV	#RSNDSZ,(R3)	:THIS BIG-14. BYTES
	CALL	RSVP	:SEND
			:AND RETURN TO SCHEDULER
D:	BIT	#BIT10,@R5	:RETRY?
	BEQ	B	:NO.
	TURTRY	REC,BCNT,DR	:YES
B:	BR	D	:ANOTHER RETRY?
	NOP		:NO

1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354

```

**
THE SELF TEST MACRO IMPLIMENTS THE COMPLETE PROTOCOL NECESSARY TO
INITIATE A 'DIAGNOSE' SEQUENCE.

```

```

SETS UP THE EXPECTED PROTOCOL RESPONSES: THE NUMBER OF PACKETS
(XSPKMN) AND THEIR FLAG BYTES AND COUNTS (XSFLG, XSCNT). CALLS
'RSVP' TO SEND EACH PACKET, AND 'CHKSUM' TO CALC. THE PACKET
CHECKSUM.

```

```

INPUTS - DEVICE BLOCK @R5
         TRBUF - BUFFER ADDRESS
         UNITS REGISTERS TEST FROM SWAPIN

```

```

OUTPUTS - SDCNT(R5) = # OF BYTES TO SEND
          XSPKMN = # OF PACKETS EXPECTED
          XSFLG = FLAG BYTE OF 1ST PACKET
          XSCNT = BYTE COUNT OF 1ST PACKET

```

```

          ***
          * SUBSEQUENT XSFLGS
          *
          * AND XSCNTS
          ***

```

```

--

```

.MACRO TUSELF

```

MOV     #TRBUF,R0           ;FORM COMMAND PACKET
MOVB    #RSCMND,@R0         ;COMMAND FLAG
MOVB    #RSMISZ,1(R0)       ;SIZE OF MESSAGE
MOVB    #RSSSLF,2(R0)       ;SELF TEST OPERATION
CLRB    3(R0)               ;NO MODIFIER.
CLR     4(R0)               ;NO DRIVE OR SWITCHES
CLR     6(R0)               ;NO SEQUENCE NUMBER
CLR     8(R0)               ;NO BYTES
CLR     10(R0)              ;NO RECORD #
MOV     #RSMISZ,R1          ;GET SIZE
TST     (R1)+                ;+2 FOR CHECKSUM
MOV     #RSSNSZ,SDCNT(R5)    ;SIZE TO SEND
CALL    CHKSUM               ;FORM CHECKSUM

MOV     R1,(R0)              ;INSERT INTO PACKET
MOV     #RSEND,XSFLG(R5)     ;EXPECT END.
MOV     #RSDSZ,XSCNT(R5)     ;THIS BIG
MOV     #1,XSPKMN(R5)        ;AND 1 PACKET
                                ;SEND
CALL    RSVP                 ;RETURN TO SCHEDULER

```

.ENDM

1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382

```

; **
; THE TEST ID MACRO INTERFACES THE SUPERVISOR'S TEST DISPATCH TO THE
; DIAGNOSTIC'S FORMAT BY IMPLEMENTING CALLS THAT: 1) INITIALIZE THE
; PC OF THE TEST CODE (TSTPC(R5)), 2) ASSIGN THE 1ST DRIVES, 3) RUN
; THE TEST, 4) SWITCH DRIVES AND REINITIALIZE, 5) RUN THE TEST AGAIN.
; --

        .MACRO TSTID    ADDRESS,?A

        .NLIST
        .LIST ME
        .LIST

                                MOV     ADDRESS,TSTTOP    ;SAVE ADDR OF TEST
                                CALL    SETUP             ;INIT UNITS TSTPC
                                CALL    SETDR             ;GET 1ST DRVS.
                                CALL    RUN               ;DO TEST
                                CALL    SWAPDR            ;GET NEXT DRVS.
                                BCC     A                 ;BR NO 2ND DRVS
                                CALL    SETUP             ;REINIT UNITS TSTPC
                                CALL    RUN               ;REPEAT TEST
                                A:                               ;DONE

        .NLIST
        .NLIST ME
        .LIST
        .ENDM
;-----

```

```

1385      .SBTTL GLOBAL SUBROUTINES SECTION
1386
1387      :++
1388      : THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES THAT ARE USED
1389      : TO LINK THE DIAGNOSTIC TO THE SUPERVISOR (THROUGH THE 1STID MACRO).
1390      :--
1391
1392      :++
1393      : SWAPDR
1394      : SUBROUTINE TO DETERMINE IF TO TEST OTHER DRIVE (FOR ALL UNITS)
1401      : INPUTS: DR(R5) - DRIVE CONFIGURATION
1402      :          BLK1BL - TOP OF DATA BLOCK ALLOCATION TABLE
1403      :          LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
1404
1405      : OUTPUTS: DR(R5) UPDATED TO TEST SAME OR OTHER DRIVE
1406      :          CARRY SET IF SECOND PASS NECESSARY
1449      :--
1461
1462 005500 005002 SWAPDR:: CLR R2 ;FOR # OF DRIVE 1'S.
1463 005502 012737 003340 005600 MOV #BLK1BL,SWPTR ;TABLE ADDR. OF 1ST UNIT
1464 005510 017705 000064 1$: MOV @SWPTR,R5 ;GET DATA BLOCK ADDR.
1465 005514 032715 100000 BIT #BIT15,@R5 ;ABORTED?
1466 005520 001013 BNE 3$ ;YES
1467 005522 032765 000001 000060 BIT #BIT0,DR(R5) ;DID DR. 0?
1468 005530 001007 BNE 3$ ;NO, DID DR. 1 1ST PASS
1469 005532 032765 001000 000060 BIT #BIT9,DR(R5) ;YES; 1 SELECTED?
1470 005540 001403 BEQ 3$ ;NO, ALL DONE
1471 005542 105265 000060 INCB DR(R5) ;YES, SWAP
1472 005546 005202 INC R2 ;ONE MORE TO TEST
1473 005550 023727 005600 003356 3$: CMP SWPTR,#LSTDEV ;LAST DEVICE?
1474 005556 103004 BHIS 4$ ;YES
1475 005560 062737 000002 005600 ADD #2,SWPTR ;NO-POINT NEXT
1476 005566 000750 BR 1$ ;DO
1477
1478 005570 005702 4$: TST R2 ; (CLEAR CARRY), MORE TO DO?
1479 005572 001401 BEQ 5$ ;NO
1480 005574 000261 SEC ;YES
1481 005576 000207 5$: RETURN ;RETURN
1482
1483 005600 000000 SWPTR: .WORD

```

```

1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497 005602 012737 003340 005656 SETDR:: MOV #BLKTBL,SETPTR ;TABLE OF ADDR. 1ST UNIT
1498 005610 017705 000042 1$: MOV @SETPTR,R5 ;GET DATA BLOCK ADDR.
1499 005614 105065 000060 CLR B DR(R5) ;PRESET AS DRO
1500 005620 032765 000400 000060 BIT #BIT8,DR(R5) ;DO DRO?
1501 005626 001002 BNE 2$ ;YES
1502 005630 105265 000060 INCB DR(R5) ;NO-USE DRIVE 1
1503 005634 023727 005656 003356 2$: CMP SETPTR,#LSTDEV ;MORE UNITS
1504 005642 103004 BHS 3$ ;NO-EXIT
1505 005644 062737 000002 005656 ADD #2,SETPTR ;YES-GET TABLE ENTRY
1506 005652 000756 BR 1$ ;CONFIGURE THAT UNIT
1507 005654 000207 3$: RETURN
1508 005656 000000 SETPTR: .WORD

```

```

1511      :
1512      : **
1513      : CLRALL - CLEARS INPUT BUFFER FOR RESPONSE FROM UNIT.
1514      :
1515      : INPUTS:      2LKTB - TOP OF DATA BLOCK ALLOCATION TABLE
1516      :              LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
1517      :
1518      : OUTPUTS:     ALL UNITS BUFFERS CLEARED.
1519      :
1520      : CALLS:      CLRBUF
1521      : --
1522 005660 012737 003340 005752 CLRALL:: MOV  #BLKTB,CLRPTR ;TOP OF TABLE OF ADDRESSES
1523 005666 017705 000060 1$:      MOV  @CLRPTR,R5 ;GET DATA BLOCK
1524 005672 004737 005720      CALL CLRBUF ;CLEAR IT'S RECEIVE BUFFER
1525 005676 023727 005752 003356      CMP  CLRPTR,#LSTDEV ;LAST DEV?
1526 005704 103004      BHS  2$ ;YES
1527 005706 062737 000002 005752      ADD  #2,CLRPTR ;-->NEXT
1528 005714 000764      BR   1$ ;CONTINUE
1529 005716 000207      2$:      RETURN

```

```

1532
1533      ;++
1534      ; CLRBUF - CLEARS 1 UNIT'S INPUT BUFFER.
1535      ; INPUTS: RCVBUF(R5) IS BUFFER START
1536      ;          RCBFSZ - SIZE OF RECEIVE BUFFER IN BYTES
1537      ;          RCBFSZ IS SIZE OF BUFFER
1538      ; OUTPUTS: CLEARED AREA.
1539      ;--
1540 005720 CLRBUF:: PUSH    R0          ;SAVE R0
1541 005722      PUSH    R4          ;SAVE R4
1542 005724 016500 000102      MOV    RCVBUF(R5),R0 ;GET ADDRESS OF BUFFER
1543 005730 012704 001036      MOV    #RCBFSZ,R4   ;SIZE IN BYTES
1544 005734 005020      1$: CLR    (R0)+          ;CLEAR IT
1545 005736 162704 000002      SUB    #2,R4        ;2 BYTES LESS
1546 005742 001374      BNE    1$                ;MORE
1547 005744      POP     R4                    ;RESTORE
1548 005746      POP     R0                    ;
1549 005750 000207      RETURN                  ;EXIT
1550 005752 000000 CLRPTR: .WORD

```

```

1553
1554      : **
1555      : SETUP - CALLED WITHIN EACH TEST TO INSERT BEGINNING ADDRESS OF THE
1556      : TEST INTO ALL UNITS TEST PC'S.
1557      : INPUTS:  TSTTOP LOADED WITH TEST ALGORITHMS STARTING ADDR.
1558      :          BLKTB1 - TOP OF DATA BLOCK ALLOCATION TABLE
1559      :          LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
1560      : OUTPUTS:  TSTPC(R5) FOR ALL UNITS
1561      :          DONE - CLEARED
1562      : --
1563
1564 005754 005037 003314  SETUP:: CLR      DONE      :NOT DONE YET
1565 005760 012737 003340 003316  MOV      #BLKTB1, IDPTR :TABLE TOP ADDR
1566 005766 017705 175324      1$:  MOV      @IDPTR, R5    :DEVICE'S DATA BLOCK
1567 005772 013765 003320 000020      MOV      TSTTOP, TSTPC(R5) :INSERT PC FOR TOP OF TEST
1568 006000 023727 003316 003356      MP      IDPTR, #LSTDEV :ALL UNITS SET?
1569 006006 103004      BHIS  2$      :YES
1570 006010 062737 000002 003316      ADD      #2, IDPTR    :NO, GET NEXT POINTER
1571 006016 00C763      BR      1$      :SET HIM UP
1572 006020 000207      2$:  RETURN    :DONE

```

```

1575
1576
1577
1578
1579
1580
1581
1582 006022 004737 006052
1583
1584 006026 005737 003314
1585 006032 001006
1586 006034 004737 006736
1587
1588 006040
1589
1590 006042 004737 010112
1591 006046 000765
1592 006050 000207

```

```

:++
: RUN - IMPLEMENTS THE CALLS TO SEND PACKETS, RECEIVE PACKETS, THEN
: CHECK ANSWERS DURING TEST RUN TIME.
: INPUTS: DONE
: OUTPUTS: NONE
:--

```

```

RUN:: CALL NXTST ;MAKE AND SEND NEXT PACK TO ALL
;UNABORTED UNITS
;COMPLETE?
;YES
;NO,GET ALL RESPONSES
;SUPERVISOR CHECK
;CHECK ALL RESPONSES
;CONTINUE TILL DONE
2$: CALL CHKANS
BR RUN
RETURN

```

```
1595 .SBTTL NXTST / THE SCHEDULER
1596
1597
1598 :**
1599 : NXTST - USING EACH UN-ABORTED UNIT'S TEST PROGRAM COUNTER
1600 : (TSTPC(R5)), EXECUTES THE TEST CODE THAT COMPRISES MAKING A
1601 : PACKET AND SENDING IT. ACTION IS ROUND ROBIN. CHECKS FIRST
1602 : FOR ANY UNIT RETRYING AND IF SO SERVICES ONLY THAT UNIT THIS
1603 : PASS. INITIS NON-RETRYING UNITS IF NECESSARY.
1604 : INPUTS: (IMPLIED) DATA BLOCKS.
1605 : BLKTL - TOP OF DATA BLOCK ALLOCATION TABLE
1606 : LSTDEV - ADDR OF LAST UNIT'S DATA BLOCK
1607 : OUTPUTS: ERRSF IF ALL UNITS ARE ABORTED.
1608 : SYSTAT IS UPDATED
1609 :--
1610
1611 006052 J12737 003340 003304 NXTST:: MOV #BLKTL,DEVPTX ;UNIT 0 TO START
1612 006060 017705 175220 1$: MOV @DEVPTX,P5 ;GET DATA BLOCK
1613 006064 032715 002000 BIT #BIT10,@R5 ;RETRYING?
1614 006070 001422 BEQ 2$ ;NOT THIS GUY
1615 006072 005715 TST @R5 ;YES, ABORTED THO?
1616 006074 100420 BMI 2$ ;YES ON TO NEXT UNIT
1617 006076 052737 000002 003300 BIS #BIT1,SYSTAT ;NOT ABORTED-SET RETRY STATUS
1618 006104 SWAPIN ;GET DEVICE REGISTERS
1619 006130 004775 000020 JSR PC,@TSTPC(R5) ;DO TEST FOR
1620 006134 000477 BR NXTRET ;THIS UNIT ONLY-EXIT
1621 006136 023727 003304 003356 2$: CMP DEVPTX,#LSTDEV ;TRY NEXT UNIT?
1622 006144 103004 BHIS NXTST2 ;NO
1623 006146 062737 000002 003304 ADD #2.,DEVPTX ;YES,->NEXT
1624 006154 000741 BR 1$ ;GET BLOCK
1625
1626 006156 005037 006336 NXTST2: CLR ABONM ;HERE=NO RETRIES TO DO, NO UNIT ABORTED YET
1627 006162 012737 003340 003304 MOV #BLKTL,DEVPTX ;-->UNIT 0 STORAGE BLOCK
1628 006170 017705 175110 PERDEV: MOV @DEVPTX,R5 ;R5-->NEXT DEVICE STORAGE BLOCK
1629
1630 006174 005715 3$: TST @R5 ;ABORTED?
1631 006176 100426 PMI 4$ ;YES
1632 006200 032715 040000 BIT #BIT14,@R5 ;SEND BREAK?
1633 006204 001407 BEQ 6$ ;NO
1634 006206 004737 013222 CALL DOBRK ;YES
1635 006212 032715 040000 BIT #BIT14,@R5 ;SUCCESSFUL INIT?
1636 006216 001016 BNE 4$ ;NO ON TO NEXT UNIT
1637 006220 005715 TST @R5 ;ABORTED?
1638 006222 100414 BMI 4$ ;YES-ON TO NEXT UNIT
1639 006224 6$: SWAPIN ;NO,GET DEVICE REGISTERS R0-R4 CONTAINING TEST PARAMETERS
1640 006250 J04775 000020 JSR PC,@TSTPC(R5) ;INITIATE 1 PACKET TRANSMISSION AND RETURN
1641 006254 005715 4$: TST @R5 ;ABORTED?
1642 006256 100002 BPL 8$ ;NO-ON TO NEXT UNIT
1643 006260 005237 006336 INC ABONM ;YES...ONE MORE TALLIED
1644 006264 023727 003304 003356 8$: CMP DEVPTX,#LSTDEV ;ALL TU'S TRIED?
1645 006272 103004 BHIS 5$ ;YES
1646 006274 062737 000002 003304 ADD #2.,DEVPTX ;NO THE ADDRESS+2=NEXT ADDRESS
1647 006302 000732 BR PERDEV ;DO NEXT UNIT
1648 006304 022737 000010 006336 5$: CMP #8.,ABONM ;ALL ABORTED?
1649 006312 001010 BNE NXTRET ;NO
1650 J06314 ERRSF 1J0.,NOMOR ;YES
1651 006324 11$: BREAK ;SUPERVISOR BREAK
```

```
1652 006326 005237 003332      INC      ALLGON      ;SET DON'T-PRINT STATISTICS FLAG
1653 006332      DOCLN      ;EXIT
1654 006334 000207      NXTRET: RETURN
1655
1656 006336 000000      ABONM: .WORD      ;THE NUMBER OF ABORTED UNITS
1657 006340      101      114      114  NOMOR: .ASCIZ /ALL UNITS ABORTED!/
1658      .EVEN
```

.SBTTL RSVP / XOFF AND SEND A PACKET TO ALL DEVICES

RSVP - SAVES TEST CODE PROGRAM COUNTER IN TSTPC(R5) AND UNIT'S REGIS-
TERS. POINTS TO "XOFF" THAT PRECEEDS PACKET IN XMIT BUFFER
AND SENDS PACKET WITH XOFF. RETURNS TO SCHEDULER (NXTST) SO
THAT OTHER UNITS PACKETS MAY BE FORMED, TO GET ALL UNITS WORKING
AT ONCE.

INPUTS: (SP) CONTAINS UNITS PC TO SAVE SINCE RSVP WAS CALLED. THE
NUMBER PACKETS EXPECTED (XSPKNM), AND THE EXPECTED FLAGS AND
BYTE COUNTS OF EACH (XSFLG, XSCNT...) ARE LOADED BY TEST CODE
(MACROS).
SND CNT - # BYTES TO SEND
REC(R5) - RECORD #
TRBUF - BUFFER ADDR.
XSPKNM(R5) - # EXPECTED
RCVBUF(R5)

OUTPUTS: CMDSNT - UPDATED WITH PACKET OP CODE
BLKER - RECORD NUMBER STATISTICS UPDATED IF NOT RETRYING
AND COMMAND PACKET SENT.
SUCCS(R5) - PRESET CLEAR
STATUS WORD @R5 - BIT9 - DATA CHECK ERROR - CLEARED
BIT5 - "VERIFY" OPERATION
BIT4 - 0 = DATA PACK 1 = CMND
BIT8 - RD/WR OPERATION
XSPTX - POINTS TO EXPECTED FLAG
UPPER BY E OF XSPKNM IS REPLICATED.
PACKET PC'INTER (PKPTR(R5)) POINTS TO TOP OF UNITS RECEIVE BUFFER
AREA (RCVBUF(R5)) FOR CURRENT UNIT.

RSVP:: NOP ;FINISH TEST
MOV (SP)+,TSTPC(R5) ;SAVE WHERE YOU WERE IN TEST BODY AND
SWAPOW ;SAVE TEST REGISTERS

XFNSND: MOV #TRBUF-1,R0 ;CORRECT FOR RETURN TO SCHEDULER
INC SND CNT(R5) ;POINT TO XOFF
BR SND ;ONE MORE TO SEND, TOO.
NOXOFF: MOV #TRBUF,R0 ;SEND XOFF+PACKET
SND: CALL SNDBYT ;FOR NORMAL PACKET SEND
;SEND BYTE
TST @R5 ;R5--> TO STATUS BLK
BMI 6\$;ABORTED? YES...QUIT
DEC SND CNT(R5) ;NO, SEND MORE
BNE SND ;IF MORE TO SEND
MOV #TRBUF,R0 ;-- BUFFER
MOV REC(R5),BLKER ;PREPARE FOR RECEIVE
BISB XSPKNM(R5),XSPKNM+1(R5) ;REPLICATE LO. BYTE TO HI FOR GIPAKS. (HMAN)
CLR SUCCS(R5) ;NO SUCCESS YET
BIC #BIT9,@R5 ;NO DATA CHK ERROR YET
MOV RCVBUF(R5),PKPTR(R5) ;TOP OF RCV BUFFER GOES THE 1ST PACKET
MOV #XSFLG,R4 ;FORM
ADD R5,R4 ;ADDRESS
MOV R4,XSPTX(R5) ;OF 1ST XSFLG

BIC #BIT4,@R5 ;PRESET AS DATA PAK

1718	006526	121027	000002		CMPB	@R0,#RSCMND	;WAS IT COMMAND PAK?
1719	006532	001054			BNE	6\$	;NO...
1720	006534	116065	000002	000100	MOVB	2(R0),CMDSN1(R5)	;YES-SAVE COMMAND
1721	006542	052715	000020		BIS	#BIT4,@R5	;ITS CMND PAK
1722							
1723	006546	032715	002000		BIT	#BIT10,@R5	;RETRYING?
1724	006552	001044			BNE	6\$	;YES-DON : UPDATE ANY STATS OR CONDITION
1725	006554	126027	000002	000002	CMPB	2(R0),#RSSRD	;NO,A READ?
1726	006562	001012			BNE	4\$	;NO
1727	006564	042715	000400		BIC	#BIT8,@R5	;(FOR HARD/SOFT LOGGING) RD/WR FLAG-0
1728	006570	004737	013052		CALL	WHCHDR	;GET DRIVE
1729	006574	103403			BCS	8\$	
1730	006576	005265	000114		INC	RDNO(R5)	;DRIVE 0
1731	006602	000402			BR	4\$	
1732	006604	005265	000116	8\$:	INC	RDN1(R5)	;DRIVE 1
1733							
1734	006610	126027	000002	000003	4\$:	CMPB	2(R0),#RSSWR
1735	006616	001022			BNE	6\$	;A WRITE?
1736	006620	052715	000400		BIS	#BIT8,@R5	;YES, RD/WR FLAG=1
1737	006624	105760	000003		TSTR	3(R0)	;VERIFY TON?
1738	006630	001403			BEQ	21\$	;NO
1739	006632	052715	000040		BIS	#BIT5,@R5	;YES-SET VERIFY FLAG
1740	006636	000402			BR	22\$	
1741	006640	042715	000040	21\$:	BIC	#BIT5,@R5	;(NO)-RESET VERIFY FLAG
1742	006644	004737	013052	22\$:	CALL	WHCHDR	;GET DRIVE NO
1743	006650	103403			BCS	5\$	;CARRY-DRI
1744	006652	005265	000110		INC	WRINO(R5)	;# BLKS WRITTEN DRO
1745	006656	000402			BR	6\$	;EXIT
1746							
1747	006660	005265	000112	5\$:	INC	WRIN1(R5)	;# BLKS WRITTEN DRV
1748	006664	000207		6\$:	RETURN		;RETURN

```

1751 .SBTTL SNDBYT / OUTPUT A BYTE TO UNIT
1752
1753 : **
1754 : SNDBYT - TEST 'READY' ON INTERFACE. IF 'READY', SEND BYTE AND EXIT.
1755 : IF TIMED OUT, LOG ERROR.
1756 : INPUTS - R0 = POINTER TO BUFFER
1757 : - IMPLIED UNIT DATA BLOCK
1758 : - CSNRDY - TIMEOUT CONSTANT
1759 : OUTPUTS - R0 IS INCREMENTED.
1760 : ERROR - NOT-READY-TO-SEND TIME OUT
1761 : --
1762
1763 SNDBYT: PUSH R1 ;ENTER R0-->BYTE
1764 4$: MOV (CSNRDY,R1 ;GET TIMEOUT CONSTANT FOR NOT READY ERROR
1765 1$: TSTB @XMSR(R5) ;READY TO SEND?
1766 BMI 2$ ;YES
1767 PUSH R0 ;NO, SAVE R0
1768 BREAK ;MONITOR BREAK
1769 POP R0 ;RESTORE
1770
1771 DEC R1 ;ABORTED?
1772 BNE 1$ ;NO
1773 MOV #TOSNDB,R4 ;YES,SET CODE FOR TIMEOUT ERROR
1774 CALL LOG ;LOG IT
1775 BR 3$ ;QUIT
1776 2$: MOVB (R0),@XMDB(R5) ;SEND IT
1777 3$: POP R1 ;RESTORE
1778 RETURN ;DONE
  
```

```

1781 .SBTTL GETANS / GETS RESPONSES ROUND ROBIN USING "XON"
1782
1783
1784 ;**
1785 ; GETANS - IF A UNIT IS RETRYING CLEAR HIS RECEIVE BUFFER (CLRBUF) AND GET
1786 ; HIS RESPONSE (GTPKS1), ELSE, CLEAR ALL BUFFERS (CLRALL) AND
1787 ; GET ALL RESPONSES (GTPKS8).
1788 ; INPUTS: SYSTAT - SYSTEM STATUS WORD.
1789 ; OUTPUTS: SERVST = -1 IF NO RETRIES.
1790 ;--
1791
1792 006736 000240 GETANS:: NOP ;1 UNIT IF RETRY; ELSE ALL
1793 006740 32737 000002 003300 BIT #BIT1,SYSTAT ;RETRY?
1794 006746 001010 BNE 1$ ;YES
1795 006750 012737 177777 007656 MOV #-1,SERVST ;PRESET NO UNITS SERVICED
1796 006756 004737 005660 CALL CLRALL ;CLEAR ALL INPUT BUFFERS
1797 006762 004737 007214 CALL GTPKS8 ;GET ALL REPLYS
1798 006766 000404 BR 2$ ;EXIT
1799 006770 004737 005720 1$: CALL CLRBUF ;RETRY-CLEAR 1 UNIT ONLY
1800 ;R5->UNIT BY NXTST
1801 006774 004737 007004 CALL GTPKS1 ;GET 1 REPLY
1802 007000 000207 2$: RETURN ;DONE
1803
1804 007002 000000 GETPTR: .WORD

```

1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863

.SBTTL GTPKS1 / GET RETRY RESPONSE-1 UNIT

```

**
GTPKS1 - SENDS 'XON' TO UNIT, GETS FLAG BYTE (IF ANY), CHECKS IF IT IS
          WHAT WAS EXPECTED. IF IT IS, USE EXPECTED BYTE COUNT(XSCNT). IF
          NOT, CHECK IF PREMATURE-END PACK OR (SINCE MAINTENANCE MODE)
          IF IT'S A PREMATURE DATA PACK. ADJUST COUNT, GET REST OF
          PACKET, AND REPEAT ABOVE UNTIL NO MORE PACKETS.
INPUTS:  (IMPLIED) UNITS DATA BLOCK
          RSND SZ - END PACKET SIZE

OUTPUTS: SYSTAT UPPER BYTE = FLAG BYTE RECEIVED
--

```

```

GTPKS1:: NOP
          MOV     #XSFLG,R3
          ADD     R5,R3
          MOV     R3,R1
          ADD     #2,R1
          MOV     #EXON,R0
          CALL    SNDBYT
          ;R5->THE UNIT
          ;THE OFFSET VALUE OF FLAG
          ;FORM THE ABSOLUTE ADDRESS
          ;R3-->ADDR. OF EXPECTED FLAG
          ;R1-->ADDR. OF EXPECTED COUNT
          ;R0=ADDRESS
          ;XON THE DEVICE
          ;*** TIME CRITICAL
          ;***--> TO THE BUFFER
          ;***GET THE # OF PACKETS TO RECEIVE
          ;***SIGN UN-EXTEND
          ;***HOW MANY BYTES IT SHOULD BE
          ;***WHAT THE FIRST BYTE SHOULD BE
          ;***GET THE ALL IMPORTANT FLAG
          ;TIMEOUT?
          ;YES
          ;-> BYTE RECEIVED
          ;SAVE IT AS FLAG BYTE
          ;1ST BYTE WHAT WAS EXPECTED?
          ;YES
          ;NO, WAS IT END PAK?
          ;NO
          ;YES, USE END SIZE FOR COUNT
          ;AND ASSUME IT'S LAST PACKET!
          ;CONTINUE RECEIVE
          ;WAS IT DATA?
          ;NO,CHKANS MAY FIND INIT...
          ;YES, SET FOR DATA PAK SIZE
          ;ONE MORE PACK THAN EXPECTED (END PAK)

          MOV     RCVBUF(R5),R0
          MOVB    XSPKRM+1(R5),R2
          BIT     #17400,R2
          1$:    MOV     @R1,RCBCNT
          MOV     @R3,RCFLG
          CALL    GTBYTE
          BIT     #BIT15,@R5
          BNE     4$
          DEC     R0
          MOVB    @R0,SYSTAT+1
          CMPB    @R0,RCFLG
          BEQ     2$
          CMPB    @R0,#RSEND
          BNE     14$
          MOV     #RSND SZ,RCBCNT
          MOV     #1,R2
          BR      2$
          14$:    CMPB    @R0,#RSDATA
          BNE     4$
          MOV     #RSDASZ,RCBCNT
          INC     R2
          2$:    INC     R0
          5$:    DEC     RCBCNT
          BEQ     3$
          CALL    GTBYTE
          TST     DLV(R5)
          BNE     4$
          BIT     #BIT15,@R5
          BNE     4$
          BR      5$
          3$:    DEC     R2
          BEQ     4$
          ;RESTORE TO -> NEXT BYTE
          ;THAT'S ONE LESS BYTE TO GO
          ;ONE
          ;GET REST OF PACKET
          ;ERROR
          ;YES-ALL OVER
          ;OR IF ABORTED
          ;THEN QUIT
          ;CONTINUE RECEIVE
          ;ONE LESS PACKET TO GO
          ;MORE PACKETS IN TRANSACTION?
          ;YES

