

2

MODEL TC-120/128 TAPE CONTROLLER
DIAGNOSTIC MANUAL

PUBLICATION NUMBER

120013 E

western peripherals

14321 MYFORD ROAD
TUSTIN, CALIFORNIA 92680

01200138

© 1980 by Western Peripherals, Inc.
All Rights Reserved

SECTION I

DIAGNOSTIC PROGRAM

```

;*****
; WESTERN PERIPHERALS *
; MAGNETIC TAPE DIAGNOSTIC *
; LISTING NO. AL12009 *
;*****
;
; PLEASE READ
; *****
; THIS DIAGNOSTIC PROGRAM IS PROVIDED
; TO FIND CATASTROPHIC FAILURES THAT ARE
; RELATED TO BASIC OPERATIONS OF THE
; TAPE CONTROL. THE RELIABILITY PRO-
; GRAM IS PROVIDED AS A STRIGENT
; TEST TO SHOW UP INTERMITTENT AND
; PATTERN SENSITIVE PROBLEMS.
; FOR THE MOST PART THE DIAGNOSTIC
; ASSUMES THE MAGNETIC TAPE MEDIA
; IS PERFECT AND NOT THE CAUSE OF AN
; ERROR. HOWEVER, IF PARITY ERRORS
; OCCUR WHILE RUNNING THE DIAGNOSTIC
; THE TAPE SHOULD ALWAYS BE SUSPECT!
;
; *****
; *****
;MAGNETIC TAPE DIAGNOSTIC
;1. PLACE ALL TAPE UNITS OFF LINE EXCEPT
; ONE. FOR A COMPLETE TEST OF THE
; INTERFACE MAKE SUCCESSIVE RUNS OF EACH:
; 7 TRACK, 9 TRACK, 800BPI, 556 BPI, PE
;2. STARTING LOCATIONS
; 400 BASIC DIAGNOSTIC PLUS FOLLOWING 3 ROUTINES:
; 401 WRITE LOCK TEST ONLY
; 402 END TAPE TEST ONLY
; 403 DATA LATE TEST ONLY
;3. SWITCH SETTINGS
; 13-15 UNIT #0-7
; 10-12 DEVICE CODE, 2 OR 6
; 9 7TRK=0 9TRK=1
; 8 LO DENS=0 HI DENS=1(7TRK), PE=1 (NRZ=1(9TRK)
; 7 NRZ=0 PE=1
; 6 TEST WRL, TE, & EOT=0 BYPASS WRL, TE, & EOT=1
; 5 6-6 PACKING 7TRK=0 6-6-4 PACKING 7TRK=1
;
; -
; 2 PRINT FAILURE RATE
; 1 INHIBIT PRINTING
; 0 FROM ERROR, GO TO NEXT TEST
;4. PROCEDURE
;4.1 BASIC DIAGNOSTIC
; LOAD TAPE, PUSH FORWARD TO POSITION AT BOT.
; LOAD PROGRAM WITH BINARY LOADER AND START AT
; PROPER LOCATION FOR THE TEST TO BE RUN. ALL
; TESTS SHOULD BE RUN FOR A COMPLETE CHECK. IF
; A HALT OCCURS CONSULT THE PROGRAM LISTING

```

NOTE - CARTRIDGE TAPE DRIVES - SWITCH 8 ON

```

: FOR DIRECTIONS.
: IMMEDIATELY AFTER STARTING A HALT WILL OCCUR
: TO ALLOW THE SWITCH REGISTER TO BE SET FOR
: PROPER UNIT SELECT, CONTROL, ETC...
: AFTER SETTING SWITCHES PUSH CONTINUE.
: EACH TEST MAY CONTAIN SEVERAL CHECK POINTS
: WHERE AN ERROR HALT CAN OCCUR. GENERALLY
: SPEAKING, ERRORS MAY BE EXPECTED AT CHECK
: POINTS FURTHER ON WHEN ONE ERROR IS FOUND.
: IF A MALFUNCTION IS DETECTED THE PROGRAM WILL
: HALT AND AC3 WILL CONTAIN THE LOCATION OF
: THE ERROR+1. CONSULT THE LISTING AND EXAMINE
: OTHER PERTINENT ACCUMULATORS. IF SWITCHES
: 0&1 ARE DOWN, PRESSING CONTINUE WILL CAUSE A
: PRINTOUT OF THE ERROR LOCATION AND A SCOPE
: LOOP WILL BE ENTERED. WHILE IN THE SCOPE LOOP
: SETTING SWITCH 2 WILL CAUSE THE FAILURE RATE
: (IN%) TO BE PRINTED. SETTING SWITCH 0 WILL
: CAUSE THE PROGRAM TO GO ON TO THE NEXT TEST.
: EACH SUCCESSIVE TEST ASSUMES THAT ALL PREVIOUS
: TESTS WORK CORRECTLY. BYPASSING A FAILING TEST
: AND PROCEEDING ON MAY LEAD TO CONFUSING
: SITUATIONS.
: WHEN A TYPEOUT REQUEST IT, REMOVE THE
: WRITE-ENABLE RING! DON'T STOP THE PROGRAM!
: REMOUNT TAPE AND PUT UNIT BACK ON-LINE.
: WHEN REQUESTED, REPLACE RING THE SAME WAY,
:4.2 WRITE LOCK TEST
: WHEN A TYPEOUT REQUESTS IT, REMOVE WRITE ENABLE RI
: DON'T STOP THE PROGRAM. REMOUNT TAPE
: AND PUT UNIT BACK ON LINE. AN ATTEMPT IS
: MADE TO WRITE, AND STATUS IS THEN CHECKED.
: EXPECTED STATUS IS 110205:
:     ERROR
:     ILLEGAL
:     BOT
:     WRITE LOCK
:     TUR
: IF THIS STATUS IS FOUND
: THE TEST IS REPEATED. IF THE
: ABOVE STATUS IS NOT FOUND AN ERROR HALT
: OCCURS. AC1=EXPECTED STATUS
:     AC0=ACTUAL STATUS
: PUSH CONTINUE TO REPEAT TEST.
:4.3 END TAPE TEST
: 4K RECORDS ARE WRITTEN DOWN
: THE LENGTH OF THE TAPE. WHEN END TAPE STATUS
: IS DETECTED "END TAPE" IS PRINTED AND PROGRAM
: HALTS. PUSHING CONTINUE WILL CAUSE THE
: PROGRAM TO GO ON.
: IF END TAPE IS NOT FOUND STATUS IS CHECKED FOR
:     DATA LATE
:     REWINDING
:     ILLEGAL
:     EOF

```

```

;          BOT
;          BAD TAPE
;          WRITE LOCK
;          IF ANY OF THESE IS FOUND THE STATUS IS PRINTED
;          AND WRITING CONTINUES.
;4.4      DATA LATE TEST
;          FOLLOWING START THE TAPE IS REWOUND. ONE 4K
;          RECORD IS THEN WRITTEN AND THE DATA LATE STATUS
;          CHECKED. IF IT IS NOT FOUND, "NO DATA LATE BIT"
;          IS PRINTED. THE TEST THEN REPEATS.
;          SOME PROCESSORS ARE INCAPABLE OF FORCING A
;          "DATA-LATE" WITHOUT OPERATOR INTERVENTION.
;          TO CAUSE A DATA LATE STATUS TO OCCUR, STOP THE
;          PROGRAM DURING A WRITE OPERATION. THEN PUSH
;          CONTINUE. THE LONG RECORD LENGTH ALLOWS PLENTY
;          OF TIME TO INSURE THE PROGRAM STOP OCCURS DURING
;          DATA CHANNEL TRANSFERS.
;
;          .LOC 0
00000    000000    0          ;STORE PC ON INTERRUPT
00001    000002    2          ;GO TO .+1
00002    010000    ISZ 0      ;INTERRUPT RETURN WITH SKIP
00003    002000    JMP @0
;          .LOC 20
00020    000000    IDX0: 0      ;AUTO INDEX REGISTERS
00021    000000    IDX1: 0
00022    000000    IDX2: 0
00023    000000    IDX3: 0
00024    000000    IDX4: 0
;          .LOC 45
00045    000050    EGGS
;          .LOC 50
00050    000000    EGGS: 0      ;HEN FLAG
00051    000000    0          ;DEVICE CODE
00052    000000    0          ;ASR 37 FLAG
00053    000000    0          ;PASS COUNTER
00054    000000    0          ;SWITCH SETTINGS
;          ;CONSTANTS
00055    000411    FIRST: A0
00056    004333    CEND:  ENDPRG
00057    065404    CMTA:  DIB 1,0
00060    005041    OBUFF:  LAST
00061    005241    IBUFF:  LAST+200
00062    105441    PPCHN:  @LAST+400
00063    007441    PCHIN:  LAST+2400
00064    000000    SAVE@:  0
00065    000000    SAVE1:  0
;          .RDX 10
00066    001750    .1SEC:  1000
00067    011610    .5SEC:  5000
00070    000062    .50MS:  50
00071    000310    .200MS: 200
00072    000620    .400MS: 400
;          .RDX 8
00073    000001    C1:     1
00074    000002    C2:     2

```

00075	000003	C3:	3	
00076	000004	C4:	4	
00077	000005	C5:	5	
00100	000007	C7:	7	
00101	000010	C10:	10	
00102	000020	C20:	20	
00103	000022	C22:	22	
00104	000030	C30:	30	
00105	000040	C40:	40	
00106	000050	C50:	50	
00107	000060	C60:	60	
00110	000070	C70:	70	
00111	000077	C77:	77	
00112	000100	C100:	100	
00113	000103	C103:	103	
00114	000104	C104:	104	
00115	000144	C144:	144	
00116	000150	C150:	150	
00117	000200	C200:	200	
00120	000375	C375:	375	
00121	000377	C377:	377	
00122	000400	C400:	400	
00123	001000	C1000:	1000	
00124	002000	C2000:	2000	
00125	002100	C2100:	2100	
00126	020000	C20K:	20000	
00127	100401	C1004:	100401	
00130	004000	C4K:	4000	
00131	007700	C7700:	7700	
00132	037477	C37477:	37477	
00133	004104	C4104:	4104	
00134	040000	C40000:	40000	
00135	005670	D3000:	5670	:3000.
00136	176000	M2000:	-2000	
00137	000764	D500:	764	:500.
00140	010421	C10421:	10421	
00141	021042	C21042:	21042	
00142	042104	C42104:	42104	
00143	004210	C421T:	4210	
00144	000201	C201:	201	
00145	000000	RECTR:	0	
00146	160077	C160077:		160077
00147	004332	PCDCM:	CDCM	
00150	177700	C177700:		177700
00151	110000	C110000:		110000
00152	125252	C125252:		125252
00153	052525	C52525:	52525	
00154	002010	C2010:	2010	
00155	051642	C5164:	51642	
00156	100000	C1M:	100000	
00157	177776	M2:	-2	
00160	177775	M3:	-3	
00161	177774	M4:	-4	
00162	177773	M5:	-5	
00163	177754	M20.:	-24	

```

00164 177770 M8: -10
00165 177700 M100: -100
00166 177160 M400: -620 ; -400.
00167 177000 M1000: -1000
00170 177776 MSKTO: 177776
00171 000144 DL100: 144
00172 070644 C70644: 70644
;PATTERN TABLE, ODD PARITY WRITES
00173 000173 POINT: .
00174 000000 0
00175 177400 177400
00176 000377 377
00177 125252 125252
00200 052525 52525
00201 000200 LST: .-1
;PATTERN TABLE, EVEN PARITY WRITES
00202 000202 PONTP: .
00203 125253 125253
00204 052525 052525
00205 002021 002021
00206 020050 020050
00207 000206 LST1: .-1
;VARIABLES
00210 000000 X70: 0
00211 000010 C1X: 10 ;UNIT # +10
00212 000000 CX: 0 ;UNIT #
00213 000050 C5X: 50 ;UNIT # +50
00214 000000 CC5X: 0 ;UNIT # +50
00215 000000 CTR: 0 ;COUNTER
00216 177220 DILLY: 0 ;-(# ITR INC-JMP LOOP IN 1
00217 000000 WDCNT: 0 ;TEMP STORAGE
00220 000000 INST: 0 ;TEMP STORAGE
00221 000000 DEVICE: 0 ;DEVICE CODE. 22 OF 62
00222 000000 SWTSV: 0 ;SWITCHES
00223 000000 ITLP: 0 ;ITERATIONS AFTER LP
;INDIRECT ADDRESSES
00224 001231 IA28: A28
00225 001645 IA39: A39
00226 002220 IA46: A46
00227 002373 IA491: A49.1
00230 002375 IA492: A49.2
00231 002433 IA50: A50
00232 003016 IA55: A55
00233 003100 ABTOF: BTOF
00234 005004 ICRLF: CRLF
00235 003632 ICHST: CSTAT
00236 004510 LOOP: CYCLE
00237 003262 PDILL: DILLI
00240 004631 PDIVD: DIVID
00241 003772 PDLTA: DLTA
00242 004466 SETUP: ENTER
00243 004463 SETP1: ENTER-3
00244 004566 ER: ERR
00245 004107 PETTA: ETTA
00246 063077 AHALT: HALT ;PUT PC IN AC3 TO KEEP

```



```

00247 003163 INTIL: INITIAL
00250 004150 PINWL: INWLT
00251 004657 IMESS: MESS
00252 003656 ONLOC: ON+2
00253 004700 IPOCT: POCT
00254 004322 IBPPE: XBPPE
00255 003404 IBSP: XBSP
00256 003307 ICHK: XCHK
00257 003342 ICLR: XCLR
00260 003621 IERAS: XERAS
00261 003453 IGEN: XGEN
00262 003431 ILO: XLD
00263 003510 IRD: XRD
00264 003505 IRDONS: XRDONS
00265 003366 IRWD: XRWD
00266 003371 IRWS: XRWS
00267 003476 ISEL: XSEL
00270 003357 ISPC: XSPC
00271 003701 ISTAT: XSTAT
00272 004230 ITIM: XTIM
00273 004256 ITMD: XTMD
00274 004273 ITPN: XTPN
00275 003413 IWEOF: XWEOF
00276 003553 IWNS: XWNS
00277 003556 IWRT: XWRT
00300 003661 IWT: XWT
00301 001400 JMP 0,3 ;CONSISTANT WITH ERROR
00302 7442 WCPST: WCPB
;DEFINATIONS
000022 .MTA=22
006271 STATUS=JSR @ISTAT
006260 ERASE=JSR @IERAS
006277 WRITE=JSR @IWRT
006261 GEN=JSR @IGEN
006275 WEOF=JSR @IWEOF
006255 BSPACE=JSR @IBSP
006265 REWIND=JSR @IRWD
006266 RWDSTALL=JSR @IRWS
006276 WRTNS=JSR @IWNS
006300 WAIT=JSR @IWT
006270 SPACE=JSR @ISPC
006244 EHALT=JSR @ER
006263 READ=JSR @IRD
006264 REDNS=JSR @IRDONS
006257 CLEAR=JSR @ICLR
006256 CHECK=JSR @ICLK
006262 LOAD=JSR @ILD
004246 ERHLT=JSR AHALT
006267 SELECT=JSR @ISEL
006272 TIME=JSR @ITIM
006273 TMDONE=JSR @ITMD
006274 TESTPL=JSR @ITPN
006254 BYPASSPE=JSR @IBPPE
000400 .LOC 400
00400 002404 JMP @IDIAG ;MTA DIAGNOSTIC

```

```

00401 002404      JMP @IWLT          ;WRITE LOCK TEST
00402 002404      JMP @IETT         ;END TAPE TEST
00403 002404      JMP @IDLTL        ;DATA LATE TEST
00404 000410      IDIAG:  DIAG
00405 004136      IWLT:   WLT
00406 004074      IETT:   ETT
00407 003760      IDLT:   DLT
                   ;CHECK SELB BUS LINE
                   .LOC 410
00410 006247      DIAG:  JSR @INTIL
00411 006242      A0:    JSR @SETUP          ;DO SKPBZ, DEVICE 0
00412 063500      SKPBZ 0
00413 006244      EHALT          ;CHECK THE SELB BUS LINE
00414 006236      JSR @LOOP
                   ;CHECK SELD BUS LINE
00415 006242      A1:    JSR @SETUP          ;DO SKPDZ, DEVICE 0
00416 063700      SKPDZ 0
00417 006244      EHALT          ;CHECK SELD BUS LINE
00420 006236      JSR @LOOP
                   ;CHECK MTA BUSY FLOP OFF
00421 006242      A2:    JSR @SETUP          ;DO BUSY TEST ON
00422 063522      SKPBZ .MTA          ;MTA
00423 006244      EHALT          ;CANNOT RESET MT BUSY
00424 006236      JSR @LOOP
                   ;CHECK MTA DONE FLOP OFF
00425 006242      A3:    JSR @SETUP          ;DONE TEST ON MTA
00426 063722      SKPDZ .MTA
00427 006244      EHALT          ;CANNOT RESET MT DONE
00430 006236      JSR @LOOP
                   ;CHECK MT SELECT, DOR, DIB, ADDR REG. IN-OUT
00431 006242      A4:    JSR @SETUP          ;LOAD CA WITH 125252
00432 020152      LDA 0,C125252      ;READ BACK AND CHECK
00433 062022      DOB 0,.MTA
00434 065422      DIB 1,.MTA
00435 101120      MOVZL 0,0
00436 101220      MOVZR 0,0
00437 106414      SUB# 0,1,SZR
00440 006244      EHALT          ;AC0=VALUE SENT TO ADDR REG
                   ;AC1=VALUE READ BACK
                   ;CHECK THE MT SELECT, LOAD ADDR
                   ;ENABLE ADDR, AND ADDR REG.
00441 006236      JSR @LOOP
                   ;CHECK CA REGISTER IN-OUT
00442 006242      A5:    JSR @SETUP          ;LOAD CA WITH 052525
00443 020153      LDA 0,C52525      ;READ BACK AND CHECK
00444 062022      DOB 0,.MTA
00445 065422      DIB 1,.MTA
00446 106414      SUB# 0,1,SZR
00447 006244      EHALT          ;AC0=VALUE SENT TO ADDR REG
                   ;AC1=VALUE READ BACK
                   ;CHECK ADDR REG DATA PATHS
00450 006236      JSR @LOOP
                   ;CHECK FOR ILLEGAL MT SELECT
00451 020057      A6:    LDA 0,CMTA          ;LOAD CA WITH 052525
00452 040404      STA 0,A6,2          ;READ BACK USING EVERY

```

```

00453 006242 A6.1: JSR @SETUP ;DEVICE CODE EXCEPT MTA
00454 020153 LDA 0,C52525 ;DATA SHOULD NOT SE
00455 062022 DOB 0,.MTA ;FOUND!
00456 065400 A6.2: DIB 1,0 ;THIS IS CHANGED BY PROG
00457 122415 SUB# 1,0,SNR
00460 006244 EHALT ;AC0=DATA SENT TO ADDR REG
;AC1=DATA READ BACK
;PROBLEM IS IN MT SELECT
;THE DEVICE SELECT GATE.

00461 006236 JSR @LOOP
00462 020774 LDA 0,A6.2
00463 101400 A6.3: INC 0,0
00464 024111 LDA 1,C77
00465 107405 AND 0,1,SNR
00466 000406 JMP A7 ;DONE ALL GO ON
00467 030221 LDA 2,DEVICE
00470 146415 SUB# 2,1,SNR
00471 000772 JMP A6.3 ;MTA, INC AGAIN
00472 040764 STA 0,A6.2
00473 000760 JMP A6.1
;CHECK CA REGISTER, ALL DATA PATTERNS

00474 102400 A7: SUB 0,0 ;LOAD CA, READ BACK
00475 006243 JSR @SETP1 ;AND CHECK, USE
00476 062022 DOB 0,.MTA ;ALL FROM 0-077777
00477 065422 DIB 1,.MTA
00500 106414 SUB# 0,1,SZR
00501 006244 EHALT ;AC0=DATA SENT TO ADDR REG
;AC1=DATA READ BACK
;ADDR REG IS PATTERN SENSITIVE

00502 006236 JSR @LOOP
00503 101400 INC 0,0
00504 126620 SUBZR 1,1
00505 122414 SUB# 1,0,SZR
00506 000767 JMP A7+1
;CHECK TO RESET OF CA REGISTER

00507 006242 A8: JSR @SETUP ;LOAD CA WITH ALL 1'S
00510 102000 ADC 0,0 ;DO IORST, READ
00511 062022 DOB 0,.MTA ;CA AND CHECK FOR 0'S
00512 062677 IORST
00513 065422 DIB 1,.MTA
00514 125004 MOV 1,1,SZR
00515 006244 EHALT ;AC0=DATA SENT TO ADDR REG
;AC1=DATA READ BACK
;CHECK IORST, RESET, AND
;ADDRESS REGISTER RESET GATES

00516 006236 JSR @LOOP
;CHECK CA LOAD

00517 006242 A8A: JSR @SETUP ;LOAD CA WITH 010421
00520 102400 SUB 0,0 ;LOAD CA WITH ALL ZEROS
00521 024140 LDA 1,C10421 ;READ AND CHECK
00522 066022 DOB 1,.MTA
00523 062022 DOB 0,.MTA
00524 061422 DIB 0,.MTA
00525 101004 MOV 0,0,SZR
00526 006244 EHALT ;CA LOADING ERROR

```

```

00527 006236      JSR @LOOP
00530 006242  A88: JSR @SETUP      ;ACO=(CA)
00531 102400      SUB 0,0          ;(CA) SHOULD BE ZERO
00532 024141      LDA 1,C21042    ;LOAD CA WITH 21042
00533 066022      DOB 1,MTA      ;LOAD CA WITH ALL ZEROS
00534 062022      DOB 0,MTA      ;READ AND CHECK
00535 061422      DIB 0,MTA
00536 101004      MOV 0,0,SZR
00537 006244      EHALT      ;CA LOADING ERROR
                                ;ACO=(CA)
                                ;(CA) SHOULD BE ZERO

00540 006236      JSR @LOOP
00541 006242  ABC: JSR @SETUP      ;LOAD CA WITH 42104
00542 102400      SUB 0,0          ;LOAD CA WITH ALL ZEROS
00543 024142      LDA 1,C42104
00544 066022      DOB 1,MTA
00545 062022      DOB 0,MTA
00546 061422      DIB 0,MTA
00547 101004      MOV 0,0,SZR
00550 006244      EHALT      ;CA LOADING ERROR
                                ;ACO=(CA)
                                ;(CA) SHOULD BE ZERO

00551 006236      JSR @LOOP
00552 006242  ABD: JSR @SETUP      ;LOAD CA WITH 4210
00553 102400      SUB 0,0          ;LOAD CA WITH ALL ZEROS
00554 024143      LDA 1,C421T   ;READ AND CHECK
00555 066022      DOB 1,MTA
00556 062022      DOB 0,MTA
00557 061422      DIB 0,MTA
00560 101004      MOV 0,0,SZR
00561 006244      EHALT      ;CA LOADING ERROR
                                ;ACO=(CA)
                                ;(CA) SHOULD BE ZERO

00562 006236      JSR @LOOP
                                ;CHECK CA LOAD
00563 006242  ABE: JSR @SETUP
00564 020140      LDA 0,C10421
00565 024141      LDA 1,C21042
00566 030142      LDA 2,C42104
00567 034143      LDA 3,C421T
00570 062022      DOB 0,MTA
00571 066022      DOB 1,MTA
00572 072022      DOB 2,MTA
00573 076022      DOB 3,MTA
00574 061422      DIB 0,MTA
00575 162414      SUB# 3,0,SZR
00576 006244      EHALT      ;CA LOADING ERROR
                                ;ACO=(CA)
                                ;(CA) SHOULD BE 004210

00577 006236      JSR @LOOP
                                ;CHECK MT START AND BUSY FLOP
00600 006242  A9: JSR @SETUP      ;SEND START PULSE
00601 006267      SELECT

```

```

00602 060122      NIOS .MTA      ;SKIP OF BUSY SET
00603 063422      SKPBN .MTA
00604 006244      EHALT          ;MT START DID NOT SET BUSY
                                ;CHECK MT START, MT BUSY FLOP
                                ;AND SET LINE

00605 006236      JSR @LOOP
                                ;CHECK IORST OF BUSY FLOP (CLEAR COM)
00606 006242      A10: JSR @SETUP      ;SEND START PULSE
00607 006267      SELECT
00610 060122      NIOS .MTA      ;MAKE SURE BUSY IS ON
00611 063422      SKPBN .MTA      ;DO IORST AND
                                ;MAKE SURE BUSY OFF.
00612 006244      EHALT          ;START PULSE DIDN'T SET BUSY
                                ;THIS JUST WORKED IN THE
                                ;PREVIOUS TEST!

00613 062677      IORST
00614 063522      SKPBZ .MTA
00615 006244      EHALT          ;IORST DIDN'T RESET BUSY
                                ;CHECK CLEAR COM AND
                                ;MT BUSY RESET LINE

00616 006236      JSR @LOOP
                                ;CHECK CLEAR PULSE
00617 006242      A11: JSR @SETUP      ;SEND START PULSE
00620 006267      SELECT
00621 060122      NIOS .MTA      ;MAKE SURE BUSY IS ON
00622 063422      SKPBN .MTA      ;SEND CLEAR PULSE
                                ;MAKE SURE BUSY RESETS
00623 006244      EHALT          ;MT START DIDN'T SET MT BUSY
                                ;THIS HAS WORKED TWICE BEFORE!

00624 060222      NIOC .MTA
00625 063522      SKPBZ .MTA
00626 006244      EHALT          ;MT CLR DIDN'T RESET BUSY
                                ;CHECK MT CLR THRU CLEAR COM

00627 006236      JSR @LOOP
                                ;CHECK DOA-MTLC-COMMAND DECODES
00630 006242      A12: JSR @SETUP      ;SELECT REWIND
00631 006267      SELECT
00632 020211      LDA 0,C1X      ;SEND START PULSE
00633 061122      DOAS 0,.MTA    ;MAKE SURE BUSY DEWSN'T SET
00634 063522      SKPBZ .MTA
00635 006244      EHALT          ;CHECK GATE DRIVING THE
                                ;MT BUSY SET LINE. REWIND IS ON
                                ;AND MT START IS SETTING BUSY.
                                ;CHECK COMMAND DECODER.

00636 006236      JSR @LOOP
00637 062677      IORST
00640 006300      WAIT
00641 000070      .50MS
                                ;CHECK DIA AND STATUS REGISTER AT BOT
00642 006243      A13: JSR @SETP1    ;REWIND TAPE
00643 006267      SELECT
00644 006266      RWDSTALL      ;WAIT 5 SEC.
00645 006271      STATUS        ;READ AND CHECK STATUS
00646 100201      100201
00647 006244      EHALT          ;AC1=EXPECTED STATUS. ACO=BAD.

```

;THIS IS THE FIRST READ STATUS.
 ;CHECK RD STATUS. REWIND THRU TO
 ;TAPE, AND BOT LINE FORM TAPE.
 ;ALSO BOT STATUS REG. GATE

```

00650 006236      JSR @LOOP
                  ;CHECK UNIT SELECT LOGIC, ERASE COMMAND DECODE
00651 020110  A14: LDA 0,C70      ;SELECT EACH UNIT
00652 040210      STA 0,X70      ;IN SEQUENCE 0-7
00653 006242      JSR @SETUP    ;CHECK FOR TUR IF
00654 020210  A14.1: LDA 0,X70      ;UNIT IS SELECTED.
00655 061022      DOA 0,,MTA    ;OTHERWISE CHECK FOR
00656 006235      JSR @ICHST    ;NO TUR,
00657 006244      EHALT        ;AC1=EXPECTED STATUS, ACO=BAD
                              ;AC2 13-15=UNIT JUST TESTED
                              ;IF UNIT TESTED=UNIT SELECTED
                              ;PROCEED ON THE STATUS INFO.
                              ;IF NOT CHECK FOR ERRONEOUS TUR.

00660 006236      JSR @LOOP
00661 010210      ISZ X70
00662 020210      LDA 0,X70
00663 024112      LDA 1,C100
00664 122414      SUB# 1,0,SZR
00665 000767      JMP A14.1
                  ;CHECK REWIND-ERASE-REWIND SEQUENCE
00666 006243  A15: JSR @SETP1    ;REWIND WAIT 5 SEC.
00667 006266      RWDSTALL      ;ERASE WAIT 1 SEC.
00670 006260      ERASE         ;CHECK STATUS (TUR)
00671 006274      TESTPE
00672 006271      STATUS        ;REWIND-CHK STATUS
00673 000001      1
00674 006244      EHALT        ;AC1=EXPECTED STATUS, ACO=BAD IF
                              ;(BOT), ERASE DIDN'T MOVE TAPE.
                              ;CHECK COMMAND THRU TO TAPE.

00675 006266      RWDSTALL
00676 006271      STATUS
00677 100201      100201
00700 006244      EHALT        ;AC1=EXPECTED STATUS, ACO=BAD.
                              ;REWIND FOLLOWING ERASE DIDN'T
                              ;WORK. THE CLUE IS IN THE BAD
                              ;STATUS.

00701 006236      JSR @LOOP
                  ;CHECK REWINDING STATUS BIT
00702 006243  A16: JSR @SETP1    ;DO ERASE-REWIND
00703 006260      ERASE         ;CHECK STATUS
00704 006265      REWIND        ;DURING REWIND
00705 030164      LDA 2,M8      ;DELAY 8 CYCLES
00706 050215      STA 2,CTR
00707 010215      ISZ CTR
00710 000777      JMP .-1
00711 024126      LDA 1,C20K
00712 060422      DIA 0,,MTA
00713 123405      AND 1,0,SNR
00714 006244      EHALT        ;AC1=EXPECTED STATUS, ACO=BAD
                              ;CHECK REWINDING STATUS BIT GATE

00715 006266      RWDSTALL

```

```

00716 006236      JSR @LOOP
                  ;CHECK SEND CLOCK STATUS BIT
00717 006243  A17: JSR @SETP1      ;REWIND-WRITE
00720 006266      RWDSTALL      ;CHECK STATUS, SHOULD
00721 006276      WRNS          ;GET SEND CLOCK FROM
00722 000102      102          ;WRITE START DELAY.
00723 006271      STATUS
00724 100220      100220
00725 006244      EHALT          ;AC1=EXPECTED STATUS, ACO=BAD
                                  ;CHECK WRITE, START DLY, AND
                                  ;STATUS REG. GATE.

00726 006273      TMDONE
00727 002000  2000
00730 006236      JSR @LOOP
                  ;CHECK 1ST CHAR. STATUS
00731 006243  A18: JSR @SETP1      ;REWIND-ERASE-WRITE
00732 006266      RWDSTALL
00733 006260      ERASE
00734 006276      WRNS          ;WATCH SEND CLOCK ON,
00735 000102      102          ;THEN OFF. THEN WATCH
                                  ;FOR 1ST CHAR

00736 006271      STATUS
00737 000020      20
00740 006244      EHALT          ;AC1=EXPECTED STATUS, ACO=BAD
                                  ;LOOKING FOR SEND CLOCK.
                                  ;THIS MUCH WORKED IN THE
                                  ;PREVIOUS TEST

00741 152620      SUBZR 2,2
00742 050215      STA 2,CTR
00743 006271  A18.1: STATUS
00744 000020      20
00745 000404      JMP .+4        ;STATUS JUST CHANGED
00746 010215      ISZ CTR
00747 000774      JMP A18.1
00750 006244      EHALT          ;SEND CLOCK ON TOO LONG.
                                  ;CHECK WRITE START DELAY
                                  ;AT BOT.

00751 152620      SUBZR 2,2
00752 050215      STA 2,CTR
00753 030101      LDA 2,C10
00754 060422  A18.2: DIA 0,MTA
00755 143404      AND 2,0,SZR
00756 000404      JMP .+4
00757 010215      ISZ CTR
00760 000774      JMP A18.2
00761 006244      EHALT          ;TIME OUT WAITING FOR 1ST CHAR.
                                  ;CHECK 1ST CHAR FF THRU TO
                                  ;THE STATUS GATE.

00762 006273      TMDONE
00763 000400      400
00764 006236      JSR @LOOP
                  ;CHECK FIRST DATA CHANNEL OUTPUT
00765 006242  A19: JSR @SETUP      ;WRITE 2WD
00766 006261      GEN          ;WDS=25052 EACH
00767 025052  A19.1: 25052      ;READ DATA REG.

```



```

;PARITY IS ALWAYS 1.
01031 006236      JSR @LOOP
;CHECK INTERRUPT AND DISABLE
01032 006243  A21: JSR @SETP1      ;WRITE 2 WDS
;CHECK DONE
01033 006277      WRITE          ;ALLOW INTERRUPT
01034 000102      102            ;DISABLE INTERRUPT
01035 063622      SKPDN .MTA      ;ALLOW INT. AGAIN
01036 006244      EHALT          ;NO DONE FLAG! THIS HAS WORKED
;BEFORE! RESTART PROGRAM.

01037 060177      INTEN
01040 000401      JMP .+1          ;ALLOW 1 INST BEFORE INT.
01041 006244      EHALT          ;NO INTERRUPT. CHECK MT INT REQ
;INTR GATE, INTP IN GATE.

01042 061477      INTA 0
01043 024221      LDA 1,DEVICE
01044 106414      SUB# 0,1,SZR
01045 006244      EHALT          ;WRONG ADDRESS FROM MTA INT.
;CHECK INTA AND GATES
;AC1=EXPECTED ADDRESS, AC0=BAD

01046 024105      LDA 1,C40
01047 066077      MSKO 1          ;DISABLE MTA INT
01050 060177      INTEN
01051 000401      JMP .+1
01052 000402      JMP .+2
01053 006244      EHALT          ;ILLEGAL INTERRUPT! CHECK
;MT INT DISABLE FLOP, AND
;CONTROL GATES

01054 060277      INTDS
01055 006236      JSR @LOOP
;WRITE 16 CHAR RECORD. ODD PARITY
;PATTERN IN A22.2

01056 102000  A22:  ADC 0,0
01057 040405      STA 0,A22.2
01060 020173      LDA 0,POINT
01061 040021      STA 0,IDX1
01062 006243  A22.1: JSR @SETP1      ;WRITE
;CHECK DONE, CA
;STATUS.

01063 006261      GEN
01064 177777  A22.2: 177777
01065 006277      WRITE
01066 000110      110
01067 063622      SKPDN .MTA
01070 006244      EHALT          ;NO DONE FLAG! RECORDS TO NOW
;HAVE BEEN 4 CHAR. THIS ONE IS
;16 CHAR. THIS IS THE ONLY
;DIFFERENCE. READ AFTER WRITE
;DETECTS ERF,AND ERF-LOAD DELAY-
;DLY OVER-STOP-TUR-MT DONE.
;CHECK WD OVER, INCREMENT WC

01071 024060      LDA 1,OBUFF
01072 030101      LDA 2,C10
01073 147000      ADD 2,1
01074 061422      DIB 0,.MTA
01075 106414      SUB# 0,1,SZR

```

```

01076 006244      EHALT      ;CA WRONG. AC1=CORRECT
;                ;ACQ=BAD. CHECK DATA
;                ;CHAN CONTRCL AND INCREMENT
;                ;WC. 10 OCTAL WORDS WRITTEN

01077 006271      STATUS
01100 000001      1
01101 006244      EHALT      ;AC1=EXPECTED STATUS. ACO=BAD
;                ;THE CLUE IS IN THE BAD STATUS
;                ;EXPECT BAD STATUS IF ERRORS
;                ;OCCURRED PREVIOUSLY HERE.

01102 006236      JSR @LOOP
01103 020021      LDA 0,IDX1
01104 024201      LDA 1,LST
01105 106405      SUB 0,1,SNR
01106 000404      JMP A23      ;GO TO NEXT TEST
01107 022021      LDA 0,@IDX1
01110 040754      STA 0,A22.2    ;SET NEW DATA PATTERN
01111 000751      JMP A22.1
;                ;WRITE 10 CHAR RECORD. EVEN PARITY (NRZ)
;                ;PATTERN IS IN A23.2

01112 102000      A23:      ADC 0,0
01113 040405      STA 0,A23.2
01114 020202      LDA 0,PONTP
01115 040021      STA 0,IDX1
01116 006243      A23.1:    JSR @SETP1    ;WRITE
;                ;CHECK DONE. CA
;                ;STATUS

01117 006261      GEN
01120 177777      A23.2:    177777
01121 006277      WRITE
01122 100110      100110
01123 063622      SKPDN .MTA
01124 006244      EHALT      ;NO DONE FLAG! THIS HAS WORKED
;                ;MANY TIMES BEFORE IN ODD PAR.
;                ;CHECK READ AFTER WRITE ERF
;                ;DETECTION.

01125 024060      LDA 1,OBUFF
01126 030101      LDA 2,C10
01127 147000      ADD 2,1
01130 061422      DIB 0,.MTA
01131 106414      SUB# 0,1,SZR
01132 006244      EHALT      ;ADDR. REG. WRONG. AC1=CORRECT
;                ;ACO=BAD. CHECK DATA CHAN REQ.
;                ;AND INCREMENT WC.

01133 006271      STATUS
01134 000001      1
01135 006244      EHALT      ;AC1=CORRECT STATUS. ACO=BAD
;                ;PARITY ERRORS ARE LIKELY DUE
;                ;TO THE PARITY CHECKING LOGIC
;                ;(EVEN PARITY)

01136 006236      JSR @LOOP
01137 020021      LDA 0,IDX1
01140 024207      LDA 1,LST1
01141 106405      SUB 0,1,SNR
01142 000404      JMP A24
01143 022021      LDA 0,@IDX1

```

```

01144 040754      STA 0,A23.2
01145 000751      JMP A23.1
                    ;TEST ILLEGAL STATUS
01146 034222  A24:  LDA 3,SWTSV      ;GIVE START TO UNSELECTED
01147 030075      LDA 2,C3        ;TAPE
01150 024076      LDA 1,C4        ;AC2=UNIT
01151 167400      AND 3,1
01152 175400      INC 3,3
01153 173400      AND 3,2
01154 133000      ADD 1,2
01155 006242      JSR @SETUP
01156 071022      DOA 2,.MTA      ;START READ
01157 060122      NIOS .MTA
01160 060422      DIA 0,.MTA      ;READ STATUS
01161 024151      LDA 1,C110000
01162 123400      AND 1,0
01163 122414      SUB# 1,0,SZR
01164 006244      EHALT
                    ;AC1=THE 2 BITS EXPECTED
                    ;AC0=THE 2 BITS FOUND
                    ;LOOKING FOR ERROR AND ILLEGAL
                    ;STATUS. CHECK ILLEGAL AND SET
                    ;GATES. ALSO STATUS BIT

01165 006236      JSR @LOOP
                    ;TEST ILLEGAL STATUS
01166 006243  A25:  JSR @SETP1      ;REWIND. GIVE
01167 006266      RWDSTALL      ;SPACE BACK AT BOT
01170 006255      BSPACE
01171 006271      STATUS
01172 110201      110201
01173 006244      EHALT
                    ;AC1=EXPECTED STATUS. AC0=BAD.
                    ;CHECK SPACE BACKBOT GATE ON
                    ;ILLEGAL. CHECK SPACE SKD
                    ;COMMAND DECODE

01174 006236      JSR @LOOP
                    ;CHECK WRITE EOF STATUS
01175 006242  A26:  JSR @SETUP      ;WEOF
                    ;AND CHECK STATUS

01176 006275      WEOF
01177 063622      SKPDN .MTA
01200 006244      EHALT
                    ;NO DONE FLAG. CHECK READ
                    ;CIRCUITS. EOF CODE,"1ST CHAR

01201 006274      TESTPE
01202 006271      STATUS
01203 100401      100401
01204 006244      EHALT
                    ;AC1=EXPECTED STATUS. AC0=BAD
                    ;CHECKING FOR EOF,ERROR.
                    ;CHECK EOF FLIP FLOP.

01205 006236      JSR @LOOP
                    ;CHECK BAD TAPE STATUS (NRZ ONLY)
01206 006254  A27:  BYPASSPE
01207 002224      JMP @IA28
01210 006243      JSR @SETP1      ;WRITE 16 CHAR RECORD
01211 024233      LDA 1,ABTOF      ;IN EVEN PARITY WITH
01212 066022      DOB 1,.MTA      ;3 ALL ZERO CHAR
01213 024164      LDA 1,M8