```

1864	007202	022121		CMP	(R1)+,(R1)+	;POINT TO NEW EXPECTED COUNT
1865	007204	022325		CMP	(R3)+,(R3)+	;AND FLAG,
1866	007206	000717		BR	\$	;AND RECEIVE,
1867	007210	000207	4\$:	RETURN		;RETURN
1868						
1869	007212	020	EXON:	.BYTE	SXON	
1870	007213	023	EXOFF:	.BYTE	PSXOFF	

```

1873 .SBTTL GTPKS8 / GET RESPONSES (NO RETRIES)
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890 007214 000240
1891 007216 012737 003340 007660
1892 007224 017705 000430
1893 007230 032715 100000
1894 007234 001403
1895 007236 004737 007572
1896 007242 000534
1897 007244 105765 000033
1898 007250 001003
1899 007252 004737 007572
1900 007256 000526
1901 007260 105365 000033
1902 007264 017537 000106 003306
1903 007272 062765 000002 000106
1904 007300 017537 000106 003310
1905 007306 012700 007212
1906
1907 007312 004737 006666
1908 007316 016500 000104
1909 007322 004737 007662
1910 007326 032715 100000
1911 007332 001403
1912 007334 105065 000033
1913 007340 000475
1914 007342 005300
1915 007344 111037 003301
1916 007350 121037 003306
1917 007354 001436
1918 007356 105065 000033
1919 007362 121027 000002
1920 007366 001004
1921 007370 012737 000016 003310
1922 007376 000406
1923 007400 121027 000001
1924 007404 001053
1925 007406 012737 000222 003310
1926 007414 005200
1927 007416 005337 003310
1928 007422 001444
1929 007424 004737 007662

GTPKS8: NOP ;GET ALL UNITS RESPONSES XOFF IF DATA PAK (THROUGHPUT)
GTAGIN: MOV #BLKTBL,GTPTR ;->1ST
BIT @GTPTR,R5 ;GET DATA BLOCK
BEQ #BIT15,R5 ;ABORTED?
CALL SETSRV ;YES-SET 'SERVICED' AND
BR GTDOWN ;ON TO NEXT UNIT
2$: TSTB XSPKNT+1(R5) ;NO, ANY PACKETS LEFT?
BNE 3$ ;YES
CALL SETSRV ;NO-HE'S DONE
BR GTDOWN ;SO ON TO NEXT UNIT
3$: DECB XSPKNT+1(R5) ;NOW ITS ONE LESS PACKET
MOV @XSPTR(R5),RCFLG ;GET EXPECTED FLAG
ADD #2,XSPTR(R5) ;--> COUNT
MOV @XSPTR(R5),RCBCNT ;AND EXPECTED COUNT
MOV #EXON,R0 ;--> XON
;***TIME CRITICAL
;***SEND IT
;***->WHERE 1ST BYTE GOES
CALL GTBYTE ;***GET IT
BIT #BIT15,R5 ;ABORTED?
BEQ 4$ ;NO-CONTINUE
CLRB XSPKNT+1(R5) ;YES-NO MORE PACKETS EXPECTED
BR GTDOWN ;ON TO NEXT
4$: DEC R0 ;-->BYTE JUST RECEIVED
MOVB @R0,SYSTAT+1 ;SAVE IT
CMPB @R0,RCFLG ;IS IT WHAT EXPECTED?
BEQ GTOK ;YES
UNXPCT: CLRB XSPKNT+1(R5) ;NO, MUST BE LAST REPLY
CMPB @R0,#RSEND ;MAYBE AN END PAK?
BNE 4$ ;NO
MOV #RSNDSZ,RCBCNT ;YES, USE PROPER COUNT
BR GTUM ;AND GET IT
4$: CMPB @R0,#RSDATA ;IS IT DATA?
BNE GTDOWN ;NO, ALL OVER, CHKANS WILL INIT UNIT
MOV #RSDNSZ,RCBCNT ;YES, USE COUNT OF DATA + END PAK SURE TO FOLLOW
GTUM: INC R0 ;WHERE TO STUFF THE REST
5$: DEC RCBCNT ;ONE DOWN
BEQ GTDOWN ;NONE TO GO
CALL GTBYTE ;MORE TO GO

```

1930	007430	032715	100000		BIT	#BIT15,@R5	:TIMEOUT?
1931	007434	001037			BNE	GTDOWN	:YES
1932	007436	005765	000074		TST	DLV(R5)	:BUT DLV ERROR?
1933	007442	001765			BEQ	1\$	:NO
1934	007444	105065	000033		CLRB	XSPKMM+1(R5)	:YES-LAST TIME
1935	007450	000431			BR	GTDOWN	:ON TO NEXT
1936							
1937	007452	005200		GTOK:	INC	R0	:NEXT PLACE IN BUFFER
1938	007454	005337	003310	1\$:	DEC	100CNT	:MORE BYTES?
1939	007460	001413			BEQ	2\$	:NO-ALL DONE
1940	007462	004737	007662		CALL	GTBYTE	:YES-GET IT
1941	007466	032715	100000		BIT	#BIT15,@R5	:TIMEOUT?
1942	007472	001020			BNE	GTDOWN	:YES
1943	007474	005765	000074		TST	DLV(R5)	:ERROR?
1944	007500	001765			BEQ	1\$	:NO
1945	007502	105065	000033		CLRB	XSPKMM+1(R5)	:LAST TIME
1946	007506	000412			BR	GTDOWN	:EXIT
1947	007510	122775	000001 000104	2\$:	CMPE	#RSDATA,@PKPTR(R5)	:WAS DATA?
1948	007516	001006			BNE	GTDOWN	:NO, ALL DONE
1949	007520	010065	000104		MOV	R0,PKPTR(R5)	:START OF NEXT PACK NEXT TIME
1950	007524	012700	007213		MOV	#EXOFF,R0	:XOFF AND SEND TO
1951	007530	004737	006666		CALL	SNDYBT	:ENHANCE THROUGHPUT
1952	007534	062765	000002 000106	GTDOWN:	ADD	#2,XSPTR(R5)	:NEXT XSFLG FOR NEXT TRY
1953	007542	023727	007660 003356		CMPE	GTPTR,#LSTDEV	:DONE ONE CYCLE ALL UNITS?
1954	007 50	103004			BHIS	1\$	:YES
1955	007552	062737	000002 007660		ADD	#2,GTPTR	:NEXT UNIT
1956	007560	000621			BR	GTAGIN	:CONTINUE RECEIVE
1957	007562	105737	007656	1\$:	TSTB	SERVST	:DONE SERVICING ALL PAKS
1958							:FROM ALL UNITS?
1959	007566	001212			BNE	GTPKS8	:NO, KEEP TRYING
1960	007570	000207			RETURN		:YES.

```

1963      .SBTTL  SETSRV / SET UNIT SERVICED
1964
1965      : **
1966      : SETSRV - RESET THE BIT IN 'SERVST' CORRESPONDING TO THE UNIT NUMBER.
1967      : INPUTS - SERVST - 'SERVICED' WORD
1968      :           - @R5 = UNIT # (BITS 0, 1, 2)
1969      : OUTPUTS - SERVST MODIFIED
1970      : --
1971
1972 007572 SETSRV: PUSH    R5           ;SET UNIT SERVICED
1973 007574      PUSH    R0
1974 007576 011505      MOV    @R5,R5      ;GET STAT WD
1975 007600 042705 177770      BIC    #177770,R5      ;MASK UNIT #
1976 007604 012700 007636      MOV    #SRVTBL,R0      ;->TOP OF BIT TABLE
1977 007610 005705      1$:      TST    R5           ;RIGHT ONE?
1978 007612 001404      BEQ    2$           ;YES
1979 007614 062700 000002      ADD    #2,R0          ;NO, ->NEXT
1980 007620 005305      DEC    R5           ;1 LESS
1981 007622 000772      BR      1$          ;CONTINUE
1982 007624 041037 007656      2$:      BIC    @R0,SERVST ;NOW IT DOWN
1983 007630      POP     R0
1984 007632      POP     R5
1985 007634 000207      RETURN           ;RETURN
1986
1987 007636 000001      SRVTBL: .WORD   B110          ;BIT POSITION LOOKUP TABLE
1988 007640 000002      .WORD   B111
1989 007642 000004      .WORD   B112
1990 007644 000010      .WORD   B113
1991 007646 000020      .WORD   B114
1992 007650 000040      .WORD   B115
1993 007652 000100      .WORD   B116
1994 007654 000200      .WORD   B117
1995
1996 007656 000000      SERVST: .WORD
1997 007660 000000      GTPTR:  .WORD
    
```

```

2000 .SBTTL GTBYTE / GET A BYTE FROM UNIT
2001
2002
2003 : **
2004 : GTBYTE - TEST INTERFACE FOR 'READY-TO-RECEIVE' AND INPUT A BYTE, IF
2005 : SO. IF NOT, THE FOLLOWING OCCURS: SEND 'XOFF' TO UNIT IN
2006 : PREPARATION FOR ^C CHECK ('BREAK' TO SUPERVISOR). WAIT
2007 : TO SEE IF A CHARACTER SLOPS OVER DUE TO UART LATENCY. IF
2008 : ONE DOES THEN MIGHT AS WELL GET IT AND SEND 'XON' TO GET
2009 : THE REST OF THE MESSAGE, OTHERWISE, 'BREAK'. THEN SEND
2010 : 'XON', AND TEST FOR LONG TIMEOUT (A 30 SECOND REWIND). IF SO,
2011 : LOG ERROR, OTHERWISE REPEAT THE ABOVE UNTIL READY OR TIME OUT.
2012 : REMEMBER TO PRESERVE R0 SINCE THE 'BREAK' TRAP CLOBBERS IT.
2013
2014 : INPUTS - R0 POINTS TO INPUT BUFFER
2015 : - IMPLIED UNITS DATA BLOCK
2016 : - CSRCVB TIME OUT MULTIPLIER
2017
2018 : OUTPUTS - R0 IS INCREMENTED
2019 : - DLV (R5) NON-ZERO ON INTERFACE ERROR.
2020
2021 : ERROR - TIME OUT ON RECEIVE
2022 : --
2023
2024 GTBYTE:: CLR GBTMP ; TIMEOUT REGISTER
2025 MOV CSRCVB,R4 ; TIMEOUT ERROR CONSTANT (MULTIPLIER)
2026 1$: TSTB @RCSR(R5) ; READY?
2027 BPL 3$ ; NO
2028 MOV @RCDB(R5),DLV(R5) ; GET ERROR + BYTE
2029 MOVB DLV(R5),(R0)+ ; COPY BYTE TO BUFFER
2030 TST DLV(R5) ; ERROR?
2031 BMI 4$ ; YES-EXIT
2032 CLR DLV(R5) ; NO-RESET
2033 BR 4$ ; AND EXIT
2034 3$: DEC GBTMP ; DEC T.O. CONSTANT
2035 BNE 1$ ; STILL VALID
2036
2037 ; CCDE TO SEE ^C DURING LONG SEEK OR REWIND
2038 MOV R0,GBTMP2 ; HERE GBTMP=0
2039 MOV #EXOFF,R0 ; R0 MUST BE PRESERVED!
2040 CALL SNDBYT ; QUIET THE DEVICE
2041 6$: TSTB @RCSR(R5) ; BY SENDING XOFF
2042 BMI 5$ ; CHARACTER SLOP OVER?
2043 DEC GBTMP ; YES
2044 TSTB GBTMP ; NO-WAIT A WHILE
2045 BNE 6$ ; DONE WAITING?
2046 BREAK ; NO
2047 MOV #EXON,R0 ; YES-NO SLOP OVER
2048 CALL SNDBYT ; START DEVICE TALKING
2049 MOV GBTMP2,R0 ; AGAIN
2050 BR 7$ ; RESTORE R0
2051 5$: MOV GBTMP2,R0 ; END KLUGE
2052 MOV @RCDB(R5),DLV(R5) ; RESTORE R0
2053 MOVB DLV(R5),(R0)+ ; GET ERROR + BYTE
2054 TST DLV(R5) ; COPY BYTE TO BUFFER
2055 BMI 17$ ; ERROR?
2056 CLR DLV(R5) ; YES-EXIT
; NO-CLEAR

```

2057	010040	000400		BR	17\$	:EXIT
2058	010042	010037	010110	17\$: MOV	R0,GBTMP2	:AGAIN SAVE R0
2059	010046	012700	007212	MOV	#EXON,R0	:RESTORE TO TALKING STATE
2060	010052	004737	006666	CALL	SND BY	:BY SENDING 'XON'
2061	010056	013700	010110	MOV	GBTMP2,R0	:RESTORE R0
2062	010062	000410		BR	4\$	:DONE
2063	010064	005037	010106	7\$: CLR	GBTMP	
2064	010070	005304		DEC	R4	:TIMEOUT?
2065	010072	001277		NE	1\$	:NO
2066	010074	012704	000050	MOV	#TORCVB,R4	:YES
2067	010100	004737	012046	CALL	LOG	:LOG ERROR.
2068	010104	000207		4\$: F TURN		:RETURN
2069	010106	000000		GBTMP: .WORD	0	
2070	010110	000000		GBTMP2: .WORD	0	

```

2073          .SBTTL  CHKANS / CHECK DEVICE(S) RESPONSE
2074
2075          : **
2076          : CHKANS - AS IN "GETANS", IF RETRYING DO ONLY 1 UNIT ELSE DO ALL NON-
2077          :          ABORTED UNITS.
2078          : INPUTS:  IMPLIED SYSTAT BIT1 (RETRYING)
2079          :          BLKTBL - TOP OF DATA BLOCK ALLOCATION TABLE
2080          :          LSTDEV - ADDR. OF LAST UNIT'S DATA BLOCK
2081
2082          : OUTPUTS: NONE PASSED.
2083          : --
2084
2085 010112 000240          CHKANS:: NOP                      ; IF RETRY THEN CHECK ONE
2086                                     ; ELSE CHECK ALL
2087 010114 032737 000002 003300          BIT      #BIT1,SYSTAT ; RETRYING?
2088 010122 001403          BEQ      CHK8                     ; NO DO NORMAL
2089 010124 004737 10202          CALL     CHKPKS              ; YES DO BAZARRE WITH
2090                                     ; R5 -> UNIT
2091 010130 000422          BR        CHKANR                    ; ALL DONE
2092
2093 010132 012737 003340 010200  CHK8:  MOV      #BLKTBL,CHKPTR ; YOU KNOW ... TOP OF TABLE
2094 010140 017705 000034 2$:  MOV      @CHKPTR,R5             ; GET UNIT'S BLOCK ADDRESS
2095 010144 032715 100000          BIT      #BIT15,R5           ; ABORTED?
2096 010150 001000          BNE      3$                         ; YES
2097 010152 004737 10202          CALL     CHKPKS              ; NO, DO THIS GUY
2098 010156 023727 010200 003356 3$:  CMP      CHKPTR,#LSTDEV ; ALL DONE?
2099 010164 103004          BHS      CHKANR                    ; YES
2100 010166 062737 000002 010200          ADD      #2,CHKPTR   ; NO,-->NEXT DEVICE
2101 010174 000761          BR        2$                       ; DO DA
2102
2103 010176 000207          CHKANR: RETURN
2104
2105 010200 000000          CHKPTR: .WORD

```

2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125

.SBTTL CHKPKS / DECIPHERS RESPONSE OF UNIT POINTED TO BY R5 /

```

**
CHKPKS - FOR UNIT R5 AND FOR ALL PACKETS, CHECK TO SEE IF PACKET IS DATA OR
END PACK, CHECK CHECKSUMS, COMPARE DATA IF DATA PACK, CHECK
SUCCESS CODE IF END. IF UNKNOWN PACKET TYPE, CHECK FOR INTERFACE
ERROR. IF "CONTINUE" FALL THROUGH. IF "INIT" SET "SEND
BREAK" FLAG. CALL "LOG" WITH R4=ERROR NUMBER IF ERROR.
INPUTS: (IMPLIED) UNITS DATA BLOCKS
OUTPUTS: ERRORS - DLV ERROR
              - UNKNOWN FLAG BYTE ERROR
              - CHECKSUM ERROR
              - DATA COMPARE ERROR
R4 = ERROR NUMBER
SYSTAT UPPER BYTE = 1ST BYTE OF RESPONSE
--

```

2126 010202 000240
2127 010204 016500 000102
2128 010210 016502 000032
2129 010214 012703 000034
2130 010220 060503
2131 010222 010301
2132 010224 062701 000002
2133 010230 010065 000104
2134 010234 111037 003301
2135 010240 011137 003310
2136 010244 011337 003306
2137 010250 121013
2138 010252 001050
2139 010254 121027 000020
2140 010260 001516
2141
2142 010262 013704 003310
2143 010266 005744
2144 010270 004737 013162
2145 010274 103005
2146 010276 012704 000022
2147 010302 004737 012046
2148 010306 000503
2149 010310 122710 000002
2150 010314 001005
2151 010316 004737 010536
2152 010322 012702 000001
2153 010326 000473
2154 010330 122710 000001
2155 010334 001003
2156 010336 004737 013762
2157 010342 000465
2158
2159 010344 052715 040000
2160 010350 012704 000052
2161 010354 005765 000074
2162 010360 001402
2163 010362 012704 000012
2164 010366 001737 012046

```

CHKPKS:: NOP                ;CHECK WHAT WAS RECIEVED
      MOV RCVBUF(R5),R0      ;GET BUFFER ADDR.
      MOV XSPKRM(R5),R2      ;AND # OF PACKETS EXPECTED
      MOV #XSFLG,R3          ;THE OFFSET VALUE
      ADD R5,R3              ;R3-->THIS UNIT XSFLG AGAIN
      MOV R3,R1              ;COPY TO R1
      ADD #2,R1              ;R1-->XSBNT FOR 1ST PACKET
1$:   MOV R0,PKPTR(R5)        ;POINT TO PACKET
      MOVB @R0,SYSTAT+1      ;SAVE RCV'D BYTE
      MOV @R1,RCBNT          ;GET COUNT
      MOV @R3,RCFLG          ;AND FLAG
      CMPB @R0,@R3           ;1ST BYTE=EXPECTED?
      BNE 5$                 ;UH OH...
      CMPB @R0,#RSCONT       ;OK, IS IT 1 BYTE?
      BEQ 7$                 ;YES...ONTO NEXT PACK
      MOV RCBNT,R4           ;NO, SO > 1 BYTE (NEVER EXPECT INIT!)
      TST -(R4)              ;EXPECTED, SO COUNT MUST BE RIGHT
      CALL CKCKSM            ;ADJUST FROM RECEIVE COUNT TO COUNT FOR CHECKSUM
      BCC 2$                 ;CHECK CHECKSUM
      MOV #BDCHK,R4          ;NO CARRY...NO INCORRECT
      CALL LOG               ;ERROR
      BR 7$                  ;LOG IT
2$:   CMPB #RSEND,(R0)       ;ON TO NEXT PACK
      BNE 3$                 ;END PAK?
      CALL CHKEND            ;NO
      MOV #1,R2              ;YES-CHECK
      BR 7$                  ;LAST PACKET
3$:   CMPB #RSDATA,@R0        ;AND FALL THROUGH
      BNE 4$                 ;DATA PAK?
      CALL COMPAR            ;NO
      BR 7$                  ;YES-CHECK DATA
4$:   BIS #BIT14,@R5          ;ALL DONE?
      MOV #OTL,R4            ;SET 'DOBREAK' FLAG
      TST DLV(R5)            ;OUT TO LUNCH
      BEQ 7$                 ;AH,BUT DLV ERROR?
      MOV #OVRN,R4           ;NO
      CALL LOG               ;YES-USE CORRECT ERROR #
20$:  CALL LOG               ;TALLY

```

```

2165 010372 000460          BR      8$          :DONE
2166
2167          ;HERE CHECKS UNEXPECTED RESPONSE
2168
2169 010374 122710 000004      5$:  CMPB    #RSINIT,@R0      ;INIT?
2170 010400 001007          BNE     6$          ;NO
2171 010402 052715 040000      BIS     #BIT14,@R5      ;YES-SET 'DOBREAK' FLAG
2172 010406 012704 000006      MOV     #RCINIT,R4      ; WE GOT AN INIT
2173 010412 004737 012046      CALL    LOG          ;TALLY IT
2174 010416 000446          BR      8$          ;DONE
2175 010420 122710 000001      6$:  LMPB    #RSDATA,@R0      ;DATA PAK?
2176 010424 001013          BNE     9$          ;NO
2177 010426 012704 000204      MOV     #RSDASZ,R4      ;YES, USE DATA SIZE
2178 010432 005744          TST     -(R4)          ;ADJUST FOR CHKSUM
2179 010434 004737 013162      CALL    CKCKSM        ;AND CHECK
2180 010440 103421          BCS     10$         ;GOOF
2181 010442 004737 013762      CALL    COMPAR        ;OK, HOW'S THE DATA?
2182
2183
2184 010446 062700 000204          ADD     #RSDASZ,R0      ;EXPECTED END, GOT
2185 010452 000666          BR      1$          ;DATA + END.
2186
2187 010454 122710 000002      9$:  CMPB    #RSEND,(R0)      ;POINT TO FND PACK
2188 010460 001331          BNE     4$          ;CHECK IT, USE SAME XSFLG
2189
2190 010462 012704 000016          MOV     #RSSNSZ,R4      ;END?
2191 010466 005744          TST     -(R4)          ;NO-OUT TO LUNCH
2192 010470 004737 013162      CALL    CKCKSM        ;YES, TOTAL SIZE MINUS
2193 010474 103403          BCS     10$         ;TWO (THE CHKSUM)
2194 010476 004737 010536      CALL    CHKEND        ;CHECK IT
2195
2196 010502 000414          BR      8$          ;CHECK IT
2197
2198 010504 012704 000022      10$: MOV     #BDCHK,R4      ;OK,NOW TEST SUC. CODE
2199 010510 004737 012046      CALL    LOG          ;ALL DONE
2200 010514 000407          BR      8$          ;CHECKSUM ERROR
2201
2202 010516 005302          7$:  DEC     R2          ;EXIT
2203 010520 001405          BEQ     8$          ;ANY PACKETS LEFT TO CHECK?
2204 010522 063700 003310      ADD     RCBNT,R0      ;NO, ALL DONE
2205 010526 022121          CMP     (R1)+,(R1)+      ;YES, POINT TO NEXT PACKET
2206 010530 022323          CMP     (R3)+,(R3)+      ;POINT TO NEXT EXPECTED COUNT
2207 010532 000636          BR      1$          ;AND EXPECTED FLAG
2208
2209 010534 000207      8$:  RETURN          ;TRY ANOTHER,THEY'RE SMALL

```

```

2212 .SBTTL CHKEND / CHECK SUCCESS AND DETERMINE RETRY STATUS /
2213
2214
2215 **
2216 : CHKEND - IF RETRYING; CHECK SUCCESS CODE AND IF 0, PRINT RECOVERED, LOG
2217 : SOFT ERROR, END RETRY STATUS. IF NOT 0 AND WAS STILL 'DATA
2218 : CHECK' ERROR - DETERMINE WHETHER TO CONTINUE ANOTHER RETRY OR
2219 : LOG 'UNRECOVERABLE' ERROR.
2220
2221 : IF NOT RETRYING; CHECK IF 'DATA CHECK' ERROR SUCCESS CODE,
2222 : AND IF SO, START RETRY, ELSE EXIT.
2223 : INPUTS: IMPLIED UNITS DATA BLOCK
2224 : OUTPUTS: RETRY (SYSTAT BIT 1), (BIT10 @R5) SET IF RETRYING.
2225 : - DATA COMARE ERROR (BIT6 @R5) CLEARED.
2226 : - REDUCED/NORMAL GAIN (BIT7 @R5) ADJUSTED
2227 :--
2228
2228 010536 CHKEND:: PUSH R0 ;R0 --> END PAK
2229 010540 PUSH R4
2230 010542 032737 000002 003300 1$: BIT #BIT11,SYSTAT ;RETRYING?
2231 010550 001052 BNE CHKREE ;YES-BRANCH
2232 010552 004737 011536 CALL CHKSUC ;NO,GET SUCCESS CODE
2233 ;LOG ERROR...
2234 010556 032715 100000 BIT #BIT15,@R5 ;ABORTED?
2235 010552 001402 BEQ 3$ ;NO,CONTINUE
2236 010554 000137 011242 JMP CHKRET ;YES,EXIT
2237 010570 105765 000077 3$: TSTB SUCCS+1(R') ;NO; HOW'D WE DO?
2238 010574 001013 BNE CHKERR ;NOT SO GOOD.
2239 010576 032715 000100 BIT #BIT6,@R5 ;OK, MOST FIND DATA PAK ERROR?
2240 010602 001002 BNE 2$ ;YES
2241 010604 000137 011242 JMP CHKRET ;NO
2242 010610 012704 000014 2$: MOV #BDCOM,R4 ;YES; JUST BAD DATA-NO DATACHK ERR
2243 010614 004737 012046 CALL LOG ;BAD DATA IN PACKET
2244 010620 000137 011242 JMP CHKRET ;QUIT
2245 010624 032715 001000 CHKERR: BIT #BIT9,@R5 ;BAD SUCCESS; TU DATA CHK ERROR?
2246 010630 001002 BNE 1$ ;YES
2247 010632 000137 011242 JMP CHKRET ;NO. ALL DONE.
2248 010636 052715 002000 1$: BIS #BIT10,@R5 ;YES-START RETRY
2249 010642 012765 000001 000002 MOV #1,RETRY(R5) ;CALL IT 1ST
2250 010650 PRINTX #RTYRN,RETRY(R5) ;** PRINT **
2251 010674 000562 BR CHKRET ;ALL DONE
2252 010676 004737 011536 CHKREE: CALL CHKSUC ;RETRYING,GET SUCCESS
2253 010702 105765 000077 TSTB SUCCS+1(R5) ; SUCCESSFUL YET?
2254 010706 001054 BNE UNSUC ;NO, CHECK COUNT
2255 010710 PRINTX #RECOV,RETRY(R5)
2256 010734 105715 TSTB (R5) ;DETERMINE THRESHOLD
2257 010736 100411 BMI 2$ ;IT'S MODIFIED
2258 010740 PRINTX #THRSLO ;NORMAL
2259 010760 000410 BR 3$
2260 010762 2$: PRINTX #THRSHI ;ENHANCED
2261 011002 032715 000400 3$: BIT #BIT8,@R5 ;WRITE OR READ OPERATION?
2262 011006 001003 BNE 4$ ;WRITE
2263 011010 012704 000002 MOV #SFTRD,R4 ;READ
2264 011014 000402 BR 5$
2265 011016 012704 000004 4$: MOV #SFTWR,R4 ;WRITE
2266 011022 004737 012046 5$: CALL LOG
2267 011026 005065 000002 CLR RETRY(R5) ;RESTORE TO NORMAL STATE
2268 011032 042715 002200 BIC #BIT10:BIT17,@R5 ;NO RETRY, NORM THRESHOLD