```

```

01214 067022      DOC 1,.MTA
01215 024212      LDA 1,CX
01216 020116      LDA 0,C150
01217 123000      ADD 1,0
01220 061122      DOAS 0,.MTA
01221 006273      TMDONE
01222 000400      400
01223 063622      SKPDN .MTA
01224 006244      EHALT
                                ;NO DONE FLAG. THIS HAS
                                ;WORKED MANY TIMES BEFORE!

01225 006271      STATUS
01226 100041      100041
01227 006244      EHALT
                                ;AC1=EXPECTED STATUS, ACO=BAD!
                                ;EXPECTING BAD TAPE STATUS.
                                ;IF NONE CHECK BAD TAPE FLOP!
                                ;A 16 CHAR REC. IS WR. IN EVEN
                                ;PARITY! CHAR 1-8 EACH CONTAIN
                                ;1 BIT: 9,10,11, ALL ZEROS CHAR
                                ;12-16 EACH CONTAIN 1 BIT AGAIN.

01230 006236      JSR @LOOP
                                ;CHECK BACKSPACE
01231 006243      A28: JSR @SETP1      ;REWIND-WRITE 1 REC.
01232 006266      RWDSTALL      ;BACKSPACE AND CHECK
01233 006277      WRITE          ;DONE AND DESELECT
01234 000102      102            ;STATUS
01235 006255      BSPACE
01236 006273      TMDONE
01237 000400      400
01240 063622      SKPDN .MTA
01241 006244      EHALT
                                ;NO DONE FLAG FOLLOWING
                                ;BACKSPACE. CHECK SPACE
                                ;COMMAND DECODE AND DRIVER
                                ;TO TAPE.

01242 006271      STATUS
01243 000001      1
01244 006244      EHALT
                                ;AC1=EXPECTED STATUS, ACO=BAD
01245 006236      JSR @LOOP
                                ;CHECK BACKSPACE
01246 006243      A30: JSR @SETP1      ;REWIND-CHECK FOR BOT
01247 006266      RWDSTALL      ;WRITE 2 REC. CHECK
01250 006271      STATUS          ;STATUS, BACKSPACE 2
01251 100201      100201        ;CHECK STATUS AT BOT
01252 006244      EHALT
                                ;AC1=EXPECTED STATUS, ACO=BAD.
                                ;LOOKING FOR BOT FOLLOWING
                                ;REWIND. THIS SHOULD WORK!

01253 006277      WRITE
01254 000202      202
01255 102120      ADCZL 0,0
01256 063022      DOC 0,.MTA
01257 020105      LDA 0,C40
01260 024212      LDA 1,CX
01261 123000      ADD 1,0
01262 024120      LDA 1,C375
01263 066022      DCB 1,.MTA
                                ;375 TO CA
01264 061122      DOAS 0,.MTA      ;BACKSPACE 2

```

```

01265 006273      TMDONE
01266 001000      1000
01267 006271      STATUS
01270 000001      1
01271 006244      EHALT          ;AC1=EXPECTED STATUS.  ACO=BAD!
01272 061422      DIB 0, .MTA
01273 024121      LDA 1, C377
01274 122414      SUB# 1.0, SZR
01275 006244      EHALT          ;ADDRESS REG. ERROR.
                                ;2 REC SHOULD ICR IT
                                ;TWICE DURING BACKSPC

01276 006236      JSR @LOOP
                                ;CHECK BACKSPACE

01277 006243      A31: JSR @SETP1      ;REWIND-WRITE 3 REC!
01300 006266      RWDSTALL      ;BACKSPACE 4. CHECK
01301 006277      WRITE          ;BOT AND CA FOR
01302 000302      302          ;3 COUNTS
01303 024112      LDA 1, C100
01304 066022      DOB 1, .MTA
01305 006255      @SPACE
01306 024161      LDA 1, M4
01307 067022      DOC 1, .MTA
01310 006273      TMDONE
01311 002000      2000
01312 006272      TIME
01313 002000      2000
01314 000201      201
01315 000401      JMP .+1
01316 006271      STATUS
01317 100201      100201
01320 006244      EHALT          ;AC1=EXPECTED STATUS, ACO=BAD
                                ;BOT SHOULD STOP OPERATION
                                ;IN 3 BACKSPACES

01321 024113      LDA 1, C103
01322 061422      DIB 0, .MTA
01323 106414      SUB# 0.1, SZR
01324 006244      EHALT          ;CA REGISTER SHOULD BE 103
                                ;AFTER 3 "INCREMENT WC".
                                ;ACO=ACTUAL CA REG. CHECK THE
                                ;DIRECT SET LINE ON STOP FF

01325 006236      JSR @LOOP
                                ;CHECK SPACE FORWARD

01326 006243      A32: JSR @SETP1      ;REWIND-WRITE 3 REC!
01327 006266      RWDSTALL      ;REWIND-CHECK STATUS
01330 006277      WRITE          ;SPACE FORWARD-CHECK
01331 000302      302          ;FOR NOT BOT. A PROPER
01332 006266      RWDSTALL      ;ADDR. REG.
01333 006271      STATUS
01334 100201      100201
01335 006244      EHALT          ;AC1=EXPECTED STATUS. ACO=BAD
                                ;THIS HAS WORKED MANY TIMES
                                ;BEFORE! IF ERROR RESTART PROG

01336 024112      LDA 1, C100
01337 066022      DOB 1, .MTA
01340 006270      SPACE

```

```

01341 006273      TMDONE
01342 002000      2000
01343 006274      TESTPE
01344 006271      STATUS
01345 000001      1
01346 006244      EHALT                ;AC1=EXPECTED STATUS, ACO=BAD!
                                           ;THIS IS THE FIRST SPACE FOR.
                                           ;CHECK COMMAND DECODER-SPACE FWD
                                           ;COUNT WC

01347 024112      LDA 1,C100
01350 125400      INC 1,1
01351 061422      DIB 0, .MTA
01352 106414      SUB# 0,1,SZR
01353 006244      EHALT                ;ADDR REG WRONG! AC1=CORRECT
                                           ;ACO=ACTUAL! ONE INCR FROM 100
                                           ;TO 101 SHOULD HAVE OCCURED!
                                           ;CHECK SPACE FWD NOT-COUNT WC-
                                           ;INCREMENT WC.

01354 006236      JSR @LOOP
                                           ;CHECK READ INSTRUCTION. READ 16 CHAR
                                           ;RECORDS. PATTERN IN A33.2 (ODD PARITY)

01355 102000      A33:  ADC 0,0                ;REWIND-WRITE 1 REC.
01356 040406      STA 0,A33.2            ;CHECK STATUS-REWIND-
01357 020173      LDA 0,POINT           ;CHECK STATUS-READ-
01360 040021      STA 0,IDX1           ;CHECK DONE, STATUS,
01361 006243      A33.1: JSR @SETP1            ;ADDR, REG, & DATA
01362 006266      RWDSTALL          ;BUFFER
01363 006261      GEN
01364 017777      A33.2: 17777
01365 006277      WRITE
01366 000110      110
01367 006274      TESTPE
01370 006271      STATUS
01371 000001      1
01372 006244      EHALT                ;AC1=EXPECTED STATUS, ACO=BAD
                                           ;THIS MUCH HAS WORKED BEFORE,
                                           ;MANY TIMES! RESTART THE PROG.

01373 006266      RWDSTALL
01374 006271      STATUS
01375 100201      100201
01376 006244      EHALT                ;AC1=EXPECTED STATUS, ACO=BAD
                                           ;LOOKING FOR EOT BEFORE TRYING
                                           ;FIRST READ, THIS HAS WORKED
                                           ;BEFORE!

01377 006257      CLEAR
01400 006263      READ
01401 000110      110
01402 063622      SKPDN .MTA
01403 006244      EHALT                ;NO DONE FLAG FOLLOWING READ.
                                           ;THIS IS THE FIRST READ INST.
                                           ;CHECK READ-DCHI-INCREMENT WC

01404 006274      TESTPE
01405 006271      STATUS
01406 000001      1
01407 006244      EHALT                ;AC1=EXPECTED STATUS, ACO=BAD

```

;90% OF THE READ LOGIC HAS BEEN
 ;CHECKED DURING WRITE OPERATION
 ;PROBLEMS HERE ARE LIKELY IN THE
 ;PROCESSOR INTERFACE AREA.

```

01410 020061      LDA 0,IBUFF
01411 024101      LDA 1,C10
01412 107000      ADD 0,1
01413 061422      DIB 0.,MTA
01414 106414      SUB# 0,1,SZR
01415 006244      EHALT

;ADDRESS REGISTER IS WRONG.
;AC1=CORRECT, AC0=ACTUAL.
;10 WORD REC READ, CHECK
;MT DCH SEL-DCHMO-INCREMENT
;WC

01416 006256      CHECK
01417 000010      10
01420 006244      EHALT

;INPUT BUFFER CONTAINS WRONG
;DATA. INPUT BUFFER SHOULD
;OUTPUT BUFFER FOR THE FIRST 8
;WORDS. AC0=GOOD WORD, AC1=BAD
;AC2=ADDRESS OF INPUT BUFFER

01421 006236      JSR @LOOP
01422 020021      LDA 0,IDX1
01423 024201      LDA 1,LST
01424 106405      SUB 0,1,SNR
01425 000404      JMP A34
01426 022021      LDA 0,@IDX1
01427 040735      STA 0,A33.2
01430 000731      JMP A33.1
;CHECK READ INSTRUCTION! READ IS CHAR
;RECORDS. PATTERN IN A34.2 (EVEN PARITY NRZ)

01431 102000      A34:  ADC 0,0          ;REWIND-WRITE 1 REC!
01432 040406      STA 0,A34.2      ;CHECK STATUS-REWIND
01433 020202      LDA 0,PONTP      ;CHECK STATUS-READ
01434 040021      STA 0,IDX1      ;CHECK DONE,STATUS,
01435 006243      A34.1: JSR @SETP1      ;ADDRESS REG.,DATA IN
01436 006266      RWDSTALL      ;BUFFER
01437 006261      GEN
01440 177777      A34.2: 177777
01441 006277      WRITE
01442 100110      100110
01443 006274      TESTPE
01444 006271      STATUS
01445 000001      1
01446 006244      EHALT

;AC1=EXPECTED STATUS, AC0=BAD
;THIS MUCH HAS WORKED BEFORE!
;RESTART PROGRAM

01447 006266      RWDSTALL
01450 006271      STATUS
01451 100201      100201
01452 006244      EHALT

;AC1=EXPECTED STATUS, AC0=BAD
;LOOKING FOR BOT STATUS BEFORE
;TRYING FIRST EVEN PARITY READ
;THIS SHOULD WORK!

01453 006257      CLEAR

```

```

01454 006263      READ
01455 100110      100110
01456 063622      SKPDN .MTA
01457 006244      EHALT                ;NO DONE FLAG. CHECK ERF-
                                       ;DCHI-INCREMENT WC

01460 006274      TESTPE
01461 006271      STATUS
01462 000001      1
01463 006244      EHALT                ;AC1=EXPECTED STATUS, ACO=BAD
                                       ;CHECK PARITY CIRCUITS! THIS
                                       ;IS THE ONLY DIFFERENCE FROM
                                       ;THE LAST TEST.

01464 020061      LDA 0,IBUFF
01465 024101      LDA 1,C10
01466 107000      ADD 0,1
01467 061422      DIS 0,.MTA
01470 106414      SUS# 0,1,SZR
01471 006244      EHALT                ;ADDRESS REG. IS WRONG. AC1=
                                       ;CORRECT VALUE, ACO=ACTUAL.
                                       ;10 WORD REG. READ. CHECK MT
                                       ;DCH SEL-DCHMO-INCREMENT WC

01472 006256      CHECK
01473 000010      10
01474 006244      EHALT                ;INPUT BUFFER CONTAINS WRONG
                                       ;DATA! INPUT BUFFER SHOULD
                                       ;=OUTPUT BUFFER FOR FIRST 8 WRDS
                                       ;ACO=GOOD WORD. AC1=BAD WORD
                                       ;AC2=ADDRESS OF THE INPUT BUFFER

01475 006236      JSR @LOOP
01476 020021      LDA 0,IDX1
01477 024207      LDA 1,LST1
01500 106405      SUB 0,1,SNR
01501 000404      JMP A35
01502 022021      LDA 0,@IDX1
01503 040735      STA 0,A34.2
01504 000731      JMP A34.1
                                       ;READ 16 CHAR REC! WITH WC=63
01505 006243      A35: JSR @SETP1                ;REWIND- WRITE 1 REC.
01506 006266      RWDSTALL                ;CHECK STATUS-REWIND
01507 006261      GEN                    ;READ-CHECK DONE.
01510 125252      125252                ;STATUS,ADDR REG.
01511 006277      WRITE                ;AND DATA
01512 000110      110
01513 006274      TESTPE
01514 006271      STATUS
01515 000001      1
01516 006244      EHALT                ;AC1=EXPECTED STATUS, ACO=BAD
                                       ;DON'T TRY TO READ IT IF IT
                                       ;CANNOT BE WRITTEN PROPERLY!
                                       ;THIS HAS WORKED BEFORE.
                                       ;RESTART PROGRAM

01517 006266      RWDSTALL
01520 006263      READ
01521 000177      177
01522 063622      SKPDN .MTA

```



```

01523 006244      EHALT          ;NO DONE FLAG. CHECK INCREMENT
                                ;WC-ERF-STOP-TUR

01524 006274      TESTPE
01525 006271      STATUS
01526 000001      1
01527 006244      EHALT          ;AC1=EXPECTED STATUS, ACO=BAD
                                ;THIS IS THE FIRST TIME ON A
                                ;READ THAT THE WC DIDN'T MATCH
                                ;THE RECORD LENGTH. CHECK
                                ;ERF GATE ON STOP!

01530 020061      LDA 0,IBUFF
01531 024101      LDA 1,C10
01532 107000      ADD 0,1
01533 061422      DIB 0,.MTA
01534 106414      SUB# 0,1,SZR
01535 006244      EHALT          ;ADDRESS REGISTER WRONG
                                ;AC1=CORRECT ADDRESS, ACO=BAD!
                                ;CHECK INCREMENT WC

01536 006256      CHECK
01537 000010      10
01540 006244      EHALT          ;DATA ERROR!
                                ;ACO=GOOD WORD, AC1=BAD WORD
                                ;AC2=ADDRESS OF THE INPUT BUFFER

01541 006236      JSR @LOOP
                                ;READ 63 CHAR RECI WITH WC=10
01542 006243      A36: JSR @SETP1      ;REWIND-WRITE 1 REC!
01543 006266      RWDSTALL      ;CHECK STATUS-REWIND
01544 006261      GEN          ;READ-CHECK DONE.
01545 125252      125252      ;STATUS,ADDR. REG!
01546 006277      WRITE        ;AND DATA
01547 000177      177
01550 006274      TESTPE
01551 006271      STATUS
01552 000001      1
01553 006244      EHALT          ;AC1=EXPECTED STATUS, ACO=BAD
                                ;THIS MUCH HAS WORKED MANY
                                ;TIMES BEFORE. RESTART PROG.

01554 006266      RWDSTALL
01555 006263      READ
01556 000110      110
01557 063622      SKPDN .MTA
01560 006244      EHALT          ;NO DONE FLAG! CHECK INCREMENT
                                ;WC-WC OVERFLOW-STOP

01561 006274      TESTPE
01562 006271      STATUS
01563 000001      1
01564 006244      EHALT          ;AC1=EXPECTED STATUS, ACO=BAD
                                ;THIS IS THE FIRST TIME A RD. IS
                                ;STOPPED BY WC OVERFLOW ALONE.
                                ;LOOK FOR WC OVERFLOW AND CHECK
                                ;THE WC OVERFLOW GATE ON STOP

01565 020061      LDA 0,IBUFF
01566 024101      LDA 1,C10
01567 107000      ADD 0,1
01570 061422      DIB 0,.MTA

```

```

01571 106414      SUB# 0.1,SZR
01572 006244      EHALLT                ;ADDRESS REG. WRONG!  AC1=CORRECT
                                ;ADDR.  ACO=BAD.
                                ;CHECK DCHI LOGIC AND INCREMENT WC

01573 006256      CHECK
01574 000010      10
01575 006244      EHALLT                ;DATA ERROR!  INPUT BUFFER SHOULD
                                ;=OUTPUT BUFFER FOR THE FIRST 8
                                ;WORDS.
                                ;ACO=GOOD WORD,  AC1=BAD WORD
                                ;AC2=ADDRESS OF THE INPUT BUFFER

01576 006236      JSR @LOOP
                                ;WRITE 4 CHAR REC!  IN EVEN PARITY
                                ;READ IT IN ODD PARITY. (NRZ ONLY)

01577 006254      A37:  BYPASSPE
01600 002225      JMP @IA39
01601 006243      JSR @SETP1                ;REWIND-WRITE 1 REC!
01602 006261      GEN                    ;BACKSPACE-READ
01603 125252      125252                ;CHECK STATUS
01604 006266      RWDSTALL
01605 006277      WRITE
01606 100102      100102
01607 006271      STATUS
01610 000001      1
01611 006244      EHALLT                ;AC1=EXPECTED STATUS, ACO=BAD
                                ;THIS HAS WORKED MANY TIMES
                                ;BEFORE!  RESTART PROGRAM.

01612 006255      BSPACE
01613 006273      TMDONE
01614 000400      400
01615 006263      READ
01616 000102      102
01617 006271      STATUS
01620 102001      102001
01621 006244      EHALLT                ;AC1=EXPECTED STATUS, ACO=BAD
                                ;LOOKING FOR PARITY ERROR!  IF
                                ;NONE CHECK THE PARITY CHECKING
                                ;LOGIC.  (ODD)

01622 006236      JSR @LOOP
                                ;WRITE 4 CHAR RECORD IN ODD PARITY
                                ;READ IT BACK IN EVEN PARITY (NRZ ONLY)

01623 006243      A38:  JSR @SETP1                ;REWIND-WRITE 1 REC!
01624 006261      GEN                    ;BACKSPACE-READ
01625 125252      125252                ;CHECK STATUS
01626 006266      RWDSTALL
01627 006277      WRITE
01630 000102      102
01631 006271      STATUS
01632 000001      1
01633 006244      EHALLT                ;AC1=EXPECTED STATUS, ACO=BAD
                                ;THIS HAS WORKED MANY TIMES
                                ;BEFORE.  RESTART PROGRAM

01634 006255      BSPACE
01635 006273      TMDONE
01636 000400      400

```

01637	006263		READ	
01640	100102		100102	
01641	006271		STATUS	
01642	102001		102001	
01643	006244		EHALT	;AC1=EXPECTED STATUS. ACO=BAD ;LOOKING FOR PARITY ERROR! IF ;NONE CHECK THE READ PARITY ;CHECKING LOGIC. (EVEN)
01644	006236		JSR @LOOP	
			;CHECK READ EOF	
01645	006243	A39:	JSR @SETP1	;REWIND-WRITE A 16
01646	006266		RWDSTALL	;CHAR REC!. WRITE EOF
01647	006277		WRITE	;CHECK STATUS
01650	000110		110	;REWIND-READ 2 REC.
01651	006275		WEOF	;CHECK STATUS,CA
01652	006271		STATUS	
01653	100401		100401	
01654	006244		EHALT	;AC1=EXPECTED STATUS. ACO=BAD ;THIS HAS WORKED BEFORE. RESTART
01655	006266		RWDSTALL	
01656	006263		READ	
01657	000210		210	
01660	034127		LDA 3,C1004	
01661	030124		LDA 2,C2000	
01662	020222		LDA 0,SWTSV	
01663	024112		LDA 1,C100	
01664	107405		AND 0,1,SNR	
01665	157000		ADD 2,3	
01666	054402		STA 3,..+2	
01667	006271		STATUS	
01670	100401		100401	
01671	006244		EHALT	;AC1=EXPECTED STATUS. ACO=BAD ;LOOKING FOR READ EOF.
01672	024061		LDA 1,IBUFF	
01673	030222		LDA 2,SWTSV	
01674	034122		LDA 3,C400	
01675	157404		AND 2,3,SZR	;SKIP INC IF PE
01676	000402		JMP .+2	
01677	125400		INC 1,1	
01700	061422		DIB 0,..MTA	
01701	106414		SUB# 0,1,SZR	
01702	006244		EHALT	;ADDRESS REG WRONG. AC1=CORRECT ;ACO=BAD. EOF SHOULD INCR ;WC AND ADDR. BY 1
01703	006236		JSR @LOOP	
			;CHECK READ EOF	ON SPACE BACK
01704	006243	A40:	JSR @SETP1	;REWIND-WRITE 2 REC
01705	006266		RWDSTALL	;WRITE EOF THEN 3 REC
01706	006277		WRITE	;BACKSPACE 10
01707	000210		210	;CHECK STATUS AND
01710	006275		WEOF	;ADDR REG.
01711	006271		STATUS	
01712	100401		100401	
01713	006244		EHALT	;AC1=EXPECTED STATUS. ACO=BAD ;THIS HAS WORKED BEFORE!

;RESTART PROGRAM

01714 006277
 01715 000310
 01716 006255
 01717 020164
 01720 063022
 01721 020112
 01722 062022
 01723 006273
 01724 001000
 01725 006271
 01726 100401
 01727 006244

WRITE
 310
 BSPACE
 LDA 0,M8
 DOC 0,.MTA
 LDA 0,C100
 DOB 0,.MTA
 TMDONE
 1000
 STATUS
 100401
 EHALT

;AC1=EXPECTED STATUS, ACO=BAD
 ;LOOKING FOR EOF. CHECK
 ;EOF GATE ON STOP

01730 024114
 01731 061422
 01732 106414
 01733 006244

LDA 1,C104
 DIB 0,.MTA
 SUB# 0,1,SZR
 EHALT

;ADDRESS REG. IS WRONG!
 ;AC1=CORRECT VALUE, ACO=BAD
 ;CHECK "COUNT WC" AND INCREMENT
 ;SHOULD GET 4 COUNT WC
 ;BEFORE STOP

01734 006236

JSR @LOOP
 ;CHECK READ EOF ON SPACE FORWARD
 JSR @SETP1
 RWDSTALL
 WRITE
 210
 WEOF
 STATUS
 100401
 EHALT

;REWIND-WRITE 2 REC
 ;WEOF-WRITE 3 REC
 ;REWIND-SPACE 3
 ;CHECK STATUS, ADDR
 ;REG.

01735 006243 A41:
 01736 006266
 01737 006277
 01740 000210
 01741 006275
 01742 006271
 01743 100401
 01744 006244

;AC1=EXPECTED STATUS, ACO=BAD
 ;THIS HAS WORKED BEFORE.
 ;RESTART PROGRAM

01745 006277
 01746 000310
 01747 006266
 01750 006270
 01751 020162
 01752 063022
 01753 020112
 01754 062022
 01755 006273
 01756 010000
 01757 006274
 01760 006271
 01761 100401
 01762 006244

WRITE
 310
 RWDSTALL
 SPACE
 LDA 0,M5
 DOC 0,.MTA
 LDA 0,C100
 DOB 0,.MTA
 TMDONE
 10000
 TESTPE
 STATUS
 100401
 EHALT

;AC1=EXPECTED STATUS, ACO=BAD
 ;LOOKING FOR EOF STATUS. THIS IS
 ;THE SAME AS THE LAST EXCEPT
 ;THAT SPACING IS FORWARD

01763 024113
 01764 061422
 01765 106414

LDA 1,C103
 DIB 0,.MTA
 SUB# 0,1,SZR

01766 006244	EHALT	;ADDRESS ERROR! AC1=CORRECT ;ACO=BAD. TWO INCREMENT WC ;SHOULD HAVE OCCURED ;EOF GIVES STOP.
01767 006236	JSR @LOOP ;CRC-LPC CHECK ;9TRACK ONLY NRZ ONLY	
01770 034222	A42: LDA 3,SWTSV	
01771 030112	LDA 2,C100	
01772 157415	AND# 2,3,SNR	
01773 002226	JMP @IA46	;SKIP OVER 9 TAPE TESTS
01774 006254	BYPASSPE	
01775 002231	JMP @IA50	
01776 006243	JSR @SETP1	;REWIND-WRITE 1 REC
01777 006266	RWDSTALL	;CHECK STATUS-REWIND
02000 006262	LOAD	;READ(TEST MODE)
02001 003107	PAT1-1	;CHECK STATUS,CA,DATA
02002 006277	WRITE	
02003 000102	102	
02004 006271	STATUS	
02005 000001	1	
02006 006244	EHALT	;AC1=EXPECTED STATUS, ACO=BAD ;THIS HAS WORKED BEFORE. ;RESTART PROGRAM
02007 006266	RWDSTALL	
02010 006257	CLEAR	
02011 024157	LDA 1,M2	
02012 067022	DOC 1,.MTA	
02013 024061	LDA 1,IBUFF	
02014 127240	ADDOR 1,1	;SET BIT 0
02015 066022	DOB 1,.MTA	
02016 024212	LDA 1,CX	
02017 065122	DOAS 1,.MTA	
02020 006273	TMDONE	
02021 002000	2000	
02022 006271	STATUS	
02023 000001	1	
02024 006244	EHALT	;AC1=EXPECTED STATUS, ACO=BAD ;JUST READ 4 CHAR REC IN TEST ;MODE. TEST MODE IS THE ONLY ;NEW VARIABLE.
02025 061422	DIB 0,.MTA	
02026 024061	LDA 1,IBUFF	
02027 030074	LDA 2,C2	
02030 147000	ADD 2,1	
02031 106414	SUB# 0,1,SZR	
02032 006244	EHALT	;ADDR REG WRONG. AC1=CORRECT ;AC7=ACTUAL
02033 131000	MOV 1,2	
02034 066422	DIC 1,.MTA	
02035 045000	STA 1,0,2	
02036 006256	CHECK	
02037 000003	3	
02040 006244	EHALT	;DATA ERROR! 2 WORDS WERE WK. ;3 WORDS WERE READ. THE 3RD

```

;WORD IS CRC(LEFT)
;CRC(LEFT)
;THIS TEST LOOKS FOR BIT PICKUPS
;IN THE CRC GEN. DATA IS CHOSEN
;TO PRODUCE MINIMUM 1'S IN CRC
;GENERATION. THE SEQ. XOR-SHIFT
;COMPARE IS DONE ONCE FOR EACH
;OF THE 4 CHARACTERS.
;XOF 1&3 PRODUCE A 1 IN CRC1
;WHICH IS THEN SHIFTED INTO CRC2
;ALL OTHER CASES PRODUCE 0'S.
;1'S ARE WRITTEN ON TAPE BECAUSE
;OF CRC COMPLEMENT AT THE END!
;AC2=ADD3. IF IBUFF
;AC1= BAD WORD
;AC0= GOOD WORD
;EXPECTED DATA IS 000200
                000200
                ; 153600

```

```

02041 006236      JSR @LOOP
                  ;CRC-LPC CHECK
                  ;9 TRACK ONLY
02042 006243      A43: JSR @SETP1      ;REWIND-WRITE 1 REC
02043 006266      RWDSTALL      ;CHECK STATUS-REWIND
02044 006262      LOAD          ;READ( TEST MODE)
02045 003112      PAT2-1      ;CHECK STATUS,CA,DATA
02046 006277      WRITE
02047 000102      102
02050 006271      STATUS
02051 000001      1
02052 006244      EHALT      ;AC1=EXPECTED STATUS, AC0=BAD
                               ;THIS HAS WORKED BEFORE.
                               ;RESTART PROGRAM

02053 006266      RWDSTALL
02054 006257      CLEAR
02055 024157      LDA 1,M2
02056 067022      DOC 1,.MTA
02057 024061      LDA 1,IBUFF
02060 127240      ADDOR 1,1      ;SET BIT 0
02061 066022      DOB 1,.MTA
02062 024212      LDA 1,CX
02063 065122      DCAS 1,.MTA
02064 006273      TMDONE
02065 002000      2000
02066 006271      STATUS
02067 000001      1
02070 006244      EHALT      ;AC1=EXPECTED STATUS, AC0=BAD
                               ;JUST READ 4 CHAR REC IN TEST
                               ;MODE. TEST MODE IS THE ONLY
                               ;NEW VARIABLE.

02071 061422      DIB 0,.MTA
02072 024061      LDA 1,IBUFF
02073 030074      LDA 2,C2
02074 147000      ADD 2,1
02075 106414      SUB# 0,1,SZR

```



```

;MODE. TEST MODE IS THE ONLY
;NEW VARIABLE.

```

```

02135 061422      DIB 0,.MTA
02136 024061      LDA 1,IBUFF
02137 030074      LDA 2,C2
02140 147000      ADD 2,1
02141 106414      SUB# 0,1,SZR
02142 006244      EHALT

```

```

;ADDR REG WRONG!  AC1=CORRECT
;ACO=ACTUAL!

```

```

02143 131000      MOV 1,2
02144 066422      DIC 1,.MTA
02145 045000      STA 1,0,2
02146 006256      CHECK
02147 000003      3
02150 006244      EHALT

```

```

;DATA ERROR!  2 WORDS WERE
;WRITTEN.  3 WORDS WERE READ.
;THE 3RD WORD IS CRC(LEFT)
;THIS TEST IS A GATE CHECK FOR
;THE CRC GEN.  DATA IS CHOSEN TO
;PRODUCE ALL COMBINATIONS OF XDR
;ON EACH CRC STT.  THIS 4 CHAR
;REC.  CATCHES BOTH OF THE CASES!
;AC2=ADDR OF IBUFF
;AC1=BAD WORD
;ACO=GOOD WORD
;EXPECTED DATA IS 052652
;                   ; 076232
;                   ; 037232

```

```

02151 006236      JSR @LOOP
;CRC-LPC CHECK
;9 TRACK ONLY

```

```

02152 006243      A45: JSR @SETP1
02153 006266      RWDSTALL
02154 006262      LOAD
02155 003120      PAT4-1
02156 006277      WRITE
02157 000102      102
02160 006271      STATUS
02161 000001      1
02162 006244      EHALT

```

```

;REWIND-WRITE 1 REC
;CHECK STATUS-REWIND
;READ(TEST MODE)
;CHECK STATUS,CA,DATA

```

```

;AC1=EXPECTED STATUS.  ACO=BAD
;THIS HAS WORKED BEFORE.
;RESTART PROGRAM

```

```

02163 006266      RWDSTALL
02164 006257      CLEAR
02165 024157      LDA 1,M2
02166 067022      DOC 1,.MTA
02167 024061      LDA 1,IBUFF
02170 127240      ADDOR 1,1
02171 066022      DOB 1,.MTA
02172 024212      LDA 1,CX
02173 065122      DOAS 1,.MTA
02174 006273      TMDONE
02175 002000      2000
02176 006271      STATUS
02177 000001      1

```

```

;SET BIT 0

```


02200	006244	EHALT	;AC1=EXPECTED STATUS. ACC=BAD ;JUST READ 4 CHAR REC IN TEST MODE. ;TEST MODE IS THE ONLY NEW ;VARIABLE
02201	061422	DIB 0,.MTA	
02202	024061	LDA 1,IBUFF	
02203	030074	LDA 2,C2	
02204	147000	ADD 2,1	
02205	106414	SUB# 0,1,SZR	
02206	006244	EHALT	;ADDR REG WRONG! AC1=CORRECT ;AC0=ACTUAL!
02207	131000	MOV 1,2	
02210	066422	DIC 1,.MTA	
02211	045000	STA 1,0,2	
02212	006256	CHECK	
02213	000003	3	
02214	006244	EHALT	;DATA ERROR! 2 WORDS WERE WRITTEN ;3 WORDS WERE READ, THE 3RD WORD IS ;CRC(LEFT) ;THIS TEST IS A GATE CHECK FOR THE ;CRC GENERATOR. DATA IS CHOSEN TO ;PRODUCE ALL COMBINATIONS OF XDR ;ON EACH CRC BIT. THIS TEST CATCHES ;THE REMAINING OF THE CASES ;NOT COVERED IN THE PREVIOUS TEST. ;AC2=ADDR OF IBUFF ;AC1=BAD WORD ;AC0=GOOD WORD ;EXPECTED DATA IS 167141 ; 000200 ; 063000
02215	006236	JSR @LOOP	
02216	002401	JMP @.+1	
02217	002433	A50	
		;7TRK LPC CHECK	
02220	006243	A46: JSR @SETP1	;WRITE 2WD REC
02221	006262	LOAD	;READ AND CHECK
02222	003123	PAT5-1	;STATUS. READ AND
02223	006277	WRITE	;CHECK LPC
02224	000102	102	
02225	006271	STATUS	
02226	000001	1	
02227	006244	EHALT	;ERROR IN WRITING. AC1=EXPECTED ;STATUS. AC0=ACTUAL. ;RESTART
02230	006255	BSPACE	
02231	006273	TMDONE	
02232	000400	400	
02233	006257	CLEAR	
02234	024157	LDA 1,M2	
02235	067022	DOC 1,.MTA	
02236	024061	LDA 1,IBUFF	
02237	127240	ADDR 1,1	
02240	066022	DOB 1,.MTA	
02241	024212	LDA 1,CX	

02242	055122		DOAS 1,,MTA	
02243	006273		TMDONE	
02244	000400		400	
02245	006271		STATUS	
02246	000001		1	
02247	006244		EHALT	;AC1=EXPECTED STATUS. ;AC0=ACTUAL. JUST READ 4 CHAR ;RECORD INTO IBUFF IN TEST ;MODE. TEST MODE IS THE ;ONLY THING NEW.
02250	066422		DIC 1,,MTA	
02251	030061		LDA 2,IBUFF	
02252	151400		INC 2,2	
02253	151400		INC 2,2	
02254	045000		STA 1,0,2	
02255	006256		CHECK	
02256	000003		3	
02257	006244		EHALT	;DATA OR 1 PC ERROR. 2 ;WORDS WRITTEN, 2 WORDS RD. BACK ;LPC ALL ZEROS, NEVER GETS READ ;(NO RPO). AC2 POINTS TO IBUFF ;AC1= BAD WORD ;AC0= GOOD WORD ;EXPECTED DATA IS 000077 ; 030077 ; 000077
02260	006236		JSR @LOOP	
			;7TRK LPC CHECK	
02261	006243	A47:	JSR @SETP1	;WRITE 2WD REC
02262	006262		LOAD	;READ AND CHECK
02263	003126		PAT6-1	;STATUS. READ AND
02264	006277		WRITE	;CHECK LPC
02265	000102		102	
02266	006271		STATUS	
02267	000001		1	
02270	006244		EHALT	;ERROR IN WRITING. AC1=EXPECTED ;STATUS. AC0=ACTUAL. ;RESTART
02271	006255		BSPACE	
02272	006273		TMDONE	
02273	000400		400	
02274	006257		CLEAR	
02275	024157		LDA 1,M2	
02276	067022		DOC 1,,MTA	
02277	024061		LDA 1,IBUFF	
02300	127240		ADDR 1,1	
02301	066022		DOB 1,,MTA	
02302	024212		LDA 1,CX	
02303	065122		DOAS 1,,MTA	
02304	006273		TMDONE	
02305	000400		400	
02306	006271		STATUS	
02307	000001		1	
02310	006244		EHALT	;AC1=EXPECTED STATUS. ;AC0=ACTUAL. JUST READ 4 CHAR

```

;RECORD INTO Ibuff IN TEST
;MODE. TEST MODE IS THE
;ONLY THING NEW.

```

```

02311 066422
02312 030061
02313 151400
02314 151400
02315 045000
02316 006256
02317 000003
02320 006244

```

```

DIC 1,.MTA
LDA 2,IBUFF
INC 2,2
INC 2,2
STA 1,0,2
CHECK
3
EHALT

```

```

;DATA OR LPC ERROR. 2
;WORDS WRITTEN, 2 WORDS RD. BACK
;LPC ALL ZEROS. NEVER GETS READ
;(NO RPO). AC2 POINTS TO Ibuff
;AC1=BAD WORD
;AC0=GOOD WORD
;EXPECTED DATA IS 037400
; 037400
; 037400

```

```
02321 006236
```

```

JSR @LOOP
;7TRK LPC CHECK

```

```

02322 006243 A48:
02323 006262
02324 003131
02325 006277
02326 000102
02327 006271
02330 000001
02331 006244

```

```

JSR @SETP1
LOAD
PAT7-1
WRITE
102
STATUS
1
EHALT

```

```

;WRITE 2WD REC
;READ AND CHECK
;STATUS. READ AND
;CHECK LPC

```

```

;ERROR IN WRITING. AC1=EXPECTED
;STATUS, AC0=ACTUAL.
;RESTART

```

```

02332 006255
02333 006273
02334 000400
02335 006257
02336 024157
02337 067022
02340 024061
02341 127240
02342 056022
02343 024212
02344 065122
02345 006273
02346 000400
02347 006271
02350 000001
02351 006244

```

```

BSPACE
TMDONE
400
CLEAR
LDA 1,M2
DOC 1,.MTA
LDA 1,IBUFF
ADDOR 1,1
DOB 1,.MTA
LDA 1,CX
DOAS 1,.MTA
TMDONE
400
STATUS
1
EHALT

```

```

;AC1=EXPECTED STATUS.
;AC0=ACTUAL. JUST READ 4 CHAR
;RECORD INTO Ibuff IN TEST
;MODE. TEST MODE IS THE
;ONLY THING NEW.

```

```

02352 066422
02353 030061
02354 151400
02355 151400

```

```

DIC 1,.MTA
LDA 2,IBUFF
INC 2,2
INC 2,2

```

```

02356 045000      STA 1,0,2
02357 006256      CHECK
02360 000003      3
02361 006244      EHALT
;DATA OR LPC ERROR.
;WORDS WRITTEN, 2 WORDS RD. BACK
;LPC ALL ZEROS, NEVER GETS READ
;(NO RPO). AC2 POINTS TO Ibuff
;AC1=BAD WORD
;AC0=GOOD WORD
;EXPECTED DATA IS 012452
; 012452
; 012452

02362 006236      JSR @LOOP
;
;7TRK LPC CHECK
02363 006243      A49: JSR @SETP1      ;WRITE 2WD REC
02364 034222      LDA 3,SWTSV    ;READ AND CHECK
02365 030124      LDA 2,C2000   ;STATUS. READ AND
02366 157405      AND 2,3,SNR   ;CHECK LPC
02367 002227      JMP @IA491
02370 006262      LOAD
02371 003137      PAT9-1
02372 002230      JMP @IA492
02373 006262      A49.1: LOAD
02374 003134      PAT9-1
02375 006277      A49.2: WRITE
02376 000102      102
02377 006271      STATUS
02400 000001      1
02401 006244      EHALT      ;ERROR IN WRITING. AC1=EXPECTED
;STATUS, AC0=ACTUAL.
;RESTART

02402 006255      BSPACE
02403 006273      TMDONE
02404 000400      400
02405 006257      CLEAR
02406 024157      LDA 1,M2
02407 067022      OOC 1,,MTA
02410 024061      LDA 1,IBUFF
02411 127240      ADDOR 1,1
02412 066022      DOB 1,,MTA
02413 024212      LDA 1,CX
02414 066122      DOAS 1,,MTA
02415 006273      TMDONE
02416 000400      400
02417 006271      STATUS
02420 100003      100003
02421 006244      EHALT      ;AC1=EXPECTED STATUS.
;AC0=ACTUAL. JUST READ 4 CHAR
;RECORD INTO Ibuff IN TEST
;MODE! TEST MODE IS THE
;ONLY THING NEW.

02422 066422      DIC 1,,MTA
02423 030061      LDA 2,IBUFF
02424 151400      INC 2,2
02425 151400      INC 2,2

```

```

02426 045000      STA 1,0,2
02427 006256      CHECK
02430 000003      3
02431 006244      EHALT
                                ;DATA OR LPC ERROR. 3
                                ;WORDS WRITTEN, 2 WORDS+LPC
                                ;READ BACK. AC2 POINTS TO Ibuff
                                ;AC1=BAD WORD
                                ;AC0=GOOD WORD
                                ;EXPECTED DATA IS 012452
                                                ; 012477
                                                ; 012477

02432 006236      JSR @LOOP
                                ;SPACING CHECK.
                                ;DURING SPACING MANY IO COMMANDS
                                ;ISSUED TO GENERATE NOISE.
02433 006243      A50: JSR @SETP1
                                ;WRITE 500 2WD REC.
02434 006265      REWIND
                                ;SPACEBACK 500
02435 006272      TIME
                                ;CHECK CA
02436 047040      47040 ; 20000.
                                ;SPACEBACK 1 AND
02437 000201      201
                                ;CHECK FOR BOT
02440 006244      EHALT
                                ;NO BOT-TUR 20 SEC.
                                ;FOLLOWING REWIND

02441 020137      LDA 0,D500
02442 040145      STA 0,RECTR
02443 006276      A50.1: WRTNS
                                102
02444 000102      TIME
                                3720 ; 2000.
02445 006272      1
02446 003720      3720 ; 2000.
02447 000001      1
02450 006244      EHALT
                                ;NO TUR 1 SEC FOLLOWING
                                ;WRITE COMMAND

02451 014145      DSZ RECTR
02452 000771      JMP A50.1
02453 006273      TMDONE
02454 000400      400
02455 020137      LDA 0,D500
02456 100400      NEG 0,0
02457 063022      DOC 0,MTA
                                ;WC=500
02460 102400      SUS 0,0
02461 062022      DOB 0,MTA
                                ;ADDR. REG=0
02462 020212      LDA 0,CX
02463 024105      LDA 1,C40
02464 123000      ADD 1,0
02465 061122      DOAS 0,MTA
                                ;START BACKSPACE
02466 102400      SUB 0,0
02467 064422      A50.2: DIA 1,MTA
02470 064422      DIA 1,MTA
02471 101404      INC 0,0,SZR
02472 000775      JMP ,-3
02473 064422      DIA 1,MTA
02474 125203      MOVR 1,1,SNC
02475 000772      JMP A50.2
02476 061422      DIB 0,MTA
02477 024137      LDA 1,D500
02500 122414      SUB# 1,0,SZR

```

```

02501 006244          EHALT          ;ADDRESS REG. WRONG.
                                   ;SHOULD HAVE COUNTED 500
                                   ;REC ON BACKSPACE.
                                   ;AC1=EXPECTED COUNT. AC0=ACTUAL

02502 006255          BSPACE
02503 064422          DIA 1, .MTA
02504 125203          MOVR 1,1,SNC
02505 000776          JMP .-2
02506 020112          LDA 0,C100
02507 107405          AND 0,1,SNR
02510 006244          EHALT          ;BACKSPACE 1 DID NOT PRODUCE
                                   ;BOT. ADDR REG COUNTED 500.
                                   ;WC REG WAS COUNTING FASTER
                                   ;BAD WC REG.

02511 006236          JSR @LOOP
                                   ;SPACING CHECK.
                                   ;DURING SPACING MANY IO COMMANDS ARE
                                   ;ISSUED TO GENERATE NOISE.

02512 006243  A51:    JSR @SETP1    ;WRITE 500 2WD REC.
02513 006265          REWIND        ;SPACEBACK 500
02514 006272          TIME          ;CHECK CA
02515 047040  47040 ; 20000.        ;SPACEBACK 1 AND
02516 000201          201          ;CHECK FOR BOT
02517 006244          EHALT        ;NO BOT-TUR 20 SEC.
                                   ;FOLLOWING REWIND
                                   LDA 0,0500

02520 020137
02521 040145          STA 0,RECTR
02522 006276  A51.1:  WRTNS
02523 000102          102
02524 006272          TIME
02525 003720  3720 ; 2000.
02526 000001          1
02527 006244          EHALT        ;NO TUR 1 SEC FOLLOWING
                                   ;WRITE COMMAND

02530 014145          DSZ RECTR
02531 000771          JMP A51.1
02532 006273          TMDONE
02533 000400          400
02534 020137          LDA 0,0500
02535 100400          NEG 0,0
02536 063022          DOC 0, .MTA    ;WC=500
02537 102400          SUB 0,0
02540 062022          DOB 0, .MTA    ;ADDR. REG=0
02541 020212          LDA 0,CX
02542 024105          LDA 1,C40
02543 123000          ADD 1,0
02544 061122          DOAS 0, .MTA    ;START BACKSPACE
02545 102400          SUB 0,0
02546 065422  A51.2:  DIB 1, .MTA
02547 065422          DIB 1, .MTA
02550 101404          INC 0,0,SZR
02551 000775          JMP .-3
02552 064422          DIA 1, .MTA
02553 125203          MOVR 1,1,SNC
02554 000772          JMP A51.2

```

02555	061422		DIB 0,,MTA	
02556	024137		LDA 1,D500	
02557	122414		SUB# 1,0,SZR	
02560	006244		EHALT	;ADDRESS REG. WRONG. ;SHOULD HAVE COUNTED 500 ;REC ON BACKSPACE. ;AC1=EXPECTED COUNT. ACO=ACTUAL
02561	006255		BSPACE	
02562	064422		DIA 1,,MTA	
02563	125203		MOVR 1,1,SNC	
02564	000776		JMP .-2	
02565	020112		LDA 0,C100	
02566	107405		AND 0,1,SNR	
02567	006244		EHALT	;BACKSPACE 1 DID NOT PRODUCE ;BOT. ADDR REG COUNTED 500. ;WC REG WAS COUNTING FASTER ;BAD WC REG.
02570	006236		JSR @LOOP	
			;SPACING CHECK.	
			;DURING SPACIND MANY IO COMMANDS ARE	
			;ISSUED TO GENERATE NOISE.	
02571	006243	A52:	JSR @SETP1	;WRITE SEG 2WD REG.
02572	006265		REWIND	;SPACEBACK 500
02573	006272		TIME	;CHECK CA
02574	047040	47040 :	20000.	;SPACEBACK 1 AND
02575	000201		201	;CHECK FOR BOT
02576	006244		EHALT	;NO BOT-TUR 20 SEC. ;FOLLOWING REWIND
02577	020137		LDA 0,D500	
02600	040145		STA 0,RECTR	
02601	006276	A52.1:	WRTNS	
02602	000102		102	
02603	006272		TIME	
02604	003720	3720 :	2000.	
02605	000001		1	
02606	006244		EHALT	;NO TUR 1 SEC FOLLOWING ;WRITE COMMAND
02607	014145		DSZ RECTR	
02610	000771		JMP A52.1	
02611	006273		TMDONE	
02612	000400		400	
02613	020137		LDA 0,D500	
02614	100400		NEG 0,0	
02615	063022		DOC 0,,MTA	;WC=500
02616	102400		SUB 0,0	
02617	062022		DOB 0,,MTA	;ADDR. REG=0
02620	020212		LDA 0,CX	
02621	024105		LDA 1,C40	
02622	123000		ADD 1,0	
02623	061122		DOAS 0,,MTA	;START BACKSPACE
02624	102400		SUB 0,0	
02625	066422	A52.2:	DIC 1,,MTA	
02626	066422		DIC 1,,MTA	
02627	101404		INC 0,0,SZR	
02630	000775		JMP .-3	

```

02631 064422      DIA 1,,MTA
02632 125203      MOVR 1,1,SNC
02633 000772      JMP A52.2
02634 061422      DIB 0,,MTA
02635 024137      LDA 1,D500
02636 122414      SUB# 1,0,SZR
02637 006244      EHALT
;ADDRESS REG. WRONG.
;SHOULD HAVE COUNTED 500
;REC ON BACKSPACE.
;AC1=EXPECTED COUNT. ACO=ACTUAL

02640 006255      BSPACE
02641 064422      DIA 1,,MTA
02642 125203      MOVR 1,1,SNC
02643 000776      JMP .-2
02644 020112      LDA 0,C100
02645 107405      AND 0,1,SNR
02646 006244      EHALT
;BACKSPACE 1 DID NOT PRODUCE
;BOT. ADDR REG COUNTED 500.
;WC REG WAS COUNTING FASTER
;BAD WC REG.

02647 006236      JSR @LOOP
;SPACING CHECK.
;DURING SPACING MANY IO COMMANDS ARE
;ISSUED TO GENERATE NOISE.

02650 006243      A53: JSR @SETP1 ;WRITE 500 2WD REC.
02651 006265      REWIND ;SPACEBACK 500
02652 006272      TIME ;CHECK CA
02653 047040      47040 : 20000. ;SPACEBACK 1 AND
02654 000201      201 ;CHECK FOR BOT
02655 006244      EHALT ;NO BOT-TUR 20 SEC.
;FOLLOWING REWIND

02656 020137      LDA 0,D500
02657 040145      STA 0,RECTR
02660 006276      A53.1: WRTNS
02661 000102      102
02662 006272      TIME
02663 003720      3720 : 2000.
02664 000001      1
02665 006244      EHALT ;NO TUR 1 SEC FOLLOWING
;WRITE COMMAND

02666 014145      DSZ RECTR
02667 000771      JMP A53.1
02670 006273      TMDONE
02671 000400      400
02672 020137      LDA 0,D500
02673 100400      NEG 0,0
02674 063022      DOC 0,,MTA ;WC=500
02675 102400      SUB 0,0
02676 062022      DOB 0,,MTA ;ADDR. REG=0
02677 020212      LDA 0,CX
02700 024105      LDA 1,C40
02701 123000      ADD 1,0
02702 061122      DCAS 0,,MTA ;START BACKSPACE
02703 102400      SUB 0,0
02704 060322      A53.2: NIOP .MTA

```


02705	060322	NIOP .MTA	
02706	101404	INC 0,0,SZR	
02707	000775	JMP .-3	
02710	064422	DIA 1,.MTA	
02711	125203	MOVR 1,1,SNC	
02712	000772	JMP A53.2	
02713	061422	DIB 0,.MTA	
02714	024137	LDA 1,0500	
02715	122414	SUB# 1,0,SZR	
02716	006244	EHALT	;ADDRESS REG. WRONG, ;SHOULD HAVE COUNTED 500 ;REC ON BACKSPACE. ;AC1=EXPECTED COUNT. AC0=ACTUAL
02717	006255	BSPACE	
02720	064422	DIA 1,.MTA	
02721	125203	MOVR 1,1,SNC	
02722	000776	JMP .-2	
02723	020112	LDA 0,C100	
02724	107405	AND 0,1,SNR	
02725	006244	EHALT	;BACKSPACE 1 DID NOT PRODUCE ;BOT. ADDR REG COUNTED 500. ;WC REG WAS COUNTING FASTER ;BAD WC REG.
02726	006236	JSR @LOOP	
		;LONGITUDINAL PARITY CHECK (NRZ ONLY)	
02727	006254	BYPASSPE	
02730	002232	JMP @IA55	
02731	020522	LDA 0,LSTRT	;CAUSE LONG PARITY ERROR
02732	040447	STA 0,LPOIT	;IN 2 TRK PAIRS
02733	102400	SUB 0,0	;CHECK PARITY STATUS
02734	040444	STA 0,RBCT	
02735	006260	ERASE	
02736	006251	GEN	
02737	037477	37477	
02740	020060	LDA 0,OBUFF	
02741	101400	INC 0,0	
02742	040020	STA 0,IDX0	
02743	022436	LDA 0,@LPOIT	
02744	101005	MOV 0,0,SNR	
02745	000435	JMP CKED	
02746	042020	STA 0,@IDX0	
02747	102400	SUB 0,0	
02750	024166	LDA 1,M400	
02751	042020	STA 0,@IDX0	
02752	125404	INC 1,1,SZR	
02753	000776	JMP .-2	
02754	006243	JSR @SETP1	
02755	020060	LDA 0,OBUFF	
02756	062022	DOB 0,.MTA	
02757	020166	LDA 0,M400	
02760	063222	DOCC 0,.MTA	
02761	020116	LDA 0,C150	
02762	024212	LDA 1,CX	
02763	123000	ADD 1,0	
02764	061122	DOAS 0,.MTA	

```

02765 006273      TMDONE
02766 001400      1400
02767 030411      LDA 2,RBCT
02770 060422      DIA 0,.MTA
02771 024124      LDA 1,C2000
02772 123415      AND# 1,0,SNR
02773 006244      EHALT

```

```

;NO PARITY ERROR. # IN AC2
;INDICATES WHICH TRACKS SHOULD
;CAUSE ERROR. (SEE BELOW)
;0=RB2-2 0=RB2-3 14=RB0-P
;1=RB3-P 7=RB3-4 15=RB1-P
;2=RB4-P 10=RB4-5 16=RB0-1
;3=RB5-P 11=RB5-6 17=RB0-2
;4=RB6-P 12=RB6-7 20=RB1-2
;5=RB7-P 13=RB7-2

```

```

02774 006236      JSR @LOOP
02775 010404      ISZ LPOIT
02776 010402      ISZ RBCT
02777 000737      JMP A54
03000 000000      RBCT: 0
03001 000000      LPOIT: 0
03002 060422      CKED: DIA 0,.MTA
03003 024112      LDA 1,C100
03004 010775      ISZ LPOIT
03005 123405      AND 1,0,SNR
03006 000404      JMP .+4
03007 022772      LDA 0,@LPOIT
03010 101004      MOV 0,0,SZR
03011 000735      JMP A54.1
03012 006265      REWIND
03013 006271      STATUS
03014 100201      100201
03015 000770      JMP .-2
03016 020222      A55: LDA 0,SWTSV
03017 024123      LDA 1,C1000
03020 107404      AND 0,1,SZR
03021 000417      JMP RSTP
03022 006250      ETST: JSR @PINWL
03023 006265      REWIND
03024 006271      STATUS
03025 100205      100205
03026 000403      JMP .+3      ;RING IN
03027 006251      JSR @IMESS
03030 003147      PRB
03031 062677      IORST
03032 006265      REWIND
03033 006271      STATUS
03034 100201      100201
03035 000774      JMP .-4
03036 006241      JSR @PDLTA
03037 006245      JSR @PETTA
03040 006251      RSTP: JSR @IMESS
03041 003143      MCYC
03042 034045      LDA 3,45
03043 011403      ISZ 3,3

```

```

03044 000401      JMP .+1
03045 006265      REWIND
03046 006271      STATUS
03047 100201      100201
03050 000776      JMP .-2
03051 002401      JMP @.+1
03052 000411      LPRT:  A0
03053 003054      LSTRT: LST79
                        ;LONGITUDINAL PARITY DATA LIST
                        ;7-9 TRACK

03054 004000      LST79:  4000          ;R82-P
03055 010000      10000         ;3-P
03056 020000      20000         ;4-P
03057 000400      400           ;5-P
03060 001000      1000          ;6-P
03061 002000      2000          ;7-P
03062 030000      30000         ;2-3
03063 014000      14000         ;3-4
03064 006000      6000          ;4-5
03065 003000      3000          ;5-6
03066 001400      1400          ;6-7
03067 000600      600           ;7-2
03070 000000      0
                        ;9 TRACK ONLY

03071 100000      LST9:  100000        ;0-P
03072 040000      40000         ;1-P
03073 140000      140000        ;0-1
03074 120000      120000        ;0-2
03075 060000      60000         ;1-2
03076 000000      0
03077 000000      0
                        ;BAD TAPE STATUS WRITE BUFFER

03100 020101      BT0F:  020101        ;1
03101 010202      010202        ;2
03102 004404      004404        ;3
03103 003010      003010        ;4
03104 000000      000000        ;5
03105 000202      000202        ;6
03106 100420      100420        ;7
03107 020041      020041        ;8
                        ;CRC CHECK WRITE BUFFERS

03110 000200      PAT1:  000200
03111 000200      000200
03112 153727      153727
03113 177474      PAT2:  177474
03114 136074      136074
03115 112327      112327
03116 052652      PAT3:  052652
03117 076232      076232
03120 037047      037047
03121 167141      PAT4:  167141
03122 000200      000200
03123 063550      063550
03124 000077      PAT5:  77
03125 000077      77

```

```

03126 003077      77
03127 037400 PAT6: 037400
03130 037400      037400
03131 037400      037400
03132 012452 PAT7: 012452
03133 012452      012452
03134 012452      012452
03135 012452 PAT8: 012452
03136 012477      12477
03137 012425      12425
03140 012452 PAT9: 012452
03141 012477      12477
03142 010477      10477
03143 005215 MCYC: .TXTE          !<15><12>
03144 054703 CY
03145 146303 CL
03146 000305 E!
03147 005215 PRB:  .TXTE          !<215><12>
03150 052520 PU
03151 120324 T
03152 144722 RI
03153 043516 NG
03154 041240 B
03155 141501 AC
03156 120113 K
03157 047317 ON
03160 152240 T
03161 050101 AP
03162 000305 E!
; MTA DIAGNOSTIC INITIALIZATION
03163 054521 INITIAL:STA 3,IRET
03164 102440      SUBO 0,0
03165 034045      LDA 3,45
03166 041403      STA 0,3,3
03167 021400      LDA 0,0,3
03170 101005      MOV 0,0,SNR
03171 000403      JMP HENA
03172 035401      LDA 3,1,3
03173 000403      JMP HENA+2
03174 063077 HENA:  HALT          ;SET SWITCHES AND CONTINUE
03175 074477      READS 3
03176 054222      STA 3,SWTSV
03177 161000      MOV 3,0
03200 024100      LDA 1,C7
03201 030101      LDA 2,C10
03202 137400      AND 1,3          ;AC3=UNIT#
03203 173000      ADD 3,2          ; AC2=UNIT + 10
03204 050211      STA 2,C1X        ;STORE UNIT # + 10
03205 054212      STA 3,CX
03206 034105      LDA 3,C40
03207 173000      ADD 3,2
03210 050213      STA 2,C5X
03211 050214      STA 2,CC5X
03212 030103      LDA 2,C22
03213 163400      AND 3,0          ; AC0=40 OR 00

```

```

03214 143000      ADD 2,0
03215 040221      STA 0,DEVICE
;
03216 030055      LDA 2,FIRST
GMOR:             LDA 0,0,2
03217 021000      LDA 1,C160077
03220 024146      AND 1,0
03221 123400      LDA 1,@PCDCM
03222 026147      SUB# 0,1,SZR
03223 106414      JMP DEV1           ;NOT MTA
03224 000407
03225 021000      LDA 0,0,2
03226 024150      LDA 1,C177700
03227 123400      AND 1,0
03230 024221      LDA 1,DEVICE
03231 123000      ADD 1,0
03232 041000      STA 0,0,2
DEV1:             INC 2,2
03233 151400      LDA 0,CEND
03234 020056      SUB# 0,2,SZR
03235 112414      JMP GMOR
;CALCULATE DILLY#
;-(R ITRS OF INC-JMP LOOP IN 1 MS -2)
03237 062677      IORST
03240 020170      LDA 0,MSKTO
03241 062077      MSKO 0
03242 020000      LDA 0,0
03243 040442      STA 0,DILLA
03244 020001      LDA 0,1
03245 040441      STA 0,DILLB
03246 020237      LDA 0,PDILL
03247 040001      STA 0,1
03250 102440      SUBO 0,0
03251 126440      SUBC 1,1
03252 065111      DOAS 1,TTO
03253 063511      SKPBZ TTO
03254 000777      JMP .-1
03255 060211      NIOC TTO
03256 060177      INTEN
03257 065111      DOAS 1,TTO
03260 123400      INC 1,1
03261 000777      JMP .-1
DILLI:           MOVR 1,2
03262 131200      MOVL 2,3
03263 155100      LDA 3,45
03264 034045      LDA 3,2,3
03265 035402      MOV 3,3,SZR
03266 175004      ADD 2,1
03267 147000      MOVL 0,0
03270 101100      LDA 2,DL100
03271 030171      JSR @PDIVD
03272 006240      NEG 1,1
03273 124400      INC 1,1
03274 125400      INC 1,1
03275 125400      INC 1,1
03276 000401      JMP .+1
03277 020406      LDA 0,DILLA

```

```

03300 040000      STA 0,0
03301 020495      LDA 0,DILLB
03302 040001      STA 0,1
03303 002401      JMP @IRET
03304 000000      IRET: 0
03305 000000      DILLA: 0
03306 000000      DILLB: 0
;COMPARE IBUFF WITH OBUFF
;
;
;          CHECK
;          ARG
;# WORDS IS IN ARG FOLLOWING CAL
;COMPARE WORD FOR WORD
;WORD N+1 IN IBUFF MUST BE ZERO
;IF ERROR RETURN TO CALL +2
;OTHERWISE CALL+3
;EXIT WITH AC1=BAD WORD
;AC0=GOOD WORD
;AC2=ADDR. OF IBUFF
03307 054447      XCHK:  STA 3,RCHK
03310 024060      LDA 1,OBUFF
03311 044022      STA 1,IDX2
03312 014022      DSZ IDX2
03313 024061      LDA 1,IBUFF
03314 044023      STA 1,IDX3
03315 014023      DSZ IDX3
03316 031400      LDA 2,0,3          ;GET ARG
03317 050436      STA 2,CNTR
03320 034132      LDA 3,C37477      ;7TRK DATA MASK
03321 030222      LDA 2,SWTSV
03322 020125      LDA 0,C2100
03323 113400      AND 0,2          ;AC2=0 IF 7 TRK
03324 022022      XCHK.:  LDA 0,@IDX2
03325 151005      MOV 2,2,SNR
03326 163400      AND 3,0          ;MASK IF 7TRK
03327 026023      LDA 1,@IDX3
03330 106414      SUB# 0,1,SZR
03331 000406      JMP LEAV
03332 014423      DSZ CNTR
03333 000771      JMP XCHK.
03334 026023      LDA 1,@IDX3
03335 125005      MOV 1,1,SNR          ;WORD FOLLOWING LAST
;MUST BE ZERO
03336 010420      ISZ RCHK
03337 010417      LEAV:  ISZ RCHK
03340 030061      LDA 2,IBUFF
03341 002415      JMP @RCHK
;CLEAR INPUT BUFFER TO ZEROS
03342 020061      XCLR:  LDA 0,IBUFF
03343 126000      ADC 1,1
03344 123000      ADD 1,0
03345 040022      STA 0,IDX2
03346 020112      LDA 0,C100
03347 040406      STA 0,CNTR
03350 102400      SUB 0,0

```

```

03351 042022      STA 0,@IDX2
03352 014403      DSZ CNTR
03353 000776      JMP .-2
03354 001400      JMP 0,3
03355 000000      CNTR: 0
03356 000000      RCHK: 0
;
;DO A SPACE FORWARD
03357 024104      XSPC: LDA 1,C30
03360 030212      LDA 2,CX
03361 147000      ADD 2,1
03362 152000      ADC 2,2
03363 073022      DCC 2,.MTA      ;WC=1
03364 065122      DOAS 1,.MTA    ;DO IT
03365 001400      JMP 0,3
;
;REWIND SUBROUTINE
03366 024211      XRWD: LDA 1,C1X
03367 065122      DOAS 1,.MTA
03370 001400      JMP 0,3
;REWIND FOLLOWED BY A 5 SEC STALL
03371 054412      XRWS: STA 3,RRWS
03372 024211      LDA 1,C1X
03373 102400      SUB 0,0
03374 040223      STA 0,ITLP
03375 065122      DOAS 1,.MTA
03376 006272      TIME
03377 020000      20000
03400 000201      201
03401 000401      JMP .+1
03402 002401      JMP @RRWS
03403 000000      RRWS: 0
;BACKSPACE ROUTINE
03404 024212      XBSP: LDA 1,CX
03405 030105      LDA 2,C40
03406 147900      ADD 2,1
03407 102000      ADC 0,0
03410 063022      DCC 0,.MTA
03411 065122      DOAS 1,.MTA
03412 001400      JMP 0,3
;WRITE EOF
03413 054415      XWEOF: STA 3,RWEOF
03414 024212      LDA 1,CX
03415 030107      LDA 2,C60
03416 034222      LDA 3,SWTSV
03417 174000      COM 3,3
03420 020112      LDA 0,C100
03421 117400      AND 0,3
03422 173000      ADD 3,2
03423 147000      ADD 2,1
03424 065122      DOAS 1,.MTA
03425 006273      TMDONE
03426 003000      3000
03427 002401      JMP @RWEOF
03430 000000      RWEOF: 0
;LOAD OBUFF WITH 3 WORDS.
;

```

```

;
;          LOAD
;          ARG
;
;ARGUMENT=ADDR OF 3 WORD FIELD
03431 031400 XLD: LDA 2,0,3
03432 050023 STA 2,IDX3
03433 030060 LDA 2,0BUFF
03434 050024 STA 2,IDX4
03435 014024 DSZ IDX4
03436 102000 ADC 0,0
03437 030222 LDA 2,SWTSV
03440 024125 LDA 1,C2100
03441 133405 AND 1,2,SNR
03442 020132 LDA 0,C37477
03443 024160 LDA 1,M3
03444 032023 LDA 2,@IDX3
03445 113400 AND 0,2
03446 052024 STA 2,@IDX4
03447 125404 INC 1,1,SZR
03450 000774 JMP .-4
03451 175400 INC 3,3
03452 001400 JMP 0,3
;GENERATE ARGUMENT DATA INTO OBUFF
;ALWAYS GENERATE 64 WORDS ALL SAME
;
;          GEN
;          ARG
;
03453 054422 XGEN: STA 3,RGEN
03454 010421 ISZ RGEN
03455 031400 LDA 2,0,3 ;GET DATA WORD
03456 102000 ADC 0,0
03457 024222 LDA 1,SWTSV
03460 034125 LDA 3,C2100
03461 167405 AND 3,1,SNR
03462 020132 LDA 0,C37477
03463 113400 AND 0,2
03464 024060 LDA 1,0BUFF
03465 044020 STA 1,IDX0 ;SET TRD
03466 014020 DSZ IDX0
03467 024165 LDA 1,M100 ;LOAD 64 COUNTER
03470 044215 STA 1,CTR
03471 052020 STA 2,@IDX0
03472 010215 ISZ CTR
03473 000776 JMP .-2
03474 002401 JMP @RGEN
03475 000000 RGEN: 0
;WAIT 50MS THEN SELECT UNIT
03476 054406 XSEL: STA 3,RSF1
03477 006300 WAIT
03500 000070 ,50MS
03501 020212 LDA 0,CX
03502 061022 DOA 0,.MTA
03503 002401 JMP @RSF1
03504 000000 RSF1: 0

```



```

;READ NO STALL (REDNS)
;JUST LIKE (READ) EXCEPT THAT ONLY
;ONE RECORD IS READ AND A RETURN
;IS EXECUTED IMMEDIATELY.
03505 102000 XRDNS: ADC 0,0
03506 040444 STA 0,RDSW ;SET SWITCH
03507 000403 JMP XRD+2
;READ SUBROUTINE
;ARGUMENT FOLLOWING CALL SPECIFIES
;# WORDS. AND PARITY TYPE
;BITS 10-15=WORDS BIT 0= 1 FOR EVEN
;BITS 4-9=RECORDS 0 FOR ODD
;200 MS DELAY BETWEEN EACH READ.
;NO STATUS,BUSY,OR DONE CHECKING?
03510 102400 XRD: SUB 0,0
03511 040441 STA 0,RDSW
03512 054437 STA 3,RRD
03513 010436 ISZ RRD
03514 031400 LDA 2,0,3
03515 020111 LDA 0,C77
03516 113400 AND 0,2
03517 150400 NEG 2,2
03520 050217 STA 2,WDCNT ;SAVE WORD COUNT
03521 031400 LDA 2,0,3
03522 102400 SUB 0,0
03523 151112 MOVL# 2,2,SZC ;SKIP IF ODD PAR
03524 020112 LDA 0,C100
03525 024212 LDA 1,CX
03526 123000 ADD 1,0 ;AC0=READ INST.
03527 040220 STA 0,INST
03530 020131 LDA 0,C7700
03531 143400 AND 2,0 ;AC0=#RECX100
03532 024112 LDA 1,C100 ;AC1=100
03533 030217 XRD.1: LDA 2,WDCNT
03534 073022 OCC 2,.MTA ;SET WC
03535 030061 LDA 2,IBUFF
03536 072022 DOB 2,.MTA ;SET CA
03537 030220 LDA 2,INST
03540 071122 DOAS 2,.MTA ;START READ
03541 030411 LDA 2,RDSW ;CHECK SWITCH
03542 151004 MOV 2,2,SZR
03543 002406 JMP @RRD
03544 006273 TMDONE
03545 003000 3000
03546 122404 SUB 1,0,SZR
03547 000764 JMP XRD.1
03550 002401 JMP @RRD
03551 000000 RRD: 0
03552 000000 RDSW: 0
;WRITE WD STALL (WRTNS)
;JUST LIKE (WRITE) EXCEPT THAT ONLY
;ONE RECORD IS WRITTEN AND RETURN
;IS EXECUTED IMMEDIATELY
03553 102000 XWNS: ADC 0,0
03554 040444 STA 0,SWTCH

```

```

03555 000403      JMP XWRT+2
                  ;WRITE SUBROUTINE
                  ;ARGUMENT FOLLOWING CALL SPECIFIES
                  ;#RECORDS AND WORDS, AND PARITY.
                  ;BITS10-15=#WORDS      BIT 0 = 1 FOR EVEN
                  ;BITS 4-0=#RECORDS      0 FOR ODD
                  ;200 MS DELAY BETWEEN WRITES
                  ;NO STATUS,BUSY,DONE,CHECKING!

03556 102400      XWRT:  SUB 0,0
03557 040441      STA 0,SWTCH
03560 054437      STA 3,RWRT
03561 010436      ISZ RWRT
03562 031400      LDA 2,0,3
03563 020111      LDA 0,C77
03564 113400      AND 0,2
03565 150400      NEG 2,2
03566 050217      STA 2,WDCNT      ;SAVE WORD COUNT
03567 031400      LDA 2,0,3
03570 102400      SUB 0,0
03571 151112      MOVL# 2,2,SZC      ;SKIP IF ODD PAR
03572 020112      LDA 0,C100
03573 024213      LDA 1,C5X
03574 123000      ADD 1,0
03575 040220      STA 0,INST      ;SAVE INST
03576 020131      LDA 0,C7700
03577 143400      AND 2,0      ;AC=-#RFCX100
03600 024112      LDA 1,C100
03601 030217      XWRT.:  LDA 2,WDCNT
03602 073222      DOCC 2, .MTA      ;SET WC
03603 030060      LDA 2,0BUFF
03604 072022      DOB 2, .MTA      ;SET CA
03605 030220      LDA 2,INST
03606 071122      DOAS 2, .MTA      ;START WRITE
03607 030411      LDA 2,SWTCH      ;CHECK SWITCH
03610 151004      MOV 2,2,SZR
03611 002406      JMP @RWRT
03612 006273      TMDONE
03613 003000      3000
03614 122404      SUB 1,0,SZR
03615 000764      JMP XWRT.
03616 002401      JMP @RWRT
03617 000000      RWRT:  0
03620 000000      SWTCH: 0
                  ;PHASE ROUTINE
03621 020212      XERAS:  LDA 0,CX
03622 054407      STA 3,XERAR
03623 024110      LDA 1,C70
03624 123000      ADD 1,0
03625 061122      DOAS 0, .MTA
03626 006273      TMDONE
03627 003000      3000
03630 002401      JMP @XERAR
03631 000000      XERAR:  0
                  ;CHECK STATUS FOR TUR AND ERASE.
                  ;AC0@UNIT BEING SELECTED.

```

```

                                ;RETURN+1 IF ERROR
03632 054426 CSTAT: STA 3,CSTAR
03633 034222 LDA 3,SWTSV
03634 030130 LDA 2,C4K
03635 173405 AND 3,2 SNR ;LOOK FOR 2 RDY UNITS
03636 000403 JMP .+3 ;IF SW4 TRUE
03637 024075 LDA 1,C3
03640 000402 JMP .+2
03641 024100 LDA 1,C7
03642 123400 AND 1,0 ;ACO=ACTUAL UNIT BEING SEL.
03643 137400 AND 1,3 ;ACO=UNIT IN SWR
03644 162415 SUB# 3,0,SNR
03645 000407 JMP ON ;TUR SHOULD BE ON
03646 126520 SUBZL 1,1 ;UNIT NOT SEL.
03647 060422 DIA 0,.MTA
03650 123415 AND# 1,0,SNR ;CORRECT STATUS = 0
03651 010407 OKRET: ISZ CSTAR ;OK, RETURN+2
03652 030210 LDA 2,X70 ;X70=UNIT#+70
03653 002405 JMP @CSTAR
03654 006271 ON: STATUS
03655 100201 100201
03656 000774 JMP OKRET+1 ;ERROR, AC1=GOOD STAT.
03657 000772 JMP OKRET ; ACO=BAD
03660 000000 CSTAR: 0
                                ;DELAY SUBROUTINE
                                ;ARGUMENT FOLLOWING CALL IS ADDRESS
                                ;OF DELAY CONSTANT.
                                ;DELAY IN INCREMENTS IF 1MS
                                ;AC2&3 ARE USED
                                ;OVERHEAD: NOVA, SUPER, 1200, 800, SC1 MICROSEC.
                                ;46.1-2.0/MS 15.21 24.61 14.6-.6/MS 11.4-.3/MS
03661 054417 XWT: STA 3,RDLAY
03662 044415 STA 1,XWTA1
03663 024216 LDA 1,DILLY ;-(# ITRS LOOP-2)
03664 037400 LDA 3,20,3 ;#MS
03665 174400 NEG 3,3
03666 131000 MOV 1,2 ;THESE TWO INSTRUCTIONS
03667 000401 JMP .+1 ;TAKE TIME OF ONE ITERATION
03670 151404 INC 2,2,SZR
03671 000777 JMP .-1
03672 175404 INC 3,3,SZR ;THESE TWO TAKE
03673 000773 JMP .-5 ;TIME OF ONE ITERATION
03674 024403 LDA 1,XWTA1
03675 034403 LDA 3,RDLAY
03676 001401 JMP 1,3
03677 000000 XWTA1: 0 ;TEMP SAVE FOR AC1
03700 000000 RDLAY: 0 ;RETURN ADDR
                                ;
                                ; STATUS
                                ; ARG
                                ;
                                ;COMBINE ARGUMENT STATUS WITH SWR
                                ;STATUS AND CHECK AGAINST ACTUAL,
                                ;RETURN+2 IF STATUS MATCH, +1 OTHERWISE
                                ;EXIT WITH AC1=EXPECTED STATUS, ACO=ACTUAL
                                ;

```

```

03701 054456 XSTAT: STA 3,STATR
03702 010455 ISZ STATR
03703 025400 LDA 1,0,3
03704 034222 LDA 3,SWTSV
03705 030112 LDA 2,C100 ;9TRK=1
03706 157400 AND 2,3
03707 167000 ADD 3,1
03710 034222 LDA 3,SWTSV
03711 030117 LDA 2,C200 ;HI DENS=1
03712 157405 AND 2,3,SNR
03713 000403 JMP .+3
03714 034130 LDA 3,C4K
03715 167000 ADD 3,1 ;AC1=EXPECTED STATUS
03716 030252 LDA 2,ONLOC
03717 034440 LDA 3,STATR
03720 156404 SUB 2,3,SZR
03721 000413 JMP XSTA1
03722 030210 LDA 2,X70
03723 034222 LDA 3,SWTSV
03724 172400 SUB 3,2
03725 034100 LDA 3,C7
03726 173405 AND 3,2,SNR
03727 000405 JMP XSTA1
03730 030130 LDA 2,C4K
03731 133415 AND# 1,2,SNR
03732 147001 ADD 2,1,SKP
03733 146400 SUB 2,1
03734 060422 XSTA1: DIA 0,.MTA ;AC0=ACTUAL STATUS
03735 034122 LDA 3,C400
03736 030222 LDA 2,SWTSV
03737 157405 AND 2,3,SNR ;EXIT IF NOT PE
03740 000414 JMP XSTT.
03741 034101 LDA 3,C10
03742 117405 AND 0,3,SNR ;EXIT IF NOT CPE
03743 000411 JMP XSTT.
03744 152400 SUB 2,2
03745 034155 LDA 3,C5164
03746 117405 AND 0,3,SNR ;MASK ERR IF PE ONLY ERR
03747 030156 LDA 2,C1M
03750 034154 LDA 3,C2010
03751 157000 ADD 2,3
03752 174000 COM 3,3 ;FOR BIT 12
03753 163400 AND 3,0
03754 106415 XSTT.: SUB# 0,1,SNR
03755 010402 ISZ STATR ;OK
03756 002401 JMP @STATR
03757 000000 STATR: 0
;DATA LATE TEST
;CONTINUALLY WRITE 4K RECORDS
;CHECK STATUS AFTER EACH WRITE
;DO CONSOLE SWITCH
; STOP - CONTINUE TO CAJSE DATA LATE

03760 006247 DLT: JSR @INTIL
03761 004411 JSR DLTA
03762 006251 JSR @IMESS

```

03763	003143	MCYC	
03764	063511	SKPBZ TTO	
03765	000777	JMP .-1	
03766	034045	LDA 3,45	
03767	011403	ISZ 3,3	
03770	000771	JMP DLT+1	
03771	000770	JMP DLT+1	
03772	054476	DLTA: STA 3,DLTB	
03773	102440	SUBO 0,0	
03774	040475	STA 0,DLTC	
03775	040475	STA 0,DLTD	
03776	040475	STA 0,DLTE	
03777	020063	LDA 0,PCHIN	
04000	040030	STA 0,30	
04001	024136	LDA 1,M2000	
04002	152620	SUBZR 2,2	
04003	141001	MOV 2,0,SKP	
04004	020030	LDA 0,30	
04005	143000	ADD 2,0	
04006	042030	STA 0,230	
04007	125404	INC 1,1,SZR	
04010	000774	JMP .-4	
04011	006265	REWIND	
04012	024117	LDA 1,C200	
04013	060422	DIA 0,MTA	:LOOK FOR BOT
04014	123405	AND 1,0,SNR	
04015	000776	JMP .-2	
04016	006302	DLT.1: JMP WCPST	:SET CA & WC TO 0
04017	020213	LDA 0,C5X	
04020	061122	DOAS 0,MTA	:START WRITE
04021	020452	LDA 0,DLTE	
04022	101202	MOVR 0,0,SZC	
04023	022062	LDA 0,2PPCHN	:1026 MEMORY REFERANCES
04024	063622	SKPDN ,MTA	
04025	000774	JMP .-4	
04026	064422	DIA 1,MTA	
04027	030134	LDA 2,C40000	
04030	133414	AND# 1,2,SZR	:DATA LATE?
04031	010440	ISZ DLTC	:YES
04032	020441	NODL: LDA 0,DLTE	:NO
04033	010440	ISZ DLTE	
04034	101203	MOVR 0,0,SNC	
04035	000415	JMP NODLA	
04036	020433	LDA 0,DLTC	
04037	101004	MOV 0,0,SZR	
04040	000412	JMP NODLA	
04041	020431	LDA 0,DLTD	
04042	101005	MOV 0,0,SNR	
04043	000404	JMP .+4	
04044	006251	JSR @IMESS	
04045	004404	NODLB	
04046	000404	JMP NODLA	
04047	006251	JSR @IMESS	
04050	004416	NODLC	
04051	010421	ISZ DLTD	

```

04052 063511 NODLA: SKPBZ TTO
04053 000777 JMP .-1
04054 020415 LDA 0,DLTC
04055 101004 MOV 0,0,SZR ;ESCAPE IF 1 DATA LATE DET.
04056 002412 JMP @DLTB
04057 030123 LDA 2,C1000
04060 064422 DIA 1,.MTA
04061 147405 AND 2,1,SNR ;EOT?
04062 000734 JMP DLT.1 ;NO
04063 006251 JSR @IMESS
04064 004404 NODLB
04065 063511 SKPBZ TTO
04066 000777 JMP .-1
04067 002401 JMP @DLTB
04070 000000 DLTB: 0
04071 000000 DLTC: 0
04072 000000 DLTD: 0
04073 000000 DLTE: 0
;END TAPE TEST. WRITE TO EOT AND
;HALT AFTER MESSAGE. NOTE ERRORS
;ALONG THE WAY

04074 006247 ETT: JSR @INTIL
04075 004412 JSR ETTA
04076 006251 JSR @IMESS
04077 003143 MCYC
04100 063511 SKPBZ TTO
04101 000777 JMP .-1
04102 004246 ERHLT ;END TAPE OK
04103 034045 LDA 3,45
04104 011403 ISZ 3,3
04105 000770 JMP ETT+1
04106 000767 JMP ETT+1
04107 054426 ETTA: STA 3,ETTB
04110 062677 IORST ;SET CA & WC TO 0
04111 006277 WRITE
04112 000177 177 ;WRITE
04113 063522 SKPBZ .MTA
04114 000777 JMP .-1
04115 064422 DIA 1,.MTA ;READ STATUS
04116 030123 LDA 2,C1000
04117 147414 AND# 2,1,SZR
04120 000412 JMP ENDT ;ENO TAPE
04121 064422 FTT.1: DIA 1,.MTA
04122 030172 LDA 2,C70644
04123 147415 AND# 2,1,SNR
04124 000764 JMP ETTA+1 ;NO EOT, NO ERRORS
04125 006251 JSR @IMESS
04126 004434 MSTAR
04127 064422 DIA 1,.MTA
04130 006253 JSR @IPOCT
04131 000757 JMP ETTA+1
04132 006251 ENDT: JSR @IMESS
04133 004455 MENOT
04134 002401 JMP @ETTB
04135 000000 ETTB: 0

```

```

;WRITE LOCK TEST
04136 006247 WLT: JSR @INTIL
04137 004411 JSR INWLT
04140 006251 JSR @IMESS
04141 003143 MCYC
04142 063511 SKPBZ TTO
04143 000777 JMP .-1
04144 034045 LDA 3,45
04145 011403 ISZ 3,3
04146 000771 JMP WLT+1
04147 000770 JMP WLT+1
04150 054456 INWLT: STA 3,RINWL
04151 062677 IORST
04152 006265 REWIND
04153 006271 STATUS
04154 100205 100205
04155 000402 JMP .+2 ;RING IN OR OTHER TROUBLE
04156 000404 JMP .+4 ;RING OUT
04157 006271 STATUS
04160 100201 100201
04161 000772 JMP .-6 ;RING OUT OR OTHER TROUBLE
04162 006271 STATUS ;RING IN
04163 100201 100201 ;SKP IF RING IN
04164 000417 JMP INWLA
04165 006251 JSR @IMESS ;RING IN
04166 004334 INWLB
04167 020135 LDA 0,D3000
04170 040437 STA 0,INWLD
04171 006271 STATUS
04172 100201 100201
04173 000410 JMP INWLA ;GOT RING OUT OR UNRDY
04174 006300 WAIT
04175 000112 C100
04176 014431 DSZ INWLD
04177 000772 JMP .-6
04200 006251 JSR @IMESS ;WAITED LONG TIME
04201 004365 INWLH
04202 002424 JMP @RINWL
04203 020161 INWLA: LDA 0,M4
04204 040423 STA 0,INWLD
04205 062677 INWLF: IORST
04206 006265 REWIND
04207 006271 STATUS
04210 100205 100205
04211 000774 JMP .-4 ;RING IN OR OTHER
04212 006277 WRITE ;OK RING OUT
04213 000102 102
04214 063522 SKPBZ .MTA
04215 000777 JMP .-1
04216 006271 STATUS
04217 110205 110205
04220 000404 JMP INWLC ;NOT ILLEGAL OR OTHER ERROR
04221 062677 IORST ;ILLEGAL
04222 000401 .+1
04223 002403 JMP @RINWL