```

```

2269 011036 000501          3R      CHKRET      ;QUIT
2270
2271 011040 000240          UNSUC: NOP          ;RETRYING; SEE IF HARD YET
2272 011042 032715 001000      BIT      #BIT9,@R5 ;TU DATA CHECK ERROR?
2273 011046 001015          BNE      2$         ;YES
2274 011050          PRINTB  #RETErr          ;NO-"OTHER-ERROR" ERROR
2275 011070 005065 000002      CLR      RETRY(R5) ;NO RETRIES
2276 011074 042715 002200      BIC      #BIT10!BIT7,@R5 ;NO RETRY, NORM THRESHOLD
2277 011100 000460          BR      CHKRET      ;EXIT
2278 011102 023765 003322 000002 2$: CMP      MXRTRY,RETRY(R5) ;YES. DID WE GRAD JATE TO HARD?
2279 011110 001425          BEQ      HRD1       ;YES
2280 011112 005265 000002      INC      RETRY(R5) ;NO. JUST ANOTHER
2281 011116          PRINTX  #RTRYN,RETRY(R5) ;PRINT OUT
2282 011142 032715 000200      BIT      #BIT7,@R5 ;WAS NORMAL THRESHOLD?
2283 011146 001403          BEQ      1$         ;YES-REDUCE GAIN
2284 011150 042715 000200      BIC      #BIT7,@R5 ;NO-NORM
2285 011154 000432          BR      CHKRET
2286 011156 052715 000200      1$: BIS      #BIT7,@R5 ;REDUCED
2287 011162 000427          BR      CHKRET      ;DONE
2288 011164 000240          HRD1: NOP          ;HERE IS HARD ERROR!
2289 011166          PRINTX  #UNREC
2290 011206 032715 000400      BIT      #BIT8,@R5 ;RD OR WR?
2291 011212 001003          BNE      4$         ;WRITE
2292 011214 012704 000016      MOV      #HRDRD,R4 ;READ
2293 011220 000402          BR      5$         ;LOG IT
2294 011222 012704 000020      4$: MOV      #HRDWR,R4 ;WRITE
2295 011226 004737 012046      5$: CALL     LOG      ;LOG IT
2296 011232 005065 000002      CLR      RETRY(R5) ;BACK TO NORMAL
2297 011236 042715 002200      BIC      #BIT10!BIT7,@R5 ;NO RETRY, NOT REDUCED
2298
2299 011242 042737 000002 003300 CHKRET: BIC      #BIT1,SYSTAT ;NO SYSTEM RETRY NEXT PASS
2300 011250 042715 000100      BIC      #BIT5,@R5 ;NO MORE HOST DATA CHECK ERROR
2301 011254          POP      R4
2302 011256          POP      R0
2303 011260 000207          RETURN
2304
2305
2306 011262      045      101      122 RECOV: .ASCIZ  /%ARECOVERED FROM DATA CHECK ERROR RETRY # %D1%N/
2307          .EVEN
2308 011342      045      101      040 THRSLO: .ASCIZ  /%A NORMAL THRESHOLD%N/
2309          .EVEN
2310 011370      045      101      040 THRSHI: .ASCIZ  /%A MODIFIED THRESHOLD %N/
2311          .EVEN
2312 011422      045      101      122 RTRYN:  .ASCIZ  /%ARETRY # %D1%N/
2313          .EVEN
2314 011442      045      101      125 UNREC:  .ASCIZ  /%AUNRECOVERABLE%N/
2315          .EVEN
2316 011464      045      101      117 RETERR: .ASCIZ  /%AOTHER ERROR DURING RETRY : EXIT RETRY%N/
2317          .EVEN

```

```

2320 .SBTTL CHKSUC / INTERPRET SUCCESS CODE /
2321
2322
2323 : **
2324 : CHKSUC - COPY SUCCESS CODE (BYTE) TO SUCCS+1(R5). INTERPRET SUCCESS
2325 : AND IF NOT 0, LOG APPROPRIATE ERROR.
2326 : INPUTS: R0 POINTS TO END PACKET.
2327 :          @R5 - UNIT STATUS WORD
2328 :          (CDSNT(R5)) - COMMAND BYTE
2329 :
2330 : OUTPUTS: R4 IS ERROR NUMBER IF ERROR.
2331 :          SUCCS(R5) UPDATED.
2332 :          BIT9 @R5 SET ON DATA CHECK SUCCESS CODE
2333 : --
2334 CHKSUC:: NOP
2335 011536 000240      MOV      2(R0),SUCCS(R5) ;R0-->END PACKET
2336 011540 016065 000002 000076      CMPB    #ESOK,3(R0) ;GET SUCCESS BYTE
2337 011546 122760 000000 000003      BEQ     12$ ;COMPLETE SUCCESS-EXIT
2338 011554 001533
2339 011556 122760 000001 000003      CMPB    #ESTRY,3(R0) ;OK BUT RETRIES?
2340 011564 001012      BNE     20$ ;NO
2341 011566 126527 000100 000002      CMPB    (CDSNT(R5)),#RSSRD ;A READ?
2342 011574 001001      BNE     22$ ;NO
2343
2344 011576 000516      BR       10$ ;NO RETRIES IN MAINTENANCE!
2345 011600 126527 000100 000003 22$ :      CMPB    (CDSNT(R5)),#RSSWR ;A WRITE?
2346 011606 001001      BNE     20$ ;NO
2347 011610 000511      BR       10$ ;LOG IT
2348 011612 122760 177737 000003 20$ :      CMPB    #ESNOMO,3(R0) ;NO MOTOR?
2349 011620 001003      BNE     1$ ;NO
2350 011622 012704 000030      MOV     #NOMOT,R4 ;YES-
2351 011626 000504      BR       11$ ;LOG
2352
2353 011630 122760 177757 000003 1$ :      CMPB    #ESCKS,3(R0) ;"DATA CHECK" ERROR?
2354 011636 001003      BNE     2$ ;NO
2355 011640 052715 001000      BIS     #BIT9,@R5 ;SET DATA-CHK-ERROR FLAG
2356 011644 000477      BR       12$ ;DONT LOG
2357
2358 011646 126527 000100 000007 2$ :      CMPB    (CDSNT(R5)),#RSSSLF ;SELF TEST?
2359 011654 001006      BNE     3$ ;NOPE
2360 011656 105760 000003      TSTB    3(R0) ;YES, NEG. IF ERROR
2361 011662 100070      BPL      12$ ;OK
2362
2363 011664 012704 000044      MOV     #SLFER,R4 ;YES-ERROR
2364 011670 000463      BR       11$ ;LOG IT
2365
2366 011672 122760 177740 000003 3$ :      CMPB    #ESSK,3(R0) ;SEEK ERROR?
2367 011700 001003      BNE     4$ ;NO
2368 011702 012704 000024      MOV     #SKERR,R4 ;YES-
2369 011706 000454      BR       11$ ;LOG
2370
2371 011710 122760 177767 000003 4$ :      CMPB    #ESNCRT,3(R0) ;NO CART?
2372 011716 001003      BNE     5$ ;NO
2373 011720 012704 000054      MOV     #NCART,R4 ;YES-
2374 011724 000445      BR       11$ ;LOG
2375
2376 011726 122760 177720 000003 5$ :      CMPB    #ESCMD,3(R0) ;NO UNDERSTAND MOST?

```

2377	011734	001003			BNE	6\$		:NO
2378	011736	012704	000040		MOV	#CMIDER,R'		:YES-
2379	011742	000436			BR	11\$		:LOG
2380								
2381	011744	122760	177770	000003	6\$:	(MPB	#ESNONX,3(R0)	:NON EXISTENT UNIT?
2382	011752	001003			BNE	7\$		:NO
2383	011754	012704	000036		MOV	#NOUNIT,R4		:YES-
2384	011760	000427			BR	11\$		:LOG
2385								
2386	011762	122760	177765	000003	7\$:	(MPB	#ESWLOC,3(R0)	:WRITE LOCKED?
2387	011770	001003			BNE	8\$		:NO
2388	011772	012704	000026		MOV	#WRLOCK,R4		:YES-
2389	011776	000420			BR	11\$		:LOG
2390								
2391	012000	122760	177776	000003	8\$:	(MPB	#ESPART,3(R0)	:PARTIAL OP?
2392	012006	001003			BNE	9\$		:NO
2393	012010	012704	000034		MOV	#PARTL,R4		:YES-
2394	012014	000411			BR	11\$		:LOG
2395								
2396	012016	122760	177711	000003	9\$:	(MPB	#ESREC,3(R0)	:WRONG RECORD?
2397	012024	001003			BNE	10\$		:NO
2398	012026	012704	000042		MOV	#RECERR,R4		:YES-
2399	012032	000402			BR	11\$		:LOG
2400								
2401	012034	012704	000046		10\$:	MOV	#SUCOTL,R4	:UNDEFINED
2402	012040	004737	012046		11\$:	CALL	LOG	:LOG ERROR
2403	012044	000207			12\$:	RETURN		:RETURN

```

2406 .SBTTL LOG / TO LOG ERROR IN CORRECT PLACE
2407
2408
2409 : **
2410 : LOG - DETERMINE IF ERROR IS FATAL, NON-FATAL OR FATAL AFTER N TRIES
2411 : BY INDEX (ERROR #) INTO DEVICE DATA BLOCK. ADD THE DRIVE # TO
2412 : INDICATE UPPER OR LOWER BYTE AND INCREMENT THAT ERROR UNLESS
2413 : THAT BYTE WOULD OVERFLOW. DETERMINE IF EVL FLAG SET, AND IF SO,
2414 : CHECK THRESHOLD (EVLTHR) AND PRINT APPROPRIATE ERROR MESSAGE
2415 : DESCRIPTION. ABORT THE UNIT IF INDICATED THROUGH DODROP CODE.
2416 : INPUTS: R4 = ERROR CODE
2417 : OUTPUTS: ABNDX(R5) = ERROR CODE.
2418 :          DLV(R5) = 0
2419 :          L$UN = UNIT NUMBER
2420 : --
2421 LOG::  PUSH    R0
2422         PUSH    R1
2423         PUSH    R3
2424         PUSH    R4
2425
2426 012046 011537 002074      MOV    @R5,L$UN      ;GET UNIT NUMBER
2427 012062 042737 177770 002074 BIC    #177770,L$UN ;MASK IT OFF
2428 012070 010465 000004      MOV    R4,ABNDX(R5) ;SAVE INDEX IN CASE OF ABORT MESSAGE
2429 012074 012703 000120      MOV    #LGOFST,R3 ;OFFSET TO LOW ORDER BYTE (DRIVE0)
2430 012100 060403      ADD    R4,R3 ;FORM INDEX OF PARAM. TO UPDATE
2431 012102 060503      ADD    R5,R3 ;FORM ABSOLUTE ADDR. THIS UNIT
2432 012104 004737 013052      CALL   WHCHDR ;SEE WHICH DRIVE IT WAS
2433 012110 103001      BCC     Z$ ;WAS DRIVE 0
2434 012112 005203      INC     R3 ;DRIVE 1; POINT TO UPPER BYTE
2435 012114 122713 000377 Z$:  CMPB   #255.,@R3 ;POTENTIAL OVERFLOW POSSIBLE?
2436 012120 007005      BNE     LOGOK ;NO
2437 012122      LOGO:  ERDF    0.,OVRFLD,ERRDES ;YES
2438 012132 000512      BR      ABO ;ABORT UNIT
2439 012134 105213      LOGOK: INCB   @R3 ;INCREMENT THE ERROR
2440 012136 111304      MOV     @R3,R4 ;TEMP'LY SAVE IT
2441 012140 016503 000004      MOV    ABNDX(R5),R3 ;GET INDEX AGAIN
2442 012144 012701 002220      MOV    #RSNTAB,R1 ;FORM ACRS OF MSG
2443 012150 066501 000004      ADD    ABNDX(R5),R1 ;LIKE THIS
2444 012154 042701 000001      BIC    #B!TO,R1 ;INSURE WORD BOUNDARY
2445 012160 032737 000004 016154 BIT    #EVL,FLGLOC ;EVL SELECTED?
2446 012166 001414      BEQ     LOGOK2 ;NO-CONT
2447 012170 123704 002216      CMPB   EVLTHR,R4 ;YES,OVER THRESHOLD?
2448 012174 101011      BHI     LOGOK2 ;NO
2449 012176 010337 012210      MOV    R3,DFTL1+2 ;YES,LOAD ERROR #
2450 012202 011137 012212      MOV    @R1,DFTL1+4 ;AND MESSAGE ADDR
2451 012206      DFTL1: ERDF    0,DFTL1,ERRDES ;ERROR
2452 012216 000460      BR      ABO ;DROP IT
2453 012220 120327 000014      LOGOK2: CMPB   R3,#BDCOM ;'NEVER FATAL' TYPE?
2454 012224 103011      BHS     NTSFT ;NO
2455 012226 010337 012240      MOV    R3,LOG1+2 ;YES, ERROR CODE
2456 012232 011137 012242      MOV    @R1,LOG1+4 ;DESCRIPTION
2457 012236      LOG1:  ERDSOFT 0.,LOG1,ERRDES
2458 012246 000450      BR      LOGO ;EXIT
2459
2460 012250 120327 000026      NTSFT: CMPB   R3,#WRLOCK ;ONE TRY?
2461 012254 103411      BLO     MABEE ;NO, MAYBE A MULTIPLE
2462 012256 010337 012270      MOV    R3,LOG2+2. ;YES

```

2463	012262	011137	012272			
2464	012266			LOG2:	MOV @R1,LOG2+4	
2465	012276	000430			ERRHRD 0,LOG2,ERRDES	;PRINT HARD MESSAGE
2466					BR ABO	;DROP UNIT
2467	012300	042704	177400	MABEE:	BIC #177400,R4	;NEGATE SIGN EXTEND
2468	012304	163704	003312	1\$:	SUB FTLNM,R4	;SEE IF MULTIPLE OF
2469	012310	001413			BEQ HRD	;FTLNM-YES!
2470	012312	103401			BLO SFT	;NO
2471	012314	000773			BR 1\$	;NOT THERE YET
2472						
2473	012316	010337	012330	SFT:	MOV R3,LOG3+2	;ERROR CODE
2474	012322	011137	012332		MOV @R1,LOG3+4	;DESCRIPTION
2475	012326			LOG3:	ERRSOFT 0,LOG3,ERRDES	
2476	012336	000414			BR LOGO	;EXIT
2477	012340	010337	012352	HRD:	MOV R3,LOG3B+2	;HARD ERROR CODE
2478	012344	011137	012354		MOV @R1,LOG3B+4	;DESCRIPTION
2479	012350			LOG3B:	ERRHRD 0,LOG3B,ERRDES	
2480						
2481	012360	011500		ABO:	MOV @R5,0\$	;GET UNIT NUMBER
2482	012362	042700	177770		BIC #177770,R0	;UN-SIGN EXTEND
2483	012366				DODU R0	;USE LOGICAL # 0 DROP
2484	012370			LOGO:	POP R4	;RESTORE
2485	012372				POP R3	
2486	012374				POP R1	
2487	012376				POP R0	
2488	012400	000207			RETURN	;RETURN

```

2491
2492
2493
2494
2495
2496 012402
2497 012402
2498 012404
2499 012406 00500^
2500 012410 032715 000020
2501 012414 001401
2502 012416 005202
2503 012420
2504 012456 016500 000064
2505 012462 016502 000072
2506 012466
2507 012530 005765 000074
2508 012534 001414
2509 012536
2510 012562 005065 000074
2511 012566
2512 012570
2513 012572
2514 012574 045 101 104 UNIT::
2515
2516 012654 045 101 102 RECID::
2517
2518 012746 103 101 116 OVRFLO:
2519
2520 013030 045 101 040 RECID2:
2521

```

```

      BGNMSG  ERRDES      ;ERROR DESCRIPTION
      PUSH    R0
      PUSH    R2
      CLR     R2           ;PRESET TO DATA TYPE
      BIT     #BIT4,R5     ;WHAT PACK TYPE?
      BEQ     2$           ;DATA
      INC     R2           ;COMMAND
2$:    PRINTB #UNIT,<B,DR(R5)>,R2,<B,SYSTAT+1>
      MOV     REC(R5),R0   ;RECORD NUMBER
      MOV     PATTEN(R5),R2 ;DATA EXPECTED
      PRINTB #RECID,R0,<B,CMDSENT(R5)>,<B,R2>,<B,SUCCESS+1(R5)>
      TST     DLV(R5)      ;DLV ERROR?
      BEQ     3$           ;NO
      PRINTB #RECID2,DLV(R5) ;YES-PRINT
      CLR     DLV(R5)      ;RESET
3$:    POP     R2           ;RESTORE
      POP     R0
      ENDMSG      ;EXIT
      .ASCIZ  /%ADRIEN#  %01%A  PAK SENT %01%A FLAG RCVD %03%N/
      .EVEN
      .ASCIZ  /%ABLOCK#  %04%A  COMMAND %02%A EXPCID %03%A SUCCESS %03%N/
      .EVEN
      .ASCIZ  /CAN'T UPDATE ERROR OR STATISTIC:OVERFLOW PENDING/
      .EVEN
      .ASCIZ  /%A RCDB WAS %0c%N/
      .EVEN

```

```
2524                    .SETTL WHCMHDR / SEE WHICH DRIVE IS ACTIVE
2525
2526                    :++
2527                    : INPUTS: DR(R5)
2528                    : OUTPUTS: CARRY=DRIVE (1 OR 0,
2529                    :--
2530
2531
2532 013052 000241       WHCMHDR:: CLC                    ;CLEAR CARRY
2533
2534 013054 105765 000060                TSTB    DR(R5)               ;DR 0?
2535 013060 001401                       BEQ    2$                ;YES
2536 013062 000261                       SEC                    ;NO
2537
2538 013064 000207       2$:    RETURN                    ;RETURN
```

```

2541 .SBTTL CHKSUM / FORM THE PACKET CHECKSUM
2542
2543 :++
2544 : THE CHECKSUM IS A 16 BIT CHECKSUM WITH END-AROUND CARRY.
2545 :
2546 : INPUTS: R0 -> (POINTS TO) TOP OF PACKET
2547 :          R1 = # OF BYTES
2548 : OUTPUTS: R0 -> WHERE TO PUT CHECKSUM
2549 :          R1 = CHECKSUM
2550 :--
2551
2552
2553 CHKSUM:: PUSH R3
2554          PUSH R2
2555 013066 042737 000001 003300 BIC #BIT0,SYSTAT ;"CHECKSUM IS ODD" BIT
2556 013100 032701 000001 BIT #BIT0,R1 ;AN ODD # OF BYTES?
2557 013104 001403 BEQ 1$ ;NO
2558 013106 052737 000001 003300 BIS #BIT0,SYSTAT ;YES
2559
2560 013114 006001 1$: ROR R1 ;/2 FOR WORDS
2561
2562 013116 005003 2$: CLR R3 ;PREP CHECKSUM WORD
2563
2564 013120 062003 3$: ADD (R0)+,R3 ;FORM SUM
2565 013122 005503 ADC R3 ;WITH CARRY
2566 013124 005301 DEC R1 ;MORE WORDS?
2567 013126 001374 BNE 3$ ;YES
2568
2569 013130 032737 000001 003300 BIT #BIT0,SYSTAT ;WAS IT ODD
2570 013136 001405 BEQ 4$ ;NO
2571 013140 112002 MOVB (R0)+,R2 ;YES GET NEXT BYTE
2572 013142 042702 177400 BIC #177400,R2 ;UN-SIGN EXTEND
2573 013146 060203 ADD R2,R3 ;ADD IT IN
2574 013150 005503 ADC R3 ;AND CARRY JUST IN CASE
2575
2576 013152 010301 4$: MOV R3,R1 ;RETURN IT IN CORRECT PLACE
2577 013154 POP R2 ;RESTORE
2578 013156 POP R3
2579 013160 000207 RETURN ;RETURN

```

```

2582 .SBTTL CKCKSM / MODULE TO CHECK THE CHKSUMS
2583
2584 ;**
2585 ; MAKE SURE THE CHECKSUM RECEIVED = THE CHECKSUM CALCULATED.
2586 ; INPUTS: R4 = THE PACKET BYTE COUNT
2587 ;          R0 -> THE PACKET TOP
2588 ; OUTPUTS: CARRY SET IF CHECKSUM CALC'D DOES NOT EQUAL CHECKSUM SENT
2589 ;          R0 -> THE PACKET TOP
2590 ;--
2591
2592
2593 CKCKSM:: PUSH R1
2594          PUSH R0 ;SAVE
2595          MOV R4,R1 ;COPY BYTE COUNT TO CORRECT
2596          CALL CHKSUM ;REGISTER FOR CHKSUM AND
2597                   ;FORM CHECKSUM
2598
2599 ;HERE R0 --> XMITTED CHKSUM, R1-CHKSUM CALC'D
2600
2601          CMPB (R0)+,R1 ;LOWER ORDER CHECK
2602          BNE 2$ ;WRONG
2603
2604          SWAB R1 ;OK-PREP FOR
2605
2606          CMPB (R0)+,R1 ;HIGH ORDER CHECK
2607          BNE 2$ ;WRONG
2608          CLC ;OK-CLEAR SAILING
2609
2610          BR 3$ ;EXIT
2611
2612 2$: SEC ;LET ERROR BE KNOWN
2613
2614
2615 3$: POP R0
2616     POP R1
2617     RETURN ;RETURN

```

```

2620 .SBTTL DOBRK / MODULE TO INIT TU58 AND TEST INTERRUPTS
2621
2622
2623 :++
2624 : DOBRK - SEND RADIAL SERIAL "BREAK" TO DEVICE:
2625 : - SET "BREAK" ON INTERFACE.
2626 : - SEND 8. NULLS
2627 : - CLEAR "BREAK" ON INTERFACE
2628 : - SET VECTORS FOR RCV AND XMIT
2629 : - SEND 2 BYTES OF "INIT"
2630 : - RECEIVE "CONTINUE"
2631 : - IF RECEIVE GARBAGE OR TIMEOUT - ERROR
2632 : - CLEAR INTERRUPTS AND VECTORS
2633 : INPUTS: @R5 BIT14 WAS SET - (SEND BREAK)
2634 : OUTPUTS: @R5 BIT14 CLEAR IF SUCCESSFUL INIT.
2635 :          SYSTAT+1 = RECEIVED BYTE
2636 :          ERRORS R4 = ERROR CODE:
2637 :          - SEND NOT READY TIMEOUT (TOSNDB)
2638 :          - NO RESPONSE
2639 :          - DLV ERROR
2640 :          - CAN'T INIT
2641 :--
2642 013222 105037 013755 DOBRK:: CLR      INITWD+1      ;CLEAR BYTE RECEIVE ADDR
2643 013226 005037 013756      CLR      BRKTO        ;CLEAR TIME OUT CONSTANT
2644 013232 052775 000001 000026      BIS      #BIT0,@XMSR(R5) ;SET "BREAK"
2645 013240 012765 000001 ^00100      MOV      #RSSNIT,CMSNT(R5) ;SAY WE SENT "INIT"
2646 013246 052715 000020      BIS      #BIT4,@R5      ;PAK SENT TYPE =COMMAND, SORT OF
2647 013252 012704 000010      MOV      #8.,R4        ;BREAK-IT'S-BACK COUNT-8
2648 013256      1$: BREAK          ;SUPERVISOR TAKE FIVE
2649      ;FOR ^C CHECK, ETC.
2650 013260 105775 000026      TSTB      @XMSR(R5)      ;READY?
2651 013264 100410      BMI      4$          ;YES
2652 013266 005337 013756      DEC      BRKTO        ;NO, TIME OUT?
2653 013272 001371      BNE      1$          ;NO
2654 013274 012704 000056      MOV      #TOSNDB,R4      ;YES, SET ERROR CODE
2655 013300 004737 012046      CALL     LOG          ;LOG IT
2656 013304 000535      BR      3$          ;EXIT
2657 013306 113775 013752 000030 4$: MOV      BRKWD,@XMDB(R5) ;SEND NULL
2658 013314 005037 013756      CLR      BRKTO        ;RESET TIME OUT
2659 013320 005304      DEC      R4          ;MORE NULLS TO SEND?
2660 013322 001355      BNE      1$          ;YES
2661 013324 005075 000026      CLR      @XMSR(R5)      ;NO, CLEAR "BREAK"
2662 013330 017500 000024      MOV      @RCDB(R5),R0     ;HEAVE "GARBAGE" 1ST BYTE
2663 013334      SETPRI      #PRI00          ;SET TO INTERRUPT FO SURE
2664 013342      SETVEC      TUVECT(R),#RCVINT,#PRI07 ;SET VECTO INFO
2665 013370 062765 000004 000204      ADD      #4,TUVECT(R5) ;AND INC TO SND VECTOR
2666 013376      SETVEC      TUVECT(5),#SNDINT,#PRI07 ;AND SET IT
2667 013424 162765 000004 000204      SUB      #4,TUVECT(R5) ;RESET VECTOR ADDR.
2668 013432 005037 013756      CLR      BRKTO        ;RESET TIME OUT
2669 013436 012704 013754      MOV      #INITWD,R4     ;USE ADDR. FOR SNDBYT
2670 013442 010437 013760      MOV      R4,BRKPTR      ;AND SAVE FOR "WAIT"
2671 013446 052775 000100 000026      BIS      #BIT6,@XMSR(R5) ;ENABLE INTER.
2672 013454 004737 013716      CALL     WAIT        ;AND ENTER LOOP
2673 013460 005715      TST      @R5          ;ABORTED FROM TIME OUT?
2674 013462 100446      BMI      3$          ;YES-EXIT
2675
2676 013464 005037 013756      CLR      BRKTO        ;RESET TIME OUT

```

```

2677 013470 012700 013754      MOV    #INITWD,R4      ;SEND SECOND INIT
2678 013474 010437 013760      MOV    R4,BRKPTR      ;SAVE POINTER AGAIN
2679 013500 052775 000100 000026  BIS    #BIT6,@XMSR(R5) ;AND THEN ENABLE INT
2680 013506 004737 013716      CALL   WAIT          ;AND WAIT
2681 013512 005715              TST     @R5             ;IF ABORTED
2682 013514 100431              BMI     3$              ;THEN EXIT
2683
2684 013516 012704 013755      MOV    #INITWD+1,R4    ;WHERE RESPONSE WILL GO (ADDRESS)
2685 013522 010437 013760      MOV    R4,BRKPTR      ;AND FOR 'WAIT'
2686 013526 052775 000100 000022  BIS    #BIT6,@RCSR(R5) ;ENABLE RECIEVE INT.
2687 013534 004737 013716      CALL   WAIT          ;GET ANSWER
2688 013540 005715              TST     @R5             ;ABORTED?
2689 013542 100416              BMI     3$              ;YES.
2690
2691 013544 123727 013755 000020  CMPB   INITWD+1,#RSCONT ;NO, IS IT 'CONTINUE'?
2692 013552 001003              BNE     2$              ;NOPE-ERROR
2693
2694 013554 042715 040000      BIC     #BIT14,@R5      ;SUCCESSFUL, CLEAR DOBREAK FLAG
2695 013560 000407              BP      3$              ;EXIT
2696
2697 013562 113737 013755 003301 2$:  MOVB   INITWD+1,SYSTAT+1 ;SAVE BUM RESPONSE
2698 013570 012704 000032      MOV    #CNINIT,R4      ;CAN'T INIT CODE
2699 013574 004737 012046      CALL   LOG              ;LOG IT
2700                                ;SCHEDULER WILL TRY AGAIN IF NOT ABORTED
2701
2702 013600 042775 000100 000026 3$:  BIC     #BIT6,@XMSR(R5) ;CLEAR INTERRUPTS
2703 013606 042775 000100 000022  BIC     #BIT6,@RCSR(R5) ; AND FOR RECIEVE
2704 013614              CLRVEC  TUVECT(R5) ;RELEASE RECIEVE VECT.
2705 013622 062765 000004 000204  ADD     #4,TUVECT(R5) ;AND GET SEND ADDR.
2706 013630              CLRVEC  TUVECT(R5) ;AND RELEASE IT
2707 013636 162765 000004 000204  SUB     #4,TUVECT(R5) ;RESTORE POINTER
2708 013644 000207              RETURN ;RETURN

```

```

2711          .SBTTL  INTERRUPT SERVICE ROUTINES AND TIMER
2712
2713 013646      BGNSRV  SNDINT          ;"SEND" INTERRUPT SERVICE:
2714
2715 013646      042775  000100  000026  SNDHND: BIC      #BIT6,@XMSR(R5) ;DISABLE INTERRUPT
2716 013654      112475  000030          MOV      (R4)+,@XMDB(R5);OUTPUT BYTE
2717 013660      ENDSRV
2718
2719
2720
2721 013662      BGNSRV  RCVINT          ;"RCV" INTERRUPT SERVICE:
2722
2723 013662      042775  000100  000022  RCVHND: BIC      #BIT6,@RCR(R5) ;DISABLE INTS
2724 013670      017565  000024  000074  MOV      @RCDB(R5),DLV(R5) ;SAVE BYTE
2725 013676      116524  000074          MOV      DLV(R5),(R4)+ ;BYTE TO BUFFER
2726 013702      005765  000074          TST      DLV(R5) ;ERROR?
2727 013706      100402          BMI      10$ ;YES
2728 013710      005065  000074          CLR      DLV(R5) ;NO CLEAR ERROR
2729 013714      10$:
2730 013714      ENDSRV
2731
2732
2733
2734 013716      000240      WAIT:  NOP          ;WAIT LOOP FOR
2735                                ;INTERRUPT SERVICING
2736 013720      020437  013760      CMP      R4,BRKPTR ;IF=,THEN NO INTERRUPT
2737 013724      001011          BNE      1$ ;GOT ONE!
2738 013726          BREAK          ;SUPERVISOR BREAK
2739 013730          BREAK          ;KILL SOME TIME
2740 013732      005337  013756      DEC      BRKTO ;TIME OUT?
2741 013736      001367          BNE      WAIT ;NO...CONT.
2742 013740      012704  000050      MOV      #TORCVB,R4 ;YES LOAD ERROR #
2743 013744      004737  012046      CALL     LOG ;LOG IT
2744 013750      000207      1$:  RETURN ;RETURN
2745
2746 013752      000500      BRKWD:  .WORD  0 ;NULL
2747 013754      004          INITWD: .BYTE  RSINIT ;INIT COMMAND
2748 013755      000          .BYTE  0 ;RSCONT IS EXPECTED HERE
2749 013756      000000      BRKTO:  .WORD  0 ;TIME OUT
2750 013760      000000      BRKPTR: .WORD  0 ;POINTER TO INITWD
  
```

```

2753 .SBTTL COMPAR/DATA COMPARISON MODULE
2754
2755
2756 :++
2757 :COMPAR - IF "COMPARE DATA" SELECTED, COMPARE EACH DATA BYTE OF PACKET
2758 :TO PATTEN(R5). SAVE NUMBER OF BYTES NOT CORRECT. IF NOT
2759 :0, PRINT SOFT ERROR AND TOTAL # WRONG BYTES. SET 'BAD_DATA_
2760 :IN_PACKET' BIT (BIT6 @R5) FOR HIGHER LEVEL MODULES.
2761 :INPUTS: - (CMPDAT) FLAG TO NOT COMPARE (=1)
2762 :          - PKPTR(R5) POINTS TO DATA PACK.
2763 :OUTPUTS: BIT6 @R5 (BAD DATA FLAG) ADJUSTED.
2764 :          L$LUN - UNIT NUMBER
2765 :          PRNSIZ - SIZE OF PACKET
2766 :--
2767 COMPAR:: PUSH R0 ;COMPARE DATA IS DATA PACKET
2768          PUSH R4 ;TO PATTERN WRITTEN
2769          PUSH R1 ;USING BYTE COUNT IN PACKET
2770          CLR BDBYTS ;CLEAR TOTAL WRONG
2771          MOV PKPTR(R5),R4 ;GET TOP OF PACKET
2772          TST CMPDAT ;COMPARE SELECTED?
2773          BEQ 4$ ;NO-EXIT
2774          INC R4 ;YES, LOCATE COUNT
2775          MOV B @R4,R1 ;GET IT
2776          BIC #177400,R1 ;SIGN-UNEATEND
2777          ;MUST TEST BYTE-WISE...
2778          INC R4 ;-->FIRST DATA BYTE
2779          CMPB PATTEN(R5),(R4)+ ;DATA-WHAT WAS EXPECTED?
2780          BEQ 2$ ;YES
2781          INC BDBYTS ;NO, INCREMENT TOTAL WRONG
2782          DEC R1 ;MORE LEFT?
2783          BNE 1$ ;YES
2784          TST BDBYTS ;ANY WRONG?
2785          BEQ 4$ ;NO
2786          MOV @R5,L$LUN ;GET UNIT NUMBER
2787          BIC #177770,L$LUN ;MASK IT OFF
2788          ERRSUFT 0,MSBDA,ERRDES ;YES-PRINT 'BAD DATA IN PACKET' ERROR
2789          PRINTB #DESC,BDBYTS
2790          BIS #BIT6,@R5 ;LE' 'EM KNOW UPSTAIRS-BAD DATA FLAG
2791          MOV #132,PRNSIZ ;SIZE IS ONE DATA PACK
2792          CALL PRNPAK ;AND PRINT THE PACKET
2793          POP R1 ;RESTORE
2794          POP R4
2795          POP R0
2796
2797          RETURN
2798
2799 BDBYTS: .WORD
2800 DESC: .ASCII /%TOTAL BAD BYTES- %D3%A.%N/
2801 .EVEN
  
```

```

2804      .SBTTL PRNPAK/MODULE TO PRINT DATA PACKET
2805
2806      **
2807      : PRNPAK - IF PRINT DATA PACK_ON_ERROR SELECTED: PRINT EACH BYTE OF PACKET
2808      :             TO BY PKPTR(R5).
2809      : INPUTS: PRNSIZ - # OF BYTES IN PACKET.
2810      : OUTPUTS: NONE
2811      :--
2812
2813 014176 000240      PRNPAK:: NOP                      :PRINTS 1 PACKET
2814                                     :PKPTR(R5)->TOP OF PACKET
2815                                     :PRNSIZ (PASSED)=BYTE COUNT
2816 014200      PUSH      R0
2817 014202      PUSH      R4
2818 014204 105737 002210      TSTB      PRBUF          :PRINT PACKET SELECTED?
2819 014210 001451      BEQ      4$          :NO
2820 014212 016564 060104      MOV      PKPTR(R5),R4    :YES-GET TOP OF PACK
2821 014216 012737 000020 014342 1$:      MOV      #16.,LNCNT :16 BYTES PER LINE
2822 014224 112437 014344      MOVB     (R4)+,PRDAT    :AVOID SIGN EXTEND
2823 014230      PRINTF    #PRFORM,<B,PRDAT> :PRINT BYTE
2824 014256 005337 003330      DEC      PRNSIZ        :ONE LESS
2825 014262 001414      BEQ      3$          :NO MORE
2826 014264 005337 014342      DEC      LNCNT        :NEW LINE?
2827 014270 001355      BNE      2$          :NOT YET
2828 014272      PRINTF    #CARLF          :YES
2829 014312 000741      BR       1$          :NEXT LINE
2830 014314      3$:      PRINTF    #CARLF          :FINISH UP
2831 014334      4$:      POP       R4
2832 014336      POP      R0
2833 014340 000207      RETURN                :RETURN
2834
2835 014342 000000      LNCNT: .WORD
2836 014344 000000      PRDAT: .WORD
2837 014346 045 117 063 PRFORM: .ASCIZ /%03%A /
2838      .EVEN
2839 014356 045 116 000 CARLF: .ASCIZ /%N/
2840      .EVEN
2841
2842 014362      ENDMOD
2843

```

```

2856
2857
2885
2886 014362
2887
2888
2889
2890
2891
2892
2893 014362
2894 014362
2895 014364
2896 014366
2897 014370
2898 014372
2899 014374
2900
2901 014376
2902 014400 012737 003340 015010
2903 014406
2904 014426
2905 014430
2906 014450
2907 014452 017705 000332
2908 014456 032715 004000
2909 014462 001131
2910
2911 014464 011537 015006
2912 014470 042737 177770 015006
2913 014476 116501 000122
2914 014502 042701 177400
2915 014506 116502 000124
2916 014512 042702 177400
2917 014516 116503 000136
2918 014522 042703 177400
2919 014526 116504 000140
2920 014532 042704 177400
2921 014536
2922 014562
2923 014634 116501 000123
2924 014640 042701 177400
2925 014644 116502 000125
2926 014650 042702 177400
2927 014654 116503 000137
2928 014660 042703 177400
2929 014664 116504 000141
2930 014670 042704 177400
2931
2932 014674
2933 014746 023727 015010 003356 28:
2934 014754 103005
2935 014756 062737 000002 015010
2936
2937 014764 000137 014450
2938
2939 014770

.TITLE MISCELLANEOUS SECTIONS
.SBTTL REPORT CODING SECTION

BGNMOD

:++
: THE REPORT CODING SECTION CONTAINS THE
: "PRINTS" CALLS THAT GENERATE STATISTICAL REPORTS.
:--

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

BREAK
MOV #BLK1BL,RPTR ;GET 1ST DEVICE BLOCK
PRINTS #STATHD ;HEADER
BREAK ;^C CHECK
PRINTS #STHD2 ;2ND HEADER
BREAK ;^C CHECK
MOV @RPTR,R5 ;GET DEVICE BLOCK
BIT #BIT111,@R5 ;UNIT NOT TESTED?
BNE 28 ;TRUE, DON'T PRINT STATISTICS
;OK TO PRINT
;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
BIC #177400,R2
MOVB HARDR(R5),R3 ;HARD READ
BIC #177400,R3
MOVB HARDW(R5),R4 ;HARD WRITE
BIC #177400,R4
PRINTS #FM0,RLUN ;SUMMARY/UNIT #
PRINTS #FM,#0,WRINO(R5),RDNO(R5),<B,BDATA(R5)>,R1,R2,R3,R4
MOVB SOFTR+1(R5),R1 ;SAME
BIC #177400,R1 ;AS
MOVB SOFTW+1(R5),R2 ;ABOVE
BIC #177400,R2 ;THIS
MOVB HARDR+1(R5),R3 ;TIME
BIC #177400,R3 ;FOR
MOVB HARDW+1(R5),R4 ;DRIVE
BIC #177400,R4 ;ONE

PRINTS #FM,#1,WRIN1(R5),RDN1(R5),<B,BDATA+1(R5)>,R1,R2,R3,R4
CP RPTR,#LSTDEV ;ALL UNITS DONE?
P-LS 38 ;YES
ADD #2,RPTR ;NO-DO

JMP 18 ;MORE UNITS

38: POP R5

```

2940	014772				POP	R4	
2941	014774				POP	R3	
2942	014776				POP	R2	
2943	015000				POP	R1	
2944	015002				POP	R0	
2945	015004				ENDRPT		
2946	015006	000000			RLUN:	.WORD	
2947	015010	000000			RPTR:	.WORD	
2948							
2949	015012	045	116	045	STATHD:	.ASCII	/%NZA DR BLKS WR BLKS RD BDPK /
2950	015060	104	103	110		.ASCII	@DCHK/RD DCHK/WR DCHK/RD DCHK/WR%N@
2951						.EVEN	
2952	015124	045	101	125	FMO:	.ASCII	/%AUNIT %D1%N/
2953						.EVEN	
2954							
2955	015142	045	101	040	FM:	.ASCII	/%A %D1ZA %D5ZA. %D5ZA. %D3ZA. /
2956	015216	045	104	063		.ASCII	/%D3ZA. %D3ZA. %D3ZA. %D3ZA.%N/
2957						.EVEN	
2958	015266	045	101	040	STMD2:	.ASCII	/%A /
2959	015333	122	105	103		.ASCII	/RECOV RECOV UNRECOV UNRECOV%N/
2960						.EVEN	
2961	015376				ENDMOD		

```

2964
2965
2966
2967
2968
2969
2970
2971 0.5376
2972
2973 015376 000240
2974 015400 105037 016150
2975 015404
2976 015412
2977 015414 005237 016150
2978 015420 012737 003340 003304
2979 015426 005004
2980 015430 017705 165650
2981 015434 010415
2982 015436 052715 120000
2983 015442 052715 004000
2984 015446 006304
2985 015450 016465 024346 000102
2986 015456 006204
2987 015460 023727 003304 003356
2988 015466 103005
2989 015470 062737 000002 003304
2990 015476 005204
2991 015500 000753
2992
2993 015502 022737 000010 002012
2994 015510 103005
2995 015512
2996 015522
2997
2998 015524 012737 003340 003304
2999 015532 005004
3000 015534 017705 165544
3001 015540 010437 002074
3002 015544
3003 015552
3004 015554 042715 004000
3005 015560 012203
3006 015562 012265 000204
3007 015566 112265 000061
3008 015572 005202
3009 015574 012237 016152
3010 015600 052715 040000
3011 015604 032765 000400 000060
3012 015612 001011
3013 015614 032765 001000 000060
3014 015622 001005
3015 015624
3016 015634
3017
3018 015636 105737 016150
3019 015642 001412
3020

.SBTTL INITIALIZE SECTION

:++
: THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
: AT THE BEGINNING OF EACH PASS.
:--

BGNINIT

INIT: NOP
      CLRB STRT ;FOR STATS CLEAR
      REDEF #EF,START ;START COMMAND?
      BNCOMPLETE INIT2 ;NO
      INC STRT ;YES. SET START FLAG
INIT2: MOV #BLKTBL,DEVPTR ;SET ALL UNITS ABORTED:
      CLR R4 ;UNIT NUMBER
1$: MOV @DEVPTR,R5 ;GET POINTER
      MOV R4,@R5 ;INSERT UNIT #
      BIS #BIT15,BIT13,@R5 ;SET ABORTED, HALTED
      BIS #BIT11,@R5 ;SET UNIT NOT TESTED
      ASL R4 ;*2 FOR LOOK-UP
      MOV BUFTBL(R4),RCVBUF(R5) ;SETUP POINTER TO UNIT'S BUFFER
      ASR R4 ;CORRECT BACK TO UNIT #
      CMP DEVPTR,#LSTDEV ;LAST DEVICE DONE?
      BHIS CHECK ;YES
      ADD #2,DEVPTR ;NO-GET
      INC R4 ;NEXT DEVICE AND
      BR 1$ ;SERVICE

CHECK: CMP #8,,L$UNIT ;MAKE SURE NOT
      BHIS GETHRD ;TOO MANY UNITS
      ERSF 101,,TOMANY ;TOMANY-REQUEST ^C
      DJCLN ;EXIT

GETHRD: MOV #BLKTBL,DEVPTR ;INIT TABLE POINTER
      CLR R4 ;CLEAR DEVICE COUNTER
1$: MOV @DEVPTR,R5 ;GET STATUS WORD
      MOV R4,L$LUN ;UNIT NUM. IN CASE ERROR
      GPHARD R4,R2 ;GET HARD INFO
      BNCOMPLETE 3$
      BIC #BIT11,@R5 ;UNIT IS TESTED!
      MOV (R2)+,R3 ;R3=CSR
      MOV (R2)+,TUVECT(R5) ;GET VECTOR ADDRESS
      MOVB (R2)+,DR+1(R5) ;SAVE UNIT SUMMARY
      INC R2 ;GET TO WORD BOUND
      MOV (R2)+,PDTFLG ;AND GET PDT FLAG
      BIS #BIT14,@R5 ;SET SEND BREAK FLAG
      BIT #BIT8,DR(R5) ;DRIVE 0?
      BNE 13$ ;YES
      BIT #BIT9,DR(R5) ;DRIVE 1?
      BNE 13$ ;OK
      ERSF 102,,NODRVS ;NEITHER?
      DJCLN ;EXIT

13$: TSTB STRT ;START COMMAND?
      BEQ 14$ ;NO, DONT CLEAR
      ;YES-CLEAR STATS

```

```

3021 015644 012702 000202      MOV      #BLKEND,R2      ;R2-->END OF STATS
3022 015650 012701 000110      MOV      #WRTNO,R1      ;FORM ADDRESS OF START:
3023 015654 060501              ADD      R5,R1      ;R1-->START OF STATS.
3024 015656 162702 000110      SUB      #WRTNO,R2      ;FORM # 10 CLEAR
3025
3026 015662 105021              2$:      CLRB      (R1)+      ;CLEAR 'EM
3027 015664 005302              DEC      R2      ;MORE?
3028 015666 001375              BNE      2$      ;YES
3029 015670 042715 120000      14$:      BIC      #B1115!B1113,0R5 ;SET NOT ABORTED NOT HALTED
3030 015674 010365 000022      MOV      R3,RC5R(R5) ;GET DEVICE REGISTERS:
3031 015700 062703 000002      ADD      #2,R3
3032 015704 010365 000024      MOV      R3,RCDB(R5)
3033 015710 062703 000002      ADD      #2,R3
3034 015714 010365 000026      MOV      R3,XMSR(R5)
3035 015720 062703 000002      ADD      #2,R3
3036 015724 105737 016152      TSTB      PDIFLG      ;UNIT A PDT?
3037 015730 001402              BEQ      4$      ;NO
3038 015732 162703 000004      SUB      #4,R3      ;YES...RCDB=XMDB
3039 015736 010365 000030      4$:      MOV      R3,XMDB(R5)
3040 015742 005065 000072      CLR      PATTEN(R5) ;ZERO DATA PATTERN
3041 015746 005065 000002      CLR      RETRY(R5) ;NO RETRIES
3042 015752 005065 000064      CLR      REC(R5) ;NO RECORD
3043 015756 005065 000076      CLR      SUCCS(R5) ;NO SUCCESS
3044 015762 005065 000074      CLR      DLV(R5) ;NO DLV ERROR
3045 015766 005037 003332      CLR      ALLGON ;OK TO PRINT STATISTICS
3046 015772 062737 000002 003304 3$:      ADD      #2,DEVPTR ;-->NEXT DEVICE
3047 016000 005204              INC      R4      ;INCREMENT UNIT NUMBER
3048 016002 020437 002012      CMP      R4,LSUNIT ;MORE UNITS?
3049 016006 001252              RNE      1$      ;YES, GP HARD THE NEXT
3050
3051 016010 005037 003300      CLR      SYSTAT ;SYSTEM STATUS WORD
3052 016014              RFLAGS      FLGLOC ;GET USER FLAGS
3053 016022 005037 003324      5$:      CLR      BLKER ;NO ERROR
3054 016026 013737 002204 003302 SETLEN: MOV      LENGTH,TAPLEN ;GET # OF RECOR. S
3055 016034 006237 003302      ASR      TAPLEN ;GET # BLCKS PER TRACK
3056 016040 012737 000200 003326      MOV      #200,SECREC ;PRESET SECOND START AT 200
3057 016046 022737 000200 003302      CMP      #200,TAPLEN ;# BLKS > 128.?
3058 016054 101003              BHI      3$      ;NO-SWITCH TRACKS 2ND PASS
3059 016056 012737 000400 003326      MOV      #400,SECREC ;YES-START AT 400
3069
3081
3082 016064              3$:      ENDINIT
3083
3084
3085 016066      124      117      117 TOMANY: .ASCIZ /TOO MANY UNITS MAX.=8 /
3086              .EVEN
3087 016116      123      105      114 NODRVS: .ASCIZ /SELECT AT LEAST 1 DRIVE /
3088              .EVEN
3089 016150 000000      STRT:: .WORD
3090 016152 000000      PDIFLG:: .WORD ;TU58 IS IN PDT
3091 016154 000000      FLGLOC:: .WORD ;USER FLAGS

```

```

3094
3095
3096
3097
3098
3099 016156 000240
3100 016156 000240
3101 016160
3102 016206 012737 003340 016264
3103 016214 017705 000044
3104 016220 032715 104000
3105 016224 100403
3106 016226 005775 000022
3107 016232 000240
3108 016234 023727 016264 003'56 28:
3109 016242 103004
3110 016244 062737 000002 016264
3111 016252 000760
3112 016254
3113 016262
3114 016264 000000
3115
3116
3117
3118
3119
3120
3121 016266
3122 016306 011500
3123 016310 042700 177770
3124 016314
3125 016316 000002
3126 016320 045 101 101

; THE AUTO DROP CODE IS INVOKED WHEN THE ADR FLAG IS SET AND CHECKS FOR
; A VALID INTERFACE LOCATION. DROPS UNIT IF INTERFACE IS NOT THERE.
;--
BGN AUTO
NOP ; AUTO DROP ROUTINE
SEI VEC #4, #TRPHND, #PRI07 ; GET BUS TRAP VEC.
MOV #BLKTBL, TRPPTR ; GET TOP OF DATA BLOCK TABLE
18: MOV @TRPPTR, R5 ; GET DATA BLGCK
BIT #BIT15:BIT11, @R5 ; NOT TESTED OR ABORTED?
BMI 28 ; YES
TST @RCSR(R5) ; NO-VALID ADDRESS?
NOP ; YES... (TRAP IF NOT)
28: CMP TRPPTR, #LSIDEV ; MORE TO TRY?
BHS 38 ; NO
ADD #2, TRPPTR ; ON TO NEXT
BR 18 ; GET IT
38: CLR VEC #4 ; RESTORE
END AUTO
TRPPTR: .WORD

; ILLEGAL ADDRESS TRAP HANDLER:
TRPHND: PRINTF #MSAUTO ; SAY "AUTO DROPPED"
MOV @R5, R0 ; GET UNIT #
BIC #177770, R0 ; MASK IT OFF
DODU R0 ; DROP HIM
RTI
MSAUTO: .ASCIZ /%AUTO DROP: %N/

```

3129
3130
3131
3132
3133
3134
3135
3136 016340
3137 016340 005737 003332
3138 016344 001004
3139 016346 005737 002206
3140 016352 001401
3141 016354
3142
3149
3161
3162 016356

.SBTTL CLEANUP CODING SECTION

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

BGNCLN		
TST	ALLGON	;ENTRANCE FROM ALL-UNITS-ABORTED?
BNE	18	;YES-EXIT
TST	STAEOP	;NO-STATS AT EOP?
BEQ	18	;NO
DORPT		;YES

18: ENDCLN

DROP UNIT SECTION

```

3165
3166
3167
3168
3169
3170
3171
3172 016360
3173
3174 016360
3175 016362
3176 016364 004737 016424
3177 016370 052715 120000
3178 016374
3179 016376
3180 016400
3181
3187
3199
3200 016422
3201 016424 012737 003340 016454 GETR5:
3202 016432 017705 000016 1$:
3203 016436 005300
3204 016440 100404
3205 016442 062737 000002 016454
3206 016450 000770
3207 016452 000207
3208 016454 000000
3209
3210 016456 045 101 104 ABOMSG:
32 1

```

```

.SBTL DROP UNIT SECTION

:++
: THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
: TO NO LONGER BE TESTED.
:--

BGNDU

PUSH R0
PUSH R5
CALL GETR5
BIS #BIT15:BIT13,R5
POP R5
POP R0
PRINTF #ABOMSG,R0

;R0=UNIT NUMBER
;SAVE IT
;SAVE PRESENT UNIT POINTER
;GET POINTER TO UNIT
;SET ABORTED, HALTED
;RESTORE PRESENT UNIT POINTER
;RETRIEVE UNIT NUMBER

ENDDU
MOV #BLK1BL,PTR
MOV @PTR,R5
DEC '0
BMI <S
ADD #2,PTR
BR 1$
2$:
PTR:
RETURN
.WORD

;-->UNIT 0
;GET STATUS WORD
;CORRECT UNIT?
;YES
;NO,-->NEXT
;CONTINUE

.ASCIIZ /%ADROPPED UNIT %D1%N/
.EVEN

```

3214
3215
3216
3217
3218
3219
3220
3221
3222 016504
3223
3224
3230
3242
3243
3244
3245 016504
3246

.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.
;--

BGNAU

;THE INIT CODE CONTAINS ALL CODE NECESSARY TO ADD A UN.I.

ENDAU

```
3305 .SBTTL TEST 1 / DEVICE SELF-DIAGNOSTIC EXECUTION
3306
3307 016506 BGNMOD
3308 .NLIST ML,BEX
3309
3310 016506 BGNIST
3311 016506 TSTID #TST1
      016506 012737 016552 003320 MOV #TST1,TSTTOP ;SAVE ADDR OF TEST
      016514 004737 005754 CALL SETUP ;INIT UNITS TSTPC
      016520 004737 005602 CALL SETDR ;GET 1ST DRVS.
      016524 004737 006022 CALL RUN ;DO TEST
      016530 004737 005500 CALL SWAPDR ;GET NEXT DRVS.
      016534 103004 BCC 64$ ;BR NO 2ND DRVS
      016536 004737 005754 CALL SETUP ;REINIT UNITS TSTPC
      016542 004737 006022 CALL RUN ;REPEAT TEST
      016546 ;DONE
3312 016546 EXIT TST
3313
3314 016552 TST1: TUSELF
3315 016672 005237 003314 INC DONE
3316 016676 000207 RETURN
3317
3318
3319 016700 ENDTST
```

```

3322          .SBTTL TEST 2 / SEEK EOT,BOT
3323
3324 016702          BGNTEST
3325 016702          TSTID #TST2
          016702 012737 016746 003320          MOV #TST2,TSTTOP ;SAVE ADDR OF TEST
          016710 004737 005754          CALL SETUP ;INIT UNITS TSTPC
          016714 004737 005602          CALL SETDR ;GET 1ST DRVS.
          016720 004737 006022          CALL R'IN ;DO TEST
          016724 004737 005500          CALL SWAPDR ;GET NEXT DRVS.
          016730 103004          BCC 64$ ;BR NO 2ND DRVS
          016732 004737 005754          CALL SETUP ;REINIT UNITS TSTPC
          016736 004737 006022          CALL RUN ;REPEAT TEST
          016742          64$: ;DONE
3326 016742          EXIT TST
3327
3328
3329 016746 005004          TST2: CLR R4 ;R4=INDEX INTO RECORD TABLE
3330 016750 016465 017130 000064 1$: MOV RECDAT(R4),REC(R5) ;GET THE RECORD
3331
3332 016756          TUSEEK REC(R5),DR(R5) ;SEEK IT
3333
3334 017106 062704 000002          ADD #2,R4 ;POINT TO NEXT RECORD
3335 017112 026427 017130 177777          CMP RECDAT(R4),#-1. ;LAST ONE DONE?
3336 017120 001313          BNE 1$ ;NO-LOOP
3337 017122 005237 003314          INC DONE ;YES-SET DONE FLAG
3338 017126 000207          RETURN
3339
3340 017130 000000          RECDAT: 0. ;BOT
3341 017132 000200          200 ;BOT OTHER TRACK
3342 017134 000177          177 ;EOT
3343 017136 000377          377 ;EOT OTHER TRACK
3344 017140 000400          400 ;BOT AGAIN
3345 017142 177777          -1.
3346 017144          ENDTST

```

```

3349 .SBTTL TEST 3 / HIGH ACTIVITY WRITE/READ
3350
3351 ; WRITE THEN READ VARYING DATA FOR ALL PHYSICALLY ADJACENT BLOCKS AROUND
3352 ; A RECORD, GO HALF-WAY INTO REMAINING TAPE REPEAT UNTIL EOT.
3353
3354 017146 BGNTEST
3355 017146 TSTID #TST3
      017146 012777 017212 003320 MOV #TST3,TSTTOP ;SAVE ADDR OF TEST
      017154 004777 005754 CALL SETUP ;INIT UNITS TSTPC
      017160 004737 005602 CALL SEIDR ;GET 1ST DRVS.
      017164 004737 006022 CALL RUN ;DO TEST
      017170 004737 005500 CALL SWAPDR ;GET NEXT DRVS.
      017174 03004 BCC 64$ ;BR NO 2ND DRVS
      017176 004737 005754 CALL SETUP ;REINIT UNITS TSTPC
      017202 004737 006022 CALL RUN ;REPEAT TEST
      017206 64$: ;DONE
3356 017206 EXIT TST
3357
3358
3359 017212 012765 000100 000066 TST3: MOV #100,TMP(R5) ;INIT TO HALF OF REMAINING
3360 017220 005004 CLR R4 ;FOR INDEX INTO DATA TABLE
3361 017222 005065 000064 CLR REC(R5) ;START AT RECORD 0
3362 017226 016465 020522 000072 1$: MOV TST3PT(R4),PATTEN(R5) ;GET DATA
3363 017234 TUWRIT PATTEN(R5),REC(R5),#512.,DR(R5),#0
3364 020024 TUREAD REC(R5),#512.,DR(R5),#0
3365 020424 062704 000002 ADD #2,R4 ;POINT TO NEXT DATA
3366 020430 005764 020522 TST TST3PT(R4) ;END?
3367 020434 001402 BEQ 2$ ;YES
3368 020436 000137 017226 JMP 1$ ;NO-WRITE, READ NEW DATA
3369 020442 005004 2$: CLR R4 ;POINT TO FIRST DATA
3370 020444 062765 000200 000064 ADD #200,REC(R5) ;BUT NOW USE ADJACENT RECORD
3371 020452 032765 001000 000064 BIT #1000,REC(R5) ;ALL ADJACENT RECORDS DONE?
3372 020460 001002 PNE 3$ ;YES
3373 020462 000137 017226 JMP 1$ ;NO-WRITE, READ AT NEW RECORD
3374 020466 162765 001000 000064 3$: SUB #1000,REC(R5) ;RESTORE TO NEXT RECORD
3375 020474 066565 000066 000064 ADD TMP(R5),REC(R5) ;HALF INTO REST OF TAPE
3376 020502 006265 000066 ASR TMP(R5) ;HALF OF HALF FOR NEXT TIME
3377 020506 103402 BCS 4$ ;DONE?
3378 020510 000137 017226 JMP 1$ ;NO
3379 020514 005237 003314 4$: INC DONE ;YES-SET FLAG
3380 020520 000207 RETURN
3381 020522 000000 TST3PT: .WORD 000000
3382 020524 1. .WORD 125252
3383 020526 177777 .WORD 177777
3384 020530 052525 .WORD 052525
3385 020532 000000 .WORD 000000
3386
3387
3388 020534 ENDTST