```



```

; ON: MODIFY STAT TO 2X
; OFF: MODIFY STAT TO 0X
; EXIT RETURN
;
04273 054426 XTPN: STA 3,TPNR
04274 175400      INC 3,3
04275 024222      LDA 1,SWTSV
04276 030122      LDA 2,C400
04277 133404      AND 1,2,SZR      ;IS TAPE PE
04300 000407      JMP XTPN.
04301 024102 XTPN1: LDA 1,C20      ;NO
04302 124000      COM 1,1          ;CHANGE BIT 11 TO ZERO
04303 031400      LDA 2,0,3
04304 133400      AND 1,2
04305 051400      STA 2,0,3
04306 002413      JMP @TPNR
04307 030223 XTPN.: LDA 2,ITLP
04310 153404      AND 2,2,SZR
04311 000770      JMP XTPN1        ;IF NOT FIRST OP AFTER LP
04312 031400      LDA 2,0,3        ;CHANGE BIT 11 TO 1
04313 024102      LDA 1,C20
04314 124000      COM 1,1
04315 133400      AND 1,2
04316 132000      ADC 1,2
04317 051400      STA 2,0,3
04320 002401      JMP @TPNR
04321 000000 TPNR: 0
;SKIP ON NRZ MODE
;
; TEST SWITCH 7
; RETURN IF 1
; INC RETURN IF 0
;
04322 054407 XBPPE: STA 3,BPPER
04323 024222      LDA 1,SWTSV
04324 030122      LDA 2,C400
04325 133404      AND 1,2,SZR      ;EXIT IF NRZ
04326 002403      JMP @BPPER
04327 010402      ISZ BPPER
04330 002401      JMP @BPPER
04331 000000 BPPER: 0
04332 060022 CDCM: 060022
04333 000000 ENDPRG: 0
04334 005215 INWLB: ,TXTE      !<215><12>
04335 142722 RE
04336 147515 MO
04337 142526 VE
04340 153640 W
04341 144722 RI
04342 142724 TE
04343 142640 E
04344 040516 NA
04345 146102 BL
04346 120305 E
04347 144722 RI

```

04350	043516	NG		
04351	120056	.		
04352	147504	DO		
04353	152116	NT		
04354	051640	S		
04355	147724	TO		
04356	120120	P		
04357	151120	PR		
04360	043717	OG		
04361	040722	RA		
04362	027115	M.		
04363	120240			
04364	000240	!		
04365	005215	INWLH:	.TXTE	!<215><12>
04366	151327	WR		
04367	152311	IT		
04370	120305	E		
04371	147714	LO		
04372	045703	CK		
04373	041240	B		
04374	152311	IT		
04375	147640	O		
04376	143306	FF		
04377	120072	:		
04400	030261	10		
04401	131060	02		
04402	130460	01		
04403	000240	!		
04404	005215	NODLB:	.TXTE	!<215><12>
04405	147516	NO		
04406	042240	D		
04407	152101	AT		
04410	026501	A-		
04411	040714	LA		
04412	142724	TE		
04413	041240	B		
04414	152311	IT		
04415	000000	!		
04416	005215	NODLC:	.TXTE	!<215><12>
04417	151120	PR		
04420	051705	ES		
04421	120123	S		
04422	152123	ST		
04423	050317	OP		
04424	040640	A		
04425	042116	ND		
04426	141640	C		
04427	047317	ON		
04430	144724	TI		
04431	052516	NU		
04432	120305	E		
04433	000000	!		
04434	005215	MSTAR:	.TXTE	!<15><12>
04435	151305	ER		
04436	147722	RO		

```

04437 120322 R
04440 152123 ST
04441 152101 AT
04442 051525 US
04443 143240 F
04444 146317 OL
04445 147714 LO
04446 144727 WI
04447 043516 NG
04450 153640 W
04451 144722 RI
04452 142724 TE
04453 120240
04454 000000 !
04455 005215 MENDT: .TXTE !<15><12>
04456 047305 EN
04457 120104 D
04460 040724 TA
04461 142520 PE
04462 000000 !
04463 054424 STA 3,LOOPR
04464 176520 SUBZL 3,3
04465 000403 JMP .+3

;*****
; SET UP *
;*****

04466 054421 ENTER: STA 3,LOOPR ;LOOP ITERATE RETURN
04467 034410 LDA 3,ITR ;THIS ROUTINE INITIALIZES
04470 054410 STA 3,ITRCT ;EACH TEST
04471 176400 SUB 3,3
04472 054223 STA 3,ITLP
04473 054406 STA 3,ESWIT
04474 054406 STA 3,ERRCT
04475 062677 IORST ;.I/O RESET
04476 002411 JMP @LOOPR
04477 000012 ITR: 12
04500 000000 ITRCT: 0
04501 000000 ESWIT: 0
04502 000000 ERRCT: 0
04503 000000 RETURN: 0
04504 000000 SAV2: 0
04505 000000 SAV1: 0
04506 000000 SAV0: 0
04507 000000 LOOPR: 0

;*****
; LOOP *
;*****

04510 054773 CYCLE: STA 3,RETURN ;END OF TEST ITERATION
04511 050773 STA 2,SAV2 ;ROUTINE
04512 044773 STA 1,SAV1 ;SAVE THE ACS!
04513 040773 STA 0,SAV0
04514 010223 ISZ ITLP
04515 014763 DSZ ITRCT
04516 000436 JMP CYCTS ;NOT 100 T1=FS ITERATED
04517 034760 LDA 3,ITR ;RESET ITERATION CNTR

```

```

04520 054760      STA 3,ITRCT
04521 074477      READS 3
04522 030757      LDA 2,ESWIT      ;IF SWITCH 2=(1)
04523 175120      MOVZL 3,3        ;AND A ERROR HAS OCCURED
04524 175100      MOVL 3,3        ;THE ERROR RATE XTL1
04525 151005      MOV 2,2,SNR    ;BE PRINTED
04526 000417      JMP NOEX
04527 175103      MOVL 3,3,SNC
04530 000421      JMP PCENT-1
04531 006234      JSR @ICRLF      ;PPTET CARRIAGE
04532 024750      LDA 1,ERRCT
04533 030115      LDA 2,C144
04534 004507      JSR MULT        ;AC1XAC2
04535 030742      LDA 2,ITR
04536 004473      JSR DIVID      ;AC2-1/AC2
04537 006467      JSR @IPDEC      ;PRINT VALUE
04540 020412      LDA 0,PCENT    ;EXAMPLE: 307
04541 006463      JSR @ICHR
04542 063611      SKPDN TTO    ;WAIT DONE FLAG
04543 000777      JMP .-1
04544 000405      JMP PCENT-1
04545 020741      NOEX:  LDA 0,SAV0    ;NORMAL EXIT, NO ERR
04546 024737      LDA 1,SAV1
04547 030735      LDA 2,SAV2
04550 002733      JMP @RETURN
04551 102401      SUB 0,0,SKP
04552 000245      PCENT: 245      ;CHARACTER
04553 040727      STA 0,ERRCT    ;RESET ERROR COUNT
04554 020732      CYCTS:  LDA 0,SAV0    ;RESTORE ACS
04555 024730      LDA 1,SAV1
04556 030726      LDA 2,SAV2
04557 034722      LDA 3,ESWIT
04560 175004      MOV 3,3,SZR
04561 074477      READS 3
04562 062677      IORST
04563 175113      MOVL# 3,3,SNC   ;SWITCH 0
04564 002723      JMP @LOOPR      ;(1)=LOOP ROUTINE
04565 002716      JMP @RETURN    ;(0)=PROCEED TO NEXT TEST

;*****
;  EHALT  *
;*****

04566 054715      ERR:  STA 3,RETURN    ;ERROR SUBROUTINE
04567 050715      STA 2,SAV2
04570 044715      STA 1,SAV1
04571 040715      STA 0,SAV0
04572 034707      LDA 3,ESWIT
04573 175005      MOV 3,3,SNR
04574 000407      JMP ERR1
04575 030707      ERET:  LDA 2,SAV2    ;RESTORE ACS
04576 024707      LDA 1,SAV1
04577 020707      LDA 0,SAV0
04600 010702      ISZ ERRCT    ;COUNT
04601 101000      NOP
04602 002701      JMP @RETURN    ;EXIT
04603 034700      ERR1:  LDA 3,RETURN    ;ERROR. C(31)=0C

```

```

04604 063077      HALT                ;OPERATOR, SET SWITCHES!
04605 054674      STA 3,ESWIT
04606 074477      READS 3
04607 175100      MOVL 3,3
04610 175113      MOVL# 3,3,SNC      ;LOOK AT SWITCH 1
04611 004402      JSR EPRINT          ;PRINT ERROR DATA
04612 000763      JMP ERET
04613 054666      EPRINT: STA 3,ESWIT      ;ERROR MESSAGE PRINTER
04614 006234      JSR @ICRLF          ;PRINT CARRIAGE
04615 006251      JSR @IMESS          ;AND HEADER
04616 004627      HEADER
04617 020664      LDA 0,RETURN
04620 126000      ADC 1,1
04621 107000      ADD 0,1
04622 004456      JSR POCT                ;PC OF ERROR
04623 002656      JMP @ESWIT          ;RETURN TO CALL
04624 004755      ICHAR: CHAR
04625 005026      ITYPE: TYPE
04626 004703      IPDEC: PDEC
04627 041520      HEADER: .TXT !PC
04630 000011      <11>!
04631 054424      DIVID: STA 3,MSAV          ;DIVIDE
04632 034424      LDA 3,M20                ;AC2-1/AC2
04633 125120      MOVZL 1,1              ;RESULT IN AC1
04634 101100      DLOOP: MOVL 0,0
04635 142412      SUB# 2,0,SZC
04636 142400      SUB 2,0
04637 125100      MOVL 1,1
04640 175404      INC 3,3,SZR
04641 000773      JMP DLOOP
04642 002413      JMP @MSAV
04643 102460      MULT: SUBC 0,0          ;MULTIPLY
04644 054411      STA 3,MSAV          ;AC1XAC2
04645 034411      LDA 3,M20                ;ANS IN AC0-1
04646 125203      MLOOP: MOVR 1,1,SNC
04647 101201      MOVR 0,0,SKP
04650 143220      ADDZR 2,0
04651 175404      INC 3,3,SZR
04652 000774      JMP MLOOP
04653 125260      MOVCR 1,1
04654 002401      JMP @MSAV
04655 000000      MSAV: 0
04656 177760      M20: -20
;TTC NOW INTERRUPT PACKAGE
;"MESS" PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBL
;"CHAR" PRINTS ASCII CHARACTER, C(0)R,C(0)L MUST BE 0
;WILL RETURN +2 IF C(0)R=0,CORRECTS THE PARITY,AS SIMUL.
;"TYPE" PRINTS C(0)R. MUST HAVE PROPER PARITY. RETURN IS
;TO CALL+1.REPLACE THIS ROUTINE WITH INTERRUPT TYPE IF
;DESIRED "CRLF" PRINTS A CARRIAGE RETURN
;"CRLF" PRINTS A CARRIAGE RETURN
;"POCT" PRINTS C(1) IN OCTAL FOLLOWED BY A TAB
;"PDEC: PRINTS C(1) IN DECIMAL,LEADING ZEROS SUPPRESSED.
;FOLLOWED BY A TAB.
04657 054546      MESS: STA 3,MESSR          ;PRINT A TEXT MESSAGE

```

```

04660 010545      ISZ MESSR
04661 031400      LDA 2,0,3        ;C(2) POINTS TO MESSAGE
04662 024121      LDA 1,C377       ;A 8 BIT MASK
04663 021000      LDA 0,0,2       ;C(2)=DATA WORD
04664 125112      MOVL# 1,1,SZC
04665 123701      ANDS 1,0,SKP
04666 123401      AND 1,0,SKP    ;C( )=DATA CHARACTER RIGHT
04667 151400      INC 2,2        ;THEN TO NEXT WORD
04670 124000      COM 1,1        ;FLIP BASE
04671 004464      JSR CHAR        ;PRTOT
04672 000771      JMP MESS+4      ;ANOTHER
04673 063511      SKPBZ TIO
04674 000777      JMP .-1
04675 002530      JMP @MESSR     ;LAST
04676 020525      OCT:  LDA 0,CH240
04677 101001      POCT: MOV 0,0,SKP
04700 020107      POCT:  LDA 0,C60
04701 030433      POCT:  LDA 2,OCTAB    ;PRINT C(1) IN OCTAL
04702 000403      POCT:  JMP .+3
04703 030441      PDEC:  LDA 2,DECTB   ;PRINT C(1) IN DECIMAL
04704 020517      PDEC:  LDA 0,CH240   ;SUPPRESS LEADING ZEROS
04705 054447      PDEC:  STA 3,RADRET  ;BOTH ENTRYS PRINT NUMBER
04706 040445      PDEC:  STA 0,ZSUPP   ;THEN TAB TO NEXT POSITION
04707 050401      PDEC:  STA 2,..+1
04710 000000      DECOCT: 0        ;A"LDA 2, TABLE" INSTRUCTION
04711 010777      DECOCT: ISZ .-1
04712 034442      DECOCT: LDA 3,RADRET  ;SETUP "TAB" AT END
04713 020503      DECOCT: LDA 0,CHTAB
04714 151005      DECOCT: MCV 2,2,SNR   ;IF TABLE ENTRY=0
04715 000440      DECOCT: JMP CHAR        ;EXIT WITH TAB
04716 034435      DECOCT: LDA 3,ZSUPP   ;ZEROS SUPPRESS STUFF
04717 102400      DECOCT: SUB 0,0
04720 146512      DECOCT: DECOD:    SUBL# 2,1,SZC
04721 000405      DECOCT: JMP DECP
04722 146400      DECOCT: SUB 2,1    ;FORM THE DIGIT
04723 034107      DECOCT: LDA 3,C60
04724 101400      DECOCT: INC 0,0
04725 000773      DECOCT: JMP DECOD
04726 151235      DECP:  MOVZR# 2,2,SNR
04727 034107      DECP:  LDA 3,C60
04730 054423      DECP:  STA 3,ZSUPP   ;C(0)=DIGIT
04731 163000      DECP:  ADD 3,0     ;MAKE ASCII
04732 004423      DECP:  JSR CHAR        ;PRINT
04733 000755      DECP:  JMP DECOCT   ;SET EXT DIGIT
04734 030425      OCTAB:  LDA 2,..+1+.-DECOCT
04735 100000      OCTAB:  100000
04736 010000      OCTAB:  10000
04737 001000      OCTAB:  1000
04740 000100      OCTAB:  100
04741 000010      OCTAB:  10
04742 000001      OCTAB:  1
04743 000000      OCTAB:  0
04744 030435      DECTB:  LDA 2,..+1+.-DECOCT
04744 000012      DECTB:  .RDX 10
04745 023420      DECTB:  10000

```

```

04746 001750          1000
04747 000144          100
04750 000012          10
04751 000001          1
04752 000000          0
      000010      .RDX 8
04753 000000      ZSUPP: 0
04754 000000      RADRET: 0
04755 054442      CHAR:  STA 3,CHRET      ;PRINT C(0) RIGHT
04756 101325      MOVZS 0,0,SNR      ;RETURN +2 IF NULL
04757 001401      JMP 1,3
04760 040440      STA 0,CHSAV
04761 176000      ADC 3,3      ;COMPUTE THE PARITY
04762 117000      ADD 0,3
04763 163404      AND 3,0,SZR
04764 000775      JMP .-3
04765 176660      SUBCR 3,3      ;COMBINE PARITY WITH CHAR
04766 020432      LDA 0,CHSAV
04767 163300      ADDS 3,0
04770 034426      CHAR1:  LDA 3,CHTAB      ;IS THIS A TAB
04771 116405      SUB 0,3,SNR
04772 000403      JMP .+3      ;YES
04773 004433      JSR TYPE      ;NO PRINT IT
04774 002423      JMP @CHRET      ;EXIT
04775 020424      LDA 0,CHORZ      ;SIMULATE A TAB
04776 034424      LDA 3,CHARZ      ;VIA 1 TO 8 SPACES
04777 117405      AND 0,3,SNR
05000 002417      JMP @CHRET
05001 020422      LDA 0,CH240
05002 004424      JSR TYPE
05003 000772      JMP .-6
05004 034420      CRLF:  STA 3,CRLFR      ;SAVE RETURN
05005 020410      LDA 0,C215
05006 004747      JSR CHAR      ;PRINT CARRIAGE AND LF
05007 020405      LDA 0,C212
05010 004745      JSR CHAR
05011 102400      SUB 0,0
05012 040407      STA 0,CHORZ      ;CLEAR HORZ POSITION
05013 002411      JMP @CRLFR      ;EXIT
05014 000212      C212:  212
05015 000215      C215:  215
05016 000011      CHTAB:  11
05017 000000      CHRET:  0
05020 000000      CHSAV:  0
05021 000000      CHORZ:  0
05022 000007      CHARZ:  7
05023 000240      CH240:  240
05024 000000      CRLFR:  0
05025 000000      MESSR:  0
05026 054412      TYPE:  STA 3,TYPRET      ;TYPE THE C(0)R IF
05027 010772      ISZ CHORZ
05030 074477      READS 3      ;SWITCH 1(0)!
05031 175100      MOVL 3,3
05032 175102      MOVL 3,3,SZC
05033 002405      JMP @TYPRET      ;INWAIT TYPE EXIT.

```

```
05034 063511      SKPBZ TTO
05035 000777      JMP .-1
05036 061111      GOAS 0,TTO
05037 002401      JMP @TYPRET
05040 000000      TYPRET: 0
05041 000000      LAST: 0
                    WCPS:

07442 062677      IORST
07443 020136      LDA Ø, M2ØØØ
07444 063022      DOC Ø, .MTA
07445 001400      JMP Ø, 3
```

```
.END
```


000411	A0	00055	00411	03052
000415	A1	00415		
000606	A10	00606		
000617	A11	00617		
000630	A12	00630		
000642	A13	00642		
000651	A14	00651		
000654	A14.1	00654	00665	
000666	A15	00666		
000702	A16	00702		
000717	A17	00717		
000731	A18	00731		
000743	A18.1	00743	00747	
000754	A18.2	00754	00760	
000765	A19	00765		
000767	A19.1	00767	01003	
000421	A2	00421		
001011	A20	01011		
001013	A20.1	01013		
001032	A21	01032		
001056	A22	01056		
001062	A22.1	01062	01111	
001064	A22.2	01057	01064	01110
001112	A23	01106	01112	
001116	A23.1	01116	01145	
001120	A23.2	01113	01120	01144
001146	A24	01142	01146	
001166	A25	01166		
001175	A26	01175		
001206	A27	01206		
001231	A28	00224	01231	
000425	A3	00425		
001246	A30	01246		
001277	A31	01277		
001326	A32	01326		
001355	A33	01355		
001361	A33.1	01361	01430	
001364	A33.2	01356	01364	01427
001431	A34	01425	01431	
001435	A34.1	01435	01504	
001440	A34.2	01432	01440	01503
001505	A35	01501	01505	
001542	A36	01542		
001577	A37	01577		
001623	A38	01623		
001645	A39	00225	01645	
000431	A4	00431		
001704	A40	01704		
001735	A41	01735		
001770	A42	01770		
002042	A43	02042		
002106	A44	02106		
002152	A45	02152		
002220	A46	00226	02220	
002261	A47	02261		

002322	A48	02322						
002363	A49	02363						
002373	A49.1	00227	02373					
002375	A49.2	00230	02375					
000442	A5	00442						
002433	A50	00231	02217	02433				
002443	A50.1	02443	02452					
002467	A50.2	02467	02475					
002512	A51	02512						
002522	A51.1	02522	02531					
002546	A51.2	02546	02554					
002571	A52	02571						
002601	A52.1	02601	02610					
002625	A52.2	02625	02633					
002650	A53	02650						
002660	A53.1	02660	02667					
002704	A53.2	02704	02712					
002736	A54	02736	02777					
002746	A54.1	02746	03011					
003016	A55	00232	03016					
000451	A6	00451						
000453	A6.1	00453	00473					
000456	A6.2	00452	00456	00462	00472			
000463	A6.3	00463	00471					
000474	A7	00466	00474	00506				
000507	A8	00507						
000517	A8A	00517						
000530	A8B	00530						
000541	A8C	00541						
000552	A8D	00552						
000600	A9	00600						
000563	A8E	00563						
000233	ABTOF	00233	01211					
000246	AHALT	00246	00302					
004331	BPPER	04322	04326	04327	04330	04331		
006255	BSPAC	00302	01170	01235	01305	01612	01634	01716
		02230	02271	02332	02402	02502	02561	02640
		02717						
003100	BTOF	00233	03100					
006254	BYPAS	00302	01206	01577	01774	02727		
000073	C1	00073	04245	04265				
000101	C10	00101	00753	01072	01126	01411	01465	01531
		01566	03201	03741				
000112	C100	00112	00663	01303	01336	01347	01663	01721
		01753	01771	02506	02565	02644	02723	03003
		03346	03420	03524	03532	03572	03600	03705
		04175						
000123	C1000	00123	03017	04057	04116			
000127	C1004	00127	01660					
000113	C103	00113	01321	01763				
000114	C104	00114	01730					
000140	C1042	00140	00521	00564				
000151	C1100	00151	01161					
000152	C1252	00152	00432					
000115	C144	00115	04533					

000116	C150	00116	01216	02761					
000146	C1600	00146	03220						
000150	C1777	00150	03226						
000156	C1M	00156	03747						
000211	C1X	00211	00632	03204	03366	03372			
000074	C2	00074	01021	02027	02073	02137	02203		
000102	C20	00102	04301	04313					
000117	C200	00117	03711	04012					
000124	C2000	00124	01661	02365	02771				
000144	C201	00144							
000154	C2010	00154	03750						
000126	C20K	00126	00711						
000125	C2100	00125	03322	03440	03460				
000141	C2104	00141	00532	00565					
005014	C212	05007	05014						
005015	C215	05005	05015						
000103	C22	00103	03212						
000075	C3	00075	01147	03637					
000104	C30	00104	03357						
000132	C3747	00132	03320	03442	03462				
000120	C375	00120	01262						
000121	C377	00121	01273	04662					
000076	C4	00076	01150						
000105	C40	00105	01046	01257	02463	02542	02621	02700	
		03206	03405						
000122	C400	00122	01674	03735	04276	04324			
000134	C4000	00134	04027						
000133	C4104	00133							
000142	C4210	00142	00543	00566					
000143	C421T	00143	00554	00567					
000130	C4X	00130	03634	03714	03730				
000077	C5	00077							
000106	C50	00106							
000155	C5164	00155	03745						
000153	C5252	00153	00443	00454					
000213	C5X	00213	03210	03573	04017	04111			
000107	C60	00107	03415	04700	04723	04727			
000100	C7	00100	03200	03641	03725				
000110	C70	00110	00651	03625					
000172	C7064	00172	04122						
000111	C77	00111	00464	03515	03563				
000131	C7700	00131	03530	03576					
000214	CC5X	00214	03211						
004332	CDCM	00147	04332						
000056	CEND	00056	03234						
005023	CH240	04676	04704	05001	05023				
004755	CHAR	04624	04671	04715	04732	04755	05006	05010	
004770	CHAR1	04770							
005022	CHARZ	04776	05022						
006256	CHECK	00302	01416	01472	01536	01573	02036	02102	
		02146	02212	02255	02316	02357	02427		
005021	CHORZ	04775	05012	05021	05027				
005017	CHRET	04755	04774	05000	05017				
005020	CHSAV	04760	04766	05020					
005016	CHTAB	04713	04770	05016					

003002	CKED	02745	03002					
006257	CLEAR	00302	01377	01453	02010	02054	02120	02164
		02233	02274	02335	02405			
000057	CMTA	00057	00451					
003355	CNTR	03317	03332	03347	03352	03355		
005004	CRLF	00234	05004					
005024	CRLFR	05004	05013	05024				
003660	CSTAR	03632	03651	03653	03660			
003632	CSTAT	00235	03632					
000215	CTR	00215	00706	00707	00742	00746	00752	00757
		00773	00777	03470	03472			
000212	CX	00212	01215	01260	02016	02062	02126	02172
		02241	02302	02343	02413	02462	02541	02620
		02677	02762	03205	03360	03404	03414	03501
		03525	03621					
004510	CYCLE	00236	04510					
004554	CYCTS	04516	04554					
000135	D3000	00135	04167					
000137	D500	00137	02441	02455	02477	02520	02534	02556
		02577	02613	02635	02656	02672	02714	
004710	DECCC	04710	04733	04734	04744			
004720	DECOD	04720	04725					
004726	DECP	04721	04726					
004744	DECTB	04703	04744					
003233	DEV1	03224	03233					
000221	DEVIC	00221	00467	01043	03215	03230		
000410	DIAG	00404	00410					
003305	DILLA	03243	03277	03305				
003306	DILLB	03245	03301	03306				
003262	DILLI	00237	03262					
000216	DILLY	00216	03276	03663				
004631	DIVID	00240	04536	04631				
000171	DL100	00171	03271					
004634	DLOOP	04634	04641					
003760	DLT	00407	03760	03770	03771			
003772	DLTA	00241	03761	03772				
004070	DLTB	03772	04056	04067	04070			
004071	DLTC	03774	04031	04036	04054	04071		
004072	DLTD	03775	04041	04051	04072			
004073	DLTE	03776	04021	04032	04033	04073		
004016	DLT.1	04016	04062					
000050	EGGS	00045	00050					
006244	EHALT	00302	00413	00417	00423	00427	00440	00447
		00460	00501	00515	00526	00537	00550	00561
		00576	00604	00612	00615	00623	00626	00635
		00647	00657	00674	00700	00714	00725	00740
		00750	00761	01001	01005	01017	01025	01030
		01036	01041	01045	01053	01070	01076	01101
		01124	01132	01135	01164	01173	01200	01204
		01224	01227	01241	01244	01252	01271	01275
		01320	01324	01335	01346	01353	01372	01376
		01403	01407	01415	01420	01446	01452	01457
		01463	01471	01474	01516	01523	01527	01535
		01540	01553	01560	01564	01572	01575	01611
		01621	01633	01643	01654	01671	01702	01713

		01727	01733	01744	01762	01766	02006	02024
		02032	02040	02052	02070	02076	02104	02116
		02134	02142	02150	02162	02200	02206	02214
		02227	02247	02257	02270	02310	02320	02331
		02351	02361	02401	02421	02431	02440	02450
		02501	02510	02517	02527	02560	02567	02576
		02606	02637	02646	02655	02665	02716	02725
		02773						
004333	ENDPR	00056	04333					
004132	ENDT	04120	04132					
004466	ENTER	00242	00243	04466				
004613	EPRIN	04611	04613					
000244	ER	00244	00302					
006260	ERASE	00302	00670	00703	00733	02735		
004575	ERET	04575	04612					
004246	ERHLT	00302	04102	04224				
004566	ERR	00244	04566					
004603	ERR1	04574	04603					
004502	ERRCT	04474	04502	04532	04553	04600		
004501	ESWIT	04473	04501	04522	04557	04572	04605	04613
		04623						
003022	ETST	03022						
004074	ETT	00406	04074	04105	04106			
004107	ETTA	00245	04075	04107	04124	04131		
004135	ETTB	04107	04134	04135				
000055	FIRST	00055	03216					
004121	FTT.1	04121						
006261	GEN	00302	00766	01012	01063	01117	01363	01437
		01507	01544	01602	01624	02736		
003217	GMOR	03217	03236					
004627	HEADE	04616	04627					
003174	HENA	03171	03173	03174				
000224	IA28	00224	01207					
000225	IA39	00225	01600					
000226	IA46	00226	01773					
000227	IA491	00227	02367					
000230	IA492	00230	02372					
000231	IA50	00231	01775					
000232	IA55	00232	02730					
000254	IBPPE	00254	00302					
000255	IBSP	00255	00302					
000061	IBUFF	00061	01410	01464	01530	01565	01672	02013
		02026	02057	02072	02123	02136	02167	02202
		02236	02251	02277	02312	02340	02353	02410
		02423	03313	03340	03342	03535		
004624	ICHAR	04541	04624					
000256	ICLK	00256	00302					
000235	ICHST	00235	00656					
000257	ICLR	00257	00302					
000234	ICRLF	00234	04531	04614				
000404	IDIAG	00400	00404					
000407	IDLT	00403	00407					
000020	IDX0	00020	02747	02746	02751	03465	03466	03471
000021	IDX1	00021	01061	01103	01107	01115	01137	01143
		01360	01422	01426	01434	01476	01502	

000022	IDX2	00022	03311	03312	03324	03345	03351	
000023	IDX3	00023	03314	03315	03327	03334	03432	03444
000024	IDX4	00024	03434	03435	03446			
000260	IERAS	00260	00302					
000406	IETT	00402	00406					
000261	IGEN	00261	00302					
000262	ILD	00262	00302					
000251	IMESS	00251	03027	03040	03762	04044	04047	04063
		04076	04125	04132	04140	04165	04200	04615
003163	INITI	00247	03163					
000220	INST	00220	03527	03537	03575	03605		
000247	INTIL	00247	00410	03760	04074	04136		
004203	INWLA	04164	04173	04203				
004334	INWLB	04166	04334					
004224	INWLC	04220	04224					
004227	INWLD	04170	04176	04204	04221	04227		
004205	INWLF	04205	04222					
004365	INWLH	04201	04365					
004150	INWLT	00250	04137	04150				
004626	IPDEC	04537	04626					
000253	IPOCT	00253	04130					
000263	IRC	00263	00302					
000264	IRDNS	00264	00302					
003304	IRET	03163	03303	03304				
000265	IRWD	00265	00302					
000266	IRWS	00266	00302					
000267	ISEL	00267	00302					
000270	ISPC	00270	00302					
000271	ISTAT	00271	00302					
000272	ITIM	00272	00302					
000223	ITLP	00223	03374	04307	04472	04514		
000273	ITMD	00273	00302					
000274	ITPN	00274	00302					
004477	ITR	04467	04477	04517	04535			
004500	ITRCT	04470	04500	04515	04520			
004625	ITYPE	04625						
000275	IWEOF	00275	00302					
000405	IWLT	00401	00405					
000276	IWNS	00276	00302					
000277	IWRT	00277	00302					
000300	IWT	00300	00302					
005041	LAST	00060	00061	00062	00063	05041		
003337	LEAV	03331	03337					
006262	LOAD	00302	02000	02044	02110	02154	02221	02262
		02323	02370	02373				
000236	LOOP	00236	00414	00420	00424	00430	00441	00450
		00461	00502	00516	00527	00540	00551	00562
		00577	00605	00616	00627	00636	00650	00660
		00701	00716	00730	00764	01010	01031	01055
		01102	01136	01165	01174	01205	01230	01245
		01276	01325	01354	01421	01475	01541	01576
		01622	01644	01703	01734	01767	02041	02105
		02151	02215	02260	02321	02362	02432	02511
		02570	02647	02726	02774			
004507	LOOPR	04463	04466	04476	04507	04564		

003001	LPOIT	02732	02743	02775	03001	03004	03007	
003052	LPRT	03052						
000201	LST	00201	01104	01423				
000207	LST1	00207	01140	01477				
003054	LST79	03053	03054					
003071	LST9	03071						
003053	LSTRT	02731	03053					
000165	M100	00165	03467					
000167	M1000	00167						
000157	M2	00157	02011	02055	02121	02165	02234	02275
		02336	02406					
004656	M20	04632	04645	04656				
000136	M2000	00136	04001					
000163	M20.	00163						
000160	M3	00160	03443					
000161	M4	00161	01306	04203				
000166	M400	00166	02750	02757				
000162	M5	00162	01751					
000164	M8	00164	00705	01213	01717			
003143	MCYC	03041	03143	03763	04077	04141		
004455	MENDT	04133	04455					
004657	MESS	00251	04657	04672				
005025	MESSR	04657	04660	04675	05025			
004646	MLOOP	04646	04652					
004655	MSAV	04631	04642	04644	04654	04655		
000170	MSKTO	00170	03240					
004434	MSTAR	04126	04434					
004643	MULT	04534	04643					
004032	NODL	04032						
004052	NODLA	04035	04040	04046	04052			
004404	NODLB	04045	04064	04404				
004416	NOCLC	04050	04416					
004545	NOEX	04526	04545					
000060	OBUFF	00060	01020	01071	01125	02740	02755	03310
		03433	03464	03603				
004676	OCT	04676						
004734	OCTAB	04701	04734					
003651	OKRET	03651	03656	03657				
003654	ON	00252	03645	03654				
000252	ONLOC	00252	03716					
003110	PAT1	02001	03110					
003113	PAT2	02045	03113					
003116	PAT3	02111	03116					
003121	PAT4	02155	03121					
003124	PAT5	02222	03124					
003127	PAT6	02263	03127					
003132	PAT7	02324	03132					
003135	PAT8	02374	03135					
003140	PAT9	02371	03140					
000147	PCDCM	00147	03222					
004552	PCENT	04530	04540	04544	04552			
000063	PCHIN	00063	03777					
004703	PDEC	04626	04703					
000237	PDILL	00237	03246					
000240	PDIVD	00240	03272					

000241	FDLTA	00241	03036						
000245	PETTA	00245	03037						
000250	PINWL	00250	03022						
004700	POCT	00253	04622	04700					
000173	POINT	00173	01060	01357					
000202	PONTP	00202	01114	01433					
000062	PPCHN	00062	04023						
003147	PRB	03030	03147						
004754	RADRE	04705	04712	04754					
003000	RBCT	02734	02767	02776	03000				
003356	RCHK	03307	03336	03337	03341	03356			
003700	RDLAY	03661	03675	03700					
003552	RDSW	03506	03511	03541	03552				
006263	READ	00302	01400	01454	01520	01555	01615	01637	
		01656							
000145	RECTR	00145	02442	02451	02521	02530	02600	02607	
		02657	02666						
006264	REDNS	00302							
004503	RETUR	04503	04510	04550	04565	04566	04602	04603	
		04617							
006265	REWIN	00302	00704	02434	02513	02572	02651	03012	
		03023	03032	03045	04011	04152	04206		
003475	RGEN	03453	03454	03474	03475				
004226	RINWL	04150	04202	04223	04225	04226			
003551	RRD	03512	03513	03543	03550	03551			
003403	RRWS	03371	03402	03403					
003504	RSF1	03476	03503	03504					
003040	RSTP	03021	03040						
004253	RTIM	04230	04231	04233	04234	04236	04250	04251	
		04252	04253						
004271	RTMD	04256	04261	04263	04270	04271			
006266	RWDST	00302	00644	00667	00675	00715	00720	00732	
		01167	01232	01247	01300	01327	01332	01362	
		01373	01436	01447	01506	01517	01543	01554	
		01604	01626	01646	01655	01705	01736	01747	
		01777	02007	02043	02053	02107	02117	02153	
		02163							
003430	RWEOF	03413	03427	03430					
003617	RWRT	03560	03561	03611	03616	03617			
004506	SAV0	04506	04513	04545	04554	04571	04577		
004505	SAV1	04505	04512	04546	04555	04570	04576		
004504	SAV2	04504	04511	04547	04556	04567	04575		
000064	SAVE	00064							
000065	SAVE1	00065							
006267	SELEC	00302	00601	00607	00620	00631	00643		
000243	SETP1	00243	00475	00642	00666	00702	00717	00731	
		01032	01062	01116	01166	01210	01231	01246	
		01277	01326	01361	01435	01505	01542	01601	
		01623	01645	01704	01735	01776	02042	02106	
		02152	02220	02261	02322	02363	02433	02512	
		02571	02650	02754					
000242	SETUP	00242	00411	00415	00421	00425	00431	00442	
		00453	00507	00517	00530	00541	00552	00563	
		00600	00606	00617	00630	00653	00765	01011	
		01155	01175						

006270	SPACE	00302	01340	01750				
003757	STATR	03701	03702	03717	03755	03756	03757	
006271	STATU	00302	00645	00672	00676	00723	00736	00743
		01026	01077	01133	01171	01202	01225	01242
		01250	01267	01316	01333	01344	01370	01374
		01405	01444	01450	01461	01514	01525	01551
		01562	01607	01617	01631	01641	01652	01667
		01711	01725	01742	01760	02004	02022	02050
		02066	02114	02132	02160	02176	02225	02245
		02266	02306	02327	02347	02377	02417	03013
		03024	03033	03046	03654	04153	04157	04162
		04171	04207	04216				
003620	SWTCH	03554	03557	03607	03620			
000222	SWTSV	00222	01146	01662	01673	01770	02364	03016
		03176	03321	03416	03437	03437	03633	03704
		03710	03723	03736	04275	04323		
006274	TESTP	00302	00671	01201	01343	01367	01404	01443
		01460	01513	01524	01550	01561	01757	
004255	TIMCT	04232	04246	04255				
006272	TIME	00302	01312	02435	02445	02514	02524	02573
		02603	02652	02662	03376			
004272	TMOCT	04260	04266	04272				
006273	TMDON	00302	00726	00762	01006	01221	01236	01265
		01310	01341	01613	01635	01723	01755	02020
		02064	02130	02174	02231	02243	02272	02304
		02333	02345	02403	02415	02453	02532	02611
		02670	02765	03425	03544	03612	03626	
004321	TPNR	04273	04306	04320	04321			
004254	TSTAT	04235	04237	04254				
005026	TYPE	04625	04773	05002	05026			
005040	TYPRE	05026	05033	05037	05040			
006300	WAIT	00302	00640	03477	04174	04244	04264	
000217	WCCNT	00217	03520	03533	03566	03601		
006275	WEOF	00302	01176	01651	01710	01741		
004136	WLT	00405	04136	04146	04147			
006277	WRITE	00302	01014	01033	01065	01121	01233	01253
		01301	01330	01365	01441	01511	01546	01605
		01627	01647	01706	01714	01737	01745	02002
		02046	02112	02156	02223	02264	02325	02375
		04212						
006276	WRTNS	00302	00721	00734	00770	02443	02522	02601
		02660						
000210	X70	00210	00652	00654	00661	00662	03652	03722
004322	XBPPE	00254	04322					
003404	XBSP	00255	03404					
003307	XCHK	00256	03307					
003324	XCHK.	03324	03333					
003342	XCLR	00257	03342					
003631	XERAR	03622	03630	03631				
003621	XERAS	00260	03621					
003453	XGEN	00261	03453					
003431	XLD	00262	03431					
003510	XRD	00263	03507	03510				
003505	XRDNS	00264	03505					
003533	XRD.1	03533	03547					

003366	XRWD	00265	03366					
003371	XRWS	00266	03371					
003476	XSEL	00267	03476					
003357	XSPC	00270	03357					
003734	XSTA1	03721	03727	03734				
003701	XSTAT	00271	03701					
003754	XSTT.	03740	03743	03754				
004230	XTIM	00272	04230					
004251	XTIM1	04243	04251					
004237	XTIM.	04237	04247					
004256	XTMD	00273	04256					
004262	XTMD.	04262	04267					
004273	XTPN	00274	04273					
004301	XTPN1	04301	04311					
004307	XTPN.	04300	04307					
003413	XWEOF	00275	03413					
003553	XWNS	00276	03553					
003556	XWRT	00277	03555	03556				
003601	XWRT.	03601	03615					
003661	XWT	00300	03661					
003677	XWTA1	03662	03674	03677				
004753	ZSUPP	04706	04716	04730	04753			
000066	.1SEC	00066						
000071	.200M	00071						
000072	.400M	00072						
000070	.50MS	00070	00641	03500				
000067	.5SEC	00067						
000022	.MTA	00302	00422	00426	00433	00434	00444	00445
		00455	00476	00477	00511	00513	00522	00523
		00524	00533	00534	00535	00544	00545	00546
		00555	00556	00557	00570	00571	00572	00573
		00574	00602	00603	00610	00611	00614	00621
		00622	00624	00625	00633	00634	00655	00712
		00754	00774	01002	01016	01023	01035	01067
		01074	01123	01130	01156	01157	01160	01177
		01212	01214	01220	01223	01240	01256	01263
		01264	01272	01304	01307	01322	01337	01351
		01402	01413	01456	01467	01522	01533	01557
		01570	01700	01720	01722	01731	01752	01754
		01764	02012	02015	02017	02025	02034	02056
		02061	02063	02071	02100	02122	02125	02127
		02135	02144	02166	02171	02173	02201	02210
		02235	02240	02242	02250	02276	02301	02303
		02311	02337	02342	02344	02352	02407	02412
		02414	02422	02457	02461	02465	02467	02470
		02473	02476	02503	02536	02540	02544	02546
		02547	02552	02555	02562	02615	02617	02623
		02625	02626	02631	02634	02641	02674	02676
		02702	02704	02705	02710	02713	02720	02756
		02760	02764	02770	03002	03363	03364	03367
		03375	03410	03411	03424	03502	03534	03536
		03540	03602	03604	03606	03625	03647	03734
		04013	04020	04026	04060	04112	04113	04115
		04121	04127	04214	04240	04262		

SECTION II

RELIABILITY PROGRAM

```

*****
; WESTERN PERIPHERALS *
;MAGNETIC TAPE RELIABILITY PROGRAM *
; LISTING NO. AL120010 *
; *
;*****
;1.ABSTRACT
; THE TAPE CONTROL RELIABILITY IS A MAINTENANCE
; PROGRAM INTENDED FOR RIGOROUS TESTING
; OF A SYSTEM THAT HAS SUCCESSFULLY RUN
; THE DIAGNOSTIC TEST AND THE TIMING
; TEST.
;2.MACHINE REQUIREMENTS
; NOVA OR SUPERNOVA PROCESSOR
; 4K READ/WRITE CORE MEMORY
; TELETYPE
;3.SWITCH SETTINGS
; 3.1 STARTING ADDRESSES
; NOTE
; SWITH 0 MUST BE SET BEFORE START
; OF PROGRAM IF SYSTEM CONFIGURED
; FOR 6,6,4 PACKING ON 7TRK
; 400 RELIABILITY TEST
; 401 INTERCHANGE TEST (RD&WR)
; 402 INTERCHANGE TEST (RD ONLY)
; 405 TEST LOOP BUILDER
; 3.2 CONTROL
; SW0=PHASE ENCODED
; SW1=INHIBIT PRINTOUTS
; SW3=DONT PRINT PARITY ERRORS
; SW4=DONT RETRY ON WRITE PARITY ERROR
; SW5=DONT RETRY ON READ PARITY ERROR
;
;4.OPERATING PROCEDURE
; LOAD PROGRAM USING THE BINRYR LOADER
; 4.1 RELIABILITY TEST
; SET WRITE ENABLE FOR ALL UNITS TO
; BE TESTED. START. ERRORS WILL BE
; REPORTED ON THE TELETYPE
; 4.2 INTERCHANGE TEST (RD&WT)
; MOUNT TAPES ON ALL UNITS TO BE TESTED
; AND SET THEM IN WRITE PERMIT. START.
; WHEN ASKED BY PROG. REMOVE TAPES AND
; MOUNT ON UNIT TO THE RIGHT. ERRORS
; ARE REPORTED ON THE TELETYPE.
; 4.3 INTERCHANGE TEST (READ ONLY)
; MOUNT TAPES WRITTEN USING THE MODE
; DESCRIBED ABOVE IN PAR. 4.2 ONTO ALL
; UNITS TO BE TESTED. START. WHEN ASKED
; BY PROG. REMOVE TAPES AND MOUNT ON THE
; UNIT TO THE RIGHT. ERRORS ARE REPORTED
; ON THE TELETYPE.
; 4.4 TEST LOOP BUILDER
; SET THE WRITE ENABLE FOR THE UNIT TO BE
; TESTED. START. PROGRAM WILL RESPOND

```

```

; WITH "UNIT". TYPE IN THE UNIT #
; PROGRAM WILL RESPOND WITH "WC", TYPE
; IN WORD COUNT. (IN OCTAL). PROGRAM
; WILL RESPOND WITH "DATA", TYPE IN THE
; DATA CODE -
; CODES:
; RAND = RANDOM DATA
; ALL1 = ALL ONES
; ALLO = ALL ZEROS
; ALTO = ALTERNATE ONES AND ZEROS.
; STARTING WITH ZERO
; ALT1 = ALTERNATE ONES AND ZEROS
; STARTING WITH ONES
; WDN-N= FILL WRITE BUFFER WITH DATA N-N
; PROGRAM WILL THEN ASK FOR PARITY TYPE
; (PAR). ENTER "ODD" OR "EVEN"
; PROGRAM WILL RESPOND WITH "COMMAND
; STRING", KEY IN THE COMMAND STRING.
; COMMANDS -
; RD = READ
; RW = REWIND
; RU = REWIND UNLOAD
; SF = SPACE FORWARD
; SB = SPACE BACKWARDS
; WT = WRITE
; WE = WRITE EOF
; ER = ERASE
; RE = READ EOF
; LOOP * = LOOP BACK TO *
; LOOP = GO BACK TO BEGINNING OF LINE
; SAMPLE COMMAND STRING
; RW/WT,10,/RW*/RD,10,/SB,1n,/LOOP *
; THIS ROUTINE WILL REWIND, WRITE 8 RECORDS
; AND THEN READ 8, AND BACKSPACE 8,
; CONTINUALLY.
; TYPE ANY CHAR ON THE TTY TO
; TERMINATE THE TEST.
; TYPING "CARRIAGE RETURN" IN RESPONSE TO "UNIT"
; WILL CAUSE THE PREVIOUS COMMAND STRING TO
; BE EXECUTED AGAIN.
; 5. DESCRIPTION
; 5.1 RELIABILITY TEST
; THE RELIABILITY TEST ASSUMES THAT ALL
; UNIT BEING TESTED ARE FUNCTIONAL TO
; A CERTAIN EXTENT. IT SEARCHES FOR ALL
; WRITE ENABLED UNITS, REWINDS THEM AND
; WRITES A FILE MARK AT THE BOT. OF THESE
; IT PICKS ONE AT RANDOM AND PERFORMS THE
; FOLLOWING.
; A.WRITES A FILE MARK
; B.PICK RANDOM NUMBER OF RECORDS (1-7)
; C.PICK RANDOM STARTING ADDRESS
; D.GENERATE RANDOM WORD LENGTH BUFFER
; MODULO 512
; E.WRITE THE FILE FOLLOWED BY FILE MARK.