```

```

3391
3392
3393
3394 020536          BGNTST
3395 020536          TSTID  #TST4
      020536 012737 020602 003320          MOV  #TST4,TSTTOP ;SAVE ADDR OF TEST
      020544 004737 005754          CALL  SETUP ;INIT UNITS TSTPC
      020550 004737 005602          CALL  SETDR ;GET 1ST DRVS.
      020554 004737 006022          CALL  RUN ;DO TEST
      020560 004737 005500          CALL  SWAPDR ;GET NEXT DRVS.
      020564 103004          BCC 648 ;BR NO 2ND DRVS
      020566 004737 005754          CALL  SETUP ;REINIT UNITS TSTPC
      020572 004737 006022          CALL  RUN ;REPEAT TEST
      020576          ;DONE
3396 020576          EXIT TST
3397
3398
3399 020602 005065 000064 TST4: CLR REC(R5) ;START AT REC 0
3400 020606 013765 003302 007066 MOV TAPLEN,IMP(R5) ;GET THE # OF BLOCKS PER TRACK
3401 020614 005065 000062 CLR TRK(R5) ;TRK(R5)=1ST OR 2ND PASS COUNTER
3402 020620 016565 000064 000072 1$: MOV REC(R5),PATTEN(R5) ;USE RECORD NO. FOR DATA
3403 020626 005737 002214 TST DRVCHK ;ADD DR #?
3404 020632 001403 BEQ 10$ ;NO
3405 020634 066565 000060 000072 ADD DR(R5),PATTEN(R5) ;YES, ADD DRIVE ID
3406 020642 10$: TUWRT PATTEN(R5),REC(R5),#512,DR(R5),#0
3407 021432 005365 000066 DEC IMP(R5) ;DO ALL RECORDS FOR THIS TRACK?
3408 021436 001404 BEQ 2$ ;YES-GET OTHER TRACK
3409 021440 005265 000064 INC REC(R5) ;NO-ONTO NEXT RECORD
3410 021444 000137 020620 JMP 1$ ;EXECUTE THE WRITE
3411 021450 005765 000062 2$: TST TRK(R5) ;DONE 2 TRACKS?
3412 021454 001012 BNE TST4EX ;YES-EXIT
3413 021456 005265 000062 INC TRK(R5) ;NO-SET FLAG FOR NEXT PASS
3414 021462 013765 003326 000064 MOV SECNEC,REC(R5) ;GET NEW STARTING BLOCK #
3415 021470 013765 003302 000066 MOV TAPLEN,IMP(R5) ;RESET # OF BLOCKS
3416 021476 000137 020620 JMP 1$ ;AND EXECUTE
3417 021472 005237 003314 TST4EX: INC DONE ;DONE
3418 021476 000207 RETURN ;RETURN
3419
3420 021510          ENDTST

```

```

3423                                     .SBTTL TEST 5 / READ SELECTED NUMBER OF BLOCKS
3424
3425 021512                             BGNTEST
3426 021512                             TSTID #TST5
                                MOV #TST5,TSTTOP ;SAVE ADDR OF TEST
                                CALL SETUP ;INIT UNITS TSTPC
                                CALL SETDR ;GET 1ST DRVS.
                                CALL RUN ;DO TEST
                                CALL SWAPDR ;GET NEXT DRVS.
                                BCC 64$ ;BR NO 2ND DRVS
                                CALL SETUP ;REINIT UNITS TSTPC
                                CALL RUN ;REPEAT TEST
                                ;D .E
                                64$:
3427 021552                             EXIT TST
3428
3429
3430 021556 005065 000064 TST5: CLR REC(R5) ;START AT REC 0
3431 021562 013765 003302 000066 MOV TAPLEN,TMP(R5) ;GET THE # OF BLOCKS PER TRACK
3432 021570 005065 000062 CLR TRK(R5) ;TRK(R5)=1ST OR 2ND PASS
3433 021574 016565 000064 000072 1$: MOV REC(R5),PATTEN(R5) ;USE RECORD NO. AS DATA
3434 021602 005737 002214 TST DRVCHK ;ADD DR #?
3435 021606 001403 BEQ 10$ ;NO
3436 021610 066565 000060 000072 ADD DR(R5),PATTEN(R5) ;ADD IN DRIVE ID
3437 021616 10$: TUREAD REC(R5),#512.,DR(R5),#0
3438 022216 005365 000066 DEC TMP(R5) ;DO ALL RECORDS THIS TRACK?
3439 022222 001404 BEQ 2$ ;YES-GET OTHER TRACK
3440 022224 005265 000064 INC REC(R5) ;NO-NEXT RECORD
3441 022230 000137 021574 JMP 1$ ;EXECUTE THE READ
3442 022234 005765 000062 2$: TST TRK(R5) ;DONE 2 TRACKS?
3443 022240 001012 BNE TSTSE. ;YES-EXIT
3444 022242 005265 000062 INC TRK(R5) ;NO-SET FLAG FOR NEXT PASS
3445 022246 013765 003326 000064 MOV SECREC,REC(R5) ;GET NEW STARTING BLOCK #
3446 022254 013765 003302 000066 MOV TAPLEN,TMP(R5) ;RESET # OF BLOCKS
3447 022262 000137 021574 JMP 1$ ;AND EXECUTE
3448 022266 005237 003314 TSTSEX: INC DONE ;DONE
3449 022272 000207 RETURN ;RETURN
3450
3451 022274                             ENDTST

```

```

3454 .SBTTL TEST 6 / WRITE-VERIFY SELECTED NUMBER OF BLOCKS
3455
3456 022276 BGNTST
3457 022276 TSTID #TST6
      022276 012737 022342 003320 MOV #TST6,TSTTOP ;SAVE ADDR OF TEST
      022304 004737 005754 CALL SETUP ;INIT UNITS TSTPC
      022310 004737 005602 CALL SETDR ;GET 1ST DRVS.
      022314 004737 006022 CALL RUN ;DO TEST
      022320 004737 005500 CALL SWAPDR ;GET NEXT DRVS.
      022324 103004 BCC 648 ;BR NO 2ND DRVS
      022326 004737 005754 CALL SETUP ;REINIT UNITS TSTPC
      022332 004737 006022 CALL FJN ;REPEAT TEST
      022336 ;DONE
3458 022336 EXIT TST 648:
3459
3460
3461 022342 005065 000064 TST6: CLR REC(R5) ;START AT REC 0
3462 022346 013765 003302 000066 MOV TAPLEN,TMP(R5) ;GET THE # OF BLOCKS PER TRACK
3463 022354 005065 000062 CLR TRK(R5) ;TRK(R5)-1ST OR 2ND PASS
3464 022360 016565 000064 000072 1$: MOV REC(R5),PATTEN(R5) ;USE RECORD NO. FOR DATA
3465 022366 005737 002214 TST DRVCHK ;ADD DR #?
3466 022372 001403 BEQ 10$ ;NO
3467 022374 066565 000060 000072 ADD DR(R5),PATTEN(R5) ;ADD DRIVE ID
3468 022402 10$: TUWRIT PATTEN(R5),REC(R5),#512,DR(R5),#1
3469 023172 005365 000066 DEC TMP(R5) ;DO ALL RECORDS FOR THIS TRACK?
3470 023176 001404 BEQ 2$ ;YES-GET OTHER TRACK
3471 023200 005265 000064 INC REC(R5) ;NO-NEXT RECORD
3472 023204 000137 022360 JMP 1$ ;EXECUTE THE WRITE
3473 023210 005765 000062 2$: TST TRK(R5) ;DONE 2 TRACKS?
3474 023214 001012 BNE TST6EX ;YES-EXIT
3475 023216 005265 000062 INC TRK(R5) ;NO-SET FLAG FOR NEXT PASS
3476 023222 013765 003326 000064 MOV SECREC,REC(R5) ;GET NEW STARTING BLOCK #
3477 023230 013765 003302 000066 MOV TAPLEN,TMP(R5) ;RESET # OF BLOCKS
3478 023236 000137 022360 JMP 1$ ;AND EXECUTE
3479 023242 005237 003314 TST6EX: INC DONE ;DONE
3480 023246 000207 RETURN ;RETURN
3481
3482 023250 ENDT.

```

```

3485 .SBTTL TEST 7 / READ-REDUCED THRESHOLD SELECTED NUMBER OF BLOCKS
3486
3487 023252 BGNTEST
3488 023252 TSTID #TST7
      023252 012737 023316 003320 MOV #TST7,TSTTOP ;SAVE ADDR OF TEST
      023260 004737 005754 CALL SETUP ;INIT UNITS TSTPC
      023264 004737 005602 CALL SETDR ;GET 1ST DRVS.
      023270 004737 006022 CALL RUN ;DO TEST
      023274 004737 005500 CALL SWAPDR ;GET NEXT DRVS.
      023300 103004 BCC 648 ;BR NO 2ND DRVS
      023302 004737 005754 CALL SETUP ;REINIT UNITS TSTPC
      023306 004737 006022 CALL RUN ;REPEAT TEST
      023312 648: ;DONE
3489 023312 EXIT TST
3490
3491
3492 023316 005065 000064 TST7: CLR REC(R5) ;START AT REC 0
3493 023322 013765 003302 000066 MOV TAPLEN,IMP(R5) ;GET THE # OF BLOCKS PER TRACK
3494 023330 005065 000062 CLR TRK(R5) ;TRK(R5)=1ST OR 2ND PASS
3495 023334 016565 000064 000072 1$: MOV REC(R5),PATTEN(R5) ;USE RECORD NO. FOR DATA
3496 023342 005737 002214 TST DRVCHK ;ADD DR #?
3497 023346 001403 BEQ 10$ ;NO
3498 023350 066565 000060 000072 ADD DR(R5),PATTEN(R5) ;ADD DRIVE ID
3499 023356 10$: TUREAD REC(R5),#512.,DR(R5),#1
3500 023756 005365 000066 DEC IMP(R5) ;DO ALL RECORDS THIS TRACK?
3501 023762 001404 BEQ 2$ ;YES-GET OTHER TRACK
3502 023764 005265 000064 INC REC(R5) ;NO-NEXT RECORD
3503 023770 000137 023334 JMP 1$ ;EXECUTE THE READ
3504 023774 005765 000062 2$: TST TRK(R5) ;DONE 2 TRACKS?
3505 024000 001012 BNE TST7EX ;YES-EXIT
3506 024002 005265 000062 INC TRK(R5) ;NO-SET FLAG FOR NEXT PASS
3507 024006 013765 003326 000064 MOV SECRC,REC(R5) ;GET NEW STARTING BLOCK #
3508 024014 013765 003302 000066 MOV TAPLEN,IMP(R5) ;RESET # OF BLOCKS
3509 024022 000137 023334 JMP 1$ ;AND EXECUTE
3510 024026 005237 023314 TST7EX: INC DONE ;DONE
3511 024032 000207 RETURN ;RETURN
3512
3513 024034 ENDTST
  
```

3516
3517
3518
3519

000144

.SBTTL PATCH AREA
.REPT 100.
.WORD
.FNR

3522
3523
3524
3525
3526 024346 025426
3527 024350 026464
3528 024352 027522
3529 024354 030560
3530 024356 031616
3531 024360 032654
3532 024362 033712
3533 024364 034750
3534
3535
3536
3537
3538
3539 024366 023
3540 024367 023
3541
3542 024370
3543
3544
3545
3546 025426
3547 026464
3548 027522
3549 030560
3550 031616
3551 032654
3552 033712
3553 034750
3554
3555
3556
3557 036006

.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 RSKOFF
.BYTE RSKOFF ;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

ENDMOD

3581
3592
3593
3621
3622 036006
3623
3624
3625
3626
3627
3628
3629
3630
3631
3632
3633 036006
3634
3635
3636 036010
3637 036020
3638 036030
3639 036036
3640 036044
3641
3647
3648 036052
3649
3650 036052 124 125 065
3651 036063 126 105 103
3652 036100 120 104 124
3653 036131 124 105 123
3654 036146 124 105 123
3655
3656
3657

.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

GPRMA MSG1,0,0,160000,177777,YES
GPRMA MSG1B,2,0,0,776,YES
GPRML MSG1C,6,1,YES
GPRML MSG2,4,1,YES
GPRML MSG3,4,2,YES

ENDHRD

MSG1: .ASCIZ /TUSB CSR/
MSG1B: .ASCIZ /VECTOR ADDR./
MSG1C: .ASCIZ /PDT (PARALLEL) INTERFACE/
MSG2: .ASCIZ /TEST DRIVE 0/
MSG3: .ASCIZ /TEST DRIVE 1/
.FVEN

```

3664
3667
3668
3669
3670
3671
3672
3673
3674
3675
3676
3677 036164
3678
3679 036166
3680 036200
3681 036206
3682 036214
3683 036222
3684
3685 036230
3692
3693 036242
3694
3695 036242 116 125 115 MSG4: .ASCIZ 'NUMBER OF BLOCKS:TEST 4-7 (8 TO 512)'
3696 036307 101 104 104 MSG4B: .ASCIZ /ADD DR # TO DATA PATTERN:TEST 4-7/
3697 036351 123 124 101 MSG5: .ASCIZ /STATISTICS PRINTED AT EOP/
3698 036403 103 117 115 MSG6: .ASCIZ /COMPARE DATA ON READ/
3699 036430 120 122 111 MSG7: .ASCIZ /PRINT PACKET ON ERROR/
3700 036456 043 040 105 MSG8: .ASCIZ /# ERRORS = DVC FATAL IF 'EVL'SET/
3701 .EVEN

```

.SBTTL SOFTWARE PARAMETER CODING SECTION

..

THE SOFTWARE 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.

--

BGNSFT

GPRMD MSG4,0,D,1777,8.,512.,YES
GPRML MSG4B,10,1,YES
GPRML MSG5,2,1,YES
GPRML MSG6,6,1,YES
GPRML MSG7,4,1,YES

GPRMD MSG8,10.,D,377,i,254.,YES

SFTOUT: ENDSFT

3704	000016	.REPT 14.	;LASTAD CORRECTION
3705		.WORD	
3706		.ENDR	
3713	036554	LASTAD	
	036560	LSLAST::	
3714	036560	ENDMOD	
3715			
3716	036560	BGNSETUP	?
3717	036560	BGNPIAB	
3718	036564	176500	
3719	036566	300	
3720	036570	3	
3721	036572	0	
3722	036574	ENDPIAB	
3723	036574	ENDSETUP	
3724	000001	.END	

PARAMETER CODING
SYMBOL TABLE

MACRO M1110 02-AUG-79 15:53 PAGE 136-1

N 7

SEQ 0091

ABNDX = 000004 G	CHKANS 010112 G	C\$MSG = 000023	EVL = 000004 G	G\$RADA= 000140
ABO 012360	CHKEND 010536 G	C\$OPEN= 000034	EVLTHR 002216	G\$RADB= 000000
ABOMSG 016456	CHKERR 010624	C\$PNTB= 000014	EXOFF 007213	G\$RADD= 000040
ABONM 006336	CHKPKS 010202 G	C\$PNTF= 000017	EXON 007212	G\$RADL= 000120
ADR = 000020 G	CHKPTR 010200	C\$PNTS= 000016	E\$END = 002100	G\$RADO= 000020
ALLGON 003332 G	CHKREE 010676	C\$PNTX= 000015	E\$LOAD= 000035	G\$XFER= 000004
ASSEMB= 000010	CHKRET 011242	C\$QIO = 000377	FLGLUC 016154 G	G\$YES = 000010
BDATA = 000134 G	CHKSUC 011536 G	C\$RDBU= 000007	FM 015142	HARDR = 000136 G
BDBYTS 014140	CHKSUM 013066 G	C\$REFG= 000047	FMO 015174	HARDW = 000140 G
BDCHK = 000022 G	CHK8 010132	C\$RESE= 000033	FTLNM 003312	HELP = 000000
BDCOM = 000014 G	CKCKSM 01162 G	C\$REVI= 000003	F\$AU = 000015	HQE = 100000 G
BIT0 = 000001 G	CLRALL 005660 G	C\$RFLA= 000021	F\$AUTO= 000020	HRD 012340
BIT00 = 000001 G	CLRBUF 005720 G	C\$RPT = 000025	F\$BGN = 000040	HRDRD = 000016 G
BIT01 = 000002 G	CLRPTR 005752	C\$SEFG= 000046	F\$CLEA= 000007	HRDWR = 000020 G
BIT02 = 000004 G	CMD\$NT= 000100 G	C\$SPRI= 000041	F\$DU = 000016	HRD1 011164
BIT03 = 000010 G	CMNDER= 000040 G	C\$SVEC= 000037	F\$END = 000041	IBE = 010000 G
BIT04 = 000020 G	CM\$DAT 002212	C\$TPRI= 000013	F\$HARD= 000004	IDPTR 003316 G
BIT05 = 000040 G	CNINIT= 000032 G	DESC 014142	F\$HW = 000013	IDU = 000040 G
BIT06 = 000100 G	COMPAR 013762 G	DEVPT 003304 G	F\$INIT= 000006	IER = 020000 G
BIT07 = 000200 G	C\$NRDY 003334 G	DEV0 003360	F\$JMP = 000050	INIT 015376
BIT08 = 000400 G	C\$RCVB 003336 G	DEV1 003570	F\$MOD = 000000	INITWD 013754
BIT09 = 001000 G	C\$AU = 000052	DEV2 004000	F\$MSG = 000011	INIT2 015420
BIT1 = 000002 G	C\$AUTO= 000061	DEV3 004210	F\$PROT= 000021	ISR = 000100 G
BIT10 = 002000 G	C\$BRK = 000022	DEV4 004420	F\$PWR = 000017	IXE = 004000 G
BIT11 = 004000 G	C\$BSEG= 000004	DEV5 004530	F\$RPT = 000012	I\$AU = 000041
BIT12 = 010000 G	C\$BSUB= 000002	DEV6 005040	F\$SEG = 000003	I\$AUTO= 000041
BIT13 = 020000 G	C\$CEFG= 000045	DEV7 005250	F\$SOFT= 000005	I\$CLN = 000041
BIT14 = 040000 G	C\$CLCK= 000062	DFPTBL 002172 G	F\$SRV = 000010	I\$DU = 000041
BIT15 = 100000 G	C\$CLEA= 000012	DFTL1 012206	F\$SUB = 000002	I\$HRD = 000041
BIT2 = 000004 G	C\$CLOS= 000035	DIAGMC= 000000	F\$SW = 000014	I\$INIT= 000041
BIT3 = 000010 G	C\$CLP1= 000006	DLV = 000074 G	F\$TEST= 000001	I\$MOD = 000041
BIT4 = 000020 G	C\$CVEC= 000036	DO\$RK 013222 G	GBTMP 010106	I\$MSG = 000041
BIT5 = 000040 G	C\$DCLN= 000044	DONE 003314 G	GBTMP2 010110	I\$PROT= 000040
BIT6 = 000100 G	C\$DODU= 000051	DR = 000060 G	GETANS 006736 G	I\$PTAB= 000041
BIT7 = 000200 G	C\$DRPT= 000024	DRVCHK 002214	GETHRD 015524	I\$PWR = 000041
BIT8 = 000400 G	C\$DU = 000053	EF.CON= 000036 G	GETPTR 007002	I\$RPT = 000041
BIT9 = 001000 G	C\$EDIT= 000003	EF.NEW= 000035 G	CETR5 016424	I\$SEG = 000041
BLKEND= 000202 G	C\$ERDF= 000055	EF.PWR= 000034 G	GTAGIN 007224	I\$SETU= 000041
BLKER 003324 G	C\$ERHR= 000056	EF.RES= 000037 G	GTBYTE 007662 G	I\$SFT = 000041
BLKS12= 000210 G	C\$ERRO= 000060	EF.STA= 000040 G	GTDOWN 007534	I\$SRV = 000041
BLKTBL 003340 G	C\$ERSF= 000054	ERRDES 012402 G	GTOK 007452	I\$SUA = 000041
BOE = 000400 G	C\$ERSO= 000057	ESABO = 177720 G	GTPKS1 007674 G	I\$TSI = 000041
BRKPTR 013760	C\$ESCA= 000010	ESCKS = 177757 G	GTPKS8 007274 G	J\$JMP = 000167
BRKTO 013756	C\$ESEG= 000005	ESCKSM= 177757	GTPTR 007661	LENGTH 002204
BRKWD 013752	C\$ESUB= 000003	ESCMD = 177720 G	GTUM 007414	LGOFST= 000120 G
BUFTBL 024346	C\$ETST= 000001	ESNCRT= 177767 G	G\$CNT0= 000700	LNCNT 014342
BUF0 025426	C\$EXIT= 000032	ESNOMO= 177737 G	G\$DELM= 000372	LCE = 040000 G
BUF1 026464	C\$GETB= 000026	ESNONX= 177770 G	G\$ISP= 000003	LOG 012046 G
BUF2 027522	C\$GETW= 000027	ESOK = 000000 G	G\$EXCP= 000400	LOGO 012370
BUF3 030560	C\$GMAN= 000043	ESPART= 177776 G	G\$HILI= 000002	LOGOK 012134
BUF4 031616	C\$GPHR= 000042	ESRD = 177757	G\$LOLI= 000001	LOGOK2 012220
BUF5 032654	C\$GPLO= 000030	ESREC = 177711 G	G\$NO = 000000	LOGO 012122
BUF6 033712	C\$GPR1= 000040	ESSK = 177740 G	G\$OFFS= 000400	LOG1 012236
BUF7 034750	C\$INIT= 000011	ESSLF = 177777 G	G\$UFSI= 000376	LOG2 012266
CARLF 014356	C\$INLP= 000020	ESTRY = 000001 G	G\$PRMA= 000001	LOG3 012326
CHECK 015502	C\$MANI= 000050	ESWLOC= 177765 G	G\$PRMD= 000002	LOG3B 012350
CHKANR 010176	C\$MEM = 000031	ESWR = 177757	G\$PRML= 000000	LOT = 000010 G

PARAMETER CODING
SYMBOL TABLE

MACRO M1110 02-AUG-79 15:53 PAGE 136-2

B 8

SEQ 0092

LSTDEV 003356 G	L10004 013660	NCART = 000054 G	RECERR= 000042 G	STRT 016150 G
LSACP 002110 G	L10005 013714	NODRVS 016116	RECID 012654 G	SUCCS = 000076 G
LSAPT 002036 G	L10006 015004	NOMCR 006340	RECID2 013030	SUCOTL= 000046 G
LSAU 016504 G	L10007 016064	NOMOT = 000030 G	RECOV 011262	SVCGBL= 000000
LSAUT 002070 G	L10010 016262	NOUNIT= 000036 G	RETER 011464	SVCINS= 177777
LSAUTO 016156 G	L10011 016356	NOXOFF 006430	RETRY = 000002 G	SVCSUB= 177777
LSCCP 002106 G	L10012 016422	NTSFT 012250	RLUN 015006	SVCTAG= 177777
LSCLEA 016340 G	L10013 016504	NXTRET 006334	RPTR 015010	SVCTST= 177777
LSCO 002032 G	L10014 016700	NXTST 006052 G	RSCMND= 000002 G	SWAPDR 005500 G
LSDEPO 002011 G	L10015 017144	NXTST2 001156	RSCONT= 000020 G	SWPTR 005600
LSDESC 002122 G	L10016 020534	ONEFIL= 000001	RSDASZ= 000204 G	SYSTAT 003300 G
LSDESP 002176 G	L10017 021510	OTL = 000052 G	RSDATA= 000001 G	SSLSYM= 010000
LSDEVP 002060 G	L10020 022274	OV FLO 012746	RSDNSZ= 000222 G	TAPLEN 003302 G
LSDISP 002152 G	L10021 023250	QVRN = 000012 G	RSEND = 000002 G	THRSHI 011370
LSOLY 002116 G	L10022 024034	OSAPTS= 000000	RSINIT= 000004 G	THRSLO 011342
LSOTP 002040 G	L10023 036052	OSAU = 000001	RSMISZ= 000012 G	IMP = 000066 G
LSOTYP 002034 G	L10024 036242	OSBGNR= 000001	RSNDSZ= 000016 G	TOMANY 016066
LSDU 016360 G	L10025 036564	OSBGNS= 000001	RSNTAB 002220	TORCVB= 000050 G
LSDUT 002072 G	L10027 036574	OSDU = 000001	RSEND= 000100 G	TOSNDB= 000056 G
LSDVTY 005460 G	MABEE 012300	OSERRT= 000000	RSSNIT= 000001 G	TRBUF 024370
LSF 002052 G	MSAUTO 016320	OSGNSW= 000001	RSSNOP= 000000 G	TRK = 000062 G
LSENV1 002044 G	MSBDA 002332 G	OSPOIN= 000001	RSSNSZ= 000016 G	TRPHND 016266
LSFETP 002102 G	MSCMD 002676 G	OSSETU= 000001	RSSRD = 000002 G	TRPPTR 016254
LSEXP1 002046 G	MSCOM 002376 G	PARTL = 000034 G	RSSSEK= 000005 G	TSTPC = 000020 G
LSEXP4 002064 G	MSG1 036052	PATTEN= 000072 G	RSSLF= 000007 G	TSTTOP 003320
LSEXP5 002066 G	MSG1B 036063	PDTFLG 016152 G	RSSWR = 000003 G	TST1 016552
LSHARD 036010 G	MSG1C 036100	PERDEV 006170	RSVP 006364 G	TST2 016746
LSHIME 002120 G	MSG2 036131	PKPTR = 000104 G	RXXOFF= 000023 G	TST3 017212
LSHPCF 002016 G	MSG3 036146	PNT = 001000 G	RSXON = 000020 G	TST3PT 020522
LSHPTP 002022 G	MSG4 036242	PRBUF 002210	RTRYN 011422	TST4 020602
LSHW 002172 G	MSG4B 036307	PRDAT 014344	RUN 006022 G	TST4EX 021502
LSICP 002104 G	MSG5 036351	PRFORM 014346	SECREC 003326 G	TST5 021556
LSINIT 015376 G	MSG6 036403	PRI = 002000 G	SERVST 007656	TST5EX 022266
LSLADP 002026 G	MSG7 036430	PRI00 = 000000 G	SETDR 005602 G	TST6 022342
LSLAST 036560 G	MSG8 036456	PRI01 = 000040 G	SETLEN 016026	TST6EX 023242
LSLOAD 002100 G	MSHCHK 002550 G	PRI02 = 000100 G	SETPTR 005656	TST7 023316
LSLUN 002074 G	MSHDRD 003146 G	PRI03 = 000140 G	SETSRV 007572	TST7EX 024026
LSMREV 002050 G	MSHWR 003210 G	PRI04 = 000200 G	SETUP 005754 G	TUVECT= 000204 G
LSNAME 002000 G	MSNIT 002612 G	PRI05 = 000240 G	SFPTBL 002204 G	T\$ARGC= 000002
LSPRIO 002042 G	MSNLOG 002314 G	PRI06 = 000300 G	SFT 012316	T\$CODE= 005052
LSPROT 002142 G	MSNOMO 002440 G	PRI07 = 000340 G	SFTOUT 036242	T\$ERRN= 000146
LSPRT 002112 G	MSNQTP 002456 G	PRNPAK 014176 G	SFTRD = 000002 G	T\$EXCP= 000000
LSREPP 002062 G	MSNRSP 002756 G	PRNSIZ 003330 G	SFTWR = 000004 G	T\$FLAG= 000040
LSREV 002010 G	MSOVRN 003252 G	PTR 016454	SKERR = 000024 G	T\$FREE= 036574
LSRPT 014362 G	MSPART 002626 G	RCBINT 003310	SLFER = 000044 G	T\$GMAN= 000000
LSOFT 036166 G	MSQRSP 002772 G	RCBSZ= 001036 G	SND 006434	T\$HILI= 000376
LSPC 002056 G	MSREC 002712 G	RCDB = 000024 G	SNDBYT 006666 G	T\$LAST= 000001
LSPCP 002020 G	MSRNT 002530 G	RCFLG 003306 G	SNDENT= 000070 G	T\$LOLI= 000001
LSPTP 002024 G	MSELF 002356 G	RCNIT= 000006 G	SNDHND 013646	T\$LSYM= 010000
LSSTA 002030 G	MSSFRD 003046 G	RCSR = 000022 G	SNDINT 013646 G	T\$LTNG= 000007
LSW 002204 G	MSSFWR 003106 G	RCVBU= 000102 G	SOFT = 000122 G	T\$NEST= 177777
LSTEST 002114 G	MSSKER 002300 G	RCVHND 013662	SOFTW = 000124 G	T\$NSO = 000000
LSIML 002014 G	MSTOSN 003024 G	RCVINT 013662 G	SRVTBL 007636	T\$NSI = 000005
LSUNIT 002012 G	MSUNIT 002650 G	RDNO = 000114 G	STAEOP 002206	T\$PCNT= 000000
L10001 002202	MSWPRO 002506 G	FDNI = 000116 G	STATHD 015012	T\$PTAB= 010026
L10002 002220	MSWRSP 002732 G	REC = 000064 G	STATUS= 000000 G	T\$PTHV= 000001
L10003 012572	MXRTRY 003322 G	RECDAT 017130	STHD2 015266	T\$PTNU= 000001

PARAMETER CODING
SYMBOL TABLE

MACRO M1110 02-AUG-79 15:53 PAGE 136-3

C 8

SEQ 0093

TSSAVL= 177777	TSSCLE= 010011	TSSSRV= 010005	T7 = 023252 G	XFNSND 006416
TSSSEGL= 177777	TSSDAT= 010027	TSSSW = 010002	UAM = 000200 G	XMDB = 000030 G
TSSSIZE= 000006	TSSDU = 010012	TSSTES= 010022	UNIT = 012574 G	XMSR = 000026 G
TSSSUBN= 000000	TSSHAR= 010023	T1 = 016506 G	UNREC = 011442	XSCNT = 000036 G
TSTAGL= 177777	TSSHW = 010001	T1TRY = 000146 G	UNSUC = 011040	XSEIG = 000034 G
TSTAGN= 010030	TSSINI= 010007	T2 = 016702 G	UNXPCT = 007356	XSPKMM= 000032 G
TSTEMP= 000000	TSSMSG= 010003	T3 = 017146 G	WAIT = 013716	XSPTR = 000106 G
TSTEST= 000007	TSSPC = 000001	T4 = 020536 G	WHCHDR = 013052 G	X\$ALWA= 000000
TSTSTM= 177777	TSSPRO= 010000	T4TRY = 000132 G	WRLOCK= 000026 G	X\$FALS= 000040
TSTSTS= 000001	TSSPTA= 010026	T5 = 021512 G	WRINO = 000110 G	X\$OFFS= 000400
TSSAU = 010013	TSSRPT= 010006	T6 = 022276 G	WRINI = 000112 G	X\$TRLE= 000020
TSSAUT= 010010	TSSSOF= 010024			

. ABS. 036574 000
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 33125 WORDS (130 PAGES)

DYNAMIC MEMORY: 20308 WORDS (78 PAGES)

ELAPSED TIME: 00:06:55

CZTUUB.BIN/EN:AMA:ABS,CZTUUB/CR/-SP LB1-[1,1]SVC/MLB,SV:[203,230]CZTUUB.MA(

CZTUUB CREATED BY MACRC ON 2-AUG-79 AT 15:5'

PAGE 1
CREF V01

SEQ 0094

SYMBOL		VALUE	REFERENCES
ABNDX	=	000004 G	#26-819 *84-2428 84-2441 84-2443
ABO		012360	84-2438 84-2452 84-2465 #84-2481
ABOMSG		016456	109-3180 #109-3210
ABONM		006336	*60-1626 *60-1643 60-1648 #60-1656
ADR	=	000020 G	#14-567
ALLGON		003332 G	#24-787 *60-1652 *103-3045 107-3137
ASSEMB	=	000010	5-375 5-375
BDATA	=	000134 G	#28-872 101-2922 101-2932
BDBYTS		014140	*98-2770 *98-2781 98-2784 98-2789 #98-2799
BDCBK	=	000022 G	#16-595 78-2146 78-2198
BDCOM	=	000014 G	#16-592 80-2242 84-2453
BIT0	=	00 001 G	#14-567 48-1467 72-1987 84-2444 90-2555 90-2556 90-2558 90-2569 94-2644
BIT00	=	000001 G	#14-567 14-567
BIT01		000002 G	#14-567 14-567
BIT02	=	000004 G	#14-567 14-567
BIT03	=	000010 G	#14-567 14-567
BIT04	=	000020 G	#14-567 14-567
BIT05	=	000040 G	#14-567 14-567
BIT06	=	000100 G	#14-567 14-567
BIT07	=	000200 G	#14-567 14-567
BIT08	=	000400 G	#14-567 14-567
BIT09	=	001000 G	#14-567 14-567
BIT1	=	000002 G	#14-567 60-1617 66-1793 72-1988 76-2087 80-2230 80-2299
BIT10	=	002000 G	#14-567 60-1613 62-1723 80-2248 80-2268 80-2276 80-2297 118-3363 118-3363
			118-3364 120-3406 120-3406 122-3437 124-3468 124-3468 126-3499
BIT11	=	004000 G	#14-567 101-2408 103-2983 103-3004 105-3104
BIT12	=	010000 G	#14-567 118-3363 118-3363 118-3363 120-3406 120-3406 120-3406 124-3468 124-3468
			124-3468
BIT13	=	020000 G	#14-567 103-2982 103-3029 109-3177
BIT14	=	040000 G	#14-567 60-1632 60-1635 78-2159 78-2171 94-2694 103-3010
BIT15	=	100000 G	#14-567 48-1465 68-1835 68-1857 70-1893 70-1910 70-1930 70-1941 76-2095
			80-2234 103-2982 103-3029 105-3104 109-3177
BIT2	=	000004 G	#14-567 72-1989
BIT3	=	000010 G	#14-567 72-1990
BIT4	=	000020 G	#14-567 62-1717 62-1721 72-1991 84-2500 94-2646
BIT5	=	000040 G	#14-567 62-1739 62-1741 72-1992
BIT6	=	000100 G	#14-567 72-1993 80-2239 80-2300 94-2671 94-2679 94-2686 94-2702 94-2703
			96-2715 96-2723 98-2790
BIT7	=	000200 G	#14-567 72-1994 80-2268 80-2276 80-2282 80-2284 80-2286 80-2297
BIT8	=	000400 G	#14-567 50-1500 62-1727 62-1736 80-2261 80-2290 105-3011
BIT9	=	001000 G	#14-567 48-1469 62-1711 80-2245 80-2272 82-2355 105-3013
BLKEND	=	000202 G	#28-892 103-3021
BLKER		003324 G	#24-784 *62-1708 *103-3053
BLKSIZ	=	000210 G	#28-896 30-918 30-919 30-920 30-921 30-922 30-923 30-924 30-925
BLKTBL		003340 G	#30-906 48-1463 50-1497 52-1522 56-1565 60-1611 60-1627 70-1891 76-2195
			101-2902 103-2978 103-2998 105-3102 109-3201
BOE	=	000400 G	#14-567
BRKPTR		013760	*94-2670 *94-2678 *94-2685 96-1736 #96-2750
BRKTO		013756	*94-2643 *94-2652 *94-2658 *94-2668 *94-2676 *94-2740 #96-2749
BRKWD		013752	94-2657 #96-2746
BUFTBL		024346	105-2985 *130-3526
BUFO		025426	130-3526 #130-3546

CZTUUB CREATED BY MACRO ON 2-AUG-79 AT 15:57

PAGE 2
CREF V01

SEQ 0095

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
BUF1		026464	130-3527 #130-3547
BUF2		027522	130-3528 #130-3548
BUF3		030560	130-3529 #130-3549
BUF4		031616	130-3530 #130-3550
BUF5		032654	130-3531 #130-3551
BUF6		033712	130-3532 #130-3552
BUF7		034750	130-3533 #130-3553
CARLF		014356	100-2828 100-2830 #100-2839
CHECK		015502	103-2988 #103-2993
CHKANR		010176	76-2091 76-2099 #76-2103
CHKANS	G	010112	58-1590 #76-2085
CHKEND	G	010536	78-2151 78-2194 #80-2228
CHKERR		010624	80-2238 #30-2245
CHKPKS	G	010202	76-2089 76-2097 #78-2126
CHKPTR		010200	*76-2093 76-2094 76-2098 *76-2100 #76-2105
CHKREE		010676	80-2231 #80-2252
CHKRET		011242	80-2236 80-2241 80-2244 80-2247 80-2251 80-2269 80-2277 80-2285 80-2287
CHKSUC	G	011536	#80-2299 80-2232 80-2252 #82-2334
CHKSUM	G	013066	#90-2553 92-2596 114-3314 116-3332 118-3363 118-3363 118-3363 118-3363 118-3364 118-3364
			120-3406 120-3406 120-3406 122-3437 122-3437 124-3468 124-3468 124-3468 126-3499
CHK8		010132	76-2088 #76-2093
CHKSM	G	013162	78-2144 78-2179 78-2192 #92-2593
CLRAIL	G	005660	#52-1522 66-1796
CLRBUF	G	005720	52-1534 #54-1540 66-1799
CLRPTR		005752	*52-1522 52-1523 52-1525 *52-1527 #54-1550
CMDSNT	= 000100	G	#26-844 *62-1720 82-2341 82-2345 82-2358 86-2506 *94-2645
CMNDER	= 000040	G	#16-602 82-2378
CMDDAT	002212		#13-494 98-2772
CNINIT	= 000032	G	#16-599 94-2698
COMPAR	013762	G	78-2156 78-2181 #98-2767
CSNRDY	003334	G	#24-794 64-1764
CSRCVB	003336	G	#24-795 74-2024
CSAU	= 000052		#5-375 111-3245
CSAUTO	= 000061		#5-375 105-3113
CS3RK	= 000022		#5-375 58-1588 60-1651 64-1768 74-2046 94-2648 96-2738 96-2739 101-2901
			101-2904 101-2906
CSBSEG	= 000004		#5-375
CSBSUB	= 000002		#5-375
CSCEFG	= 000045		#5-375
CSCLCK	= 000062		#5-375
CSCLEA	= 000012		#5-375 107-3162
CSCLDS	= 000035		#5-375
CSCLPI	= 000006		#5-375
CSCEVC	= 000006		#5-375 94-2704 94-2706 105-3112
CSDECLN	= 000044		#5-375 60-1653 103-2996 103-3016
CSDDDU	= 000051		#5-375 84-2483 105-3124
CSDRPT	= 000024		#5-375 107-3141
CSDU	= 000053		#5-375 109-3200
CSEDIT	= 000003		#5-375 5-418
CSERDF	= 000055		#5-375 84-2437 84-2451

CZTUUB CREATED BY MACRO UN 2-AUG-79 AT 15:57 PAGE 3 F 8
 SYMBOL CROSS REFERENCE CREF V01
 SYMBOL VALUE REFERENCES

CSFRMR	= 000056	#5-375 84-2464 84-2479
CSERRO	= 000060	#5-375
CSERSF	= 000054	#5-375 60-1650 103-2995 103-3015
CSERSO	= 000057	#5-375 84-2457 84-2475 98-2788
CSESCA	= 000010	#5-375
CSSEEG	= 000005	#5-375
CSesub	= 000003	#5-375
CSSTST	= 000001	#5-375 114-3319 116-3346 118-3388 120-3420 122-3451 124-3482 126-3513
CSEXIT	= 000032	#5-375 114-3312 116-3326 118-3356 120-3396 122-3427 124-3458 126-3489
CSGETB	= 000026	#5-375
CSGETW	= 000027	#5-375
CSGMAN	= 000043	#5-375
CSGPHR	= 000042	#5-375 103-3002
CSGPLO	= 000030	#5-375
CSGPRI	= 000040	#5-375
CSINIT	= 000011	#5-375 103-3082
CSINLP	= 000020	#5-375
CSMANI	= 000050	#5-375
CSMEM	= 000031	#5-375
CSMSG	= 000023	#5-375 86-2513
CSOPEN	= 000034	#5-375
CSNTB	= 000014	#5-375 80-2274 86-2503 86-2506 86-2509 98-2789
CSNTF	= 000017	#5-375 100-2823 100-2828 100-2830 105-3121 109-3180
CSPTS	= 000016	#5-375 101-2903 101-2905 101-2921 101-2922 101-2932
CSPTX	= 000015	#5-375 80-2250 80-2255 80-2258 80-2260 80-2281 90-2289
CSQIO	= 000377	#5-375
CSRDPJ	= 000007	#5-375
CSREFG	= 000047	#5-375 103-2975
CSRESE	= 000033	#5-375 #5-375
CSREVI	= 000003	#5-375 5-418
CSRFLA	= 000021	#5-375 103-3052
CSRPT	= 000025	#5-375 101-2945
CSSEFG	= 000046	#5-375
CSSPRI	= 000041	#5-375 94-2663
CSVEEC	= 000037	#5-375 94-2664 94-2666 105-3101
CSIPRI	= 000013	#5-375
DESC	014142	98-2789 #98-2800
DEVPTX	003304 G	#24-776 #60-1611 60-1612 60-1621 #60-1623 #60-1627 60-1628 60-1644 #60-1646
		#103-2978 103-2980 103-2987 #103-2989 #103-2998 103-3000 #103-3046
DEV0	003360	30-906 #30-918
DEV1	003570	30-907 #30-919
DEV2	004000	30-908 #30-920
DEV3	004210	30-909 #30-921
DEV4	004420	30-910 #30-922
DEV5	004630	30-911 #30-923
DEV6	005040	30-912 #30-924
DEV7	005250	30-913 #30-925
DFPTBL	02172 G	#11-466
DFTL1	012206	#84-2449 #84-2450 #84-2451 84-2451
DIAGMC	= 000000	5-375
DLV	= 000074 G	#26-842 68-1855 70-1932 70-1943 #74-2027 74-2028 74-2029 #74-2031 #74-2032
		74-2053 74-2054 #74-2056 78-2161 86-2507 86-2509 #86-2510 #96-2724 96-2725

SEQ 0096

PAGE 4

REF V01

SEO 0097

SYMBOL	CROSS REFERENCE	SYMBOL	VALUE	REFERENCES	REF	V01
DOBRK	013222	G		96-2726	*96-2728	*103-3044
DONE	003314	G		60-1634	#94-2642	
				#24-780	*56-1564	58-1584
DR	= 000060	G		*126-3510		*114-3315
				#26-835	48-1467	*116-3337
				*103-3007	48-1469	*118-3379
				120-3406	*48-1471	*120-3417
				126-3499	*50-1499	*122-3448
					50-1500	*124-3479
					*50-1502	86-2503
					118-3363	88-2534
					118-3364	120-3405
					124-3467	126-3498
					124-3468	
DRVCHK	002214			#13-495	120-3403	122-3434
EF.CON	= 000036	G		#14-567		124-3465
EF.NEW	= 000035	G		#14-567		126-3496
EF.PWR	= 000034	G		#14-567		
EF.RES	= 000037	G		#14-567		
EF.STA	= 000040	G		#14-567	103-2975	
ERRDES	012402	G		84-2437	84-2451	84-2457
ESABO	= 177720	G		#18-661		84-2464
ESCKS	= 177757	G		#18-672	18-674	84-2475
ESCKSM	= 177757			#18-674		84-2479
ESCMD	= 177720	G		#18-670		#86-2496
ESNCRT	= 177767	G		#18-662		98-2788
ESNOMO	= 177737	G		#18-669		
ESNONX	= 177770	G		#18-663		
ESOK	= 000000	G		#18-664		
ESPART	= 177776	G		#18-665		
ESRD	= 177757			#18-671		
ESREC	= 177711	G		#18-666		
ESSK	= 177740	G		#18-673		
ESSLF	= 177777	G		#18-667		
ESTRY	= 000001	G		#18-668		
ESWLOC	= 177765	G		#18-675		
ESWR	= 177757			#14-567		
EVL	= 000004	G		#13-496		
EVLTHR	002216			#68-1870		
EXOFF	007213			68-1826	#68-1869	
EXON	007212			#5-375		
ESEND	= 002100			#5-375		
ESLOAD	= 000035			5-418		
FLGLOC	016154	G		84-2445	*103-3052	
FM	015142			101-2922	#103-3091	
FMO	015124			101-2921	#101-2955	
FTLNM	003312			#24-779		
FSAU	= 000015			#5-375		
FSAUTO	= 000020			#5-375		
FBBGN	= 000040			#5-375		
				101-2886		
				114-3310		
				120-3394		
				126-3487		
				136-3717		
ISCLEA	= 000007			#5-375		
FSDU	= 000016			#5-375		
FSEND	= 000041			#5-375		

CZTUUB		CREATED BY		MACRO ON 2-AUG-79 AT 15:57		PAGE 5		H 8				SEQ 0098	
SYMBOL		CROSS REFERENCE		REFERENCES		PAGE 5		H 8					
SYMBOL		VALUE		REFERENCES		CREF		V01					
				5-375	5-375	5-375	5-375	5-375	5-375	5-375	5-375	5-401	
				13-507	14-560	86-2513	96-2717	96-2730	100-2842	101-2886	101-2945	101-2961	
				103-3082	105-3113	107-3162	104-3200	111-245	114-3307	114-3310	114-3310	114-3310	
				114-3312	114-3319	114-3319	116-3324	116-3324	116-3324	116-3326	116-3346	116-3346	
				118-3354	118-3354	118-3354	118-3356	118-3388	118-3388	120-3394	120-3394	120-3394	
				120-3396	120-3420	120-3420	122-3425	122-3425	122-3425	122-3427	122-3451	122-3451	
				124-3456	124-3456	124-3456	124-3458	124-3482	124-3482	126-3487	126-3487	126-3487	
				126-3489	126-3513	126-3513	130-3557	132-3622	132-3648	134-3693	136-3714	136-3716	
				136-3717	136-3722	136-3723							
FSHARD	=	000004		#5-375	132-3633	132-3648							
FSHW	=	000013		#5-375	11-466	11-479							
FSINIT	=	000006		#5-375	103-2971	103-3082							
FSJMP	=	000050		#5-375	114-3312	116-3326	118-3356	120-3396	122-3427	124-3458	126-3489		
FSMOD	=	000000		#5-375	5-401	13-507	14-560	100-2842	101-2886	101-2961	114-3307	130-3557	
				132-3622	136-3714								
FSMSG	=	000011		#5-375	86-2496	86-2513							
FSPROT	=	000021		#5-375	7-428	7-432							
FSRWR	=	000017		#5-375									
FRPT	=	000012		#5-375	101-2893	101-2945							
FSFG	=	000003		#5-375									
FSJFT	=	000005		#5-375	134-3677	134-3693							
FSRV	=	000010		#5-375	96-2713	96-2717	96-2721	96-2730					
FSUB	=	000002		#5-375									
FSW	=	000014		#5-375	13-489	1-505							
FSTEST	=	000001		#5-375	114-3310	114-3319	116-3324	116-3346	118-3354	118-3388	120-3394	120-3420	
				122-3425	122-3451	124-3456	124-3482	126-3487	126-3513				
GBTMP	010106			*74-2023	*74-2033	*74-2043	74-2044	*74-2063	*74-2069				
GBTMP2	010110			*74-2038	74-2049	74-2051	*74-2058	74-2061	*74-2070				
GETANS	006736	G		58-1586	#66-1792								
GETHRD	01574			103-2994	#103-2998								
GETPTR	007002			#66-1804									
GETRS	016424			109-3176	#109-3201								
GTAGIN	007224			#70-1892	70-1956								
G1BYTE	007662	G		68-1834	68-1854	70-1909	70-1929	70-1940	*74-2023				
GTDOWN	007534			70-1896	70-1900	70-1913	70-1924	70-1928	70-1931	70-1935	70-1942	70-1946	
				70-1948	#70-1952								
G1OK	007452			70-1917	#70-1937								
G1PKS1	007004	G		66-1801	#68-1821								
G1PKS8	007214	G		66-1797	#70-1890	70-1959							
G1PTR	007660			*70-1891	70-1892	70-1953	*70-1955	#72-1997					
GTUM	007414			70-1922	#70-1926								
GSCNT0	=	000200		#5-375									
GSDLM	=	000372		#5-375									
GSDISP	=	000003		#5-375									
GSEKCP	=	000400		#5-375									
GSHLI	=	000002		#5-375									
GSLOLI	=	000001		#5-375									
G8NO	=	000000		#5-375									
G8OFFS	=	000400		#5-375	132-3636	132-3637	132-3638	132-3639	132-3640	134-3679	134-3680	134-3681	
				134-3682	134-3683	134-3685							
G8OFFS1	=	000376		#5-375	132-3636	132-3637	132-3638	132-3639	132-3640	134-3679	134-3680	134-3681	
				134-3682	134-3683	134-3685							

CZTUUB		CREATED BY		MACRO ON 2-AUG-79 AT 15:57		PAGE 6		CREF		V01		SEQ 0099	
SYMBOL		CROSS REFERENCE		VALUE		REFERENCES							
GSPRMA	=	000001		#5-375	132-3636	132-3637							
GSPRMD	=	000002		#5-375	134-3679	134-3685							
GSPRML	=	000000		#5-375	132-3638	132-3639	132-3640	134-3680	134-3681	134-3682	134-3683		
GSRADA	=	000140		#5-375									
GSRADB	=	000000		#5-375									
GSRADD	=	000040		#5-375	134-3679	134-3685							
GSRADL	=	000120		#5-375	132-3638	132-3639	132-3640	134-3680	134-3681	134-3682	134-3683		
GSRADO	=	000020		#5-375	132-3636	132-3637							
G%XFER	=	000004		#5-375									
G%YES	=	000010		#5-375	132-3636	132-3637	132-3638	132-3639	132-3640	134-3679	134-3680	134-3681	
				134-3682	134-3683	134-3685							
HARDR	=	000136	G	#28-873	101-2917	101-2927							
HARDW	=	000140	G	#28-874	101-2919	101-2929							
HELP	=	000000		#5-360	5-370	5-392	5-410	9-435	9-450	11-473	13-498	#14-512	
				14-550	14-569	30-926	30-932	32-948	32-953	32-961	32-968	32-973	
				32-979	48-1395	48-1407	48-1412	48-1418	48-1423	48-1429	48-1437	48-1444	
				48-1450	48-1456	#101-2848	103-3060	103-3070	107-3143	107-3150	109-3182	109-3188	
				111-3225	111-3231	#112-3251	112-3292	112-3298	130-3558	130-3563	130-3573	#132-3584	
				132-3642	132-3658	134-3686	136-3707						
MOE	=	100000	G	#14-567									
MRD	=	012340		84-2469	#84-2477								
MPDRD	=	000016	G	#16-593	80-2292								
MRDWR	=	000020	G	#16-594	80-2294								
MRD1	=	011164		80-2279	#80-2288								
IBE	=	010000	G	#14-567									
IDPTR	=	003316	G	#24-781	*56-156	56-1566	56-1568	*56-1570					
IDU	=	000040	G	#14-567									
IER	=	020000	G	#14-567									
INIT	=	015376		#103-2973									
INITWD	=	013754		*94-2642	94-2669	94-2677	94-2684	94-2691	94-2697	#96-2747			
INIT2	=	015420		103-2976	#103-2978								
ISR	=	000100	G	#14-567									
IXE	=	004000	G	#14-567									
ISAU	=	000041		#5-375	#111-3222	#111-3245							
ISAUTO	=	000041		#5-375	#105-3099	#105-3113							
ISCLN	=	000041		#5-375	#107-3136	#107-3162							
ISDU	=	000041		#5-375	#107-3172	#109-3200							
ISHRD	=	000041		#132-3633	#132-3648								
ISINIT	=	000041		#5-375	#103-2971	#103-3082							
ISMOD	=	000041		#5-375	5-401	#5-401	13-507	#13-507	14-560	#14-560	100-2842	#100-2842	
				101-2886	#101-2886	101-2961	#101-2961	14-3307	#114-3307	130-3557	#130-3557	132-3622	
				#132-3622	136-3714	#136-3714							
ISMSG	=	000041		#5-375	#86-2496	#86-2513							
ISPROT	=	000040		#5-375	#7-428								
ISPTAB	=	000041		#5-375	136-3717	#136-3717	136-3722	#136-3722					
ISPR	=	000041		#5-375									
ISRPT	=	000041		#5-375	#101-2893	#101-2945							
ISSEG	=	000041		#5-375	114-3310	116-3324	118-3354	120-3394	122-3425	124-3456	126-3487		
ISSETU	=	000041		#5-375	136-3716	#136-3716	136-3717	136-3723	#136-3723				
ISFT	=	000041		#134-3677	#134-3693								
ISSRV	=	000041		#5-375	#96-2713	#96-2717	#96-2721	#96-2730					
ISSUB	=	000041		#5-375	114-3310	116-3324	118-3354	120-3394	122-3425	124-3456	126-3487		

CZTUUB CREATED BY MACRO ON 2-AUG-79 AT 15:57

PAGE 7
CREF V01

SEB 0100

SYMBOL	VALUE	REFERENCE	REFERENCES	REF	V01					
ISTST	= 000041		#5-375 114-3310 #114-3310 114-3312 114-3319 #114-3319 #114-3319 116-3324 #116-3324							
			116-3326 116-3346 #116-3346 #116-3346 118-3354 #118-3354 118-3356 118-3388 #118-3388							
			#118-3388 120-3394 #120-3394 120-3396 120-3420 #120-3420 #120-3420 122-3425 #122-3425							
			122-3427 122-3451 #122-3451 #122-3451 124-3456 #124-3456 124-3458 124-3482 #124-3482							
			#124-3482 126-3487 #126-3487 126-3489 126-3513 #126-3513 #126-3513							
J\$JMP	= 000161		#5-375							
LENGTH	002204		#13-491 103-3054							
LGOFST	= 000120	G	#28-863 84-2429							
LNCNT	014342		*100-2821 *100-2826 #100-2835							
LOE	= 040000	G	#14-567							
LOG	012046	G	64-1774 74-2067 78-2147 78-2164 78-2173 78-2199 80-2243 80-2266 80-2295							
			82-2402 #84-2421 94-2655 94-2699 96-2743							
LOGO	012370		84-2458 84-2476 #84-2484							
LOGOK	012134		84-2436 #84-2439							
LOGOK2	012220		84-2446 84-2448 #84-2453							
LOGO	012122		#84-2437							
LOG1	012236		*84-2455 *84-2456 #84-2457 84-2457							
LOG2	012266		*84-2462 *84-2463 #84-2464 84-2464							
LOG3	012326		*84-2473 *84-2474 #84-2475 84-2475							
LOG3B	012350		*84-2477 *84-2478 #84-2479 84-2479							
LOT	= 000010	G	#14-567							
LSTDEV	003356	G	#30-913 48-1473 50-1503 52-1525 56-1568 60-1621 60-1644 70-1953 76-2098							
			101-2933 103-2987 105-3108							
LSACP	002110	G	#5-418							
LSAPT	002036	G	#5-418							
LSAU	016504	G	5-418 #111-3222							
LSAUT	002070	G	#5-418							
LSAUTO	016156	G	5-418 #105-3099							
LSCIP	002106	G	#5-418							
LSCLEA	016340	G	5-418 #107-3136							
LSCO	002032	G	#5-418							
LSDEPO	002011	G	#5-418							
LSDESC	002122	G	5-418 #5-420							
LSDESP	012076	G	#5-418							
LSDEVP	012060	G	#5-418							
LSDISP	002152	G	5-418 #9-448							
LSDLV	002116	G	#5-418							
LSDTP	002040	G	#5-418							
LSDTYP	002034	G	#5-418							
LSDU	016360	G	5-418 #105-3172							
LSDUT	002072	G	#5-418							
LSDVTV	005460	G	5-418 #32-946							

CZTUUB
 SYMBOL CROSS REFERENCE
 SYMBOL VALUE REFERENCES
 LSHW 002172 G 5-418 11-466 #11-466
 L\$ICP 002104 G #5-418
 L\$INIT 015376 G 5-418 #103-2971
 L\$ADP 002026 G #5-418
 L\$LAST 036560 G 5-418 #136-3713 136-3723
 L\$LOAD 002100 G #5-418
 L\$LUN 002074 G #5-418 *84-2426 *84-2427 *98-2786 *98-2787 *103-3001
 L\$MREV 002050 G #5-418
 L\$NAME 002000 G #5-418
 L\$PRIO 002042 G #5-418
 L\$PROT 002142 G 5-418 #7-42E
 L\$PRT 002112 G #5-418
 L\$REPP 002062 G #5-418
 L\$REV 002010 G #5-418
 L\$RPT 014362 G 5-418 #101-2893
 L\$SOFT 036166 G 5-418 124-3677 #134-3677
 L\$SPC 002056 G #5-418
 L\$SPCP 002020 G #5-418
 L\$SPTP 002024 G #5-418
 L\$STA 002030 G #5-418
 L\$SW 002204 G 5-418 13-489 #13-489
 L\$TEST 002114 G #5-418
 L\$TIML 002014 G #5-418
 L\$UNIT 002012 G #5-418 103-2993 103-3048
 L10001 002202 11-466 #11-479
 L10002 002220 13-489 #13-505
 L10003 012572 #86-2513
 L10004 013660 #96-2717
 L10005 013714 #96-2730
 L10006 015004 #101-2945
 L10007 016064 #103-3082
 L10010 016262 #105-3113
 L10011 016356 #107-3162
 L10012 016422 #109-3200
 L10013 016504 #111-3245
 L10014 016700 114-3312 #114-3319
 L10015 017144 116-3326 #116-3346
 L10016 020534 118-3356 #118-3388
 L10017 021510 120-3396 #120-3420
 L10020 022274 122-3427 #122-3451
 L10021 023250 124-3458 #124-3482
 L10022 024034 126-3489 #126-3513
 L10023 036052 132-3633 #132-3648
 L10024 036242 134-3677 #134-3693
 L10025 036564 #136-3717
 L10027 036574 136-3717 #136-3722
 MABEE 012300 84-2461 #84-2467
 MSAUTO 016320 105-3121 #105-3126
 MSBDA J02332 G #22-717 98-2788
 MSCMD 002676 G 20-701 #22-739
 MSGNM 002376 G 20-691 #22-721
 MSG1 036052 132-3636 #132-3650