```

```

; F.BACKSPACE OVER THE FILE
; G.READ AND CHECK THE FILE.
; H.PICK ANOTHER UNIT AND START AGAIN.
; 5.2 INTERCHANGE TEST (RD&WT)
; THIS IS A WORST CASE SKEW TEST THAT
; IS PERFORMED TO TEST THE INTERCHANG-
; ABILITY OF TAPES WRITTEN ON ONE UNIT
; TO READING ON ANOTHER. IT PROCEEDS
; AS FOLLOWS.
; A.FIND AVAILABLE WRITE ENA2LED UNITS
; B.WRITE THE SKEW PATTERN 100 TIMES
; ON EACH.
; C.WRITE 100 RANDOM DATA REZORDS FOLLOWED
; BY 2 EOF, ON EACH UNIT.
; D.REWIND ALL UNTS.
; E.READ AND CHECK THE SKEW PATTERN
; ALL UNITS.
; F.READ AND CHECK THE RANDOM!! DATA AND EOF'S
; ON ALL UNITS.
; G.TYPE MESSAGE ON TTY TO HAVE OPERATOR
; MOVE TAPES TO ANOTHER UNIT FOR THE
; READ TEST AGAIN. GO BACZ TO STEP D.
; 5.3 INTERCHANGE TEST (READ ONLY)
; THIS IS THE SAME AS THE PREVIOUSLY
; DISCUSSED TEST IN PAR. 5.2. EXCEPT
; THAT STEPS A THRU C ARE BYPASSED
; 5.4 TEST LOOP BUILDER
; THIS PROGRAM IS ENTENDED FOR USE BY
; THE TECHNICIAN AS A DEBUGGING AID.
; IF A CERTAIN SEQUENCE OF EVENTS IS
; NEEDED TO SHOW UP A PARTICULAR PROBLEM
; THAT SEQUENCE CAN BY TYPED IN AS A COMMAND
; STRING WHICH WILL THEN BE PERFORMED.
; THE SAME BASIC TAPE SUBROUTINES USED
; IN THE RELIABILITY TEST ARE USED HERE,
; LINKED TOGETHER BY THE COMMAND STRING
; INTERPRETER.
; 5.5 GENERAL
; THE HEART OF THE PROGRAM IS COMPOSED
; OF SEVERAL TAPE HANDLING SUBROUTINES.
; ERROR CHECKING IN THESE SU2R. IS DESCRIBED
; BELOW.
; READ SUBROUTINE
; 1. GIVE C PULSE
; 2. IF DATA LATE,REWIND,ILLEGAL OR PARITY
; STATUS, PRINT MESSAGE AND EXIT
; 3. READ AND WAIT IN SKPBZ LOOP
; 4. IF DATA LATE. REWIND OR ILLEGAL, PRINT
; MESSAGE AND EXIT
; 5. IF PARITY PRINT MESSAGE FIRST TIME ONLY
; AND RETRY UP TO 8 TIMES.
; CORR PARITY ERR ARE IGNORED (PE)
; 6. IF ECT SET SOFTWARE FLAG.
; 7. EXIT
; CHECK SUBROUTINE

```

```

; 1. CHECK DATA IN INPUT BUFFER AGAINST DATA
; THAT WAS WRITTEN. PRINT MAXIMUM OF 3
; DIFFERENCES IN EACH RECORD.
; 2. CHECK CRC IF 9TRK, ODD PAR. CHECK LPC
; UNLESS 9 TRK EVEN PAR. PRINT MESSAGE.
; IF ERROR.
; WRITE SUBROUTINE
; 1. GIVE CLEAR PULSE
; 2. IF DATA LATE, WRITE LOCK, ILLEGAL, OR
; PARITY, PRINT MESSAGE AND EXIT
; 3. WRITE AND WAIT FOR INTERRUPT (RUN TIMER)
; 4. IF NO INTERRUPT PRINT MESSAGE AND EXIT
; IF INTERRUPT, CHECK DISABLE FLAG AND PRINT
; MESSAGE IF IT FAILS.
; 6. IF PARITY ERR PRINT MESSAGE ONCE AND RETRY
; UP TO 8 TIMES. IF NO-GO ERASE TAPE
; AND TRY AGAIN.
; 7. IF EOT SET SOFTWARE FLAG
; 8. EXIT
; SPACE SUBROUTINE
; 1. LOAD WC AND SPACE WAITING IN SKPBZ LOOP
; 2. IF ADDRESS REG. COUNTS DO NOT AGREE
; WITH #REC SPECIFIED, PRINT MESSAGE.
; 3. IF DATA LATE, REWIND, ILLEGAL, OR
; PARITY PRINT MESSAGE.
; 4. EXIT.

```

```

000000 .LOC 0
00000 000000 0 ;STORE PC IN INT.
00001 000407 SERINT ;ADDR. OF SER RUT.
00002 000000 IS0: 0 ;ON INT, SAVE ACO
00003 000000 IS1: 0 ;AC1
00004 000000 IS2: 0 ;AC2
00005 000000 IS3: 0 ;AC3
00006 000000 ISC: 0 ;CARRY
000020 .LOC 20
00020 000000 IDX0: 0
00021 000000 IDX1: 0
00022 000000 IDX2: 0
00040 000040 .LOC 40
00040 005042 IGET: GETCH
00041 004021 TYPE
00042 000605 ITTI: TTINT
00043 000622 IERR: INTER
00044 005061 TIBUF: LAST+1
00045 005062 LAST+2
00046 000001 AVAIL: 1
00047 000002 2
00050 000004 4
00051 000010 10
00052 000020 20
00053 000040 40
00054 000100 100
00055 000200 200
000010 WPAR: .BLK 10 ;WRITE PARITY
000010 PERMW: .BLK 10 ;PERM WRITE ERROR

```

000010	REC:	.BLK 10	;RECORD # COUNTERS
000010	RPAR:	.BLK 10	;READ PARITY
000010	PERMR:	.BLK 10	;PERM READ ERRORS
000010	DATER:	.BLK 10	;DATA ERRORS.
000020	WDSW:	.BLK 20	;WORDS WRITTEN
000020	WDSR:	.BLK 20	;WORDS READ
00176	000000	KEYRET:	0
00177	000177	C177:	177
00200	000015	C15:	15
00201	000022	C22:	22
00202	000215	C215:	215
00203	000212	C212:	212
00204	000040	C40:	40
00205	000074	C74:	74
00206	000703	C703:	703
00207	000727	C727:	727
00210	000000	PARITY:	0
00211	001226	IWE1.1:	WE1.1
00212	001271	IWE6:	WE6
00213	000000	CLRET:	0
00214	000056	CWPAR:	WPAR
00215	001000	PLACE:	1000
00216	004714	IINP:	INP
00217	005007	I.OBIN:	.CBIN
00220	000000	HISRET:	0
00221	004001	ICRLF:	CRLF
00222	000000	DMPTM:	0
00223	003673	IZOCT:	ZOCT
00224	003542	I.DBD:	.DBD
00225	000000	DMPRET:	0
00226	003656	IMESS:	MESS
00227	004232	IMORE:	MORE
00230	000010	C10:	10
00231	000000	TAPRET:	0
00232	000000	ROTRY:	0
00233	172000	C172K:	172000
00234	005061	BUFF:	LAST+1
00235	177400	WC:	-400
00236	070000	C70K:	70000
00237	002000	C2K:	2000
00240	001000	C1000:	1000
00241	100000	PKFLG:	100000
00242	000000	EOTFLG:	0
00243	000000	UNIT:	0
00244	010000	INHFLG:	10000
00245	003677	IPDEC:	PDEC
00246	003674	IPOCT:	POCT
00247	002437	IRECN:	RELREC
00250	000000	WORD:	0
00251	000010	RRTRY:	10
00252	000000	WRTRY:	0
00253	000004	C4:	4
00254	000050	C50:	50
00255	000010	RETRY:	10
00256	000000	BAD:	0

;READ RETRY CONSTANT.

;WRITE RETRY CONSTANT

00257	000000	SACRET:	0
00260	000030	C30:	30
00261	000000	ERARET:	0
00262	000070	C70:	70
00263	000100	C100:	100
00264	000200	C200:	200
00265	000060	C60:	60
00266	004000	C4K:	4000
00267	177400	C1774:	177400
00270	176000	CB1:	176000
00271	001760	CB2:	1760
00272	000017	CB3:	17
00273	000020	C20:	20
00274	000000	CTR:	0
00275	000000	HOLD:	0
00276	000000	LPC:	0
00277	000000	CRC:	0
00300	000000	SAV0:	0
00301	000000	SAV1:	0
00302	000000	SAV3:	0
00303	000000	FSTGC:	0
00304	000000	GRET:	0
00305	000000	PATT:	0
00306	037477	C37477:	37477
00307	037400	C37400:	37400
00310	000000	DATMSK:	0
00311	000000	GWC:	0
00312	000000	CKRET:	0
00313	000000	GOOD:	0
00314	000017	C17:	17
00315	000007	C7:	7
00316	000000	UCTR:	0
00317	000000	RANDOM:	0
00320	000000	RELTRAN:	0
00321	003400	C3400:	3400
00322	000000	RELTEM:	0
00323	005060	CEND:	LAST
00324	000777	C777:	777
00325	000000	INTEM:	0
00326	000000	INTRET:	0
00327	177634	M144:	-144
00330	012345	C12345:	12345
00331	000377	C377:	377
00332	000000	INTRAN:	0
00333	000000	ALLRET:	0
00334	000000	RETS6:	0
00335	002471	ITEST:	INTW
00336	002311	RTEST:	REL
00337	002470	ITST:	INT
00340	000464	HIST:	HISTORY
00341	001530	ITOK:	TOK
00342	002151	.RAND:	RAND
00343	072000	C72K:	72000
00344	002643	IALL:	ALL
00345	177400	M400:	-400

```

00346 177000 M1000: -1000
00347 177600 M2000: -200
00350 000433 IKEY: KEY
00351 007600 CMEND: 7600 ;TOP OF MEMORY
00352 001777 C1777: 1777
00353 000532 ISM: SMEND
00354 000452 ICLBIN: CLBIN
00355 000631 XREAD: RD
00356 001030 XWRIT: WR
00357 001301 XSPAC: BAC
00360 001451 XFILE: WEOF
00361 001422 XERAS: ERAS
00362 001540 XREW: REW
00363 001551 XUNLD: UNLD
00364 001556 XGEN: G
00365 001555 XCHECK: C
00366 001276 XSPC: SPACHK
00367 000000 DIGIT: 0
00370 004014 ICHORZ: CHORZ
00371 000000 RR: 0
00372 000400 C400: 400
00373 070400 C704: 70400
00374 002040 C2040: 2040
00375 010000 C10K: 10000
00376 004034 IBLD: BUILD
      000022 .MTA=22
      006357 SPACE=JSR @XSPAC
      006355 READ=JSR @XREAD
      006356 WRITE=JSR @XWRIT
      006360 WFILE=JSR @XFILE
      006361 ERASE=JSR @XERAS
      006362 REWIND=JSR @XREW
      006363 UNLOAD=JSR @XUNLD
      006365 CHECK=JSR @XCHECK
      006364 GEN=JSR @XGEN
      006040 GET=JSR @IGET
      006366 SPCHK=JSR @XSPC
00377 000000 DMPT: 0
      000400 .LOC 400
00400 002336 JMP @RTEST ;START RELIABILITY TEST
00401 006335 JSR @ITEST ;START INTERCHANGE TEST
00402 006337 JSR @ITST ;START INTERCHANGE TEST(NO WR.)
00403 000776 JMP .-2
00404 000776 JMP .-2
00405 002401 JMP @.+1 ;START TEST LOOP BUILDER
00406 004034 BUILD
      ;INTERRUPT SERVICE ROUTINE
00407 040002 SERINT: STA 0,IS0
00410 044003 STA 1,IS1
00411 050004 STA 2,IS2
00412 054005 STA 3,IS3
00413 101200 MOVR 0,0
00414 040006 STA 0,ISC
00415 102400 SUB 0,0
00416 061477 INTA 0 ;DEVICE CODE TO ACO

```

```

00417 024201      LDA 1,C22      ;IS IT TAPE
00420 122415      SUB# 1,0,SNR
00421 000410      JMP TAPINT     ;YES
00422 024230      LDA 1,C10     ;IS IT TTI
00423 122415      SUB# 1,0,SNR
00424 002042      JMP @ITTI     ;YES
00425 125400      INC 1,1      ;IS TI TTO
00426 122415      SUB# 1,0,SNR
00427 002042      JMP @ITTI     ;YES
00430 002043      JMP @IERR     ;ERRONEOUS INTERRUPT
00431 010000      TAPINT: ISZ 0 ;TAPE INTERRUPT
00432 002000      JMP @0
00433 054176      KEY:  STA 3,KEYRET ;CARRIAGE FLAERS ERRORS.
00434 060610      DIAC 0,TTI   ;SPACE PRINTS ERRORS
00435 024177      LDA 1,C177   ;ANY OTHER KEY PRINTS AND
00436 123400      AND 1,0     ;THEN CLEARS.
00437 024200      LDA 1,C15
00440 106415      SUB# 0,1,SNR
00441 000407      JMP KEY1+1   ;CARRIAGE
00442 024204      LDA 1,C40
00443 106414      SUB# 0,1,SZR
00444 000403      JMP KEY1     ;OTHER
00445 006340      JSR @HIST   ;SPACE
00446 002176      JMP @KEYRET
00447 006340      KEY1: JSR @HIST
00450 004402      JSR CLBIN
00451 002176      JMP @KEYRET
00452 054213      CLBIN: STA 3,CLRET ;CLEAR ERROR BINS
00453 102401      SUB 0,0,SKP
00454 177660      -120
00455 024777      LDA 1,..-1
00456 030214      LDA 2,CWPAR
00457 041000      STA 0,0,2
00460 151400      INC 2,2
00461 125404      INC 1,1,SZR
00462 000775      JMP .-3
00463 002213      JMP @CLRET
00464 054220      HISTRY: STA 3,HISRET ;PRINT THE HISTORY
00465 006221      JSR @ICRLF
00466 006221      JSR @ICRLF
00467 004460      JSR DMPDAT-2
00470 000420      JMP H0      ;UNIT
00471 004460      JSR DMPDAT
00472 000420      JMP H1      ;WRITE PARITY
00473 004456      JSR DMPDAT
00474 000420      JMP H2      ;READ PARIY
00475 004454      JSR DMPDAT
00476 000420      JMP H3      ;PERM WRITE ERR
00477 004452      JSR DMPDAT
00500 000420      JMP H4      ;PERM READ ERR
00501 004450      JSR DMPDAT
00502 000420      JMP H5      ;DATA ERROR
00503 004446      JSR DMPDAT
00504 000420      JMP H6      ;WORDS READ
00505 004444      JSR DMPDAT

```

```

00506 000421      JMP H7           ;WORDS WRITTEN
00507 002220      JMP @HISRET
00510 024222 H0:   LDA 1,DMPTEM      ;PRINT UNIT #
00511 002223      JMP @IZOCT
00512 025056 H1:   LDA 1,WPAR,2      ;WRITE PARITY
00513 002245      JMP @IPDEC
00514 025106 H2:   LDA 1,RPAR,2      ;READ PARITY
00515 002245      JMP @IPDEC
00516 025066 H3:   LDA 1,PERMW,2     ;PERM WRITE ERR
00517 002245      JMP @IPDEC
00520 025116 H4:   LDA 1,PERMR,2     ;PERM READ ERR
00521 002245      JMP @IPDEC
00522 025126 H5:   LDA 1,DATER,2     ;DATA ERR
00523 002245      JMP @IPDEC
00524 025156 H6:   LDA 1,WDSR,2      ;WORDS READ
00525 031166      LDA 2,WDSR+10,2
00526 002224      JMP @I.DBD
00527 025136 H7:   LDA 1,WOSW,2
00530 031146      LDA 2,WOSW+10,2
00531 002224      JMP @I.DBD
                   ;FIND THE TOP OF MEMORY (4K SEGMENTS)
00532 152400 SMEND: SUB 2,2
00533 020375      LDA 0,C10K
00534 113000 S.1:  ADD 0,2
00535 051000      STA 2,0,2      ;STORE INT' UPPER
00536 025000      LDA 1,0,2      ;GET IT BACK
00537 146404      SUB 2,1,SZR
00540 000403      JMP S.2
00541 151113      MOVL# 2,2,SNC   ;CHECK IF 32K
00542 000772      JMP S.1
00543 024177 S.2:  LDA 1,C177      ;WENT TOO FAR
00544 132400      SUB 1,2
00545 050351      STA 2,CMEND     ;SET END MEM
00546 001400      JMP 0,3
00547 020425      LDA 0,CMH0      ;PRINT OUT A LINE
00550 040377      STA 0,DMP      ;OF DATA FOR 8 UNITS.
00551 054225 DMPDAT: STA 3,DMPRET
00552 022377      LDA 0,@DMP
00553 040403      STA 0,.,+3
00554 006221      JSR @ICRLF      ;CARRIAGE RETURN
00555 006226      JSR @IMESS      ;PRINT MESSAGE TO
00556 000000      0           ;START EACH LINE
00557 132400      SUB 2,2
00560 050222      STA 2,DMPTEM
00561 010377      ISZ DMP      ;ADVANCE TO NEXT MESSAGE
00562 021076 DMPD1: LDA 0,REC,2
00563 101112      MOVL# 0,0,SZC   ;IF THE UNITS ARE ACTIVE
00564 006225      JSR @DMPRET   ;EXIT TO DECIMAL PRINTERS.
00565 010222      ISZ DMPTEM
00566 030222      LDA 2,DMPTEM   ;TEST FOR LAST UNIT.
00567 024230      LDA 1,C10
00570 132414      SUB# 1,2,SZR
00571 000771      JMP DMPD1      ;ANOTHER UNIT
00572 010225      ISZ DMPRET   ;EXIT TO NEXT DATA.
00573 002225      JMP @DMPRET

```

```

00574 000575 CMHO:  ,+1
00575 003321      MH0
00576 003324      MH1
00577 003330      MH2
00600 003334      MH3
00601 003341      MH4
00602 003346      MH5
00603 003353      MH6
00604 003357      MH7

;TELETYPE INTERRUPT
00605 024414 TTINT: LDA 1,CNIOC
00606 123000      ADD 1,0
00607 040401      STA 0,.,+1
00610 000000      0
00611 020006      LDA 0,ISC
00612 101100      MOVL 0,0          ;RESET CARRY
00613 034005      LDA 3,IS3          ;RESTORE AC3
00614 030004      LDA 2,IS2          ; AC2
00615 024003      LDA 1,IS1          ; AC1
00616 020002      LDA 0,ISO          ; AC0
00617 060177      INTEN
00620 002000      JMP @0
00621 060200 CNIOC: 060200          ;NIOC 0,0

;INTERUPT ERROR
00622 006226 INTER: JSR @IMESS          ;PRINT ERR MSG
00623 004625      MIER
00624 105000      MOV 0,1
00625 040763      STA 0,TTINT+3      ;SAVE AC0
00626 006223      JSR @IZOCT          ;PRINT INT DEV CODE
00627 020761      LDA 0,TTINT+3
00630 000755      JMP TTINT
00631 054231 RD:   STA 3,TAPRET          ;READ FROM TAPE
00632 102400      SUB 0,0
00633 040232      STA 0,RDTRY          ;CLEAR RETRY COUNTER
00634 006341 RD.1: JSR @ITOK          ;WAIT FOR INIT.
00635 060222      NIOC ,MTA
00636 064422      DIA 1,.,MTA
00637 030343      LDA 2,C72K          ;AFTER A "C" PULSE
00640 133414      AND# 1,2,SZR          ;LATE,REWIND,ILLEGAL
00641 002464      JMP @IWE2          ;PARITY IS FATAL ERR.
00642 020243 RD1:  LDA 0,UNIT
00643 024210      LDA 1,PARITY
00644 123000      ADD 1,0
00645 061022      DOA 0,.,MTA          ;SELECT UNIT & PARITY
00646 024234      LDA 1,BUFF          ;SEND ADDRESS+MA INT BIT
00647 131100      MOVL 1,2
00650 151240      MOVOR 2,2          ;CHECK READ BACK.
00651 060422      DIA 0,.,MTA
00652 034263      LDA 3,C100
00653 117405      AND 0,3,SNR          ;EXIT IF 7 TRK
00654 000406      JMP RD1.
00655 034266      LDA 3,C4K
00656 117404      AND 0,3,SZR          ;EXIT IF NRZ
00657 000403      JMP RD1.
00660 151100      MOVL 2,2

```

```

00661 151220          MOVZR 2,2
00662 072022 RD1.:   DCB 2,.MTA
00663 061422          DIB 0,.MTA
00664 106414          SUB# 0,1,SZR
00665 002441          JMP @IWE3
00666 024235          LDA 1,WC
00667 067122          DOCS 1,.MTA          ;START UNIT
00670 063522 RD2.:   SKPBZ .MTA
00671 000777          JMP RD2
00672 030243          LDA 2,UNIT
00673 021076          LDA 0,REC,2
00674 101500          INCL 0,0
00675 101240          MOVOR 0,0
00676 041076          STA 0,REC,2          ;UPDATE RECORD COUNT
00677 064422          DIA 1,.MTA
00700 020373          LDA 0,C704          ;IF ITS LATE,REWIND
00701 107414          AND# 0,1,SZR          ;OR ILLEGAL ITS A
00702 000521          JMP RE6          ;FATAL ERROR.
00703 060422 RD3.:   DIA 0,.MTA          ;CHECK FOR PARITY
00704 024374          LDA 1,C2040          ;ERROR.
00705 107414          AND# 0,1,SZR
00706 000421          JMP RD5          ;READ PARITY ERROR.
00707 060422 RD4.:   DIA 0,.MTA
00710 024240          LDA 1,C1000
00711 107620          ANDZR 0,1
00712 125300          MOVS 1,1          ;C(EOTFLG).0 IF NOT EOT.
00713 044242          STA 1,EOTFLG          ;C(EOTFLG)=1 IF EOT.
00714 030243          LDA 2,UNIT
00715 021166          LDA 0,WDSR+10,2
00716 024235          LDA 1,WC          ;SUM THE WORDS
00717 122422          SUBZ 1,0,SZC          ;READ ON EACH UNIT.
00720 011156          ISZ WDSR,2
00721 041166          STA 0,WDSR+10,2
00722 063710          SKPDZ TTI
00723 006350          JSR @IKEY          ;INPUT KEY STRUCK,
00724 002231          JMP @TAPRET
00725 001254 IWE2.:  WE2
00726 001244 IWE3.:  WE3
00727 060422 RD5.:   DIA 0,.MTA          ;IGNORE IF CRE
00730 024230          LDA 1,C10
00731 107404          AND 0,1,SZR
00732 000755          JMP RD4
00733 024232          LDA 1,RDTRY
00734 125004          MOV 1,1,SZR
00735 000445          JMP RD6          ;NOT FIRST ERROR.
00736 030243          LDA 2,UNIT
00737 011106          ISZ RPAR,2          ;INC READ PARITY.
00740 020244          LDA 0,INHFLG
00741 064477          READS 1
00742 107414          AND# 0,1,SZR          ;PRINT PARITY MESSAGE
00743 000437          JMP RD6          ;SWITCH,
00744 006221          JSR @ICRLF
00745 020204          LDA 0,C40
00746 064422          DIA 1,.MTA
00747 107415          AND# 0,1,SNR

```

```

00750 000403      JMP .+3
00751 006226      JSR @IMESS
00752 002751      BADTAP          ;"BAD TAPE"
00753 020237      LDA 0,C2K
00754 064422      DIA 1,.MTA
00755 107415      AND# 0,1,SNR
00756 000403      JMP .+3
00757 006226      JSR @IMESS
00760 002677      MRPE
00761 006226      JSR @IMESS
00762 003321      MH0
00763 024243      LDA 1,UNIT
00764 006223      JSR @IZOCT
00765 006226      JSR @IMESS          ;"RECORD#"
00766 002707      MRECN
00767 030243      LDA 2,UNIT
00770 025076      LDA 1,REC,2
00771 125100      MOVL 1,1
00772 125220      MOVZR 1,1
00773 006245      JSR @IPDEC          ;THE RECORD NUMBER.
00774 066422      DIC 1,.MTA
00775 020267      LDA 0,C1774
00776 107400      AND 0,1
00777 006246      JSR @IPOCT
01000 006226      JSR @IMESS
01001 002714      MRCRC          ;"=CHECK WORD"
01002 020237      RD6: LDA 0,C2K
01003 064477      READS 1
01004 107404      AND 0,1,SZR          ;BYPASS RETRY IF SW5 SET
01005 000702      JMP RD4
01006 010232      ISZ RDTRY          ;SHALL A ATTEMPT
01007 020232      LDA 0,RDTRY          ;TO RE-READ BE MADE.
01010 024251      LDA 1,RRETR
01011 106415      SUB# 0,1,SNR
01012 000404      JMP RD7          ;NO GIVE UP
01013 006357      SPACE          ;YES BACK SPACE.
01014 177777      -1
01015 000617      JMP RD.1
01016 030243      RD7: LDA 2,UNIT          ;PERM READ ERROR.
01017 011116      ISZ PERMR,2          ;INCREMENT PERM ERROR
01020 015106      DSZ RPAR,2          ;DECREMENT READ PARITY
01021 101000      MOV# 0,0          ;GO TO EXIT ROUTINE
01022 000665      JMP RD4
01023 006221      RE6: JSR @ICRLF
01024 006246      JSR @IPOCT
01025 006226      JSR @IMESS
01026 002722      MRE6          ;"=STATUS AFTER A READ
01027 002211      JMP @IWE1.1          ;COMMAND ON UNIT"
01030 054231      WR: STA 3,TAPRET          ;WRITE ON TAPE
01031 102400      SUB 0,0
01032 040252      STA 0,WRTRY          ;CLEAR RETRY COUNTER.
01033 006341      WR.1: JSR @ITOK          ;WAIT FOR UNIT.
01034 064622      DIAC 1,.MTA
01035 030253      LDA 2,C4
01036 133414      AND# 1,2,SZR          ;IF UNIT WRITE LOCKED

```

```

01037 000564      JMP WE1           ;ITS A FATAL ERROR
01040 064422      DIA 1,.MTA
01041 030343      LDA 2,C72K       ;AFTER A "C" PULSE
01042 133414      AND# 1,2,SZR    ;LATE,REWIND,ILLEGAL,PARITY
01043 002662      JMP @IWE2       ;ITS A FATAL ERROR
01044 024254      WR1: LDA 1,C50
01045 107000      ADD 0,1
01046 020210      LDA 0,PARITY
01047 107000      ADD 0,1
01050 065022      DOA 1,.MTA     ;SEND WRITE COMMAND
01051 024234      LDA 1,BUFF    ;AND MEMORY ADDRESS.
01052 066022      DOB 1,.MTA
01053 061422      DIB 0,.MTA    ;IS ADDRESS SENT
01054 106414      SUB# 0,1,SZR  ;DIDNT GET THERE ITS
01055 002651      JMP @IWE3     ;A FATAL ERROR,
01056 024235      LDA 1,WC
01057 020320      LDA 0,RELRA  ;RANDOM PRELOAD TO TRY
01060 063022      DOC 0,.MTA   ;TO CAUSE LOAD FAILURE
01061 067122      DOCS 1,.MTA  ;SEND WC AND START
;INTERRUPT CHECKING
01062 060177      INTEN
01063 000400      JMP .
01064 020204      LDA 0,C40
01065 062077      MSKO 0
01066 060177      INTEN ;ALLOW INT.
01067 102400      SUB 0,0
01070 000402      JMP .+2
01071 000540      JMP ILLI      ;ERROR RETURN BAD INT
01072 060277      INTDS
01073 062077      MSKO 0
01074 060177      INTEN
01075 000401      JMP .+1
01076 000540      JMP NOIT     ;ERROR, NO INT
01077 064422      WR2.1: DIA 1,.MTA
01100 020240      LDA 0,C1000
01101 123620      ANDZR 1,0
01102 101300      MOVS 0,0    ;C(EOTFLG)=0 IF NOT EOT
01103 040242      STA 0,EOTFLG ;C(EOTFLG)=1 IF EOT
01104 101004      MOV 0,0,SZR ;SKIP IF NOT EOT
01105 002231      JMP @TAPRET  ;EXIT IF EOT
01106 020373      LDA 0,C704   ;IF ITS LAYE,REWIND
01107 107414      AND# 0,1,SZR ;OR ILLEGAL ITS A
01110 002212      JMP @IWE6    ;FATAL ERROR.
01111 030243      LDA 2,UNIT  ;INCREMENT RECORD
01112 021076      LDA 0,REC,2
01113 101500      INCL 0,0
01114 101240      MOVOR 0,0
01115 041076      STA 0,REC,2
01116 020235      LDA 0,WC
01117 024234      LDA 1,BUFF
01120 106400      SUB 0,1     ;START ADDRESS+WC
01121 061422      DIB 0,.MTA  ;SHOULD EQUAL THE
01122 106414      SUB# 0,1,SZR ;CONTENTS OF THE
01123 000536      JMP WES     ;TAPE ADDRESS REGISTER.
01124 030243      WR3: LDA 2,UNIT

```



```

01125 024374      LDA 1,C2040
01126 060422      DIA 0.,MTA
01127 107415      AND# 0,1,SNR
01130 000462      JMP WR6           ;NO PARITY ERROR.
01131 024252      LDA 1,WRTRY
01132 125004      MOV 1,1,SZR
01133 000442      JMP WR5           ;NOT FIRST ERROR.
01134 011056      WR4: ISZ WPAR,2
01135 020244      LDA 0,INHFLG
01136 064477      READS 1
01137 107414      AND# 0,1,SZR    ;IS SWITCH 0, PRINT
01140 000435      JMP WR5           ;PARITY MESSAGE.
01141 006221      JSR @ICRLF
01142 020204      LDA 0,C40
01143 064422      DIA 1.,MTA
01144 107415      AND# 0,1,SNR
01145 000403      JMP .+3
01146 006226      JSR @IMESS       ;"BAD TAPE"
01147 002751      BADTAP
01150 020237      LDA 0,C2K
01151 064422      DIA 1.,MTA
01152 107415      AND# 0,1,SNR
01153 000403      JMP .+3
01154 006226      JSR @IMESS       ;"WRITE PARITY"
01155 002741      MWPE
01156 006226      JSR @IMESS
01157 003321      MHO
01160 024243      LDA 1,UNIT
01161 006223      JSR @IZOCT
01162 006226      JSR @IMESS       ;"RECORD #"
01163 002707      MRECN
01164 030243      LDA 2,UNIT
01165 025076      LDA 1,REC,2
01166 125100      MOVL 1,1
01167 125220      MOVZR 1,1
01170 006245      JSR @IPDEC       ;THE RECORD NUMBER.
01171 020266      LDA 0,C4K
01172 064477      READS 1
01173 107404      AND 0,1,SZR     ;BYPASS RETRY IF SW4 SET
01174 000416      JMP WR6
01175 004504      WR5: JSR @AC
01176 177777      -1              ;SPACE OVER THE
01177 010252      ISZ WRTRY       ;ERROR RECORD.
01200 020252      LDA 0,WRTRY
01201 024255      LDA 1,RETRY
01202 106414      SUB# 0,1,SZR    ;TEST RETRY COUNTER
01203 000630      JMP WR,1        ;TRY AGAIN SAME SPOT.
01204 030243      LDA 2,UNIT
01205 011066      ISZ PERMW,2    ;INCREMENT PERM ERROR
01206 015056      DSZ WPAR,2     ;REMOVE RECOVERY PARITY AND
01207 101000      MOV# 0,0       ;ERASE TAPE.
01210 006301      ERASE
01211 000620      JMP WR+1
01212 030243      WR6: LDA 2,UNIT ;ONE GETS HERE AFTER
01213 021146      LDA 0,WDSW+10,2 ;A GOOD WRITE.