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
MSG1B		036063	132-3637 #132-3651
MSG1C		036100	132-3638 #132-3652
MSG2		036131	132-3639 #132-3653
MSG3		036146	132-3640 #132-3654
MSG4		036242	134-3679 #134-3695
MSG4B		036307	134-3680 #134-3696
MSG5		036351	134-3681 #134-3697
MSG6		036403	134-3682 #134-3698
MSG7		036430	134-3683 #134-3699
MSG8		036456	134-3685 #134-3700
MSHCHK		002550 G	20-694 #22-731
MSHDRD		003146 G	20-692 #22-755
MSHDWR		003210 G	20-693 #22-757
MSNIT		002612 G	20-698 #22-733
MSNLOG		002314 G	20-685 20-689 #22-715
MSNOMO		002440 G	20-697 #22-723
MSNOTP		002456 G	20-707 #22-725
MSNRSP		002756 G	20-705 #22-745
MSOVRN		003252 G	20-690 #22-759
MSPART		002626 G	20-699 #22-735
MSQRSP		002772 G	20-706 #22-747
MSREC		002712 G	20-702 #22-741
MSRNIT		002530 G	20-688 #22-729
MSSELF		002356 G	20-703 #22-719
MSSFED		003046 G	20-686 #22-751
MSSFWR		003106 G	20-687 #22-753
MSSKER		002300 G	20-695 #22-713
MSIOSN		003024 G	20-708 #22-749
MSUNIT		002650 G	20-700 #22-737
MSWPRO		002506 G	20-696 #22-727
MSWRSP		002732 G	20-704 #22-743
MXRTY		003322 G	#24-783 80-2278
NCART	=	000054 G	#16-608 82-2373
NODRVS		016116	103-3015 #103-3087
NOMOR		006340	60-1650 #60-1657
NUMOT	=	000030 G	#16-598 82-2350
NOUNIT	=	000036 G	#16-601 82-2383
NOXOFF		006430	#62-1701
NTSFT		012250	84-2454 #84-2460
NXTRET		006334	60-1620 60-1649 #60-1654
NXTST		006052 G	58-1582 #60-1611
NXTST2		006156	60-1622 #60-1626
ONEFIL	=	000001	#2-4 2-8 4-356 5-357 5-396 13-508 14-509 14-522 100-2844 101-2845 101-2858 111-3247 111-3248 112-3259 131-3579 132-3580 132-3594
OTL	=	000052 G	#16-607 78-2160
OVRFLD		012746	84-2437 #86-2518
OVRN	=	000012 G	#16-591 78-2163
OSAPTS	=	000000	#5-375 5-418
OSAU	=	000001	#5-375 #5-408 5-418
OSGMR	=	000001	#5-375 #5-408 5-418
OSBGNS	=	000001	#5-375 #5-408 5-418
OSDU	=	000001	#5-375 #5-408 5-418

CZTUUB		CREATED BY	MACRO ON 2-AUG-79 AT 15:57		PAGE 10	M 8				
SYMBOL	CROSS REFERENCE	VALUE	REFERENCES	REF	V01					
OSERRT	=	000000	#5-375	5-418						
OSGNSW	=	000001	#5-375	#5-408	5-418					
OSPOIN	=	000001	#5-375	#5-408	#5-408	#5-408	#5-408	#5-408	5-408	5-418
OSSETU	=	000001	#5-375	#5-408	5-418	136-3713				
PARTL	=	000034	G	#16-600	82-2393					
PATTEN	=	000072	G	#26-841	86-2505	98-2779	*103-3040	*118-3362	118-3363	*120-3402
				*122-3433	*122-3436	*124-3464	*124-3467	*124-3468	*126-3495	*126-3498
				*103-3009	103-3036	#103-3090				
PDTFLG		016152	G	#60-1628	60-1647					
PERDEV		006170		#26-847	*62-1712	70-1908	70-1947	*70-1949	*78-2133	98-2771
PKPTR	=	000104	G	#14-567						100-2820
PNT	=	001000	G	#13-493	100-2818					
PRBUF		002210		*100-2822	100-2823	#100-2836				
PRDAT		014344		100-2823	#100-2837					
PRFORM		014346		#14-567						
PR1	=	002000	G	#14-567	94-2663					
PR100	=	000000	G	#14-567						
PR101	=	000040	G	#14-567						
PR102	=	000100	G	#14-567						
PR103	=	000140	G	#14-567						
PR104	=	C00200	G	#14-567						
PR105	=	000240	G	#14-567						
PR106	=	000300	G	#14-567						
PR107	=	000340	G	#14-567	94-2664	94-2666	105-3101			
PRNPAK		014176	G	98-2792	#100-2813					
PRNSIZ		003330	G	#24-786	*98-2791	*100-2824				
PIR		016454		*109-3201	109-3202	*109-3205	#109-3208			
RCBCNT		003310		#24-778	*68-1832	*68-1843	*68-1848	*68-1852	*70-1904	*70-1921
				*70-1938	*78-2135	78-2142	78-2204			
				#18-645	54-1543	130-3542	130-3546	130-3547	130-3548	130-3549
				130-3552	130-3553					
RCDB	=	000024	G	#26-827	74-2027	74-2052	94-2662	96-2724	*103-3032	
RCFLG		003306	G	#24-777	*68-1833	68-1839	*70-1902	70-1916	*78-2136	
RCINIT	=	000006	G	#16-590	78-2172					
RCSR	=	000022	G	#26-826	74-2025	74-2041	94-2686	94-2703	96-2723	*103-3030
RCVBUF	=	000102	G	#26-846	54-1542	62-1712	68-1829	78-2127	*103-2985	105-3106
RCVHND		013662		#96-2723						
RCVINT		013662	G	94-2664	#96-2721					
RDNO	=	000114	G	#26-851	*62-1730	101-2922				
RDNI	=	000116	G	#26-852	*62-1732					

CZTUUB
 SYMBOL CROSS REFERENCE
 SYMBOL VALUE
 RLUN 015006
 RPTR 015010
 RSCMD = 000002 G
 RSCNT = 000020 G
 RSDASZ = 000204 G
 RSDATA = 000001 G
 RSDNSZ = 000222 G
 RSEND = 000002 G
 RSINI = 000004 G
 RMSIZ = 000012 G
 RSNDZ = 000016 G
 RSNTAB 002220
 RSSEND = 000100 G
 RSSNIT = 000001 G
 RSSNOP = 000000 G
 RSSNSZ = 000016 G
 RSSRD = 000002 G
 RSSSEK = 000005 G
 RSSSLF = 000007 G
 RSSWR = 000003 G
 RSVP 006364 G
 RSXOFF = 000023 G
 RSXON = 000020 G
 RTRYN 011422
 RUN 006022 G
 SECREC 003326 G
 SERVST 007656
 SETDR 005602 G
 SETLEN 016026
 SETPTR 005656
 SETSRV 007572
 SETUP 005754 G
 SFPTBL 002204 G
 SFT 012316
 SFTOUT 036242
 SFTRD = 000002 G
 SFTWR = 000004 G
 SKTRR = 000024 G
 SLFEM = 000044 G

CREATED BY MACRO ON 2-AUG-79 AT 15:57
 REFERENCES
 *101-2911 *101-2912 101-2921 #101-2946
 *101-2902 101-2907 101-2933 *101-2935 #101-2947
 #18-627 18-633 62-1718 114-3314 116-3332 116-3332 118-3363 118-3363 118-3364
 118-3364 120-3406 120-3406 122-3437 122-3437 124-3468 124-3468 126-3499 126-3499
 #18-628 78-2139 94-2691 118-3363 118-3363 120-3406 120-3406 124-3468 124-3468
 #18-640 18-642 18-645 68-1848 78-2177 78-2184
 #18-632 68-1846 70-1923 70-1947 78-2154 78-2175 118-3363 118-3363 118-3364
 118-3364 120-3406 120-3406 122-3437 122-3437 124-3468 124-3468 126-3499 126-3499
 #18-642 70-1925
 #18-633 68-1841 70-1919 78-2149 78-2187 114-3314 118-3363 118-3363 118-3364
 118-3364 120-3406 120-3406 122-3437 122-3437 124-3468 124-3468 126-3499 126-3499
 #18-631 78-2169 96-2747
 #18-638 18-644 114-3314 114-3314 116-3332 116-3332 118-3363 118-3363 118-3363
 118-3363 118-3364 118-3364 120-3406 120-3406 120-3406 120-3406 120-3406
 122-3437 122-3437 122-3437 122-3437 124-3468 124-3468 124-3468 124-3468 126-3499
 126-3499 126-3499 126-3499
 #18-636 18-642 18-645 68-1843 70-1921 114-3314 116-3332 118-3363 118-3363
 118-3364 118-3364 120-3406 120-3406 122-3437 122-3437 124-3468 124-3468 126-3499
 126-3499
 #20-685 84-2442
 #18-651
 #18-656 94-2645
 #18-655
 #18-644 78-2190 114-3314 116-3332 118-3363 118-3363 118-3364 118-3364 120-3406
 120-3406 122-3437 122-3437 124-3468 124-3468 126-3499 126-3499 126-3499 126-3499
 #18-653 62-1725 82-2341 118-3363 118-3364 118-3364 120-3406 122-3437 122-3437
 124-3468 126-3499 126-3499
 #18-654 116-3332
 #18-657 82-2358 114-3314
 #18-652 62-1734 82-2345 118-3363 120-3406 124-3468
 #62-1093 114-3314 116-3332 118-3363 118-3363 118-3363 118-3364 118-3364 120-3406
 120-3406 120-3406 122-3437 122-3437 124-3468 124-3468 124-3468 126-3499 126-3499
 #18-630 68-1870 130-3539 130-3540
 #18-629 68-1869
 80-2250 80-2281 #80-2312
 #58-1582 58-1591 114-3311 114-3311 116-3325 116-3325 118-3355 118-3355 120-3395
 120-3395 122-3426 122-3426 124-3457 124-3457 126-3488 126-3488
 #24-785 *103-3056 *103-3059 120-3414 122-3445 124-3476 126-3507
 *66-1795 70-1957 *72-1982 #72-1996
 #50-1497 114-3311 116-3325 118-3355 120-3395 122-3426 124-3457 126-3488
 #103-3054
 *50-1497 50-1498 50-1503 *50-1505 #50-1508
 70-1895 70-1899 #72-1972
 #56-1564 11-3311 114-3311 16-3325 116-3325 118-3355 118-3355 120-3395 120-3395
 122-3426 12-3426 124-3457 124-3457 126-3488 126-3488
 #13-489
 84-2470 #84-2473
 #134-3633
 #16-588 80-2263
 #16-589 80-2265
 #16-596 82-2368
 #16-604 82-2363

PAGE 11
 CREF V01
 N 8
 SEQ 0104

SYMBOL CROSS REFERENCE

CREF VOI

SEO 0106

SYMBOL	VALUE	REFERENCES	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503
		86-2503	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503	86-2503
		86-2506	86-2506	86-2506	86-2506	86-2506	86-2506	86-2506	86-2506	86-2506
		86-2509	86-2509	86-2509	86-2509	86-2509	86-2509	86-2509	86-2509	86-2509
		94-2664	94-2664	94-2664	94-2664	94-2664	94-2664	94-2664	94-2664	94-2664
		94-2704	94-2704	94-2704	94-2704	94-2704	94-2704	94-2704	94-2704	94-2704
		98-2788	98-2788	98-2788	98-2788	98-2788	98-2788	98-2788	98-2788	98-2788
		100-2823	100-2823	100-2823	100-2823	100-2823	100-2823	100-2823	100-2823	100-2823
		100-2828	100-2828	100-2828	100-2828	100-2828	100-2828	100-2828	100-2828	100-2828
		101-2903	101-2903	101-2903	101-2903	101-2903	101-2903	101-2903	101-2903	101-2903
		101-2905	101-2905	101-2905	101-2905	101-2905	101-2905	101-2905	101-2905	101-2905
		101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922
		101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922	101-2922
		101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932
		101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932	101-2932
		101-2945	103-2975	103-2975	103-2975	103-2975	103-2975	103-2975	103-2975	103-2975
		103-3002	103-3002	103-3002	103-3002	103-3002	103-3002	103-3002	103-3002	103-3002
		103-3052	103-3052	103-3052	103-3052	103-3052	103-3052	103-3052	103-3052	103-3052
		-3112	105-3112	105-3112	105-3112	105-3112	105-3112	105-3112	105-3112	105-3112
		3141	107-3162	109-3180	109-3180	109-3180	109-3180	109-3180	109-3180	109-3180
		111-3245	114-3312	114-3312	114-3312	114-3312	114-3312	114-3312	114-3312	114-3312
		118-3388	120-3396	120-3396	120-3396	120-3396	120-3396	120-3396	120-3396	120-3396
		124-3482	126-3489	126-3489	126-3489	126-3489	126-3489	126-3489	126-3489	126-3489
		132-3637	132-3637	132-3637	132-3637	132-3637	132-3637	132-3637	132-3637	132-3637
		132-3639	132-3640	132-3640	132-3640	132-3640	132-3640	132-3640	132-3640	132-3640
		134-3679	134-3679	134-3680	134-3680	134-3680	134-3680	134-3680	134-3680	134-3680
		134-3682	134-3682	134-3683	134-3683	134-3683	134-3683	134-3683	134-3683	134-3683
		134-3685	134-3693	136-3715	136-3715	136-3715	136-3715	136-3715	136-3715	136-3715
SVCSUB	= 177777	#5-375	#5-383							
SVCTAG	= 177777	#5-375	#5-385	11-479	11-479	11-479	13-505	13-505	13-505	86-2513
		86-2513	86-2513	96-2717	96-2717	96-2717	96-2730	96-2730	96-2730	101-2945
		101-2945	101-2945	103-3082	103-3082	103-3082	105-3113	105-3113	105-3113	107-3162
		107-3162	107-3162	109-3200	109-3200	109-3200	111-3245	111-3245	111-3245	114-3319
		114-3319	114-3319	116-3346	116-3346	116-3346	118-3388	118-3388	118-3388	120-3420
		120-3420	120-3420	122-3451	122-3451	122-3451	124-3482	124-3482	124-3482	126-3513
		126-3513	126-3513	132-3648	132-3648	132-3648	134-3693	134-3693	134-3693	136-3717
		136-3717	136-3717	136-3722	136-3722	136-3722				
SVCTST	= 177777	#5-375	#5-382	114-3310	114-3310	114-3310	116-3324	116-3324	116-3324	118-3354
		118-3354	118-3354	120-3394	120-3394	120-3394	122-3425	122-3425	122-3425	124-3456
		124-3456	124-3456	126-3487	126-3487	126-3487				
SWAPDR	005500 G	#48-1462	114-3311	116-3325	118-3355	120-3395	122-3426	124-3457	126-3488	
SWPTR	005600	#48-1463	48-1464	48-1473	#48-1475	#48-1483				
SVSTAT	003300 G	#24-766	#60-1617	66-1793	#68-1838	#70-1915	76-2087	#78-2134	80-2230	#80-2299
		86-2503	#90-2555	#90-2558	90-2569	#94-2697	#103-3057			
SBLSYM	= 010000	#5-375	#11-479	#13-505	#86-2513	#96-2717	#96-2730	#101-2945	#103-3082	#105-3113
		#107-3162	#109-3200	#111-3245	#114-3319	#116-3346	#118-3388	#120-3420	#122-3451	#124-3482
		#126-3513	#132-3648	#134-3693						
TAPLEN	003302 G	#24-775	#103-3054	#103-3055	#103-3057	120-3400	120-3415	122-3431	122-3446	124-3462
		124-3477	126-3493	126-3508						
THRSHI	011370	80-2260	#80-2310							
THRSLO	011342	80-2258	#80-2308							
IMP	000066 G	#26-839	#118-3359	118-3375	#118-3376	#120-3400	#120-3407	#120-3415	#122-3431	#124-3438
		#122-3446	#124-3462	#124-3469	#124-3477	#126-3493	#126-3500	#126-3508		

CZTUUB CREATED BY MACRO ON 2-AUG-79 AT 15:57

PAGE 14
CREF V01

SEQ 0107

SYMBOL	VALUE	CROSS REFERENCE	REFERENCES
SYMBOL			
TOMANY	016066		103-2995 #103-3085
TORCVB	= 000050	G	#16-606 74-2066 96-2742
TOSNDB	= 000056	G	#16-609 64-1773 94-2654
TRBUF	024370		62-1698 62-1701 62-1707 114-3314 116-3332 118-3363 118-3363 118-3363 118-3363
			118-3364 118-3364 120-3406 120-3406 120-3406 120-3406 122-3437 122-3437 124-3468
			124-3468 124-3468 124-3468 126-3499 126-3499 #130-3542
TRK	= 000062	G	#26-836 *120-3401 120-3411 *120-3413 *122-3432 122-3442 *122-3444 *124-3463 124-3473
			*124-3475 *126-3494 126-3504 *126-3506
TRPHND	016266		105-3101 #105-3121
TRPPT	016264		*105-3102 105-3103 105-3108 *105-3110 #105-3114
TSTPC	000020	C	#26-825 *56-1567 60-1619 60-1640 *62-1694
TSTTOP	003320		#24-782 56-1567 *114-3311 *116-3325 *118-3355 *120-3395 *122-3426 *124-3457 *126-3498
TST1	016552		114-3311 #114-3314
TST2	016746		116-3325 #116-3329
TST3	017212		118-3355 #118-3359
TST3PT	020522		118-3362 118-3366 #118-3381
TST4	020602		120-3395 #120-3399
TST4EX	021502		120-3412 #120-3417
TST5	021556		122-3426 #122-3430
TST5EX	022266		122-3443 #122-3448
TST6	022342		124-3457 #124-3461
TST6EX	023242		124-3474 #124-3479
TST7	023316		126-3488 #126-3492
TST7EX	024026		126-3505 #126-3510
TUVECT	= 000204	G	#28-894 94-2664 *94-2665 94-2666 *94-2667 94-2704 *94-2705 94-2706 *94-2707
			*103-3006
TSARGC	= 000002		#5-418 5-418 #5-418 5-418 5-419 #5-418 5-418 5-418 #5-418 5-418 5-418 #5-418
			5-418 5-418 #5-418 5-418 5-418 #5-418 5-418 5-418 #5-418 5-418 5-418 #5-418
			80-2250 #80-2250 80-2250 80-2250 #80-2255 80-2255 #80-2255 80-2255 80-2255 #80-2255 80-2255 80-2255 #80-2255
			#80-2258 80-2258 80-2258 #80-2260 80-2260 80-2260 #80-2274 80-2274 80-2274 #80-2274 80-2274 80-2274 #80-2274
			#80-2281 80-2281 #80-2281 80-2281 80-2281 #80-2289 80-2289 80-2289 #80-2289 80-2289 80-2289 #80-2289
			86-2503 #86-2503 86-2503 #86-2503 86-2503 #86-2503 86-2503 86-2503 #86-2503 86-2503 86-2503 #86-2503
			86-2506 #86-2506 86-2506 #86-2506 86-2506 #86-2506 86-2506 86-2506 #86-2506 86-2506 86-2506 #86-2506
			86-2506 #86-2509 86-2509 #86-2509 86-2509 #86-2509 86-2509 86-2509 #86-2509 86-2509 86-2509 #86-2509
			98-2789 98-2789 #100-2823 100-2823 #100-2823 100-2823 100-2823 #100-2823 100-2823 100-2823 #100-2823
			100-2828 #100-2830 100-2830 100-2830 #101-2903 101-2903 101-2903 #101-2903 101-2903 101-2903 #101-2903
			101-2905 #101-2921 101-2921 #101-2921 101-2921 101-2921 #101-2921 101-2921 101-2921 #101-2921 101-2921 101-2921 #101-2921
			101-2922 #101-2922 101-2922 #101-2922 101-2922 #101-2922 101-2922 101-2922 #101-2922 101-2922 101-2922 #101-2922
			#101-2922 101-2922 #101-2922 101-2922 #101-2922 101-2922 101-2922 #101-2922 101-2922 101-2922 #101-2922
			#101-2932 101-2932 #101-2932 101-2932 #101-2932 101-2932 101-2932 #101-2932 101-2932 101-2932 #101-2932
			101-2932 #101-2932 101-2932 #101-2932 101-2932 #101-2932 101-2932 101-2932 #101-2932 101-2932 101-2932 #101-2932
			105-3121 105-3121 #109-3180 109-3180 #109-3180 109-3180 109-3180 #109-3180 109-3180 109-3180 #109-3180
TSRDE	= 005052		#132-3636 132-3636 #132-3636 132-3636 #132-3636 132-3636 #132-3636 132-3636 #132-3636 132-3636 #132-3636
			132-3637 #132-3637 132-3637 #132-3637 132-3637 #132-3637 132-3637 132-3637 #132-3637 132-3637 132-3637 #132-3637
			#132-3639 132-3639 #132-3639 132-3639 #132-3639 132-3639 #132-3639 132-3639 #132-3639 132-3639 #132-3639
			132-3640 #132-3640 132-3640 #134-3679 134-3679 #134-3679 134-3679 #134-3679 134-3679 #134-3679 134-3679 #134-3679
			#134-3680 134-3680 #134-3680 134-3680 #134-3680 134-3680 #134-3680 134-3680 #134-3680 134-3680 #134-3680
			134-3681 #134-3681 134-3681 #134-3682 134-3682 #134-3682 134-3682 #134-3682 134-3682 #134-3682 134-3682 #134-3682
			#134-3683 134-3683 #134-3683 134-3683 #134-3683 134-3683 #134-3683 134-3683 #134-3683 134-3683 #134-3683
			134-3685 #134-3685 134-3685
TSFRN	= 000146		#5-375 #60-1650 60-1650 #84-2437 84-2437 #84-2451 84-2451 #84-2457 84-2457 #84-2457 84-2457 #84-2457
			#84-2464 84-2464 #84-2475 84-2475 #84-2479 84-2479 #98-2788 98-2788 #103-2995

CZTUUB
SYMBOL CROSS REFERENCE
SYMBOL VALUE

MACRO ON 2-AUG-79 AT 15:57

PAGE 15
REF V01

E 9

SFO 0108

TSEXCP = 0000C7
TSFLAG = 000040

REFERENCES

103-2995 #103-3015 103-3015
#132-3636 132-3636 #132-3637 132-3637 #134-3679 134-3679 #134-3685 134-3685
#114-3312 #114-3312 114-3312 114-3312 #116-3326 #116-3326 116-3326 116-3326 #118-3356
#118-3356 118-3356 118-3356 #120-3396 #120-3396 120-3396 120-3396 #122-3427 #122-3427
122-3427 122-3427 #124-3458 #124-3458 124-3458 124-3458 #126-3489 #126-3489 126-3489
126-3489

TSFREE = 036574
TSGMAN = 000000
TSMILI = 000376
TSLAST = 000001
TSLOLI = 000001
TSLSYM = 010000

136-3713 #136-3723
#5-375
#132-3636 132-3636 #132-3637 132-3637 #134-3679 134-3679 #134-3685 134-3685
#5-375 #136-3713 136-3716
#132-3636 132-3636 #132-3637 132-3637 #134-3679 134-3679 #134-3685 134-3685
#5-375 5-375 11-479 13-505 86-2513 96-2717 96-2730 101-2945 103-3082
105-3113 107-3162 109-3200 111-3245 114-3319 116-3346 118-3388 120-3420 122-3451
124-3482 126-3513 132-3693 134-3693

TSLINO = 000007
TSNEST = 177777

#136-3713
#5-375 5-401 #5-401 5-401 7-428 #7-428 7-428 7-432 7-432
7-432 #7-432 11-466 #11-466 11-466 11-479 11-479 11-479 #11-479
13-489 #13-489 13-489 13-505 13-505 13-505 #13-505 13-507 13-507
13-507 #13-507 14-560 #14-560 14-560 86-2496 #86-2496 86-2496 86-2513
86-2513 86-2513 #86-2513 96-2713 #96-2713 96-2713 96-2717 96-2717 96-2717
#96-2717 96-2721 #96-2721 96-2721 96-2730 96-2730 96-2730 #96-2730 100-2842
100-2842 100-2842 #100-2842 101-2886 #101-2886 101-2886 101-2893 #101-2893 101-2893
101-2945 101-2945 101-2945 #101-2945 101-2961 101-2961 101-2961 #101-2961 103-2971
#103-2971 103-2971 103-3082 103-3082 103-3082 #103-3082 105-3099 #105-3099 105-3099
105-3113 105-3113 105-3113 #105-3113 107-3136 #107-3136 107-3136 107-3162 107-3162
107-3162 #107-3162 109-3172 #109-3172 109-3172 109-3200 109-3200 #109-3200
111-3222 #111-3222 111-3222 111-3245 111-3245 111-3245 #111-3245 114-3307 #114-3307
114-3307 114-3310 #114-3310 114-3310 114-3319 114-3319 114-3319 #114-3319 116-3324
#116-3324 116-3324 116-3346 116-3346 116-3346 #116-3346 118-3354 #118-3354 118-3354
118-3388 118-3388 118-3388 #118-3388 120-3394 #120-3394 120-3394 120-3420 120-3420
120-3420 #120-3420 122-3425 #122-3425 122-3425 122-3451 122-3451 122-3451 #122-3451
124-3456 #124-3456 124-3456 124-3482 124-3482 124-3482 #124-3482 126-3487 #126-3487
126-3487 126-3513 126-3513 126-3513 #126-3513 130-3557 130-3557 130-3557 #130-3557
132-3622 #132-3622 132-3622 132-3633 #132-3633 132-3633 132-3648 132-3648 132-3648
#132-3648 134-3677 #134-3677 134-3677 134-3693 134-3693 134-3693 #134-3693 136-3714
136-3714 136-3714 #136-3714

TSNSO = 000000

#5-401 13-507 #14-560 100-2842 #101-2886 101-2961 #103-2971 103-3082 #105-3099
105-3113 #107-3136 107-3162 #109-3172 109-3200 #111-3222 111-3245 #114-3307 130-3557
#132-3622 136-3714

TSN = 000005

#7-428 7-432 #11-466 11-479 #13-489 13-505 #86-2496 86-2513 #96-2713
96-2717 #96-2721 96-2730 #101-2893 101-2945 #114-3310 114-3319 #116-3324 116-3346
#118-3354 118-3388 #120-3394 120-3420 #122-3425 122-3451 #124-3456 124-3482 #126-3487
126-3513 #132-3633 132-3648 #134-3677 134-3693

TSPCNT = 000000
TSPTAB = 010026
TSPTHV = 000001
TSPINU = 000001
TSSAVL = 177777
TSSFGL = 177777
TSSIZE = 000006
TSSUBN = 000000
TSTAGL = 177777

#136-3716 136-3717 #136-3717 136-3717 136-3723 136-3723
#136-3717 136-3717 #136-3717 136-3723 136-3723
#5-418 #136-3723
#5-375 136-3717 #136-3717 136-3723 136-3723
#5-375
#5-375
136-3713 #136-3723
#5-375 #114-3310 #116-3324 #118-3354 #120-3394 #122-3425 #124-3456 #126-3487
#5-375

SYMBOL CROSS REFERENCE
SYMBOL VALUE
TSTAGN = 010030

REFERENCES

#5-375	7-428	7-428	#7-428	11-466	11-466	#11-466	13-489	13-489
#13-489	86-2496	86-2496	#86-2496	96-2713	96-2713	#96-2713	96-2721	96-2721
#96-2721	101-2893	101-2893	#101-2893	103-2971	103-2971	#103-2971	105-3099	105-3099
#105-3099	107-3136	107-3136	#107-3136	109-3172	109-3172	#109-3172	111-3222	111-3222
#111-3222	114-3310	114-3310	#114-3310	116-3324	116-3324	#116-3324	118-3354	118-3354
#118-3354	120-3394	120-3394	#120-3394	122-3425	122-3425	#122-3425	124-3456	124-3456
#124-3456	126-3487	126-3487	#126-3487	132-3633	132-3633	#132-3633	134-3677	134-3677
#134-3677	136-3716	136-3716	#136-3716	136-3717	136-3717	#136-3717	136-3717	136-3717
#136-3717								

TSTEMP = 000000

#7-432	7-432	#9-448	9-448	9-448	#9-448	9-448	9-448	#9-448
9-448	9-448	#9-448	9-448	9-448	#9-448	9-448	9-448	#9-448
9-448	9-448	#9-448	9-448	9-448	#9-448	#11-479	11-479	#13-505
13-505	#13-507	13-507	#86-2513	86-2513	#96-2717	96-2717	#96-2730	96-2730
#100-2842	100-2842	#101-2945	101-2945	#101-2961	101-2961	#103-3082	103-3082	#105-3113
105-3113	#107-3162	107-3162	#109-3200	109-3200	#111-3245	111-3245	#114-3312	114-3312
#114-3319	114-3319	#116-3326	116-3326	#116-3346	116-3346	#118-3356	118-3356	#118-3388
118-3388	#120-3396	120-3396	#120-3420	120-3420	#122-3427	122-3427	#122-3451	122-3451
#124-3458	124-3458	#124-3482	124-3482	#126-3489	126-3489	#126-3513	126-3513	#130-3557
130-3557	#132-3636	132-3636	#132-3636	132-3636	#132-3636	132-3636	#132-3637	132-3637
#132-3637	132-3637	#132-3637	132-3637	#132-3638	132-3638	#132-3638	132-3638	#132-3638
132-3638	#132-3639	132-3639	#132-3639	132-3639	#132-3639	132-3639	#132-3640	132-3640
#132-3640	132-3640	#132-3640	132-3640	#132-3648	132-3648	#134-3679	134-3679	#134-3679
134-3679	#134-3679	134-3679	#134-3680	134-3680	#134-3680	134-3680	#134-3680	134-3680
#134-3681	134-3681	#134-3681	134-3681	#134-3681	134-3681	#134-3682	134-3682	#134-3682
134-3682	#134-3682	134-3682	#134-3683	134-3683	#134-3683	134-3683	#134-3683	134-3683
#134-3685	134-3685	#134-3685	134-3685	#134-3685	134-3685	#134-3693	134-3693	#136-3714
136-3714								

TBTST = 000007

#5-375	114-3310	#114-3310	114-3310	116-3324	#116-3324	116-3324	118-3354	#118-3354
118-3354	120-3394	#120-3394	120-3394	122-3425	#122-3425	122-3425	124-3456	#124-3456
124-3456	126-3487	#126-3487	126-3487	136-3713				

TBTSTM = 177777

#5-375	58-1588	60-1650	60-1651	60-1653	64-1768	74-2046	80-2250	80-2255
80-2258	80-2260	80-2274	80-2281	80-2289	84-2437	84-2451	84-2457	84-2464
84-2475	84-2479	84-2483	86-2503	86-2506	86-2509	86-2513	94-2648	94-2663
94-2664	94-2666	94-2704	94-2706	96-2738	96-2739	98-2788	98-2789	100-2823
100-2828	100-2830	101-2901	101-2903	101-2904	101-2905	101-2906	101-2921	101-2922
101-2932	101-2945	103-2975	103-2995	103-2996	103-3002	103-3015	103-3016	103-3052
103-3082	105-3101	105-3112	105-3113	105-3121	105-3124	107-3141	107-3162	109-3180
109-3200	111-3245	114-3312	114-3319	116-3326	116-3346	118-3356	118-3388	120-3396
120-3420	122-3427	122-3451	124-3458	124-3482	126-3489	126-3513		

TBTSTS = 000001

#5-375	#114-3310	#116-3324	#118-3354	#120-3394	#122-3425	#124-3456	#126-3487
--------	-----------	-----------	-----------	-----------	-----------	-----------	-----------

TSSAU = 010013

#111-3222	111-3245
-----------	----------

TSSAUT = 010010

#105-3099	105-3113
-----------	----------

TSSCLE = 010011

#107-3136	107-3162
-----------	----------

TSSDAT = 010027

#136-3717	136-3717	136-3722
-----------	----------	----------

TSSDU = 010012

#109-3172	109-3200
-----------	----------

TSSMAR = 010023

#132-3633	132-3633	132-3648
-----------	----------	----------

TSSHW = 010001

#11-466	11-466	11-479
---------	--------	--------

TSSINI = 010007

#103-2971	103-3082
-----------	----------

TSSMSG = 010003

#86-2496	86-2513
----------	---------

TSSPC = 000001

#136-3716	136-3723
-----------	----------

TSSPRO = 010000

#7-428

TSSPTA = 010026

#136-3716	136-3717	#136-3717
-----------	----------	-----------

CZTUUB CREATED BY MACRO ON 2-AUG-79 AT 15:57

PAGE 17
CREF V01

SEQ 0110

SYMBOL	CROSS REFERENCE	VALUE	REFERENCES
TSSRPT	=	010006	#101-2893 101-2945
TSSSOF	=	010024	#134-3677 134-3677 134-3693
TSSSRV	=	010005	#96-2713 96-2717 #96-2721 96-2730
TSSSW	=	010002	#13-489 13-489 13-505
TSSTES	=	010022	#111-3310 114-3312 114-3319 #116-3324 116-3326 116-3346 #118-3354 118-3356 118-3388 #120-3394 120-3396 120-3420 #122-3425 122-3427 122-3451 #124-3456 124-3458 124-3482 #126-3487 126-3489 126-3513
T1		016506	G 9-448 #114-3310
T1TRY	=	000146	G #28-877
T2		016702	G 9-448 #116-3324
T3		017146	G 9-448 #118-3354
T4		020536	G 9-448 #120-3394
T4TRY	=	000132	G #28-871
T5		021512	G 9-448 #122-3425
T6		022276	G 9-448 #124-3456
T7		023252	G 9-448 #126-3487
UAM	=	000200	G #14-567
UNIT		012574	G 86-2503 #86-2514
UNREC		011442	80-2289 #80-2314
UNSUC		011040	80-2254 #80-2271
UNXPCT		007356	#70-1918
WAIT		013716	94-2672 94-2680 94-2687 #96-2734 96-2741
WHCHDR		013052	G 62-1728 62-1742 84-2432 #88-2532
WRLOCK	=	000026	G #16-597 82-2388 84-2460
WRINO	=	000110	G #26-849 *62-1744 101-2922 103-3022 103-3024
WRINI	=	000112	G #26-850 *62-1747 101-2932
XFNSND		006416	#62-1698
XMDB	=	000030	G #26-829 64-1776 94-2657 96-2716 *103-3039
XMSR	=	000026	G #26-828 64-1745 94-2644 94-2650 94-2661 94-2671 94-2679 94-2702 96-2715
			*103-3034
XSCNT	=	000036	G #26-832 *114-3314 *116-3332 *118-3363 *118-3363 *118-3363 *120-3406 *120-3406 *120-3406 *124-3468 *124-3468 *124-3468
XFLG	=	000034	G #26-831 62-1713 68-1822 78-2129 *114-3314 *116-3332 *118-3363 *118-3363 *118-3363 118-3363 118-3364 118-3364 *120-3406 *120-3406 *120-3406 120-3406 122-3437 122-3437 *124-3468 *124-3468 *124-3468 124-3468 126-3499 126-3499
XSPKMM	=	000032	G #26-830 62-1709 *62-1709 68-1830 70-1897 *70-1901 *70-1912 *70-1918 *70-1934 *70-1945 78-2128 *114-3314 *116-3332 *118-3363 *118-3363 *118-3363 *118-3363 *118-3364 *118-3364 *120-3406 *120-3406 *120-3406 *120-3406 *122-3437 *122-3437 *124-3468 *124-3468 *124-3468 *124-3468 *126-3499 *126-3499
XSPTR	=	000106	G #26-848 *62-1715 70-1902 *70-1903 70-1904 *70-1952
XBALWA	=	000000	#5-375
XSFALS	=	000040	#5-375
XBOFFS	=	000400	#5-375
XSTRUE	=	000020	#5-375

CZTUUB CREATED BY MACRO ON 2-AUG-79 AT 15:57

PAGE 18
CREF V01

SEQ 0111

[illegible]

MACRO CROSS REFERENCE

CREF V01

SEQ 0113

MACRO NAME REFERENCES

MACRO NAME	REFERENCES
MSGETS	#101-2945 #101-2945 #103-2971 #103-2971 #103-3082 #103-3082 #105-3099 #105-3099 #105-3113 #105-3113 #107-3136 #107-3136 #107-3162 #107-3162 #109-3172 #109-3172 #109-3200 #109-3200 #111-3222 #111-3222 #111-3245 #111-3245 #114-3310 #114-3310 #114-3319 #114-3319 #116-3324 #116-3324 #116-3346 #116-3346 #118-3354 #118-3354 #118-3388 #118-3388 #120-3394 #120-3394 #120-3420 #120-3420 #122-3425 #122-3425 #122-3451 #122-3451 #124-3456 #124-3456 #124-3482 #124-3482 #126-3487 #126-3487 #126-3513 #126-3513 #132-3633 #132-3633 #132-3648 #132-3648 #134-3677 #134-3677 #134-3693 #134-3693 #136-3713 #136-3713 #136-3717 #136-3717 #136-3722 #136-3722 #7-432 7-432 #11-479 11-479 #13-505 13-505 #13-507 13-507 #86-2513 86-2513 #96-2717 96-2717 #96-2730 96-2730 #100-2842 100-2842 #101-2945 101-2945 #101-2961 101-2961 #103-3082 103-3082 #105-3113 105-3113 #107-3162 107-3162 #109-3200 109-3200 #111-3245 111-3245 #114-3319 114-3319 #116-3346 116-3346 #118-3388 118-3388 #120-3420 120-3420 #122-3451 122-3451 #124-3482 124-3482 #126-3513 126-3513 #130-3557 130-3557 #132-3648 132-3648 #134-3693 134-3693 #136-3714 136-3714
MSGETT	#114-3312 #116-3326 #118-3356 #120-3396 #122-3427 #124-3458 #126-3489
MSGNGB	#5-401 #5-418 #7-428 #9-448 #9-448 #11-466 #11-466 #11-466 #13-489 #13-489 #13-489 #14-560 #32-946 #32-946 #86-2496 #86-2496 #96-2713 #96-2713 #96-2721 #96-2721 #101-2886 #101-2893 #101-2893 #103-2971 #103-2971 #105-3099 #105-3099 #107-3136 #107-3136 #109-3172 #109-3172 #111-3222 #111-3222 #114-3307 #132-3622 #132-3633 #132-3633 #134-3677 #134-3677 #136-3713 #136-3713
MSGNIN	#5-418 5-418 #5-420 #5-420 5-420 5-420 #9-448 9-448 #9-448 9-448 #9-448 9-448 #9-448 9-448 #9-448 9-448 #9-448 9-448 #9-448 9-448 #9-448 9-448 #11-466 11-466 #13-489 13-489 #32-946 32-946 32-946 32-946 #58-1588 58-1588 #60-1650 #60-1650 60-1650 #60-1650 60-1650 #60-1650 60-1650 #60-1650 60-1650 60-1651 #60-1653 60-1653 #64-1768 64-1768 #74-2046 74-2046 #80-2250 80-2250 #80-2250 80-2250 #80-2250 80-2250 #80-2250 80-2250 #80-2250 80-2250 80-2255 #80-2255 80-2255 #80-2255 80-2255 #80-2255 80-2255 #80-2255 80-2255 #80-2258 80-2258 #80-2258 80-2258 #80-2258 80-2258 #80-2258 80-2258 80-2260 #80-2260 80-2260 #80-2260 80-2260 #80-2260 80-2260 #80-2260 80-2260 #80-2274 80-2274 #80-2274 80-2274 #80-2274 80-2274 #80-2281 80-2281 80-2281 #80-2281 80-2281 #80-2281 80-2281 #80-2281 80-2281 #80-2281 80-2281 #80-2289 80-2289 #80-2289 80-2289 #80-2289 80-2289 #84-2437 84-2437 84-2437 #84-2437 84-2437 #84-2437 84-2437 #84-2437 84-2437 #84-2437 84-2437 #84-2451 84-2451 #84-2451 84-2451 #84-2451 84-2451 #84-2451 84-2451 84-2457 #84-2457 84-2457 #84-2457 84-2457 #84-2457 84-2457 #84-2457 84-2457 #84-2464 84-2464 #84-2464 84-2464 #84-2464 84-2464 #84-2464 84-2464 84-2475 #84-2475 84-2475 #84-2475 84-2475 #84-2475 84-2475 #84-2475 84-2475

MACRO CROSS REFERENCE

CREF V01

SEG 0114

MACRO NAME	REFERENCES								
	#84-2483	#84-2483	84-2483	#86-2503	#86-2503	86-2503	86-2503	#86-2503	86-2503
	86-2503	86-2503	#86-2503	86-2503	#86-2503	86-2503	86-2503	#86-2503	86-2503
	#86-2506	#86-2506	86-2506	86-2506	#86-2506	86-2506	86-2506	#86-2506	86-2506
	#86-2506	86-2506	#86-2506	86-2506	#86-2506	86-2506	86-2506	#86-2506	86-2506
	#86-2509	#86-2509	86-2509	#86-2509	86-2509	#86-2509	86-2509	#86-2509	86-2509
	86-2509	#86-2513	86-2513	#94-2648	94-2648	#94-2663	94-2663	#94-2663	94-2663
	#94-2664	94-2664	#94-2664	94-2664	#94-2664	94-2664	#94-2664	94-2664	94-2664
	94-2664	#94-2666	#94-2666	94-2666	#94-2666	94-2666	#94-2666	94-2666	94-2666
	#94-2666	94-2666	94-2666	#94-2704	94-2704	#94-2704	94-2704	#94-2706	94-2706
	94-2706	#96-2717	96-2717	#96-2730	96-2730	#96-2738	96-2738	#96-2739	96-2739
	#98-2788	98-2788	#98-2788	98-2788	#98-2788	98-2788	#98-2788	98-2788	#98-2789
	98-2789	#98-2789	98-2789	#98-2789	98-2789	98-2789	#98-2789	98-2789	#100-2823
	#100-2823	100-2823	100-2823	#100-2823	100-2823	#100-2823	100-2823	100-2823	100-2823
	100-2823	#100-2828	#100-2828	100-2828	#100-2828	100-2828	#100-2828	100-2828	100-2828
	#100-2830	#100-2830	100-2830	#100-2830	100-2830	100-2830	#100-2830	100-2830	#101-2901
	101-2901	#101-2903	#101-2903	101-2903	#101-2903	101-2903	#101-2903	101-2903	101-2903
	#101-2904	101-2904	#101-2905	#101-2905	101-2905	#101-2905	101-2905	#101-2905	101-2905
	101-2905	#101-2906	101-2906	#101-2921	#101-2921	101-2921	#101-2921	101-2921	101-2921
	101-2921	#101-2921	101-2921	101-2921	#101-2922	101-2922	#101-2922	101-2922	#101-2922
	101-2922	#101-2922	101-2922	#101-2922	101-2922	101-2922	#101-2922	101-2922	101-2922
	#101-2922	101-2922	#101-2922	101-2922	#101-2922	101-2922	#101-2922	101-2922	101-2922
	#101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	#101-2932
	101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	#101-2932	101-2932	101-2932
	#101-2932	101-2932	101-2932	#101-2932	101-2932	101-2932	#101-2945	101-2945	#103-2975
	#103-2975	103-2975	#103-2976	103-2976	#103-2995	103-2995	#103-2995	103-2995	#103-2995
	103-2995	#103-2995	103-2995	#103-2996	103-2996	#103-3002	103-3002	#103-3002	#103-3002
	103-3002	#103-3003	103-3003	#103-3015	103-3015	103-3015	#103-3015	103-3015	103-3015
	#103-3015	103-3015	#103-3016	103-3016	#103-3052	103-3052	#103-3052	103-3052	#103-3082
	#105-3101	#105-3101	105-3101	#105-3101	105-3101	#105-3101	105-3101	#105-3101	105-3101
	105-3101	105-3101	#105-3112	105-3112	#105-3112	105-3112	#105-3113	105-3113	#105-3121
	105-3121	#105-3121	105-3121	105-3121	#105-3121	105-3121	105-3121	#105-3124	105-3124
	#107-3141	107-3141	#107-3162	107-3162	#109-3180	109-3180	#109-3180	109-3180	#109-3180
	109-3180	109-3180	#109-3180	109-3180	109-3180	#109-3200	109-3200	#111-3245	111-3245
	114-3312	#114-3312	114-3312	#114-3319	114-3319	#116-3326	116-3326	#116-3326	116-3326
	116-3346	#118-3356	118-3356	#118-3356	118-3356	#118-3388	118-3388	#120-3396	120-3396
	120-3396	#120-3420	120-3420	#122-3427	122-3427	#122-3427	122-3427	#122-3451	122-3451
	124-3458	#124-3458	124-3458	#124-3482	124-3482	#126-3489	126-3489	#126-3489	126-3489
	126-3513	#132-3633	132-3633	#132-3636	132-3636	132-3636	132-3636	#132-3636	#132-3637
	132-3637	132-3637	132-3637	#132-3638	132-3638	132-3638	132-3638	#132-3639	132-3639
	132-3639	#132-3640	132-3640	132-3640	132-3640	#132-3648	132-3648	#134-3677	134-3677
	134-3679	134-3679	134-3679	134-3679	134-3679	#134-3680	134-3680	134-3680	#134-3681
	134-3681	134-3681	134-3681	#134-3682	134-3682	134-3682	134-3682	#134-3683	134-3683
	134-3683	#134-3685	134-3685	134-3685	134-3685	134-3685	134-3685	#134-3693	134-3693
	136-3713	136-3713	136-3713	#136-3717	#136-3717	136-3717	136-3717		
MSGNTA	#11-479	#11-479	#13-505	#13-505	#86-2513	#86-2513	#96-2717	#96-2717	#96-2730
	#101-2945	#101-2945	#103-3082	#103-3082	#105-3113	#105-3113	#107-3162	#107-3162	#109-3200
	#111-3245	#111-3245	#114-3319	#114-3319	#116-3346	#116-3346	#118-3388	#118-3388	#120-3420
	#122-3451	#122-3451	#124-3482	#124-3482	#126-3513	#126-3513	#132-3648	#132-3648	#134-3693
	#136-3717	#136-3717	#136-3722	#136-3722					
MSGNTE	#114-3310	#114-3310	#116-3324	#116-3324	#118-3354	#118-3354	#120-3394	#120-3394	#122-3425
	#124-3456	#124-3456	#126-3487	#126-3487					
MSGNPT	#5-418	#5-418							

MACRO NAME	REFERENCES									
MSHAP	#5-418	#5-418								
MSINCR	#5-401	#5-401	#7-428	#7-428	#7-428	#7-428	#11-466	#11-466	#11-466	#11-466
	#13-489	#13-489	#13-489	#13-489	#14-560	#14-560	#58-1588	#60-1650	#60-1651	#60-1653
	#64-1768	#74-2046	#80-2250	#80-2255	#80-2258	#80-2260	#80-2274	#80-2281	#80-2289	#84-2437
	#84-2451	#84-2457	#84-2464	#84-2475	#84-2479	#84-2483	#86-2496	#86-2496	#86-2496	#86-2496
	#86-2503	#86-2506	#86-2509	#86-2513	#94-2648	#94-2663	#94-2664	#94-2666	#94-2704	#94-2706
	#96-2713	#96-2713	#96-2713	#96-2713	#96-2721	#96-2721	#96-2721	#96-2721	#96-2738	#96-2739
	#98-2788	#98-2789	#100-2823	#100-2828	#100-2830	#101-2886	#101-2886	#101-2893	#101-2893	#101-2893
	#101-2893	#101-2901	#101-2903	#101-2904	#101-2905	#101-2906	#101-2921	#101-2922	#101-2932	#101-2945
	#103-2971	#103-2971	#103-2971	#103-2971	#103-2975	#103-2995	#103-2996	#103-3002	#103-3015	#103-3016
	#103-3052	#103-3082	#105-3099	#105-3099	#105-3099	#105-3099	#105-3101	#105-3112	#105-3113	#105-3121
	#105-3124	#107-3136	#107-3136	#107-3136	#107-3136	#107-3141	#107-3162	#109-3172	#109-3172	#109-3172
	#109-3172	#109-3180	#109-3200	#111-3222	#111-3222	#111-3222	#111-3222	#111-3245	#114-3307	#114-3307
	#114-3310	#114-3310	#114-3310	#114-3310	#114-3310	#114-3310	#114-3312	#114-3319	#116-3324	#116-3324
	#116-3324	#116-3324	#116-3324	#116-3324	#116-3326	#116-3346	#118-3354	#118-3354	#118-3354	#118-3354
	#118-3354	#118-3354	#118-3356	#118-3388	#120-3394	#120-3394	#120-3394	#120-3394	#120-3394	#120-3394
	#120-3396	#120-3420	#122-3425	#122-3425	#122-3425	#122-3425	#122-3425	#122-3425	#122-3427	#122-3451
	#124-3456	#124-3456	#124-3456	#124-3456	#124-3456	#124-3456	#124-3458	#124-3482	#126-3487	#126-3487
	#126-3487	#126-3487	#126-3487	#126-3487	#126-3489	#126-3513	#132-3622	#132-3622	#132-3633	#132-3633
	#132-3633	#132-3633	#134-3677	#134-3677	#134-3677	#134-3677	#136-3716	#136-3716	#136-3717	#136-3717
	#136-3717	#136-3717								
MSLDRO	#84-2483	84-2483	#94-2663	94-2663	#94-2704	94-2704	#94-2706	94-2706	#103-2975	103-2975
	#103-3002	103-3002	#105-3112	105-3112	#105-3124	105-3124				
MSMCHI	#5-375	#5-375								
MSMCLD	#5-375	#5-375								
MSPOP	#7-432	7-432	#11-479	11-479	#13-505	13-505	#13-507	13-507	#86-2513	86-2513
	#96-2717	96-2717	#96-2730	96-2730	#100-2842	100-2842	#101-2945	101-2945	#101-2961	101-2961
	#103-3082	103-3082	#105-3113	105-3113	#107-3162	107-3162	#109-3200	109-3200	#111-3245	111-3245
	#114-3319	114-3319	#116-3346	116-3346	#118-3388	118-3388	#120-3420	120-3420	#122-3451	122-3451
	#124-3482	124-3482	#126-3513	126-3513	#130-3557	130-3557	#132-3648	132-3648	#134-3693	134-3693
	#136-3714	136-3714								
MSPRIN	#80-2250	#80-2250	#80-2255	#80-2255	#80-2258	#80-2258	#80-2260	#80-2260	#80-2274	#80-2274
	#80-2281	#80-2281	#80-2289	#80-2289	#86-2503	#86-2503	#86-2506	#86-2506	#86-2509	#86-2509
	#98-2789	#98-2789	#100-2823	#100-2823	#100-2828	#100-2828	#100-2830	#100-2830	#101-2903	#101-2903
	#101-2905	#101-2905	#101-2921	#101-2921	#101-2922	#101-2922	#101-2932	#101-2932	#105-3121	#105-3121
	#109-3180	#109-3180								
MSPUSH	#5-401	#5-401	#7-428	#7-428	#11-466	#11-466	#13-489	#13-489	#14-560	#14-560
	#86-2496	#86-2496	#96-2713	#96-2713	#96-2721	#96-2721	#101-2886	#101-2886	#101-2893	#101-2893
	#103-2971	#103-2971	#105-3099	#105-3099	#107-3136	#107-3136	#109-3172	#109-3172	#111-3222	#111-3222
	#114-3307	#114-3307	#114-3310	#114-3310	#116-3324	#116-3324	#118-3354	#118-3354	#120-3394	#120-3394
	#122-3425	#122-3425	#124-3456	#124-3456	#126-3487	#126-3487	#132-3622	#132-3622	#132-3633	#132-3633
	#134-3677	#134-3677								
MSPUT	#80-2250	#80-2250	#80-2250	#80-2250	#80-2255	#80-2255	#80-2255	#80-2255	#80-2258	#80-2258
	#80-2258	#80-2260	#80-2260	#80-2260	#80-2274	#80-2274	#80-2274	#80-2281	#80-2281	#80-2281
	#80-2281	#80-2289	#80-2289	#80-2289	#86-2503	#86-2503	#86-2503	#86-2503	#86-2503	#86-2503
	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2506	#86-2509	#86-2509	#86-2509
	#86-2509	#94-2664	#94-2664	#94-2664	#94-2664	#94-2664	#94-2666	#94-2666	#94-2666	#94-2666
	#94-2666	#98-2789	#98-2789	#98-2789	#98-2789	#98-2789	#100-2823	#100-2823	#100-2823	#100-2828
	#100-2828	#100-2828	#100-2830	#100-2830	#100-2830	#101-2903	#101-2903	#101-2903	#101-2905	#101-2905
	#101-2905	#101-2921	#101-2921	#101-2921	#101-2921	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922
	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2922	#101-2932	#101-2932	#101-2932	#101-2932
	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#101-2932	#105-3101	#105-3101	#105-3101

REFERENCES

MSWORD

POINTE
POPPRINTB
PRINTF
PRINTS
PRINTX
PUSHREADEF
RFLAGS
SETPRI
SETVEC
SVC
SWAPIN
SWAPOW
TSTID
TUREAD
TURTRY
TUSEEK
TUSELF
TUWRIT
XFER

64-1768	#74-2046	74-2046	#80-2250	80-2250	#80-2255	80-2255	#80-2258	80-2258	#80-2260
80-2260	#80-2274	80-2274	#80-2281	80-2281	#80-2289	80-2289	#84-2437	#84-2437	84-2437
#84-2451	#84-2451	84-2451	#84-2457	#84-2457	84-2457	#84-2464	#84-2464	84-2464	#84-2475
#84-2475	84-2475	#84-2479	#84-2479	84-2479	#84-2483	84-2483	#86-2503	86-2503	#86-2506
86-2506	#86-2509	86-2509	#86-2513	86-2513	#94-2648	94-2648	#94-2663	94-2663	#94-2664
94-2664	#94-2666	94-2666	#94-2704	94-2704	#94-2706	94-2706	#96-2738	96-2738	#96-2739
96-2739	#98-2788	#98-2788	98-2788	#98-2789	98-2789	#100-2823	100-2823	#100-2828	100-2828
#100-2830	100-2830	#101-2901	101-2901	#101-2903	101-2903	#101-2904	101-2904	#101-2905	101-2905
#101-2906	101-2906	#101-2921	101-2921	#101-2922	101-2922	#101-2932	101-2932	#101-2945	101-2945
#103-2975	103-2975	#103-2995	#103-2995	103-2995	#103-2996	103-2996	#103-3002	103-3002	#103-3015
#103-3015	103-3015	#103-3016	103-3016	#103-3052	103-3052	#103-3082	103-3082	#105-3101	105-3101
#105-3112	105-3112	#105-3113	105-3113	#105-3121	105-3121	#105-3124	105-3124	#107-3141	107-3141
#107-3162	107-3162	#109-3180	109-3180	#109-3200	109-3200	#111-3245	111-3245	#114-3312	114-3312
#114-3319	114-3319	#116-3326	116-3326	#116-3346	116-3346	#118-3356	118-3356	#118-3388	118-3388
#120-3396	120-3396	#120-3420	120-3420	#122-3427	122-3427	#122-3451	122-3451	#124-3458	124-3458
#124-3482	124-3482	#126-3489	126-3489	#126-3513	126-3513				
#5-418	#5-418	#9-448	#9-448	#9-448	#9-448	#9-448	#9-448	#9-448	#9-448
#9-448	#60-1650	#60-1650	#60-1650	#60-1650	#84-2437	#84-2437	#84-2437	#84-2437	#84-2451
#84-2451	#84-2451	#84-2451	#84-2457	#84-2457	#84-2457	#84-2457	#84-2464	#84-2464	#84-2464
#84-2464	#84-2475	#84-2475	#84-2475	#84-2475	#84-2479	#84-2479	#84-2479	#84-2479	#98-2788
#98-2788	#98-2788	#98-2788	#103-2995	#103-2995	#103-2995	#103-2995	#103-3015	#103-3015	#103-3015
#103-3015	#114-3312	#116-3326	#118-3356	#120-3396	#122-3427	#124-3458	#126-3489	#132-3636	#132-3636
#132-3637	#132-3637	#132-3638	#132-3638	#132-3639	#132-3639	#132-3640	#132-3640	#134-3679	#134-3679
#134-3680	#134-3680	#134-3681	#134-3681	#134-3682	#134-3682	#134-3683	#134-3683	#134-3685	#134-3685
#136-3717	#136-3717								
5-408									
#34-993	54-1547	54-1548	64-1769	64-1777	72-1983	72-1984	80-2301	80-2302	84-2484
84-2485	84-2486	84-2487	86-2511	86-2512	90-2577	90-2578	92-2615	92-2616	98-2793
98-2794	98-2795	100-2831	100-2832	101-2939	101-2940	101-2941	101-2942	101-2943	101-2944
109-317	109-3179								
#80-2274	#86-2503	#86-2506	#86-2509	#98-2789					
100-2823	100-2828	100-2830	105-3121	109-3180					
#101-2903	#101-2905	#101-2921	#101-2922	#101-2932					
80-2250	80-2255	80-2258	80-2260	80-2281	80-2289				
#34-989	54-1540	54-1541	64-1763	64-1767	72-1972	72-1973	80-2228	80-2229	84-2421
84-2422	84-2423	84-2424	86-2497	86-2498	90-2553	90-2554	92-2593	92-2594	98-2767
98-2768	98-2769	100-2816	100-2817	101-2894	101-2895	101-2896	101-2897	101-2898	101-2899
109-3174	109-3175								
#103-2975									
#103-3052									
94-2663									
94-2664	94-2666	105-3101							
#5-374	5-375								
#34-1002	60-1618	60-1639							
#34-1015	#62-1695								
#46-1364	#114-3311	#116-3325	#118-3355	#120-3395	#122-3426	#124-3457	#	7488	
#42-1262	118-3364	122-3437	126-3499						
#40-1195	118-3363	118-3364	120-3406	122-3437	124-3468	126-3499			
#38-1143	116-3332								
#44-1331	114-3314								
#36-1048	#118-3363	#120-3406	#124-3468						
#114-3312	#116-3326	#118-3356	#120-3396	#122-3427	#124-3458	#126-489			