```

```

01214 024235      LDA 1,WC
01215 122422      SUBZ 1,0,SZC
01216 011136      ISZ WDSW,2      ;SUM THE WORDS
01217 041146      STA 0,WDSW+10,2 ;WRITTEN ON EACH
01220 063710      SKPOZ TTI        ;TRANSPORT.
01221 006350      JSR @IKEY       ;KEY STRUCK.
01222 002231      JMP @TAPRET
01223 006221      WE1:   JSR @ICRLF       ;"WRITE LOCK ON UNIT"
01224 006226      JSR @IMESS
01225 002757      MLOCK
01226 024243      WE1.1: LDA 1,UNIT      ;EXIT WRITE COMMAND
01227 006223      JSR @IZOCT      ;WITH FATAL ERROR.
01230 002231      JMP @TAPRET
01231 006226      ILLI:  JSR @IMESS
01232 003514      MBI
01233 024243      LDA 1,UNIT
01234 006223      JSR @IZOCT
01235 000642      JMP WR2.1
01236 060277      NOIT:  INTDS
01237 006226      JSR @IMESS
01240 003472      MTOE
01241 024243      LDA 1,UNIT
01242 006223      JSR @IZOCT
01243 000634      JMP WR2.1
01244 040256      WE3:   STA 0,BAD
01245 006221      JSR @ICRLF
01246 006246      JSR @IPOCT      ;ADDRESS SENT
01247 024256      LDA 1,BAD
01250 006246      JSR @IPOCT      ;ADDRESS RCVD
01251 006226      JSR @IMESS      ;"=SENT,RECEIVED. ADDRESS
01252 002771      MWAE        ;REGISTER FAILS TO LOAD. UNIT"
01253 000753      JMP WE1.1
01254 006221      WE2:   JSR @ICRLF
01255 006246      JSR @IPOCT      ;PRINT STATUS
01256 006226      JSR @IMESS      ;"=ERROR STATUS AFTER
01257 003020      MSTAT      ;A "C" PULSE. UNIT"
01260 000746      JMP WE1.1
01261 040256      WE5:   STA 0,BAD
01262 006221      JSR @ICRLF
01263 006246      JSR @IPOCT      ;PRINT GOOD DATA.
01264 024256      LDA 1,BAD
01265 006246      JSR @IPOCT      ;PRINT BAD DATA
01266 006226      JSR @IMESS      ;"ADDRESS+WC,DIB READING. CNTL
01267 003044      MWFAE      ;FAILURE WRITING UNIT"
01270 000736      JMP WE1.1
01271 006221      WE6:   JSR @ICRLF
01272 006246      JSR @IPOCT
01273 006226      JSR @IMESS      ;"=STATUS AFTER A WRITE
01274 003067      MWE6        ;COMMAND ON UNIT"
01275 000731      JMP WE1.1
                                ;SPACE TO EOF
01276 126520      SPACHK: SUBZL 1,1      ;AC1=1
01277 044522      STA 1,FILFLG
01300 000403      JMP BAC+2
                                ;SPACE SUBROUTINE

```

```

01301 102400 BAC: SUB 0,0
01302 040517 STA 0,FILFLG
01303 054257 STA 3,BACRET ;SPACE UNIT "N" RECORDS
01304 006341 JSR @ITOK ;IF ARGUMENT FOLLOWING
01305 030243 LDA 2,UNIT ;CALL IS (-) SPACE IS
01306 024260 LDA 1,C30 ;BACKWARD. (+)=FORWARD.
01307 022257 LDA 0,@BACRET
01310 101113 MOVL# 0,0,SNC
01311 100401 NEG 0,0,SKP
01312 024204 LDA 1,C40 ;RVS COMMAND
01313 147000 ADD 2,1
01314 065022 DOA 1,.MTA
01315 075422 DIS 3,.MTA ;SAVE NUMBER OF RECORDS.
01316 063122 DOCS 0,.MTA ;START THE TAPE MOVING.
01317 063522 BACK1: SKPBZ .MTA
01320 000777 JMP BACK1
01321 061422 DIS 0,.MTA
01322 116400 SUB 0,3 ;AC3=-#REC COUNTED BY MTA
01323 025076 LDA 1,REC,2
01324 022257 LDA 0,@BACRET
01325 101112 MOVL# 0,0,SZC ;UPDATE THE RECORD #
01326 167001 ADD 3,1,SKP
01327 166400 SUB 3,1
01330 045076 STA 1,REC,2 ;AC0=N, AC1=-#REC SPACED
01331 024470 LDA 1,FILFLG
01332 125004 MOV 1,1,SZR
01333 000421 JMP BAC.1 ;SPACE TO EOF
01334 101112 MOVL# 0,0,SZC ;N SHOULD=WC
01335 100400 NEG 0,0
01336 117014 ADD# 0,3,SZR ;SKIP IF N=WC
01337 000425 JMP BAC.2 ;ERROR
01340 010257 BAC.5: ISZ BACRET ;INCREMENT RETURN
01341 064422 DIA 1,.MTA
01342 020343 LDA 0,C72K
01343 107415 AND# 0,1,SNR ;CHECK THE STATUS.
01344 002257 JMP @BACRET
01345 006221 JSR @ICRLF ;PRINT STATUS.
01346 006246 JSR @IPOCT
01347 006226 JSR @IMESS ;"=STATUS AFTER SPACE, UNIT "
01350 003106 MSPST
01351 024243 LDA 1,UNIT
01352 006223 JSR @IZOCT ;PRINT UNIT #
01353 002257 JMP @BACRET
;EXPECTING EOF TO STOP SPACE, ALRGE WC
;AC0=LARGE WC, AC3=(-)#REC SPACED
;RELREC= #REC IN FILE(+ OR -)
01354 026247 BAC.1: LDA 1,@IRECN
01355 125112 MOVL# 1,1,SZC ;MAKE AC1 POSITIVE
01356 124400 NEG 1,1
01357 125400 INC 1,1
01360 137015 ADD# 1,3,SNR
01361 000757 JMP BAC.5 ;COUNT OK
01362 101112 MOVL# 0,0,SZC ;MAKE 0 POSITIVE
01363 100400 NEG 0,0
;REC SPACED NOT EQUAL THAT SPECIFIED

```

```

;AC0=# SPECIFIED (+)< AC3= #SPACED (-)
;AC1= !#RECI IN FILE
01364 040300 BAC.2: STA 0,SAV0 ;SAVE AC0
01365 044301 STA 1,SAV1 ;SAVE AC1
01366 054302 STA 3,SAV3 ;SAVE AC3
01367 006226 JSR @IMESS
01370 003363 MSPC1 ;"SPACING ERROR, UNIT"
01371 024243 LDA 1,UNIT
01372 006223 JSR @IZOCT ;PRINT UNIT #
01373 006226 JSR @IMESS
01374 003377 MSPC2 ;"COMMAND VS SPACE"
01375 022257 LDA 0,@BACRET
01376 101113 MOVL# 0,0,SN0
01377 000403 JMP .+3
01400 006226 JSR @IMESS
01401 003411 MSPC3 ;"- "
01402 024300 LDA 1,SAV0
01403 006223 JSR @IZOCT ;PRINT # REC
01404 006226 JSR @IMESS
01405 003412 MSPC4 ;"RECORDS DETECTED="
01406 024302 LDA 1,SAV3
01407 124400 NEG 1,1
01410 006223 JSR @IZOCT ;PRINT #RE% SPACED
01411 024410 LDA 1,FILFLG
01412 125005 MOV 1,1,SNR
01413 000725 JMP BAC.5
01414 006226 JSR @IMESS
01415 003425 MSPC5 ;"RECORDS IN FILE="
01416 024301 LDA 1,SAV1
01417 006223 JSR @IZOCT
01420 000720 JMP BAC.5
01421 000000 FILFLG: 0 ;=1 IF NOT SPACING TO EOF
01422 054261 ERAS: STA 3,ERARET ;ERASE TAPE
01423 004505 JSR TOK ;WAIT UNIT READY.
01424 064422 DIA 1,.MTA
01425 020253 LDA 0,C4
01426 107414 AND# 0,1,SZR
01427 000454 JMP WEOF1 ;UNIT IS WRITE LOCKED.
01430 030243 LDA 2,UNIT
01431 024262 LDA 1,C70
01432 147000 ADD 2,1
01433 065122 DOAS 1,.MTA ;START THE TAPE
01434 063522 SKPBZ .MTA
01435 000777 JMP .-1 ;WAIT CNTL FREE.
01436 004464 JSR XEOT ;TEST END OF TAPE FLAG
01437 064422 DIA 1,.MTA
01440 020343 LDA 0,C72K
01441 107414 AND# 0,1,SZR ;CHECK STATUS FOR ERROR
01442 000402 JMP ERA1
01443 002261 JMP @ERARET
01444 006221 ERA1: JSR @ICRLF
01445 006246 JSR @IPOCT
01446 006226 JSR @IMESS ;"=STATUS AFTER ERASE
01447 003125 ERA1 ;ON UNIT"
01450 000436 JMP WEOFF

```

```

01451 054261 WEOF:   STA 3,ERARET   ;WRITE EOF'..
01452 004456       JSR TOK
01453 064422       DIA 1,.MTA
01454 020253       LDA 0,C4
01455 107414       AND# 0,1,SZR
01456 000425       JMP WEOF1       ;WRITE LOCKED UNIT.
01457 020263       LDA 0,C100
01460 107000       ADD 0,1        ;CALCULATE PARITY
01461 107400       AND 0,1        ;FOR 7-9 TRACK
01462 020265       LDA 0,C60
01463 107000       ADD 0,1
01464 030243       LDA 2,UNIT
01465 147000       ADD 2,1
01466 065122       DOAS 1,.MTA   ;START XPORT
01467 063522       SKPBZ .MTA
01470 000777       JMP .-1        ;WAIT CNTL FREE.
01471 004431       JSR XEOT
01472 011076       ISZ REC,2
01473 064422       DIA 1,.MTA
01474 020343       LDA 0,C72K
01475 107414       AND# 0,1,SZR   ;CHECK FOR STATUS
01476 000413       JMP WEOF2       ;ERROR,
01477 020372       LDA 0,C400
01500 107415       AND# 0,1,SNR
01501 000415       JMP WEOF3       ;NO EOF STATUS.
01502 002261       JMP @ERARET
01503 006221 WEOF1:   JSR @ICRLF
01504 006226       JSR @IMESS     ;"WRITE LOCK ON UNIT"
01505 002757       MLOCK
01506 024243 WEOFF:   LDA 1,UNIT     ;PRINT UNIT#
01507 006223       JSR @IZOCT
01510 002261       JMP @ERARET
01511 006221 WEOF2:   JSR @ICRLF
01512 006246       JSR @IPOCT     ;PRINT STATUS
01513 006226       JSR @IMESS     ;"-STATUS AFTER A WRITE
01514 003144 MEOF3:   MEOF3       ;EOF COMMAND ON UNIT"
01515 000771       JMP WEOFF
01516 006221 WEOF3:   JSR @ICRLF
01517 006226       JSR @IMESS     ;"NO EOF STATUS DETECTED
01520 003164 MEOF4:   MEOF4       ;ON WRITING A EOF."
01521 000765       JMP WEOFF
01522 060422 XEOT:   DIA 0,.MTA   ;SET OR CLEAR SOFT
01523 024240       LDA 1,C1000   ;EOT FLAG.
01524 107620       ANDZR 0,1
01525 125300       MOVS 1,1
01526 044242       STA 1,EOTFLG
01527 001400       JMP 0,3
01530 063522 TOK:   SKPBZ .MTA   ;TAPE UNIT O.K.?
01531 000777       JMP .-1      ;THIS ROUTINE WILL
01532 020243       LDA 0,UNIT   ;WAIT UNTIL THE
01533 061022       DOA 0,.MTA   ;UNIT BECOMES READY.
01534 064422       DIA 1,.MTA
01535 125213       MOVR# 1,1,SNC
01536 000776       JMP .-2
01537 001400       JMP 0,3

```

```

01540 054231 REW: STA 3,TAPRET ;REWIND...
01541 004767 JSR TOK ;WAIT UNIT READY
01542 024230 LDA 1,C10
01543 030243 REW1: LDA 2,UNIT
01544 147000 ADD 2,1
01545 065122 DOAS 1,.MTA ;START THE REWIND.
01546 102620 SUBZR 0,0
01547 041076 STA 0,REC,2
01550 002231 JMP @TAPRET
01551 054231 UNLD: STA 3,TAPRET ;REWIND/UNLOAD...
01552 004756 JSR TOK
01553 024273 LDA 1,C20
01554 000767 JMP REW1
01555 102541 C: SUBOL 0,0,SKP ;CHECK DATA...
01556 102520 G: SUBZL 0,0 ;GENERATE DATA...
01557 040303 STA 0,FSTGC ;C(BUFF)=START ADDRESS
01560 054304 STA 3,GRET ;C(WC) =NUMBER OF WORDS
01561 021400 LDA 0,0,3 ;WORD FOLLOWING THE CALL
01562 040305 STA 0,PATT ;DEFINES THE PATTERN.
01563 010304 ISZ GRET
01564 020234 LDA 0,BUFF
01565 040020 STA 0,IDX0
01566 014020 DSZ IDX0
01567 020235 LDA 0,WC
01570 040311 STA 0,GWC
01571 030243 LDA 2,UNIT ;SELECT UNIT TO
01572 071022 DOA 2,.MTA ;DETERMINE THE
01573 070422 DIA 2,.MTA ;7-9 CHANNEL MASK
01574 024263 LDA 1,C100 ;FOR CHECKING DATA.
01575 133404 AND 1,2,SZR
01576 000404 JMP GG.2
01577 064477 READS 1
01600 030241 LDA 2,PKFLG ;CHECK IF PACK MODE
01601 133404 AND 1,2,SZR
01602 102001 GG.2: ADC 0,0,SKP
01603 020306 LDA 0,C37477
01604 040310 STA 0,DATMSK
01605 014303 DSZ FSTGC ;GEN/CHECK FIRST SWITCH
01606 000510 JMP CC.2
01607 006305 GG: JSR @PATT ;GENERATE DATA
01610 030310 LDA 2,DATMSK
01611 143400 AND 2,0
01612 030210 LDA 2,PARITY ;CHECK PARITY SELECT
01613 024263 LDA 1,C100
01614 147405 AND 2,1,SNR ;IS PARITY SELECT ODD?
01615 000414 JMP GG.1-1 ;EXIT IF ODD PARITY
01616 064477 READS 1
01617 030241 LDA 2,PKFLG ;CHECK IF PACK MODE
01620 133404 AND 1,2,SZR
01621 000420 JMP GG.4 ;GO TO PACK MODE ZERO DET
01622 024416 LDA 1,C5012
01623 030331 LDA 2,C377
01624 143415 AND# 2,0,SNR
01625 004410 JSR GG.3 ;ADD UI IF LSS ZERO
01626 150000 COM 2,2

```

```

01627 143415      AND# 2,0,SNR
01630 004405      JSR GG.3          ;ADD 12 IF MSB ZERO
01631 042020      STA 0,@IDX0      ;STORE IN MEMORY
01632 010311      GG.1: ISZ GWC
01633 000754      JMP GG
01634 002304      JMP @GRET
01635 147400      GG.3: AND 2,1
01636 123000      ADD 1,0
01637 001400      JMP 0,3
01640 005012      C5012: 5012
01641 024413      GG.4: LDA 1,CPMZ
01642 030272      LDA 2,CB3
01643 143415      AND# 2,0,SNR    ;IS LSB ZERO?
01644 004771      JSR GG.3        ;ADD 12 TO ZERO CHAR
01645 030271      LDA 2,CB2
01646 143415      AND# 2,0,SNR    ;IS MIDDLE CHAR ZERO
01647 004766      JSR GG.3        ;ADD 12 TO ZERO CHAR
01650 030270      LDA 2,CB1
01651 143415      AND# 2,0,SNR    ;IS MSB ZERO?
01652 004763      JSR GG.3        ;ADD 12 TO ZERO CHAR
01653 000756      JMP GG.1-1
01654 024252      CPMZ: 24252
01655 054312      CKER: STA 3,CKRET    ;A ERROR DETECTED ON
01656 044256      STA 1,BAD
01657 010303      ISZ FSTGC
01660 000407      JMP CK1         ;DONT PRINT HEADER
01661 006221      JSR @ICRLF     ;"ADDR GOOD BAD WORD
01662 006221      JSR @ICRLF     ;REC.# UNIT"
01663 006226      JSR @IMESS
01664 003212      MHEAD
01665 030243      LDA 2,UNIT
01666 011126      ISZ DATER,2
01667 030303      CK1: LDA 2,FSTGC
01670 176120      ADCZL 3,3
01671 157036      ADDZ# 2,3,SEZ
01672 002312      JMP @CKRET     ;EXIT IF>3 ERRORS.
01673 006221      CK2: JSR @ICRLF
01674 024020      LDA 1,IDX0
01675 006223      JSR @IZOCT    ;PRINT ADDRESS
01676 024313      LDA 1,GOOD
01677 006246      JSR @IPOCT    ;GOOD DATA
01700 024256      LDA 1,BAD
01701 006246      JSR @IPOCT    ;BAD DATA
01702 020234      CK3: LDA 0,BUFF
01703 024020      LDA 1,IDX0
01704 106400      SUB 0,1
01705 006223      JSR @IZOCT    ;WORD#
01706 030243      LDA 2,UNIT
01707 025076      LDA 1,REC,2
01710 125100      MOVL 1,1
01711 125220      MOVZR 1,1
01712 006245      JSR @IPDEC    ;RECORD#
01713 024243      LDA 1,UNIT
01714 006223      JSR @IZOCT
01715 002312      JMP @CKRET

```

```

01716 102400 CC.2: SUB 0,0
01717 040276 STA 0,LPC
01720 040277 STA 0,CRC
01721 006305 CC: JSR @PATT ;CHECK THE DATA
01722 026020 LDA 1,@IDX0
01723 030310 LDA 2,DATMSK ;MASK FOR 7-9
01724 143400 AND 2,0 ;CHANNEL TAPE
01725 040313 STA 0,GOOD
01726 106414 SUB# 0,1,SZR
01727 004726 JSR CKER ;CHECK ERROR
01730 020313 LDA 0,GOOD
;SEPARATE BYTES AND COMPUTE PARITY
;7TRK OR 9TRK
01731 064422 DIA 1,.MTA
01732 034263 LDA 3,C100
01733 137404 AND 1,3,SZR ;IF 9TRK
01734 000444 JMP CRCG ;GEN CRC
;7TRK LPC GEN
01735 064477 READS 1
01736 030241 LDA 2,PKFLG
01737 133404 AND 1,2,SZR ;IF PACK MODE
01740 000401 JMP .+1
;7TRK NON PACK LPC GEN
01741 024331 LDA 1,C377
01742 107400 AND 0,1
01743 004424 JSR LPCG
01744 024267 LDA 1,C1774
01745 107700 ANDS 0,1
01746 004421 JSR LPCG
01747 000444 JMP CC.3
;7TRK PACK LPC GEN
01750 024270 PCKG: LDA 1,CB1
01751 107700 ANDS 0,1
01752 127620 ANDZR 1,1
01753 127620 ANDZR 1,1
01754 004413 JSR LPCG
01755 024271 LDA 1,CB2
01756 107620 ANDZR 0,1
01757 127620 ANDZR 1,1
01760 127620 ANDZR 1,1
01761 127620 ANDZR 1,1
01762 004405 JSR LPCG
01763 024272 LDA 1,CB3
01764 107400 AND 0,1
01765 004402 JSR LPCG
01766 000425 JMP CC.3
;7TRK GENERATE LPC
01767 054410 LPCG: STA 3,LPCGR
01770 030276 LDA 2,LPC
01771 135000 MOV 1,3
01772 157520 ANDZL 2,3
01773 133000 ADD 1,2
01774 172400 SUB 3,2
01775 050276 STA 2,LPC
01776 002401 JMP @LPCGR

```



```

01777 000000  LPCGR:  0
                    ;9TRK CRC GEN
                    ;LEAVE BYTE 1 IN AC1, BYTE 2 IN AC2
02000 024331  CRCG:  LDA 1,C377
02001 107400          AND 0,1
02002 004500          JSR PAR
02003 131000          MOV 1,2
02004 024267          LDA 1,C1774
02005 107700          ANDS 0,1
02006 004474          JSR PAR
                    ;GENERATE CRC
                    ;ALGORITHM FOR DATA P01234567
                    ;U.OLD XOR NEW. START WITH OLD=0
                    ;I.SHIFT RIGHT END AROUND (9BITS)
                    ;3.IF P=1 COMP. 2345
                    ;4.AT END COMP. ALL BUT 284
02007 020277  CC.4:  LDA 0,CRC
02010 004516          JSR GCRC          ;GEN CRC
02011 145000          MOV 2,1          ;MOV 2ND BYTE UP
02012 004514          JSR GCRC
02013 010311  CC.3:  ISZ GWC          ;COUNT WORDS
02014 000705          JMP CC
02015 064422          DIA 1,MTA
02016 030263          LDA 2,C100
02017 147405          AND 2,1,SNR          ;SKIP IF 9TRK
02020 000446          JMP CC.5
                    ;COMP ALL BUT 284
02021 024254          LDA 1,C50
02022 030207          LDA 2,C727
02023 107400          AND 0,1          ;AC1=50 PART
02024 100000          COM 0,0          ;COMP ALL
02025 143400          AND 2,0          ;MASK ALL BUT 284
02026 107000          ADD 0,1          ;ADD 284 BACK IN
02027 020331          LDA 0,C377
02030 123700          ANDS 1,0
02031 040277          STA 0,CRC
02032 030267  CC.6:  LDA 2,C1774
02033 066422          DIC 1,MTA
02034 147400          AND 2,1
02035 122415          SUB# 1,0,SNR
02036 002304          JMP @GRET          ;EXIT...
02037 040300  CC.1:  STA 0,SAV0          ;ERROR IN CHECK CHAR(S)
02040 044301          STA 1,SAV1
02041 060477          READS 0          ;SKIP CRC IF PE
02042 024241          LDA 1,PK FLG
02043 107404          AND 0,1 SZR
02044 002304          JMP @GRET
02045 006226          JSR @IMESS
02046 003437          MCKC          ;"ERROR IN CHECK CHARACTER
                    ;RECORD #"
02047 030243          LDA 2,UNIT
02050 025076          LDA 1,REC,2          ;GET REC #
02051 125100          MOVL 1,1
02052 125220          MOVZR 1,1
02053 006245          JSR @IPDEC

```

```

02054 006226      JSR @IMESS
02055 003462      MGD                      ;"GOOD"
02056 024300      LDA 1,SAV0
02057 006246      JSR @IPOCT          ;PRINT GOOD DATA
02060 006226      JSR @IMESS
02061 003466      MBD                      ;"BAD"
02062 024301      LDA 1,SAV1
02063 006246      JSR @IPOCT          ;PRINT BAD DATA
02064 002304      JMP @GRET            ;EXIT...
02065 000752      JMP CC.1              ;ERROR IN CHECK CHAR(S)
02066 020276      CC.5:  LDA 0,LPC
02067 101305      MOVS 0,0,SNR
02070 002304      JMP @GRET            ;ALL ZEROS, DONT CHECK
02071 024307      LDA 1,C37400
02072 123400      AND 1,0
02073 070477      READS 2
02074 034241      LDA 3,PKFLG
02075 157405      AND 2,3,SNR
02076 000734      JMP CC.6
02077 103120      ADDZL 0,0
02100 030270      LDA 2,CB1
02101 000732      JMP CC.6+1
                      ;COMPUTE PARITY ON AC1
02102 054423      PAR:  STA 3,PRET
02103 176400      SUB 3,3
02104 054444      STA 3,PARCT          ;CLEAR PARITY COUNTER
02105 135000      MOV 1,3              ;PUT CHAR IN AC3
02106 175222      MOVZR 3,3,SZC        ;FOR PARITY GENERATION
02107 010441      ISZ PARCT          ;COUNT THE 1'S
02110 175004      MOV 3,3,SZR
02111 000775      JMP , -3
02112 034436      LDA 3,PARCT
02113 175200      MOVR 3,3
02114 034210      LDA 3,PARITY
02115 175006      MOV 3,3,SEZ          ;SKIP IF C OR R=0
02116 000404      JMP .+4            ;PAR=1
02117 175004      MOV 3,3,SZR
02120 176400      SUB 3,3              ;PAR=0
02121 175003      MOV 3,3,SNC
02122 034372      LDA 3,C400          ;PAR=1
02123 167000      ADD 3,1
02124 002401      JMP @PRET
02125 000000      PRET:  0
                      ;CRC SUBROUTINE
                      ;ENTER ACO=CRC, AC1=NEXT BYTES
02126 054421      GCRC:  STA 3,RCRC
                      ;AC1 XOR ACO
02127 135000      MOV 1,3
02130 117520      ANDZL 0,3
02131 123000      ADD 1,0
02132 162400      SUB 3,0              ;RESULT IN ACO
                      ;SHIFT RIGHT 1
02133 034372      LDA 3,C400
02134 101223      MOVZR 0,0,SNC        ;SKIP IF P=1
02135 000410      JMP CRDON

```

```

02136 163000      ADD 3,0          ;SHIFTED RESULT IN ACO
                  ;COMP 2345
02137 034205      LDA 3,C74
02140 024206      LDA 1,C703
02141 107400      AND 0,1          ;AC1=703 PART
02142 100000      COM 0,0         ;COMP ALL
02143 163400      AND 3,0         ;ACO=COMP OF 2345
02144 123000      ADD 1,0         ;ACO=RESULT
02145 040277      CRDON: STA 0,CRC
02146 002401      JMP @RCRC
02147 000000      RCRC: 0
02150 000000      PARCT: 0
02151 054371      RAND: STA 3,RR
02152 004403      JSR RAN
02153 000317      RANDOM
02154 002371      JMP @RR
02155 054530      RAN: STA 3,.UD03 ;GENERATE A RANDOM
02156 010527      ISZ .UD03       ;NUMBER IN C(0)
02157 031400      LDA 2,0,3
02160 021000      LDA 0,0,2       ;OLD VALUE TO ACO
02161 004405      JSR .UD50
02162 034525      LDA 3,.UD20
02163 163000      ADD 3,0
02164 041000      STA 0,0,2       ;STORE NEW VALUE.
02165 002520      JMP @.UD03
02166 024522      .UD50: LDA 1,.UD21 ;RANDOM CONTINUED
02167 044517      STA 1,.UD10
02170 105120      MOVZL 0,1
02171 125120      MOVZL 1,1
02172 014514      DSZ .UD10
02173 000776      JMP .-2
02174 107000      ADD 0,1
02175 125120      MOVZL 1,1
02176 125120      MOVZL 1,1
02177 123000      ADD 1,0
02200 001400      JMP 0,3
02201 030413      FZ: LDA 2,CFZTAB ;FLOATING ZERO
02202 101001      MOV 0,0,SKP
02203 030411      FO: LDA 2,CFZTAB ;FLOATING ONE
02204 020234      LDA 0,BUFF
02205 024020      LDA 1,IDX0
02206 106400      SUB 0,1         ;C(1)=WORD#
02207 020314      LDA 0,C17       ;MODULO 17
02210 107400      AND 0,1
02211 133000      ADD 1,2
02212 031000      LDA 2,0,2       ;PICK A TABLE ENTRY.
02213 001400      JMP 0,3         ;EXIT.
02214 002215      CFZTAB: .+1
02215 077777      077777
02216 137777      137777
02217 157777      157777
02220 167777      167777
02221 173777      173777
02222 175777      175777
02223 176777      176777

```

02224	177377		177377	
02225	177577		177577	
02226	177677		177677	
02227	177737		177737	
02230	177757		177757	
02231	177767		177767	
02232	177773		177773	
02233	177775		177775	
02234	177776		177776	
02235	100000		100000	
02236	040000		40000	
02237	020000		20000	
02240	010000		10000	
02241	004000		4000	
02242	002000		2000	
02243	001000		1000	
02244	000400		400	
02245	000200		200	
02246	000100		100	
02247	000040		40	
02250	000020		20	
02251	000010		10	
02252	000004		4	
02253	000002		2	
02254	000001		1	
02255	102401	ZEROS:	SUB 0,0,SKP	
02256	102000	ONES:	ADC 0,0	;ZEROS/ONES
02257	001400		JMP 0,3	
02260	020020	SKEW:	LDA 0,IDX0	;BIT DITTLE/SKEW TEST
02261	024234		LDA 1,BUFF	;SINGLE BIT FOLLOWED BY
02262	122400		SUB 1,0	;EVERY OTHER. C(0)=COUNT.
02263	030315		LDA 2,C7	;PATTERN IS DESIGNED FOR
02264	113400		AND 0,2,	;WORST CASE SKEW.
02265	024415		LDA 1,CFBIT	
02266	133000		ADD 1,2	
02267	025000		LDA 1,0,2	;BIT FOR 1 CHARACTER.
02270	030262		LDA 2,C70	
02271	113620		ANDZR 0,2	
02272	151220		MOVZR 2,2	
02273	151220		MOVZR 2,2	
02274	020406		LDA 0,CFBIT	;BIT FOR OTHER CHAR.
02275	113000		ADD 0,2	;PUT BITS TOGETHER
02276	021000		LDA 0,0,2	
02277	101300		MOVS 0,0	;AND EXIT
02300	123000		ADD 1,0	
02301	001400		JMP 0,3	
02302	002245	CFBIT:	CFZTAB+31	
02303	000000	.UD01:	0	
02304	000000	.UD02:	0	
02305	000000	.UD03:	0	
02306	000000	.UD10:	0	
02307	033031	.UD20:	33031	
02310	000010	.UD21:	10	
02311	006353	REL:	JSR @ISM	;SIZE MEMORY
02312	102400		SUB 0,0	;RELIABILITY TEST

```

02313 040210      STA 0,PARITY
02314 040316      STA 0,UCTR          ;FOR ALL READY,WRITE
02315 006344      JSR @IALL          ;ENABLED UNITS
02316 002460      RELXPT
02317 006354      JSR @ICLBIN      ;CLEAR ERROR BINS.
02320 006344      JSR @IALL
02321 001540      REW              ;REWIND
02322 006344      JSR @IALL
02323 001451      WEOF              ;WRITE FILE MARK
02324 020316      LDA 0,UCTR
02325 101004      MOV 0,0,SZR
02326 000406      JMP REL1
02327 006221      JSR @ICRLF      ;"NO READY WRITE ENABLED
02330 006226      JSR @IMESS      ;UNITS AVAILABLE"
02331 003231      MNOACT
02332 063077      HALT
02333 000756      JMP REL
02334 006342      REL1: JSR @.RAND      ;GENERATE RANDOM
02335 030753      LDA 2,.UD21
02336 024314      LDA 1,C17
02337 147400      AND 2,1
02340 044750      STA 1,.UD21
02341 010747      ISZ .UD21
02342 030321      LDA 2,C3400      ;PICK A UNIT AT
02343 113700      ANOS 0,2        ;RANDOM FROM THE
02344 025046      LDA 1,AVAIL,2    ;AVAILABLE UNITS.
02345 034316      LDA 3,UCTR
02346 167415      AND# 3,1,SNR
02347 000765      JMP REL1        ;UNIT NOT AVAIL
02350 050243      REL2: STA 2,UNIT
02351 006360      WFILE          ;WRITE A FILE MARK
02352 006342      JSR @.RAND
02353 040320      STA 0,RELRAN
02354 024315      LDA 1,C7
02355 107405      AND 0,1,SNR
02356 124001      COM 1,1,SKP
02357 124400      NEG 1,1        ;NUMBER OF RECORDS
02360 044457      STA 1,RELREC    ;IN FILE MODULO 7.
02361 044322      STA 1,RELTEM
02362 024352      REL.2: LDA 1,C1777 ;MAX WC
02363 030351      LDA 2,CMEND
02364 107022      ADDZ 0,1,SZC
02365 000411      JMP REL.4        ;CA+1777=OVERFLOW
02366 132423      SUBZ 1,2,SNC
02367 000407      JMP REL.4        ;CA+1777>MEMORY END
02370 024323      REL.3: LDA 1,CEND
02371 106423      SUBZ 0,1,SNC
02372 000407      JMP REL3--1
02373 024372      LDA 1,C400      ;CA<PROGRAM END
02374 125000      ADD 1,0
02375 000773      JMP REL.3
02376 024372      REL.4: LDA 1,C400
02377 122400      SUB 1,0
02400 000762      JMP REL.2
02401 040234      STA 0,BUFF      ;START ADDRESS

```

```

02402 006342 REL3: JSR @.RAND ;GENERATE A RANDOM
02403 024324 LDA 1,C777 ;MODULE 1024 WORDS.
02404 107405 AND 0,1,SNR
02405 124121 COMZL 1,1,SKP
02406 124520 NEGZL 1,1
02407 044235 STA 1,WC
02410 006364 GEN ;ILLL CORE WITH
02411 002151 RAND ;RANDOM
02412 006356 WRITE
02413 014242 DSZ EOTFLG
02414 101001 MOV 0,0,SKP
02415 000451 JMP REL5 ;END OF TAPE
02416 010322 ISZ RELTEM
02417 000763 JMP REL3
02420 006360 WFILE ;WRITE A FILE MARK
02421 014242 DSZ EOTFLG
02422 101001 MOV 0,0,SKP
02423 000443 JMP REL5 ;END OF TAPE
02424 006357 SPACE ;BACK OVER FILE
02425 177777 -1 ;JUST WRITZEN.
02426 020317 LDA 0,RANDOM
02427 101112 MOVL# 0,0,SZC
02430 000406 JMP REL3.
02431 006366 SPCHK ;SPACE BACK TO
02432 177000 -1000 ;A FILE MARK.
02433 006357 SPACE ;FORWARD OVER THE
02434 000001 1 ;FILE MARK
02435 000403 JMP REL4-2
02436 006357 REL3.: SPACE ;SPACE BACK THE
02437 000000 RELREC: 0 ;"N" RECORDS
02440 020320 LDA 0,RELREC ;RESET RANDOM FOR
02441 040317 STA 0,RANDOM ;READING THE FILE.
02442 006342 REL4: JSR @.RAND
02443 024324 LDA 1,C777
02444 107405 AND 0,1,SNR
02445 124121 COMZL 1,1,SKP
02446 124520 NEGZL 1,1
02447 044235 STA 1,WC
02450 006355 READ ;READ AND CHECK THE
02451 006363 CHECK ;RECORDS IN THE FILE.
02452 002151 RAND
02453 010764 ISZ RELREC
02454 000766 JMP REL4
02455 006357 SPACE ;SPACE OVER THE FILE MARK.
02456 000001 1
02457 000655 JMP REL1
02460 030243 RELXPT: LDA 2,UNIT ;ASSIGN THE READY XPORTS
02461 020316 LDA 0,UCTR
02462 025046 LDA 1,AVAIL,2
02463 123000 ADD 1,0
02464 040316 STA 0,UCTR
02465 001400 JMP 0,3
02466 006340 REL5: JSR @HIST
02467 000622 JMP REL
02470 102541 INT: SUBOL 0,0,SKP ;INTERCHANGE TAPES TEST...

```

```

02471 102520 INTW:  SUBZL 0,0          ;HERE IF WRITING THE TAPE.
02472 040325      STA 0,INTEM
02473 054326      STA 3,INTRET
02474 006354      JSR @ICLBIN
02475 020345      LDA 0,M400
02476 040235      STA 0,WC
02477 102400      SUB 0,0
02500 040210      STA 0,PARITY
02501 004542 INT1:  JSR ALL          ;REWIND THE UNITS.
02502 001540      REW
02503 102400      SUB 0,0
02504 040316      STA 0,UCTR
02505 004536      JSR ALL          ;COUNT THE READY UNITS
02506 002517      INTISZ          ;IN THE SYSTEM.
02507 020316      LDA 0,UCTR
02510 101004      MOV 0,0,SZR
02511 000410      JMP INT2
02512 006221      JSR @ICRLF
02513 006226      JSR @IMESS          ;"NO READY WRITE ENABLED
02514 003231      MNOACT          ;UNITS AVAILABLE"
02515 063077      HALT             ;PLACE UNITS WRITE ENABLE,
02516 000763      JMP INT1         ;AND ON LINE. PRESS CONT,
02517 010316 INTISZ: ISZ UCTR      ;UNIT COUNTER
02520 001400      JMP 0,3
02521 014325 INT2:  DSZ INTEM
02522 000437      JMP INT5          ;DONT WRITE ON UNITS
02523 020347      LDA 0,M2000
02524 040325      STA 0,INTEM
02525 006364      GEN
02526 002260      SKEW
02527 004514      JSR ALL          ;WRITE SKEW PATTERN
02530 001030      WR              ;100 TIMES ON ALL UNITS.
02531 010325      ISZ INTEM
02532 000775      JMP .-3
02533 020347 INT3:  LDA 0,M2000
02534 040325      STA 0,INTEM
02535 020330      LDA 0,C12345     ;INIT RANDOM
02536 040317      STA 0,RANDOM
02537 024351 INT4:  LDA 1,C377
02540 107405      AND 0,1,SNR
02541 124121      COMZL 1,1,SKP
02542 124520      NEGZL 1,1
02543 044233      STA 1,WC
02544 006364      GEN
02545 002151      RAND
02546 004475      JSR ALL          ;WRITE ALL UNITS
02547 001030      WR              ;WITH THIS RAND PATTERN,
02550 020317      LDA 0,RANDOM     ;WRITE 100 RECORDS
02551 010325      ISZ INTEM        ;ON EACH UNIT
02552 000765      JMP INT4
02553 004470      JSR ALL          ;WRITE 2 FILE MARKS.
02554 001451      WEOF            ;AND REWIND.
02555 004466      JSR ALL
02556 001451      WEOF
02557 004464      JSR ALL

```

```

02560 001540      REW
02561 020347  INT5:  LDA 0,M2000      ;THE UNITS ARE WRITTEN.
02562 040325      STA 0,INTEM
02563 020345      LDA 0,M400
02564 040235      STA 0,WC
02565 004456      JSR ALL
02566 002627      INT55      ;READ AND CHECK SKEW
02567 010325      ISZ INTEM      ;PATTERN ON EACH UNIT.
02570 000773      JMP .-3
02571 020347      LDA 0,M2000
02572 040325      STA 0,INTEM
02573 020330      LDA 0,C12345      ;INIT RANDOM
02574 040332  INT6:  STA 0,INTRAN
02575 024331      LDA 1,C377
02576 107405      AND 0,1,SNR
02577 124121      COMZL 1,1,SKP
02600 124520      NEGZL 1,1
02601 044235      STA 1,WC      ;GET RANDOM WORD COUNT.
02602 004441      JSR ALL      ;INIT RAND,READ AND
02603 002634      INT66      ;CHECK RANDOM ON ALL UNITS.
02604 020317      LDA 0,RANDOM
02605 010325      ISZ INTEM      ;DO IT 100 TIMES
02606 000766      JMP INT6
02607 014316      DSZ UCTR
02610 000404      JMP INT7
02611 006340      JSR @HIST      ;MORE INTERCHANGE REQUIRED.
02612 010326      ISZ INTRET      ;LAST INTERCHANGE.
02613 002326      JMP @INTRET
02614 004427  INT7:  JSR ALL      ;REWIND
02615 001540      REW
02616 060210      NIOC TTI
02617 006340      JSR @HIST      ;PRINT ERROR DATA
02620 006221      JSR @ICRLF      ;"ROTATE TAPES FROM UNIT
02621 006226      JSR @IMESS      ;1 TO 2, 2 TO 3, ETC. PRESS
02622 003253      MROTATE      ;TTI KEY TO CONTINUE."
02623 063610      SKPDN TTI      ;WAIT FOR OPERATOR
02624 000777      JMP .-1      ;TO PRESS KEY.
02625 060210      NIOC TTI
02626 000733      JMP INT5
02627 054334  INT55:  STA 3,RET56      ;READ AND CHECK
02630 006355      READ      ;THE SKEW PATTERN.
02631 006365      CHECK
02632 002260      SKEW
02633 002334      JMP @RET56
02634 054334  INT66:  STA 3,RET56      ;READ AND CHECK
02635 020332      LDA 0,INTRAN      ;RANDOM#
02636 040317      STA 0,RANDOM
02637 006355      READ
02640 006365      CHECK
02641 002151      RAND
02642 002334      JMP @RET56
02643 054333  ALL:  STA 3,ALLRET      ;EXECUTE THE ROUTINE
02644 063522      SKPBZ .MTA      ;POINTED TO BY THE
02645 000777      JMP .-1      ;WORD FOLLOWING THE CALL.
02646 102400      SUB 0,0      ;IF THE UNIT IS O.K.

```



```

02647 040243          STA 0,UNIT
02650 020243  ALL1.:  LDA 0,UNIT
02651 061022          DOA 0,.MTA
02652 060422          DIA 0,.MTA
02653 105120          MOVZL 0,1
02654 127102          ADDL 1,1,SZC
02655 000415          JMP ALL3          ;WAIT FOR REWIND
02656 101223          MOVZR 0,0,SNC
02657 000404          JMP ALL2          ;UNIT NOT READY.
02660 101220          MOVZR 0,0
02661 101223          MOVZR 0,0,SNC  ;CHECK WRITE LOCK
02662 007400          JSR @0,3          ;EXIT OT OPERATOR.
02663 034333  ALL2.:  LDA 3,ALLRET
02664 010243          ISZ UNIT          ;INCREMENT THE UNIT
02665 020243          LDA 0,UNIT
02666 024230          LDA 1,C10
02667 106414          SUB# 0,1,SZR  ;TEST FOR LAST UNIT
02670 000760          JMP ALL1.          ;ANOTHER UNIT
02671 001401          JMP 1,3          ;LAST UNIT, EXIT
02672 060422  ALL3.:  DIA 0,.MTA
02673 101213          MOVR# 0,0,SNC
02674 000776          JMP ,-2
02675 000753          JMP ALL1.          ;WAIT FOR READY, AFTER REW.
02676 000011  MTAB:   .TXTE I<11>I
02677 151240  MRPE:   .TXTE I R
02700 040705  EA
02701 120104  D
02702 040520  PA
02703 144722  RI
02704 054724  TY
02705 120254  ,
02706 000000  !
02707 151240  MREC#:   .TXTE I R
02710 141705  EC
02711 151317  OR
02712 121504  D#
02713 000240  !
02714 141675  MRCRC:  .TXTE I=C
02715 142510  HE
02716 045703  CK
02717 153640  W
02720 151317  OR
02721 000104  O!
02722 051675  MRE6:   .TXTE I=S
02723 040724  TA
02724 052724  TU
02725 120123  S
02726 143101  AF
02727 142724  TE
02730 120322  R
02731 120101  A
02732 142722  RE
02733 042101  AD
02734 147640  O
02735 120116  N

```

02736 047125 UN
02737 152311 IT
02740 000240 I
02741 153640 MWPE: .TXTE I W
02742 144722 RI
02743 142724 TE
02744 050240 P
02745 151101 AR
02746 152311 IT
02747 126131 Y,
02750 000240 I
02751 041240 BADTAP: .TXTE I B
02752 042101 AD
02753 152240 T
02754 050101 AP
02755 126305 E,
02756 000240 I
02757 151327 MLOCK: .TXTE IWR
02760 152311 IT
02761 120305 E
02762 147714 LO
02763 045703 CK
02764 147640 O
02765 120116 N
02766 047125 UN
02767 152311 IT
02770 000240 I
02771 051675 MWAE: .TXTE I=S
02772 047305 EN
02773 126324 T,
02774 142722 RE
02775 053303 CV
02776 027104 D.
02777 040640 A
03000 042104 DD
03001 142722 RE
03002 051523 SS
03003 151240 R
03004 043705 EG
03005 143240 F
03006 144501 AI
03007 120314 L
03010 147724 TO
03011 146240 L
03012 040717 OA
03013 027104 D.
03014 052640 U
03015 144516 NI
03016 120324 T
03017 000000 I
03020 142675 MSTAT: .TXTE I=E
03021 151322 RR
03022 151317 OR
03023 051640 S
03024 040724 TA

03025 052724 TU
03026 120123 S
03027 143101 AF
03030 142724 TE
03031 120322 R
03032 120101 A
03033 141442 "C
03034 120042 "
03035 052520 PU
03036 051714 LS
03037 027305 E.
03040 052640 U
03041 144516 NI
03042 120324 T
03043 000000 !
03044 040640 MWFAE: .TXTE I A
03045 042104 DD
03046 142722 RE
03047 051523 SS
03050 153453 +W
03051 126303 C,
03052 144504 DI
03053 120102 B
03054 142722 RE
03055 042101 AD
03056 047311 IN
03057 027107 G.
03060 153640 W
03061 144722 RI
03062 142724 TE
03063 052640 U
03064 144516 NI
03065 120324 T
03066 000000 !
03067 051675 MWE6: .TXTE I=S
03070 040724 TA
03071 052724 TU
03072 120123 S
03073 143101 AF
03074 142724 TE
03075 120322 R
03076 120101 A
03077 151327 WR
03100 152311 IT
03101 027305 E.
03102 052640 U
03103 144516 NI
03104 120324 T
03105 000000 !
03106 051675 MSPST: .TXTE I=S
03107 040724 TA
03110 052724 TU
03111 120123 S
03112 143101 AF
03113 142724 TE

03114 120322 R
03115 120101 A
03116 050123 SP
03117 141501 AC
03120 027305 E.
03121 052640 U
03122 144516 NI
03123 120324 T
03124 000000 !
03125 051675 MERA1: .TXTE !=S
03126 040724 TA
03127 052724 TU
03130 120123 S
03131 143101 AF
03132 142724 TE
03133 120322 R
03134 120101 A
03135 151305 ER
03136 051501 AS
03137 027305 E.
03140 052640 U
03141 144516 NI
03142 120324 T
03143 000000 !
03144 051675 MEOF3: .TXTE !=S
03145 040724 TA
03146 052724 TU
03147 120123 S
03150 143101 AF
03151 142724 TE
03152 120322 R
03153 151327 WR
03154 152311 IT
03155 120305 E
03156 147705 EO
03157 027306 F.
03160 052640 U
03161 144516 NI
03162 120324 T
03163 000000 !
03164 147516 MEOF4: .TXTE INC
03165 142640 E
03166 143317 OF
03167 051640 S
03170 040724 TA
03171 052724 TU
03172 120123 S
03173 142504 DE
03174 142724 TE
03175 152303 CT
03176 042305 ED
03177 142640 O
03200 120116 N
03201 151327 WR
03202 152311 IT

03203 120305 E
03204 147705 EO
03205 027306 F.
03206 052640 U
03207 144516 NI
03210 120324 T
03211 000000 !
03212 042101 MHEAD: .TXTE IAD
03213 151104 DR
03214 043411 <11>G
03215 147717 OO
03216 004504 D<11>
03217 040502 BA
03220 004504 D<11>
03221 147727 WO
03222 042322 RD
03223 151011 <11>R
03224 141705 EC
03225 004643 #<11>
03226 047125 UN
03227 152311 IT
03230 000000 !
03231 147516 MNOACT: .TXTE INO
03232 151240 R
03233 040705 EA
03234 054504 DY
03235 153640 W
03236 144722 RI
03237 142724 TE
03240 142640 E
03241 040516 NA
03242 146102 BL
03243 042305 ED
03244 052640 U
03245 144516 NI
03246 051724 TS
03247 040640 A
03250 040526 VA
03251 146311 IL
03252 000000 !
03253 147722 MROTAT: .TXTE IRO
03254 040724 TA
03255 142724 TE
03256 152240 T
03257 050101 AP
03260 051705 ES
03261 143240 F
03262 147722 RO
03263 120115 M
03264 047125 UN
03265 152311 IT
03266 030240 O
03267 152240 T
03270 120317 C
03271 126261 1.

03272	130640	1		
03273	152240	T		
03274	120317	O		
03275	126262	2,		
03276	142640	E		
03277	141724	TC		
03300	120056	.		
03301	151120	PR		
03302	051705	ES		
03303	120123	S		
03304	142724	TE		
03305	142714	LE		
03306	054724	TY		
03307	142520	PE		
03310	045640	K		
03311	054705	EY		
03312	152240	T		
03313	120317	O		
03314	147703	CO		
03315	152116	NT		
03316	047311	IN		
03317	142525	UE		
03320	000056	!		
03321	047125	MH0:	.TXTE	IUN
03322	152311	IT		
03323	000011	<11>!		
03324	040520	MH1:	.TXTE	IPA
03325	120322	R		
03326	151327	WR		
03327	000011	<11>!		
03330	040520	MH2:	.TXTE	IPA
03331	120322	R		
03332	042322	RD		
03333	000011	<11>!		
03334	142520	MH3:	.TXTE	IPE
03335	046722	RM		
03336	153640	W		
03337	004722	R<11>		
03340	000000	!		
03341	142520	MH4:	.TXTE	IPE
03342	046722	RM		
03343	151240	R		
03344	004504	D<11>		
03345	000000	!		
03346	040504	MH5:	.TXTE	IDA
03347	040724	TA		
03350	142640	E		
03351	004722	R<11>		
03352	000000	!		
03353	042327	MH6:	.TXTE	IWD
03354	120123	S		
03355	042322	RD		
03356	000011	<11>!		
03357	042327	MH7:	.TXTE	IWD
03360	120123	S		

03361	151327	WR	
03362	000011	<11>!	
03363	005215	MSPC1:	.TXTE !<15><12>
03364	050123	SP	
03365	141501	AC	
03366	047311	IN	
03367	120107	G	
03370	151305	ER	
03371	147722	RO	
03372	125322	R,	
03373	052640	U	
03374	144516	NI	
03375	120324	T	
03376	000000	!	
03377	005215	MSPC2:	.TXTE !<15><12>
03400	147703	CO	
03401	046515	MM	
03402	047101	AN	
03403	120104	D	
03404	051711	IS	
03405	051640	S	
03406	040520	PA	
03407	142703	CE	
03410	000240	!	
03411	000055	MSPC3:	.TXTE !-!
03412	005215	MSPC4:	.TXTE !<15><12>
03413	142722	RE	
03414	147703	CO	
03415	042322	RD	
03416	120123	S	
03417	142504	DE	
03420	142724	TE	
03421	152303	CT	
03422	042305	ED	
03423	136640	=	
03424	000000	!	
03425	005215	MSPC5:	.TXTE !<15><12>
03426	142722	RE	
03427	147703	CO	
03430	042322	RD	
03431	120123	S	
03432	047311	IN	
03433	143240	F	
03434	146311	IL	
03435	120305	E	
03436	000275	=!	
03437	005215	MCKC:	.TXTE !<15><12>
03440	151305	ER	
03441	147722	RO	
03442	120322	R	
03443	047311	IN	
03444	141640	C	
03445	142510	HE	
03446	045703	CK	
03447	141640	C	

03450	040510	HA	
03451	040722	RA	
03452	152303	CT	
03453	151305	ER	
03454	120254	*	
03455	142722	RE	
03456	147703	CO	
03457	042322	RD	
03460	121640	#	
03461	000240	!	
03462	005215	MGD:	.TXTE !<15><12>
03463	147507	GO	
03464	042317	OD	
03465	000240	!	
03466	005215	MBD:	.TXTE !<15><12>
03467	040502	BA	
03470	120104	D	
03471	000240	!	
03472	005215	MTOE:	.TXTE !<15><12>
03473	144515	MI	
03474	051523	SS	
03475	042305	ED	
03476	144640	I	
03477	152116	NT	
03500	151305	ER	
03501	052722	RU	
03502	152120	PT	
03503	147640	O	
03504	120116	N	
03505	151327	WR	
03506	152311	IT	
03507	126305	E,	
03510	052640	U	
03511	144516	NI	
03512	120324	T	
03513	000000	!	
03514	005215	MBI:	.TXTE !<15><12>
03515	047311	IN	
03516	142724	TE	
03517	151322	RR	
03520	050125	UP	
03521	120324	T	
03522	143101	AF	
03523	142724	TE	
03524	120322	R	
03525	144504	DI	
03526	040523	SA	
03527	146102	BL	
03530	120305	E	
03531	146306	FL	
03532	043501	AG	
03533	051640	S	
03534	152305	ET	
03535	120056	.	
03536	052640	U	


```

03537 144516 NI
03540 120324 T
03541 000000 !
;BINARY TO DECIMAL ASCII CONVERT
;CONVERTS A DOUBLE PRECISION, TWO'S COMPLEMENT NUMBER
;TO AN ASCII DECIMAL CHARACTER STRING
;INPUT: D IN AC1, AC2 (HIGH, LOW)
;OUTPUT: ASCII CHARACTER STRING, TERMINATED BY A
;
; BIT 8 = 0. IN AC0 TO USER
;
; ROUTINE WHOSE ADDRESS MUST BE
;
; STORED IN LOCATION 41 OF PAGE 0
;
; STRING OF FORM:
;
; +NNNNNNNNNN(NULL)
;
; OR -NNNNNNNNNN(NULL)
;CALLING SEQUENCE
; JSR .DBD
; RETURN
;DESTROYED: AC1, AC2, AC3, CARRY
;UNCHANGED: AC0
03542 054456 .DBD: STA 3, .FD03 ;SAVE RETURN
03543 040454 STA 0, .FD00 ;SAVE AC0
03544 020501 LDA 0, .FD30 ;POINT TO HIGH ORDER POWER IN
;TABLE
03545 040505 STA 0, .FD12
03546 176400 SUB 3,3
03547 054367 STA 3, DIGIT
03550 044477 .FD99: STA 1, .FD10 ;SAVE ABS(NUMBER)
03551 050477 STA 2, .FD10+1
03552 034367 LDA 3, DIGIT
03553 175004 MOV 3,3, SZR
03554 006041 JSR @.FD40 ;PUT OUT SIGN OR DIGIT
03555 024472 LDA 1, .FD10 ;RESTORE ABS(NUMBER)
03556 030472 LDA 2, .FD10+1
03557 020476 LDA 0, .FD22 ;GET OCTAL 57
03560 040471 STA 0, .FD11 ;COUNT IT UP IN STORAGE
03561 034471 LDA 3, .FD12 ;CURRENT POINTER TO POWER OF
;10 TABLE
;LOW ORDER WORD
03562 021401 .FD98: LDA 0,1,3
03563 101235 MOVZR# 0,0, SNR
03564 010367 ISZ DIGIT
03565 101005 MOV 0,0, SNR ;TEST FOR END OF TABLE
03566 000426 JMP .FD97 ;DONE
03567 112420 SUBZ 0,2
03570 021400 LDA 0,0,3 ;HIGH ORDER WORD
03571 101003 MOV 0,0, SNC
03572 106001 ADC 0,1, SKP
03573 106400 SUB 0,1
03574 010455 ISZ .FD11 ;COUNT UP DIGIT
03575 125113 MOVL# 1,1, SNC ;TEST FOR <0
03576 000764 JMP .FD98 ;KEEP SUBTRACTING
03577 021401 LDA 0,1,3 ;RESTORE POSITIVE VALUE
03600 113022 ADDZ 0,2, SZC
03601 125400 INC 1,1
03602 021400 LDA 0,0,3
03603 107000 ADD 0,1

```

```

03604 175400      INC 3,3          ;BUMP AC3 TO NEXT TABLE ENTRY
03605 175400      INC 3,3
03606 054444      STA 3,.FD12
03607 020442      LDA 0,.FD11      ;GET DIGIT
03610 034265      LDA 3,C60
03611 116414      SUB# 0,3,SZR
03612 010367      ISZ DIGIT
03613 000735      JMP .FD99          ;PUT IT OUT
03614 006226      .FD97: JSR @IMESS
03615 002676      MTAB
03616 002402      JMP @.FD03      ;RETURN
03617 000000      .FD00: 0          ;SAVE ACO
03620 000000      .FD03: 0          ;SAVE RETURN
03621 035632      .FD05: 35632      ;10**9
03622 145000      145000
03623 002765      2765          ;10**8
03624 160400      160400
03625 000230      230          ;10**7
03626 113200      113200
03627 000017      17          ;10**6
03630 041100      41100
03631 000001      1          ;10**5
03632 103240      103240
          000012      .RDX 10
03633 000000      0          ;10**4
03634 023420      10000
03635 000000      0          ;10**3
03636 001750      1000
03637 000000      0          ;10**2
03640 000144      100
03641 000000      0          ;10**1
03642 000012      10
03643 000000      0          ;10**0
03644 000001      1
03645 003621      .FD30: .FD05      ;POINTER TO CONVERSION TABLE
03646 000000      0          ;END OF TABLE INDICATION
          000010      .RDX 8
          000002      .FD10: .BLK 2      ;SAVE CURRENT DOUBLE WORD
03651 000000      .FD11: 0          ;COUNT UP DIGIT WORD
03652 000000      .FD12: 0          ;POINTER TO POWER OF TEN ENTRY
03653 000053      .FD20: "+"        ;ASCII "+"
03654 000055      .FD21: "-"        ;ASCII "-"
03655 000057      .FD22: 57        ;ASCII "0" -1
          000041      .FD40=41      ;PAGE 0 PUT CHARACTER ADDRESS
          ;TTO NON INTERRUPT PACKAGE
          ;"MESS" PRINTS ASCII MESSAGES AS SPECIFIED BY ASSEMBLR
          ;"CHAR" PRINTS ASCII CHARACTER, C(0)R,C(0)L MUST BE 0
          ;WILL RETURN +2 IF C(0)R=/,CORRECTS THE PARITY,33 SIMUL.
          ;"TYPE" PRINTS C(/)R. MUST HAVE PROPER PARITY. RET. IS
          ;TO CALL+1.REPLACE THIS ROUTINE WITH INT. TYPE IF DES
          ;"CRLF" PRINTS A CARRIAGE RETURN
          ;"POCT" PRINTS C(1) IN OCTAL FOLLOWED BY A TAB
          ;"PDEC" PRINTS C(1) IN DECIMAL,LEADING ZEROS SUPPRESSED,
          ;FOLLOWED BY A TAB.
03656 054542      MESS: STA 3,MESSR      ;PRINT A TEXT MESSAGE

```

```

03657 010541      ISZ MESSR
03660 031400      LDA 2,0,3        ;C(2) POINTS TO MESSAGE
03661 024331      LDA 1,C377      ;A 8 BIT MASK
03662 021000      LDA 0,0,2        ;C(2)=DATA WORD
03663 125112      MOVL# 1,1,SZC
03664 123701      ANDS 1,0,SKP
03665 123401      AND 1,0,SKP      ;C(0)=DATA CHARACTER RIGHT
03666 151400      INC 2,2        ;INC TO NEXT WORD
03667 124000      COM 1,1        ;FLIP MASK
03670 004462      JSR CHAR.      ;PRINT
03671 000771      JMP MESS+4      ;ANOTHER
03672 002526      JMP @MESSR      ;LAST
03673 102401      ZOCT:  SUB 0,0,SKP
03674 020265      POCT:  LDA 0,C60
03675 030434      LDA 2,OCTAB      ;PRINT C(1) IN OCTAL
03676 000403      JMP .+3
03677 030442      PDEC:  LDA 2,DECTB      ;PRINT C(1) IN DECIMAL
03700 102400      SUB 0,0
03701 054450      STA 3,RADRET      ;BOTH ENTRIES PRINT NUMBER
03702 040446      STA 0,ZSUPP      ;THEN TAB TO NEXT POSITION
03703 050401      STA 2,+.1
03704 000000      DECOCT: 0        ;A"LDA 2, TABLE" INSTRUCTION
03705 010777      ISZ .-1
03706 034443      LDA 3,RADRET      ;SETUP "TAB" AT END
03707 020502      LDA 0,CHTAB
03710 151005      MOV 2,2,SNR      ;IF TABLE ENTRY=0
03711 000441      JMP CHAR.      ;EXIT WITH TAB
03712 034436      LDA 3,ZSUPP      ;ZEROS SUPPRESS STUF
03713 102400      SUB 0,0
03714 146512      DECOT:  SUBL# 2,1,SZC
03715 000405      JMP DECP
03716 106400      SUB 2,1        ;FORM THE DIGIT
03717 034265      LDA 3,C60
03720 101400      INC 0,0
03721 000773      JMP DECOT
03722 151235      DECP:  MOVZR# 2,2,SNR
03723 034265      LDA 3,C60
03724 054424      STA 3,ZSUPP      ;C(0)=DIGIT
03725 163000      ADD 3,0        ;MAKE ASCII
03726 175004      MOV 3,3,SZR
03727 004423      JSR CHAR.      ;PRINT
03730 000754      JMP DECOCT      ;GET NEXT DIGIT
03731 030426      OCTAB:  LDA 2,+.1+.-DECOCT
03732 100000      100000
03733 010000      10000
03734 001000      1000
03735 000100      100
03736 000010      10
03737 000001      1
03740 000000      0
03741 030436      DECTB:  LDA 2,+.1+.-DECOCT
03742 000012      .RDX 10
03742 023420      10000
03743 001750      1000
03744 000144      100

```

```

03745 000012      10
03746 000001      1
03747 000000      0
          000010    .RDX 8
03750 000000    ZSUPP: 0
03751 000000    RADRET: 0
03752 054440    CHAR.: STA 3,CHRET      ;PRINT C(0) RIGHT
03753 101325      MOVZS 0,0,SNR      ;RETURN +2 IF NULL
03754 001401      JMP 1,3
03755 040436      STA 0,CHSAV
03756 176000      ADC 3,3          ;COMPUTE THE PARITY
03757 117000      ADD 0,3
03760 163404      AND 3,0,SZR
03761 000775      JMP .-3
03762 176660      SUBCR 3,3        ;COMBIND PARITY WITH CHAR
03763 020430      LDA 0,CHSAV
03764 163300      ADDS 3,0
03765 034424    CHAR1: LDA 3,CHTAB      ;IS THIS A TAB
03766 116405      SUB 0,3,SNR
03767 000407      JMP .+7         ;YES
03770 004431      JSR TYPE       ;NO PRINT IT
03771 002421      JMP @CHRET     ;EXIT
03772 020422      LDA 0,CHORZ     ;SIMULATE A TAB
03773 034422      LDA 3,CHAR7     ;VIA 1 TO 8 SPACES
03774 117405      AND 0,3,SNR
03775 002415      JMP @CHRET
03776 020420      LDA 0,CH240
03777 004422      JSR TYPE
04000 000772      JMP .-6
04001 054416    CRLF: STA 3,CRLFR      ;SAVE RETURN
04002 020202      LDA 0,C215
04003 004747      JSR CHAR.     ;PRINT CARRIAGE AND LF
04004 020203      LDA 0,C212
04005 004745      JSR CHAR.
04006 102400      SUB 0,0
04007 040405      STA 0,CHORZ     ;CLEAR HORZ POSITION
04010 002407      JMP @CRLFR     ;EXIT
04011 000011    CHTAB: 11
04012 000000    CHRET: 0
04013 000000    CHSAV: 0
04014 000000    CHORZ: 0
04015 000007    CHAR7: 7
04016 000240    CH240: 240
04017 000000    CRLFR: 0
04020 000000    MESSR: 0
04021 054412    TYPE: STA 3,TYPRET     ;TYPE THE C(0)R IF
04022 010772      ISZ CHORZ
04023 074477      READS 3          ;SWITCH 1(0).
04024 175100      MOVL 3,3
04025 175102      MOVL 3,3,SZC
04026 002405      JMP @TYPRET     ;INHIBIT TYPE EXIT.
04027 063511      SKPBZ TTO
04030 000777      JMP .-1
04031 061111      DOAS 0,TTO
04032 002401      JMP @TYPRET

```

```

04033 000000  TYPRET: 0
          ;TEST LOOP BUILDER
04034 006226  BUILD:  JSR @IMESS           ;TYPE "UNIT"
04035 004562           MUNIT
04036 006216           JSR @IINP           ;GET UNIT NUMBER
04037 005060           LAST
04040 026044           LDA 1,@TIBUF       ;PLACE # TYPED IN UNIT
04041 020331           LDA 0,C377
04042 123405           AND 1,0,SNR
04043 000561           JMP STRT
04044 030315           LDA 2,C7
04045 133400           AND 1,2
04046 050243           STA 2,UNIT
04047 030465           LDA 2,TBLAD-1       ;RESET TABLE ADDR.
04050 050465           STA 2,TBLAD
04051 006354           JSR @ICLBIN
04052 006226  BLD.1:  JSR @IMESS           ;TYPE "WC  "
04053 004566           MWC
04054 006216           JSR @IINP           ;GET WORD COUNT
04055 005060           LAST
04056 020777           LDA 0,.-1         ;LOAD BUFFER BASE
04057 040021           STA 0,IDX1         ;ADDR.-1 INTO IDX1
04060 126400           SUB 1,1         ;0 TO HOLD
04061 044275           STA 1,HOLD
04062 006217           JSR @I.OBIN       ;CONVERT WC TO BINARY
04063 030352           LDA 2,C1777       ;MOD 1024
04064 133400           AND 1,2
04065 151235           MOVZR# 2,2,SNR
04066 000764           JMP BLD.1
04067 150400           NEG 2,2
04070 050235           STA 2,WC
04071 006226  GETDA:  JSR @IMESS           ;ASK FOR DATA TYPE
04072 004571           MDATA
04073 006216           JSR @IINP           ;GET INPUT
04074 005060           LAST
04075 022044           LDA 0,@TIBUF       ;GET FIRST 2 INPUT CHAR
04076 024446           LDA 1,CAL
04077 106415           SUB# 0,1,SNR
04100 000416           JMP AL           ;ALL OR ALT
04101 024444           LDA 1,CRA
04102 106415           SUB# 0,1,SNR
04103 000473           JMP LOAD+5       ;RANDOM
04104 024450           LDA 1,CWD
04105 106414           SUB# 0,1,SZR
04106 000763           JMP GETDA       ;ILLEGAL DATA SELECTION
04107 020044           LDA 0,TIBUF
04110 040021           STA 0,IDX1
04111 102400           SUB 0,0
04112 040275           STA 0,HOLD
04113 006217           JSR @I.OBIN       ;GET BIN WORD IN AC1
04114 044250           STA 1,WORD
04115 000454           JMP LOAD
04116 022045  AL:     LDA 0,@TIBUF+1     ;TAKE A LOOK AT THE
04117 024427           LDA 1,CLO         ;NEXT 2 INPUT CHAR
04120 122415           SUB# 1,0,SNR

```

```

04121 000451      JMP LOAD+1      ;ALL ZEROS
04122 024425      LDA 1,CL1
04123 122415      SUB# 1,0,SNR
04124 000447      JMP LOAD+2      ;ALL ONES
04125 024423      LDA 1,CT0
04126 122415      SUB# 1,0,SNR
04127 000445      JMP LOAD+3      ;ALTERNATE 0'S-1'S
04130 024421      LDA 1,CT1
04131 106415      SUB# 0,1,SNR
04132 000443      JMP LOAD+4      ;ALTERNATE 1'S-0'S
04133 000736      JMP GETDA      ;ILLEGAL DATA SELECTION

;LOCAL TABLES
04134 004136      .+2
04135 004136      TBLAD: .+1      ;ADDR. OF DATA
04136 002151      RAND      ;GEN SUBROUTINES
04137 004157      ALT1
04140 004161      ALTO
04141 004163      ALL1
04142 004165      ALLO
04143 004167      WD

;LOCAL CONSTANTS
04144 046101      CAL: 046101      ;ASCII L-A
04145 040522      CRA: 040522      ; A-R
04146 030114      CL0: 030114      ; C-L
04147 030514      CL1: 030514      ; 1-L
04150 030124      CT0: 030124      ; O-T
04151 030524      CT1: 030524      ; 1-T
04152 053105      CEV: 053105      ; V-E
04153 042117      COD: 042117      ; D-O
04154 042127      CWD: 042127      ; W-D

;LOCAL INDIRECT ADDRESSES
04155 004366      IRDRT: RDARG
04156 004531      IWTRT: WTARG

;DATA GENERATING SUBROUTINES
04157 020267      ALT1: LDA 0,C1774
04160 001400      JMP 0,3
04161 020331      ALTO: LDA 0,C377
04162 001400      JMP 0,3
04163 102000      ALL1: ADC 0,0
04164 001400      JMP 0,3
04165 102400      ALLO: SUB 0,0
04166 001400      JMP 0,3
04167 020250      WD: LDA 0,WORD
04170 001400      JMP 0,3

;
;DATA SELECTION ROUTINE
;
04171 010744      LOAD: ISZ TBLAD      ;ENTER SPC WORD
04172 010743      ISZ TBLAD      ;ENTER ALLO
04173 010742      ISZ TBLAD      ;ENTER ALL1
04174 010741      ISZ TBLAD      ;ENTER ALT2
04175 010740      ISZ TBLAD      ;ENTER ALT1
04176 026737      LOA 1,@TBLAD      ;ENTER RANDOM
04177 046756      STA 1,@IRDRT      ;FIX READ-WRITE ROUTINES
04200 046756      STA 1,@IWTRT      ;FOR DATA ACQUISITION

```

```

:GET PARITY SELECTION
04201 006226 GTPAR: JSR @IMESS
04202 004605 MPAR
04203 006216 JSR @IINP
04204 005060 LAST
04205 024746 LDA 1,COD
04206 022044 LDA 0,@TIBUF ;FIRST 2 CHAR TYPED
04207 106415 SUB# 0,1,SNR
04210 000406 JMP ODD ;ODD PARITY
04211 024741 LDA 1,CEV
04212 122414 SUB# 1,0,SZR
04213 000766 JMP GTPAR ;ILLEGAL PARITY ENTRY
04214 020263 LDA 0,C100 ;EVEN PAR
04215 000402 JMP .+2
04216 102400 ODD: SUB 0,0
04217 040210 STA 0,PARITY
04220 006226 AFCS: JSR @IMESS ;ASK FOR COMMAND STRING
04221 004574 MCOM
04222 006216 JSR @IINP ;GET COMMAND STRING
04223 007060 INADR: LAST+2000 ;SAVE IT IN UPPER CORE
;BIT 0 IS A BYTE FLAG
;
;THE FOLLOWING ROUTINE SCANS THE COMMAND
;STRING, EXECUTING COMMANDS AS THEY ARE
;ENCOUNTERED. IDX1 IS THE BYTE POINTER
;
04224 024777 STRT: LDA 1,INADR ;LOAD BYTE POINTER
04225 044021 STA 1,IDX1 ;FOR STRING SCAN
04226 126400 SUB 1,1
04227 044275 STA 1,HOLD ;0 TO HOLD
04230 024323 LDA 1,CEND
04231 044234 STA 1,BUFF
04232 006040 MORE: GET ;GET A CHAR IN ACO
04233 101005 MOV 0,0,SNR
04234 000600 JMP BUILD ;END OF COMMAND STRING
04235 024200 LDA 1,C15
04236 122415 SUB# 1,0,SNR
04237 002376 JMP @IBLD ;END OF COMMAND STRING
04240 024465 LDA 1,SLH
04241 122415 SUB# 1,0,SNR
04242 000770 JMP MORE ;BYPASS SLASH, GET ANOTHER
04243 024445 LDA 1,R
04244 122415 SUB# 1,0,SNR
04245 000425 JMP CHAR ;=R
04246 024446 LDA 1,S
04247 122415 SUB# 1,0,SNR
04250 000536 JMP CHAS ;=S
04251 024441 LDA 1,W
04252 122415 SUB# 1,0,SNR
04253 002415 JMP @ICHAW ;=W
04254 024444 LDA 1,E
04255 122415 SUB# 1,0,SNR
04256 000557 JMP CHAE ;=E
04257 024442 LDA 1,L
04260 122415 SUB# 1,0,SNR

```

```

04261 000566      JMP CHAL          ;=L
04262 024444      LDA 1,STAR
04263 122415      SUB# 1,0,SNR
04264 002405      JMP @ICHA,        ;=*
04265 006226      INERR: JSR @IMESS      ;ILLEGAL COMMAND STRING
04266 004610      MERR
04267 000731      JMP AFCS          ;ASK AGAIN
04270 004503      ICHAW: CHAW
04271 004474      ICHA.: CHA.
;
;SECOND CHARACTER ROUTINES
;
04272 006040      CHAR: GET          ;GET A CHAR IN ACO
04273 024416      LDA 1,D
04274 122415      SUB# 1,0,SNR
04275 000453      JMP CHR0          ;=RD (READ)
04276 024414      LDA 1,W
04277 122415      SUB# 1,0,SNR
04300 000502      JMP CHRW          ;=RW (REWIND)
04301 024412      LDA 1,U
04302 122415      SUB# 1,0,SNR
04303 000501      JMP CHRU          ;=RU (UNLOAD)
04304 024414      LDA 1,E
04305 122415      SUB# 1,0,SNR
04306 000422      JMP CHRE          ; = RE, READ EOF
04307 000756      JMP INERR        ;ILLEGAL COMMAND STRING
;
;LOCAL CONSTANTS
;
04310 000122      R:          122
04311 000104      D:          104
04312 000127      W:          127
04313 000125      U:          125
04314 000123      S:          123
04315 000106      F:          106
04316 000102      B:          102
04317 000124      T:          124
04320 000105      E:          105
04321 000114      L:          114
04322 000117      O:          117
04323 000040      SP:         40
04324 000054      COM.:       54
04325 000057      SLH:        57
04326 000052      STAR:       52
04327 070400      C70.4K:    70400
;RE (READ EOF)
04330 006040      CHRE: GET
04331 024774      LDA 1,SLH
04332 122414      SUB# 1,0,SZR
04333 000732      JMP INERR          ;NO (/) AFTER RE
04334 020236      LDA 0,C70K      ;ALLOW REOF IN
04335 040373      STA 0,C704      ;READ SUBROUTINE
04336 005355      READ
04337 025770      LDA 0,C70.4K    ;RESTORE
04340 040373      STA 0,C704

```



```

04341 060422      DIA 0,MTA
04342 024372      LDA 1,C400
04343 123414      AND# 1,0,SZR
04344 000656      JMP MORE          ;EOF FOUND
04345 006226      JSR @IMESS       ;NO EOF STATUS
04346 004673      MRET
04347 000663      JMP MORE
;COMMAND IS RD (READ)
04350 006040      CHR0:  GET          ;GET COMMA IN ACC
04351 024753      LDA 1,COM.
04352 122414      SUB# 1,0,SZR
04353 000712      JMP INERR        ;NO (,) AFTER RD
04354 006217      JSR @I,OBIN     ;GET # READS IN AC1
04355 030747      LDA 2,COM.     ;(,) SHOULD BE IN ACC
04356 142414      SUB# 2,0,SZR
04357 000706      JMP INERR        ;NO COMMA
04360 044421      STA 1,CTR1     ;SET READ COUNTER
04361 024320      LUPRD:  LDA 1,RELRA  ;RESET RANDOM # GEN.
04362 044317      STA 1,RAND0M  ;JUST IN CASE IT IS USED
04363 004563      JSR CLEAR
04364 006355      READ
04365 006365      CHECK
04366 000000      RDARG:  0          ;ADDR OF DATA GENERATOR
04367 014242      DSZ EOTFLG    ;CHECK EOT
04370 000402      JMP .+2        ;NO EOT
04371 000404      JMP REOT      ;PRINT EOT MESSAGE THEN QUIT
04372 014407      DSZ CTR1     ;DONE WITH READS
04373 000766      JMP LUPRD    ;NO
04374 000636      JMP MORE     ;YES, GET NEXT COMMAND
04375 006226      REOT:  JSR @IMESS
04376 004642      MREOT
04377 002401      JMP @.+1
04400 004034      BUILD
04401 000000      CTR1:  0          ;COUNTER
;COMMAND IS RW (REWIND)
04402 006362      CHR0:  REWIND
04403 000627      JMP MORE     ;GET NEXT COMMAND
;COMMAND IS RU (UNLOAD)
04404 006363      CHR0:  UNLOAD
04405 000625      JMP MORE     ;GET NEXT COMMAND
;
;FIRST COMMAND CHARACTER IS S
;
04406 006040      CHAS:  GET          ;GET NEXT CHAR IN ACC
04407 024706      LDA 1,F
04410 122415      SUB# 1,0,SNR
04411 000416      JMP CHSF      ;=SF (SPACE FORWARD)
04412 024704      LDA 1,B
04413 122414      SUB# 1,0,SZR
04414 000651      JMP INERR    ;ILLEGAL COMMAND
04415 006040      GET          ;=SB (SPACE BACK)
04416 024706      LDA 1,COM.
04417 122404      SUB 1,0,SZR
04420 000645      JMP INERR    ;ILLEGAL FORMAT
04421 006217      JSR @I,OBIN  ;GET # SPACES IN AC1

```

```

04422 124400      NEG 1,1
04423 044402  DOSP:  STA 1,.,+2
04424 006357      SPACE
04425 000000      0
04426 000604      JMP MORE          ;GET NEXT COMMAND
;COMMAND IS SF (SPACE FORWARD)
04427 006040  CHSF:  GET          ;GET COMMA IN ACO
04430 024674      LDA 1,COM.
04431 122414      SUB# 1,0,SZR
04432 000633      JMP INERR        ;ILLEGAL COMMAND
04433 006217      JSR @I,OBIN     ;GET # SPACES IN AC1
04434 000767      JMP DOSP

;
;FIRST COMMAND CHAR IS E
04435 006040  CHAE:  GET          ;GET NEXT IN ACO
04436 024652      LDA 1,R
04437 122414      SUB# 1,0,SZR
04440 000625      JMP INERR        ;ILLEGAL COMMAND
04441 006361      ERASE          ;COMMAND IS ER (ERASE)
04442 014242      DSZ EOTFLG    ;CHECK EOT
04443 002227      JMP @IMORE     ;GET NEXT COMMAND
04444 006226      JSR @IMESS    ;EOT DURING ERASE
04445 004656      MEEOT
04446 002227      JMP @IMORE

;
;FIRST CHARACTER IS L
;
04447 006040  CHAL:  GET          ;SKIP OVER THE OOP
04450 006040      GET
04451 006040      GET
04452 006040      GET          ;GET SPACE OR CR OR *
04453 101005      MOV 0,0,SNR
04454 002417      JMP @ISTRT     ;GO TO FIRST COMMAND
04455 024200      LDA 1,C15
04456 122415      SUB# 1,0,SNR
04457 002414      JMP @ISTRT     ;GO TO FIRST COMMAND
04460 024643      LDA 1,SP
04461 122415      SUB# 1,0,SNR
04462 000770      JMP CHAL+3     ;SKIP OVER A SPACE
04463 024643      LDA 1,STAR
04464 122414      SUB# 1,0,SZR
04465 000600      JMP INERR        ;ILLEGAL COMMAND
04466 024413      LDA 1,KHOLD    ;=LOOP *
04467 044275      STA 1,HOLD
04470 024412      LDA 1,KIDX1
04471 044021      STA 1,IDX1
04472 002227      JMP @IMORE     ;GET NEXT COMMANDAT *
04473 004224  ISTRT:  STRT

;
;FIRST CHARACTER IS A *
;
04474 024275  CHA.:  LDA 1,HOLD    ;SAVE HOLD
04475 044404      STA 1,KHOLD
04476 024021      LDA 1,IDX1    ;SAVE IDX1
04477 044403      STA 1,KIDX1

```

```

04500 002227          JMP @IMORE          ;DO NEXT COMMAND
04501 000000  KHOLD:  0
04502 000000  KIDX1:  0
;
;FIRST CHARACTER IS W
;
04503 006040  CHAW:  GET          ;GET NEXT CHAR IS ACC
04504 024613          LDA 1,T
04505 122415          SUB# 1,0,SNR
04506 000412          JMP CHWT          ;=WT (WRITE)
04507 024611          LDA 1,E
04510 122414          SUB# 1,0,SZR
04511 002434          JMP @IINER        ;ILLEGAL COMMAND
04512 006360          WFILE          ;=WE (WRITE EOF)
04513 014242          DSZ EOTFLG        ;CHECK FOR EOT
04514 002227          JMP @IMORE        ;GET NEXT COMMAND
04515 005226          JSR @IMESS        ;EOT DURING WEOF
04516 004664          MEFEOT
04517 002227          JMP @IMORE
;COMMAND IS WR (WRITE)
04520 006040  CHWT:  GET          ;GET COMMA IN ACC
04521 024603          LDA 1,COM.
04522 122414          SUB# 1,0,SZR
04523 000422          JMP IINER        ;ILLEGAL COMMAND
04524 006217          JSR @I,OBIN        ;GET # RECORDS IN ACC1
04525 044654          STA 1,CTR1        ;SET WRITE COUNTER
04526 024317          LDA 1,RANDOM        ;SAVE RANDOM NUMBER
04527 044520          STA 1,RELRAN
04530 006364          GEN          ;GENERATE DATA PATTERN
04531 000000  WTARG:  0          ;ADDR OF GEN ROUTINE
04532 006356          WRITE
04533 014242          DSZ EOTFLG        ;CHECK EOT
04534 000402          JMP .+2          ;NONE
04535 000404          JMP ETFND        ;EOT DURING WRT
04536 014643          DSZ CTR1          ;COUNT WRITES
04537 000773          JMP WTARG+1        ;DO ANOTHER
04540 002227          JMP @IMORE        ;GET NEXT COMMAND
04541 006226  ETFND:  JSR @IMESS
04542 004650          MWEOT
04543 002401          JMP @.+1
04544 004034          BUILD
04545 004265  IINER:  INERR
04546 034413  CLEAR:  STA 3,CLR
04547 034234          LDA 3,BUFF
04550 054022          STA 3,IDX2
04551 014022          DSZ IDX2
04552 176400          SUB 3,3
04553 030405          LDA 2,M2K
04554 056022          STA 3,@IDX2
04555 151404          INC 2,2,SZR
04556 000776          JMP .-2
04557 002402          JMP @CLR
04560 176000  M2K:  -2000
04561 000000  CLR:  0
;

```

;MESSAGES

;
04562 005215 MUNIT: .TXTE I<15><12>
04563 047125 UN
04564 152311 IT
04565 000240 !
04566 141727 MWC: .TXTE IWC
04567 120240
04570 000240 !
04571 040504 MDATA: .TXTE IDA
04572 040724 TA
04573 000240 !
04574 147703 MCOM: .TXTE ICO
04575 046515 MM
04576 047101 AN
04577 120104 D
04600 152123 ST
04601 144722 RI
04602 043516 NG
04603 005215 <15><12>
04604 000000 !
04605 040520 MPAR: .TXTE IPA
04606 120322 R
04607 000240 !
04610 146311 MERR: .TXTE IIL
04611 142714 LE
04612 040507 GA
04613 120314 L
04614 147703 CO
04615 046515 MM
04616 047101 AN
04617 120104 D
04620 152123 ST
04621 144722 RI
04622 043516 NG
04623 005215 <15><12>
04624 000000 !
04625 005215 MIER: .TXTE I<15><12>
04626 047311 IN
04627 142724 TE
04630 151322 RR
04631 050125 UP
04632 120324 T
04633 151306 FR
04634 046717 OM
04635 042240 D
04636 053305 EV
04637 141711 IC
04640 120305 E
04641 000000 !
04642 005215 MREOT: .TXTE I<15><12>
04643 142722 RE
04644 042101 AD
04645 142640 E
04646 152317 OT

```

04647 000000  !
04650 005215  MWEOT:  .TXTE !<15><12>
04651 151327  WR
04652 152311  IT
04653 120305  E
04654 147705  EO
04655 000324  TI
04656 005215  MEEOT:  .TXTE !<15><12>
04657 151305  ER
04660 051501  AS
04661 120305  E
04662 147705  EO
04663 000324  TI
04664 005215  MEFEOT: .TXTE !<15><12>
04665 142727  WE
04666 144706  FI
04667 142714  LE
04670 142640  E
04671 152317  OT
04672 000000  !
04673 005215  MRET:   .TXTE !<15><12>
04674 147516  NO
04675 142640  E
04676 143317  OF
04677 051640  S
04700 040724  TA
04701 052724  TU
04702 120123  S
04703 143101  AF
04704 142724  TE
04705 120322  R
04706 142722  RE
04707 141640  C
04710 046717  OM
04711 040515  MA
04712 042116  ND
04713 000000  !
;TELETYPE INPUT ROUTINE, NON INTERRUPT.  SUBROUTINE CALL
;IS FOLLOWED BY INPUT BUFFER ADDRESS-1.  INPUT WILL BE
;PACKED 2 CHAR/WORD, R TO L, BIT 3 ALWAYS ZERO, EXCEPT
;FOR CR(215).  LAST BUFFER WORD IS EITHER (215)L(0)R, OR
;(215)L(LAST CHAR)R.  KJBOUT DELETES PREVIOUS CHAR.
;ALL ACCUMULATORS ARE USED AND NONE ARE SAVED
;
;AC0=377 OR 177
;AC1 ACCEPTS CHAR FORM TTI,  AC2 PACKS BYTES
;AC3=215
;
04714 054461  INP:    STA 3,RINP      ;SAVE RETURN
04715 010460      ISZ RINP
04716 031400      LDA 2,0,3          ;AC2=BUFFER ADDR-1
04717 050020      STA 2,IDX0       ;LOAD AUTO INDEX REG.
04720 050457      STA 2,BASE       ;REMEMBER BASE ADDR.
04721 034202      LDA 3,C215
04722 152440      SUBO 2,2          ;G TO AC2, 0 TO C

```

```

04723 060210      NIOC TTI      ;CLEAR TTI
04724 063610 TTIWT:  SKPDN TTI      ;WAIT FOR INPUT
04725 000777      JMP .-1
04726 064610      DIAC 1,TTI      ;READ CHARACTER
04727 136415      SUB# 1,3,SNR      ;CR TYPED
04730 000435      JMP CRFND      ;YES, END OF INPUT
04731 020331      LDA 0,C377      ;NO: CHECK RUBOUT
04732 122415      SUB# 1,0,SNR
04733 000412      JMP DELCH      ;DELETE LAST
04734 065111      DOAS 1,TT0      ;ECHO CHAR
04735 020177      LDA 0,C177
04736 107400      AND 0,1      ;MASK OFF BIT 8
04737 133300      ADDS 1,2      ;PACK BYTES

```

```

;
;USE CARRY AS BYTE INDICATOR. IF C=0, AC2=0 AND
;LAST 2 BYTES ARE IN MEMORY. IF C=1 AC2=LAST BYTE.
;

```

```

04740 101062      MOV C 0,0,SZC      ;SKIP EVERY OTHER TIME
04741 000763      JMP TTIWT      ;GET ANOTHER BYTE
04742 052020      STA 2,@IDX0      ;STORE 2 BYTES
04743 152460      SUBC 2,2      ;CLEAR AC2, DON'T CHANGE C
04744 000750      JMP TTIWT      ;GET ANOTHER BYTE
04745 101002 DELCH:  MOV 0,0,SZC      ;CHECK BYTE POSITION
04746 000415      JMP DLCH1      ;LAST BYTE IN AC2
04747 024020      LDA 1,IDX0
04750 030427      LDA 2,BASE
04751 132415      SUB# 1,2,SNR      ;CHECK BUFFER POINTER
04752 000750      JMP TTIWT-2      ;NOTHING TO DELETE
04753 014020      DSZ IDX0      ;GET LAST BYTE FROM MEM.
04754 032020      LDA 2,@IDX0
04755 113440      ANDO 0,2      ;DELETE CHAR, SET C TO 1
04756 151300      MOVS 2,2
04757 014020      DSZ IDX0      ;RESET BUFFER POINTER
04760 024416 ECSL:  LDA 1,BSLH
04761 065111      DOAS 1,TT0      ;ECHO (1)
04762 000742      JMP TTIWT
04763 152440 DLCH1:  SUBO 2,2      ;CLEAR BYTE IN AC2, 0 TO C
04764 000774      JMP ECSL
04765 065111 CRFND:  DOAS 1,TT0      ;ECHO CHAR
04766 133300      ADDS 1,2      ;PACK BYTE
04767 052020      STA 2,@IDX0      ;STORE IN BUFFER
04770 063611      SKPDN TT0
04771 000777      JMP .-1
04772 024203      LDA 1,C212
04773 065111      DOAS 1,TT0      ;PRINT LF AUTOMATICALLY
04774 002401      JMP @RINP      ;SUBR RETURN
04775 000000 RINP:  0
04776 000334 BSLH:  334
04777 000000 BASE:  0

```

```

;CONVERT AN ASCII OCTAL CHARACTER STRING TO A BINARY
;NUMBER
;INPUT:
;
;CALLS A GET CHARACTER ROUTINE WHOSE
ADDRESS MUST BE STORED
;
;IN LOCATION 40 OF PAGE 0.
;CHARACTERS MUST BE RETURNED.
;

```

```

;
; RIGHT ADJUSTED IN ACO WITH BIT 3=0
; INPUT OF FORM:
;         00...0(BREAK)
; WHERE "0" REPRESENTS AN OCTAL DIGIT AND
; BREAK IS ANY OTHER CHARACTER
; OUTPUT:  ACO CONTAINS THE BREAK CHARACTER
;         AC1 CONTAINS THE BINARY NUMBER (MOD
;         200000 OCTAL)
; CALLING SEQUENCE:
;         JSR         .OBIN
;         RETURN
; IF AN INDICATION IS DESIRED TO SIGNAL CHARACTERS ARE
; REQUESTED, CALLING SEQUENCE:
;         JSR         .OBNI
;         RETURN
; AN ASCII "0" FOLLOWED BY A NULL CHARACTER
; WILL BE TRANSMITTED VIA ACO
; TO USER PUT CHARACTER ROUTINE WHOSE
; ADDRESS MUST BE STORED IN LOCATION 41 OF PAGE 0
; CAUTION:  RESULT IS N MOD 200000 (OCTAL)
;         E.G. 576452* CONVERTS TO 176452
; DESTROYED:  ACO, AC1, AC3, CARRY
; UNCHANGED:  AC2
05000 054435 .OBNI:  STA 3,.EE03      ;SAVE RETURN
05001 050433          STA 2,.EE02      ;SAVE AC2
05002 020437          LDA 0,.EE22
05003 006041          JSR @.EE41        ;SEND "0"
05004 102400          SUB 0,0
05005 006041          JSR @.EE41        ;SEND NULL
05006 000403          JMP .+3
05007 054426 .OBIN:  STA 3,.EE03      ;SAVE RETURN
05010 050424          STA 2,.EE02      ;SAVE AC2
05011 126400          SUB 1,1          ;CLEAR RESULT WORD
05012 044424          STA 1,.EE10
05013 006040 .EE98:  JSR @.EE40        ;GET A DIGIT
05014 030423          LDA 2,.EE20      ;OCTAL 60
05015 034423          LDA 3,.EE21      ;OCTAL 67
05016 162033          ADCZ# 3,0,SNC    ;TEST FOR 10 <=N<= 67
05017 112032          ADCZ# 0,2,SZC
05020 000411          JMP .EE99        ;NO - MUST BE BREAK CHARACTER
05021 142400          SUB 2,0          ;PUT N IN RANGE 0-7
05022 024414          LDA 1,.EE10
05023 125120          MOVZL 1,1        ;SHIFT SUM
05024 125120          MOVZL 1,1
05025 125120          MOVZL 1,1
05026 107000          ADD 0,1
05027 044407          STA 1,.EE10
05030 000763          JMP .EE98        ;LOOP TILL BREAK RECEIVED
05031 030403 .EE99:  LDA 2,.EE02      ;RESTORE AC2
05032 024404          LDA 1,.EE10      ;ANSWER TO AC1
05033 002402          JMP @.EE03      ;AND RETURN
05034 000000 .EE02:  0          ;SAVE AC2
05035 000000 .EE03:  0          ;SAVE RETURN
05036 000000 .EE10:  0          ;STORAGE FOR RESULTS
05037 000060 .EE20:  60         ;ASCII "0"

```

```

05040 000057 .EE21: 67 ;ASCII "7"
05041 000117 .EE22: "0" ;ASCII "0"
      000040 .EE40=40 ;PAGE 0 ADDRESS OF GET A
      ;CHARACTER ROUTINE
      000041 .EE41=41 ;PAGE 0 ADDRESS OF PUT A
      ;CHARACTER ROUTINE
;GET CHARACTER ROUTINE. CHARACTERS RETRIEVED
;FROM MEMORY, 2 BYTES/WORD, PACKED R-L.
;SET IDX1 TO BUFF-1
;SUBR RETURN WITH CHAR RIGHT JUSTIFIED IN ACO,
;BIT 8 ALWAYS ZERO. SET HOLD TO ZERO BEFORE
;INITIAL CALL.
;
05042 054415 GETCH: STA 3,GTCHR ;SAVE RETURN
05043 020275 LDA 0,HOLD ;GET CURRENT DATA
05044 101004 MOV 0,0,SZR ;ANY ?
05045 000405 JMP GT.1 ;YES USE IT
05046 022021 LDA 0,@IDX1 ;NO, GET SOME MORE
05047 040275 STA 0,HOLD
05050 034177 GT,2: LDA 3,C177
05051 163400 AND 3,0 ;MASK 7 BITS
05052 002405 JMP @GTCHR
05053 101300 GT,1: MOVS 0,0
05054 176400 SUB 3,3
05055 054275 STA 3,HOLD
05056 000772 JMP GT.2
05057 000000 GTCHR: 0
05060 000000 LAST: 0
      .END

```

ERRORS =

004220	AFCS	04220	04267					
004116	AL	04100	04116					
002643	ALL	00344	02501	02505	02527	02546	02553	02555
		02557	02565	02602	02614	02643		
004165	ALLO	04142	04165					
004163	ALL1	04141	04163					
002650	ALL1.	02650	02670	02675				
002663	ALL2	02657	02663					
002672	ALL3	02655	02672					
000333	ALLRE	00333	02643	02663				
004161	ALTO	04140	04161					
004157	ALT1	04137	04157					
000046	AVAIL	00046	02344	02462				
004316	B	04316	04412					
001301	BAC	00357	01175	01300	01301			
001317	BACK1	01317	01320					
000257	BACRE	00257	01303	01307	01324	01340	01344	01353
		01375						
001354	BAC.1	01333	01354					
001364	BAC.2	01337	01364					
001340	BAC.5	01340	01361	01413	01420			
000256	BAD	00256	01244	01247	01261	01264	01656	01700
002751	BADTA	00752	01147	02751				
004777	BASE	04720	04750	04777				
004052	BLD.1	04052	04066					
004776	BSLH	04760	04776					
000234	BUFF	00234	00646	01051	01117	01564	01702	02204
		02261	02401	04231	04547			
004034	BUILD	00376	00406	04034	04234	04400	04544	
001555	C	00365	01555					
000230	C10	00230	00422	00567	00730	01542	02666	
000263	C100	00263	00652	01457	01574	01613	01732	02016
		04214						
000240	C1000	00240	00710	01100	01523			
000375	C10K	00375	00533					
000330	C1234	00330	02535	02573				
000200	C15	00200	00437	04235	04455			
000314	C17	00314	02207	02336				
000233	C172K	00233						
000177	C177	00177	00435	00543	04735	05050		
000267	C1774	00267	00775	01744	02004	02032	04157	
000352	C1777	00352	02362	04063				
000273	C20	00273	01553					
000264	C200	00264						
000374	C2040	00374	00704	01125				
000203	C212	00203	04004	04772				
000202	C215	00202	04002	04721				
000201	C22	00201	00417					
000237	C2K	00237	00753	01002	01150			
000260	C30	00260	01306					
000321	C3400	00321	02342					
000307	C3740	00307	02071					
000306	C3747	00306	01603					
000331	C377	00331	01623	01741	02000	02027	02537	02575
		03661	04041	04161	04731			

000253	C4	00253	01035	01425	01454		
000204	C40	00204	00442	00745	01064	01142	01312
000372	C400	00372	01477	02122	02133	02373	02376 04342
000266	C4K	00266	00655	01171	02042		
000254	C50	00254	01044	02021			
001640	C5012	01622	01640				
000265	C60	00265	01462	03610	03674	03717	03723
000315	C7	00315	02263	02354	04044		
000262	C70	00262	01431	02270			
000206	C703	00206	02140				
000373	C704	00373	00700	01106	04335	04340	
000236	C70K	00236	04334				
004327	C70.4	04327	04337				
000207	C727	00207	02022				
000343	C72K	00343	00637	01041	01342	01440	01474
000205	C74	00205	02137				
000324	C777	00324	02403	02443			
004144	CAL	04076	04144				
000270	CB1	00270	01650	01750	02100		
000271	CB2	00271	01645	01755			
000272	CB3	00272	01642	01763			
001721	CC	01721	02014				
002037	CC.1	02037	02065				
001716	CC.2	01606	01716				
002013	CC.3	01747	01766	02013			
002007	CC.4	02007					
002066	CC.5	02020	02066				
002032	CC.6	02032	02076	02101			
000323	CEND	00323	02370	04230			
004152	CEV	04152	04211				
002302	CFBIT	02265	02274	02302			
002214	CFZTA	02201	02203	02214	02302		
004016	CH240	03776	04016				
004435	CHAE	04256	04435				
004447	CHAL	04261	04447	04462			
004272	CHAR	04245	04272				
003765	CHAR1	03765					
004015	CHAR7	03773	04015				
003752	CHAR.	03670	03711	03727	03752	04003	04005
004406	CHAS	04250	04406				
004503	CHAW	04270	04503				
004474	CHA.	04271	04474				
006365	CHECK	00377	02451	02631	02640	04365	
004014	CHORZ	00370	03772	04007	04014	04022	
004350	CHRD	04275	04350				
004330	CHRE	04306	04330				
004012	CHRET	03752	03771	03775	04012		
004404	CHRU	04303	04404				
004402	CHRW	04300	04402				
004013	CHSAV	03755	03763	04013			
004427	CHSF	04411	04427				
004011	CHTAB	03707	03765	04011			
004520	CHWT	04506	04520				
001667	CK1	01660	01667				
001673	CK2	01673					

001702	CK3	01702							
001655	CKER	01655	01727						
000312	CKRET	00312	01655	01672	01715				
004146	CL0	04117	04146						
004147	CL1	04122	04147						
000452	CLBIN	00354	00450	00452					
004546	CLEAR	04363	04546						
004561	CLR	04546	04557	04561					
000213	CLRET	00213	00452	00463					
000351	CMEND	00351	00545	02363					
000574	CMHO	00547	00574						
000621	CMIOC	00605	00621						
004153	COO	04153	04205						
004324	COM.	04324	04351	04355	04416	04430	04521		
001654	CPMZ	01641	01654						
004145	CRA	04101	04145						
000277	CRC	00277	01720	02007	02031	02145			
002000	CRCG	01734	02000						
002145	CRDON	02135	02145						
004765	CRFND	04730	04765						
004001	CRLF	00221	04001						
004017	CRLFR	04001	04010	04017					
004150	CT0	04125	04150						
004151	CT1	04130	04151						
000274	CTR	00274							
004401	CTR1	04360	04372	04401	04525	04536			
004154	CWD	04104	04154						
000214	CWPAR	00214	00456						
004311	D	04273	04311						
000126	DATER	00126	00522	01666					
000310	DATMS	00310	01604	01610	01723				
003704	DECOG	03704	03730	03731	03741				
003714	DECOT	03714	03721						
003722	DECP	03715	03722						
003741	DECTB	03677	03741						
004745	DELCH	04733	04745						
000367	DIGIT	00367	03547	03552	03564	03612			
004763	DLOH1	04746	04763						
000562	DMPD1	00562	00571						
000551	DMPDA	00467	00471	00473	00475	00477	00501	00503	
		00505	00551						
000225	DMPRE	00225	00551	00564	00572	00573			
000377	DMPT	00377	00550	00552	00561				
000222	DMPTE	00222	00510	00560	00565	00566			
004423	DOSP	04423	04434						
004320	E	04254	04304	04320	04507				
004760	ECSL	04760	04764						
000242	EOTFL	00242	00713	01103	01526	02713	02421	04367	
		04442	04513	04533					
001444	ERA1	01442	01444						
000261	ERARE	00261	01422	01443	01451	01502	01510		
001422	ERAS	00361	01422						
006361	ERASE	00377	01210	04441					
004541	ETFND	04535	04541						
004315	F	04315	04407						

001421	FILFL	01277	01302	01331	01411	01421			
002203	FO	02203							
000303	FSTGC	00303	01557	01605	01657	01667			
002201	FZ	02201							
001556	G	00364	01556						
002126	GCRC	02010	02012	02126					
006364	GEN	00377	02410	02525	02544	04530			
006040	GET	00377	04232	04272	04330	04350	04406	04415	
		04427	04435	04447	04450	04451	04452	04503	
		04520							
005042	GETCH	00040	05042						
004071	GETOA	04071	04106	04133					
001607	GG	01607	01633						
001632	GG.1	01615	01632	01653					
001602	GG.2	01576	01602						
001635	GG.3	01625	01630	01635	01644	01647	01652		
001641	GG.4	01621	01641						
000313	GOOD	00313	01676	01725	01730				
000304	CRET	00304	01560	01563	01634	02036	02044	02064	
		02070							
005057	GTCHR	05042	05052	05057					
004201	GTPAR	04201	04213						
005053	GT.1	05045	05053						
005050	GT.2	05050	05056						
000311	GWC	00311	01570	01632	02013				
000510	H0	00470	00510						
000512	H1	00472	00512						
000514	H2	00474	00514						
000516	H3	00476	00516						
000520	H4	00500	00520						
000522	H5	00502	00522						
000524	H6	00504	00524						
000527	H7	00506	00527						
000220	HISRE	00220	00464	00507					
000340	HIST	00340	00445	00447	02466	02611	02617		
000464	HISTR	00340	00464						
000275	HOLD	00275	04061	04112	04227	04467	04474	05043	
		05047	05055						
000344	IALL	00344	02315	02320	02322				
000376	IBLD	00376	04237						
004270	ICAW	04253	04270						
004271	ICHA.	04264	04271						
000370	ICHR	00370							
000354	ICLBI	00354	02317	02474	04051				
000221	ICRLF	00221	00465	00466	00554	00744	01023	01141	
		01223	01245	01254	01262	01571	01345	01444	
		01503	01511	01516	01661	01662	01673	02327	
		02512	02620						
000020	IDX0	00020	01565	01566	01631	01574	01703	01722	
		02205	02260	04717	04742	04747	04753	04754	
		04757	04767						
000021	IDX1	00021	04057	04110	04225	04471	04476	05046	
000022	IDX2	00022	04550	04551	04554				
000043	IERR	00043	00430						
000040	IGET	00040	00377						

004545	IINER	04511	04523	04545				
000216	IINP	00216	04036	04054	04073	04203	04222	
000350	IKEY	00350	00723	01221				
001231	ILLI	01071	01231					
000226	IMESS	00226	00555	00622	00751	00757	00761	00765
		01000	01025	01146	01154	01156	01162	01224
		01231	01237	01251	01256	01266	01273	01347
		01367	01373	01400	01404	01414	01446	01504
		01513	01517	01663	02045	02054	02060	02330
		02513	02621	03614	04034	04052	04071	04201
		04220	04265	04345	04375	04444	04515	04541
000227	IMORE	00227	04443	04446	04472	04500	04514	04517
		04540						
004223	INADR	04223	04224					
004265	INERR	04265	04307	04333	04353	04357	04414	04420
		04432	04440	04465	04545			
000244	INHFL	00244	00740	01135				
004714	INP	00216	04714					
002470	INT	00337	02470					
002501	INT1	02501	02516					
002521	INT2	02511	02521					
002533	INT3	02533						
002537	INT4	02537	02552					
002561	INT5	02522	02561	02626				
002627	INT55	02566	02627					
002574	INT6	02574	02606					
002634	INT66	02603	02634					
002614	INT7	02610	02614					
000325	INTEM	00325	02472	02521	02524	02531	02534	02551
		02562	02567	02572	02605			
000622	INTER	00043	00622					
002517	INTIS	02506	02517					
000332	INTRA	00332	02574	02635				
000326	INTRE	00326	02473	02612	02613			
002471	INTW	00335	02471					
000245	IPDEC	00245	00513	00515	00517	00521	00523	00773
		01170	01712	02053				
000246	IPOCT	00246	00777	01024	01246	01250	01255	01263
		01265	01272	01346	01445	01512	01677	01701
		02057	02063					
004155	IRDRT	04155	04177					
000247	IRECN	00247	01354					
000002	IS0	00002	00407	00616				
000003	IS1	00003	00410	00615				
000004	IS2	00004	00411	00614				
000005	IS3	00005	00412	00613				
000006	ISC	00006	00414	00611				
000353	ISM	00353	02311					
004473	ISTRT	04454	04457	04473				
000335	ITEST	00335	00401					
000341	ITOK	00341	00634	01033	01304			
000337	ITST	00337	00402					
000042	ITTI	00042	00424	00427				
000211	IWE1.	00211	01027					
000725	IWE2	00641	00725	01043				

000726	IWE3	00665	00726	01055				
000212	IWE6	00212	01119					
004156	IWTRT	04156	04200					
000223	IZOCT	00223	00511	00625	00764	01161	01227	01234
		01242	01352	01372	01403	01410	01417	01507
		01675	01705	01714				
000224	I,DBD	00224	00526	00531				
000217	I,CBI	00217	04062	04113	04354	04421	04433	04524
000433	KEY	00350	00433					
000447	KEY1	00441	00444	00447				
000176	KEYRE	00176	00433	00446	00451			
004501	KHOLD	04466	04475	04501				
004502	KIDX1	04470	04477	04502				
004321	L	04257	04321					
005060	LAST	00044	00045	00234	00323	04037	04055	04074
		04204	04223	05060				
004171	LOAD	04103	04115	04121	04124	04127	04132	04171
000276	LPC	00276	01717	01770	01775	02066		
001767	LPCG	01743	01746	01754	01762	01765	01767	
001777	LPCGR	01767	01776	01777				
004361	LUPRD	04361	04373					
000346	M1000	00346						
000327	M144	00327						
000347	M2000	00347	02523	02533	02561	02571		
004560	M2K	04553	04560					
000345	M400	00345	02475	02563				
003466	MBD	02061	03466					
003514	MBI	01232	03514					
003437	MCKC	02046	03437					
004574	MCOM	04221	04574					
004571	MDATA	04072	04571					
004656	MEECT	04445	04656					
004664	MEFEO	04516	04664					
003144	MEOF3	01514	03144					
003164	MEOF4	01520	03164					
003125	MERA1	01447	03125					
004610	MERR	04266	04610					
003656	MESS	00226	03656	03671				
004020	MESSR	03656	03657	03672	04020			
003462	MGD	02055	03462					
003321	MH0	00575	00762	01157	03321			
003324	MH1	00576	03324					
003330	MH2	00577	03330					
003334	MH3	00600	03334					
003341	MH4	00601	03341					
003346	MH5	00602	03346					
003353	MH6	00603	03353					
003357	MH7	00604	03357					
003212	MHEAD	01664	03212					
004625	MIER	00623	04625					
002757	MLOCK	01225	01505	02757				
003231	MNOAC	02331	02514	03231				
004232	MORE	00227	04232	04242	04344	04347	04374	04403
		04405	04426					
004605	MPAR	04202	04605					

002714	MRCRC	01001	02714						
002722	MRE6	01026	02722						
002707	MRECN	00766	01163	02707					
004642	MRECT	04376	04642						
004673	MRET	04346	04673						
003253	MROTA	02622	03253						
002677	MRPE	00760	02677						
003363	MSPC1	01370	03363						
003377	MSPC2	01374	03377						
003411	MSPC3	01401	03411						
003412	MSPC4	01405	03412						
003425	MSPC5	01415	03425						
003106	MSPST	01350	03106						
003020	MSTAT	01257	03020						
002676	MTAB	02676	03615						
003472	MTGE	01240	03472						
004562	MUNIT	04035	04562						
002771	MWAE	01252	02771						
004566	MWC	04053	04566						
003067	MWE6	01274	03067						
004650	MWEOT	04542	04650						
003044	MWFAE	01267	03044						
002741	MWPE	01155	02741						
001236	NOIT	01076	01236						
004322	C	04322							
003731	OCTAB	03675	03731						
004216	CDD	04210	04216						
002256	GNES	02256							
002102	PAR	02002	02006	02102					
002150	PARCT	02104	02107	02112	02150				
000210	PARIT	00210	00643	01046	01612	02114	02313	02500	
		04217							
000305	PATT	00305	01562	01607	01721				
001750	PCKG	01740	01750						
003677	PDEC	00245	03677						
000116	PERMR	00116	00520	01017					
000066	PERMW	00066	00516	01205					
000241	PKFLG	00241	01600	01617	01736	02074			
000215	PLACE	00215							
003674	POCT	00246	03674						
002125	PRET	02102	02124	02125					
004310	R	04243	04310	04436					
003751	RADRE	03701	03706	03751					
002155	RAN	02152	02155						
002151	RAND	00342	02151	02411	02452	02545	02641	04136	
000317	RANOO	00317	02153	02426	02441	02536	02550	02604	
		02636	04362	04526					
002147	RCRC	02126	02146	02147					
000631	RD	00355	00631						
000642	RD1	00642							
000662	RD1.	00654	00657	00662					
000670	RD2	00670	00671						
000703	RD3	00703							
000707	RD4	00707	00732	01005	01022				
000727	RD5	00706	00727						

001002	RD6	00735	00743	01002					
001016	RD7	01012	01016						
004366	RDARG	04155	04366						
000232	RDTRY	00232	00633	00733	01006	01007			
000634	RD.1	00634	01015						
001023	RE6	00702	01023						
006355	READ	00377	02450	02630	02637	04336	04364		
000076	REC	00076	00562	00673	00676	00770	01112	01115	
		01165	01323	01330	01472	01547	01707	02050	
002311	REL	00335	02311	02333	02467				
002334	REL1	02326	02334	02347	02457				
002350	REL2	02350							
002402	REL3	02372	02402	02417					
002436	REL3.	02430	02436						
002442	REL4	02435	02442	02454					
002466	REL5	02415	02423	02466					
000320	RELRA	00320	01057	02353	02440	04361	04527		
002437	RELRE	00247	02360	02437	02453				
000322	RELTE	00322	02361	02416					
002460	RELXP	02316	02460						
002362	REL.2	02362	02400						
002370	REL.3	02370	02375						
002376	REL.4	02365	02367	02376					
004375	REOT	04371	04375						
000334	RET56	00334	02627	02633	02634	02642			
000255	RETRY	00255	01201						
001540	REW	00362	01540	02321	02502	02560	02615		
001543	REW1	01543	01554						
006362	REW1N	00377	04402						
004775	RINP	04714	04715	04774	04775				
000106	RPAR	00106	00514	00737	01020				
000371	RR	00371	02151	02154					
000251	RRETR	00251	01010						
000336	RTEST	00336	00400						
004314	S	04246	04314						
000300	SAV0	00300	01364	01402	02037	02056			
000301	SAV1	00301	01365	01416	02040	02062			
000302	SAV3	00302	01366	01406					
000407	SERIN	00001	00407						
002260	SKEW	02260	02526	02632					
004325	SLH	04240	04325	04331					
000532	SMEND	00353	00532						
004323	SP	04323	04460						
006357	SPACE	00377	01013	02424	02433	02436	02455	04424	
001276	SPACH	00366	01276						
006366	SPCHK	00377	02431						
004326	STAR	04262	04326	04463					
004224	STRT	04043	04224	04473					
000534	S.1	00534	00542						
000543	S.2	00540	00543						
004317	T	04317	04504						
000431	TAPIN	00421	00431						
000231	TAPRE	00231	00631	00724	01030	01105	01222	01230	
		01540	01550	01551					
004135	TBLAD	04047	04050	04135	04171	04172	04173	04174	

		04175	04176					
000044	TIBUF	00044	04040	04075	04107	04116	04206	
001530	TOK	00341	01423	01452	01530	01541	01552	
000605	TTINT	00042	00605	00625	00627	00650		
004724	TTIWT	04724	04741	04744	04752	04762		
004021	TYPE	00041	03770	03777	04021			
004033	TYPRE	04021	04026	04032	04033			
004313	U	04301	04313					
000316	UCTR	00316	02314	02324	02345	02461	02464	02504
		02507	02517	02607				
000243	UNIT	00243	00642	00672	00714	00736	00763	00767
		01016	01111	01124	01160	01164	01204	01212
		01226	01233	01241	01305	01351	01371	01430
		01464	01506	01532	01543	01571	01665	01706
		01713	02047	02350	02460	02647	02650	02664
		02665	04046					
001551	UNLD	00363	01551					
006363	UNLOA	00377	04404					
004312	W	04251	04276	04312				
000235	WC	00235	00666	00716	01056	01116	01214	01567
		02407	02447	02476	02543	02564	02601	04070
004167	WD	04143	04167					
000156	WDSR	00156	00524	00525	00715	00720	00721	
000136	WDSW	00136	00527	00530	01213	01216	01217	
001223	WE1	01037	01223					
001226	WE1.1	00211	01226	01253	01260	01270	01275	
001254	WE2	00725	01254					
001244	WE3	00726	01244					
001261	WE5	01123	01261					
001271	WE6	00212	01271					
001451	WEOF	00360	01451	02323	02554	02556		
001503	WEOF1	01427	01456	01503				
001511	WEOF2	01476	01511					
001516	WEOF3	01501	01516					
001506	WEOFF	01450	01506	01515	01521			
006360	WFILE	00377	02351	02420	04512			
000250	WORD	00250	04114	04167				
000056	WPAR	00056	00214	00512	01134	01206		
001030	WR	00356	01030	01211	02530	02547		
001044	WR1	01044						
001077	WR2.1	01077	01235	01243				
001124	WR3	01124						
001134	WR4	01134						
001175	WR5	01133	01140	01175				
001212	WR6	01130	01174	01212				
006356	WRITE	00377	02412	04532				
000252	WRTRY	00252	01032	01131	01177	01200		
001033	WR.1	01033	01203					
004531	WTARG	04156	04531	04537				
000365	XCHECK	00365	00377					
001522	XEOT	01436	01471	01522				
000361	XERAS	00361	00377					
000360	XFILE	00360	00377					
000364	XGEN	00364	00377					
000355	XREAD	00355	00377					

000362	XREW	00362	00377					
000357	XSPAC	00357	00377					
000366	XSPC	00366	00377					
000363	XUNLD	00363	00377					
000356	XWRIT	00356	00377					
002255	ZEROS	02255						
003673	ZOCT	00223	03673					
003750	ZSUPP	03702	03712	03724	03750			
003542	.DBD	00224	03542					
005034	.EE02	05001	05010	05031	05034			
005035	.EE03	05000	05007	05033	05035			
005036	.EE10	05012	05022	05027	05032	05036		
005037	.EE20	05014	05037					
005040	.EE21	05015	05040					
005041	.EE22	05002	05041					
000040	.EE40	05013	05042					
000041	.EE41	05003	05005	05042				
005013	.EE98	05013	05030					
005031	.EE99	05020	05031					
003617	.FD00	03543	03617					
003620	.FD03	03542	03616	03620				
003621	.FD05	03621	03645					
003647	.FD10	03550	03551	03555	03556	03647		
003651	.FD11	03560	03574	03607	03651			
003652	.FD12	03545	03561	03606	03652			
003653	.FD20	03653						
003654	.FD21	03654						
003655	.FD22	03557	03655					
003645	.FD30	03544	03645					
000041	.FD40	03554	03656					
003614	.FD97	03566	03614					
003562	.FD98	03562	03576					
003550	.FD99	03550	03613					
000022	.MTA	00377	00635	00636	00645	00651	00662	00663
		00667	00670	00677	00703	00707	00727	00746
		00754	00774	01034	01040	01050	01052	01053
		01060	01061	01077	01121	01126	01143	01151
		01314	01315	01316	01317	01321	01341	01424
		01433	01434	01437	01453	01466	01467	01473
		01522	01530	01533	01534	01545	01572	01573
		01731	02015	02033	02041	02644	02651	02652
		02672	04341					
005007	.OBIN	00217	05007					
005000	.OBNI	05000						
000342	.RAND	00342	02334	02352	02402	02442		
002303	.UD01	02303						
002304	.UD02	02304						
002305	.UD03	02155	02156	02165	02305			
002306	.UD10	02167	02172	02306				
002307	.UD20	02162	02307					
002310	.UD21	02166	02310	02335	02340	02341		
002166	.UD50	02161	02166					

DEVICE CODE
LOCATIONS

DEVICE CODE

APPENDIX A

SAMPLE DRIVER PROGRAM

TC-120 MAGNETIC TAPE DRIVERS

The Western Peripherals Nova Magnetic Tape Unit Drivers allow the user to perform read, write, space, write end of file and rewind functions. The individual drivers are described on the following pages. Error conditions and calling sequences are explained in each subroutine writeup. The status and command registers are described in Appendix B.

The present source program begins at location 1200 with a ".LOC 1200" statement. If the user wants to relocate the drivers, he must replace the ".LOC 1200" statement with the desired starting address and re-assemble the program. The starting address of each subroutine is given in page zero of memory locations 160 - 170. Each subroutine may be called by linking indirect through the branch vectors, i.e. (JSR @READ where READ=161), the user has the option of using the NOVA mag tape drivers with or without interrupts. If interrupts are desired, the drivers are initialized by calling MTINT.

No registers are saved by the driver subroutines.

The proper procedure in using the tape drivers is as follows:

- 1) If interrupt control is desired, a JSR to MTINT to initialize the drivers followed by INTEN to enable the interrupts.

1) (Continued)

it is the users responsibility to handle interrupts and to call the mag tape interrupt handler when the interrupt is caused by the mag tape device 22.

A JSR @MTHND will execute the mag tape handler. A sample of interrupt service routine is given in Appendix A.

2) Prior to any driver calls, a JSR to UNIT should be made to initialize and select the unit and parity, select 9 track or 7 track. If a different mag tape unit is to be addressed, another call to UNIT can be made.

3) A call to any driver will cause the selected function to be initiated; control is then returned to the user. Then prior to further calls to the drivers, the user should do a JSR to CMPLTE. CMPLTE will wait until the tape system is not busy and advise the user of error or end of file conditions.

UNIT (Unit Select Subroutine)

The Unit subroutine is called to select unit number, formatter, parity, density, edit mode, threshold control and specified 7 or 9 track unit. UNIT must be called before executing any of the driver subroutines and when changing control from one unit to another.

READ (Read Subroutine) (Continued)

Calling Sequence:

JSR @READ	(READ=161)
WORD 1	Memory buffer address
WORD 2	Word count (negative)
WORD 3	Error return (JMP ERROR)

Return from subroutine.

At this point the user may continue his program until such time that he requires the data transfer to be complete; he then must perform a "JSR" to the "CMPLTE" subroutine which will return to him when the tape function is complete (including error retries).

The error return address is utilized when the control unit or tape unit is not ready. ACO equals the status read from the tape controller.

WRITE (Write Subroutine)

The Write Subroutine starts the mag tape to write a record onto mag tape. Arguments given the subroutine are buffer core address, number of words in the record to write and an error

WRITE (Write Subroutine) (Continued)

return address.

Calling sequence:

JSR @WRITE	(WRITE=162)
WORD 1	Memory buffer address
WORD 2	Word count (negative)
WORD 3	Error return (JMP ERROR)

Return from subroutine.

At this point the user may continue his program until such time that he requires the data transfer to be complete; he then must perform a "JSR" to the "CMLPTE" subroutine which will return to him when the tape function is complete (including error retries).

The error return address is utilized when the control unit is off-line or the write ring is not in. ACO equals status read from the tape controller on error returns.

WEOF (Write End of File Subroutine)

The WEOF subroutine writes an end of file mark. EOF status is indicated when an EOF is written, thus the EOF return address

WEOF (Write End of File Subroutine) (Continued)

is utilized when the user performs a "JSR" to "CMLPTE" following a "JSR" to "WEOF."

Calling sequence:

```
JSR @WEOF          (WEOF=163)
.WORD              (Error return address)
                  Return from subroutine
```

The error return address is utilized when the tape control unit or tape unit is not ready or there is no write ring.

SPACE (Space Forward/Backward Subroutine)

The SPACE subroutine performs space forward and space backward functions. It does not perform retries in the event of error. Spacing functions are terminated when an End of Tape, End of File or Load Point is encountered. The user may examine the contents of the address counter (DIB -,MTA) to determine the number of records actually spaces. The register is initially loaded with the negative (two's complement) value of the number of records to be spaced.

SPACE (Space Forward/Backward Subroutines) (Continued)

Calling Sequence:

JSR @ SPACE	(SPACE=164)
.WORD	Space count, positive forward, negative reverse.
.WORD	Error return address
	Return from subroutine

The error return address is utilized when the tape control unit or tape unit is not ready or is positioned at load point for a reverse operation.

REWIND (Rewind Subroutine)

The REWIND subroutine performs a rewind function.

The CMPLTE subroutine should not be called following a "JSR" to "REWIND." Rewind waits until tape has been rewound before returning.

Calling Sequence:

JSR @REWIND	(REWIND=165)
	Return from subroutine.

CMPLTE (Complete Subroutine)

The CMPLTE subroutine performs two functions:

CMPLTE (Complete Subroutine) (Continued)

- 1) It waits until the prior read, write, space or write end of file function is completed.
- 2) It indicates (in a generalized fashion) the status of that function.

Calling Sequence:

JSR @CMPLTE	(CMPLTE=166)
.WORD	Error return address
.WORD	End of file return address
	Return, no error or EOF

The error return address is utilized if the prior command was

- 1) A space forward or backward and an End of Tape, Load point or parity was encountered (without an end of file).
- 2) A read (without EOF) or a write.
 - (a) A data date, illegal, end of tape, bad tape or odd character.
 - (b) An unclearable (8 tries) parity or bad tape error.

The End-of-File return address is utilized if the prior command resulted in the EOF status being set. This return takes precedence over any other conditions hence the user should check for EOT after the EOF return is made.

MTHND (Mag Tape Interrupt Handler)

This subroutine is called by the interrupt service routine if interrupts are utilized. If no interrupts are used, it is called by the complete (CMPLTE) subroutine.

The handler reads the status of the last tape operation just completed and clears the done flag. If the error flag is not set (bit 0 of status) it zeros error flag (EFG) and busy flag (BSY) and returns. If the error is caused by End-of-File only the status word is stored in error flag (EFG) and busy is zeroed. If the error was caused by parity or bad tape only and the original command was a read or write, seven retries are attempted to recover from the error condition before the final error condition is indicated. In the case of a write command 2.5 inches of tape are erased before retrying the write command. Retries are not attempted on write end-of-file or space commands or for errors other than parity or bad tape.

Calling sequence:

JSR @MTHND	(MTHND=170) call made from service routine.
	Return
JSR MTHD1	Call made from complete (CMPLTE)

MTINT (Mag Tape Interrupts)

To initialize the drivers to use interrupts, a call is made to MTINT. This subroutine no-op's the call (JSR MTHD1) by the complete (CMLTE) subroutine to the mag tape handlers. The mag tape handler is entered by the interrupt service routine instead. A sample of an interrupt service routine is given in Appendix A.

Calling Sequence:

JSR @MTINT (MTINT=167)

INTEN Return, interrupts can be enabled at this time.

INTERRUPT SERVICE ROUTINE

APPENDIX A

```

                                MTHND=170
00001 000001  .LOC          1
00001 002110  INTSER
00001 002000  .LOC          2110
02110 040423  INTSER:      STA 0,RO          ;SAVE ACO-AC3, CARRY
02111 044423          STA 1, R1
02112 050423          STA 2, R2
02113 054423          STA 3, R3
02114 101200          MOVR 0,0
02115 040415          STA 0, CR
02116 061477          INTA 0          ;ACKNOWLEDGE
02117 024420          LAD 1,C22          ;MTA = DEVICE 22
02120 106405          SUB 0,1,SNR          ;SKIP IF NOT MTA
02121 006170          JSR @MTHND          ;MAG TAPE HANDLER
02122 020410          LDA 0,CR
02123 101100          MOVL 0,0          ;RESTORE CARRY, ACO-3
02124 034412          LDA 3,R3
02125 030410          LDA 2,R2
02126 024406          LDA 1,R1
02127 020404          LDA 0,RO
02130 060177          INTEN          ;ENABLE
02131 002000          JMP @0
02132 000000  CR:          0
02133 000000  RO:          0
02134 000000  R1:          0
02135 000000  R2:          0
02136 000000  R3:          0
02137 000022  C22:        22
                                .END
```

Western Peripherals 120026

```

; NOVA MAGNETIC TAPE DRIVER CALLING SEQUENCE
;
; JSR MTINT ; INITIALIZE DRIVER FOR INTERRUPTS
; INTEN ; ENABLE INTERRUPTS
;
; JSR UNIT ; UNIT SELECT
; -XØY ; X=PARITY(Ø=ODD 1=EVEN) Y=UNIT#
; Z ; Z=1ØØ FOR 7 TRACK, Z=Ø FOR 9TK
;
; JSR READ ; READ MAGTAPE RECORD
; ... ; ADDRESS
; ... ; WORD COUNT (NEG)
; ... ; ERROR RETURN (OFF LINE)
; ... ; RETURN
;
; JSR WRITE ; WRITE MAGTAPE RECORD
; ... ; ADDRESS
; ... ; WORD COUNT (NEG)
; ... ; ERROR RETURN(WRITE LOCK OR OFF LINE)
; ... ; RETURN
;
; JSR WEOF ; WRITE END OF FILE
; ... ; ERROR (WRITE LOCK OR UNIT NOT READY)
; ... ; RETURN
;
; JSR SPACE ; SPACE RECORDS
; ... ; RECORD COUNT (NEG FOR REVERSE)
; ... ; (LARGE COUNT WILL SPACE A FILE)
; ... ; ERROR RETURN (ILLEGAL)
; ... ; RETURN
;
; JSR REWIND ; REWIND TAPE UNIT
;
; JSR CMLTE ; COMPLETE PREVIOUS COMMAND
; ... ; ERROR RETURN
; ... ; EOF RETURN
; ... ; RETURN

```

NOTE: AØØ = STATUS ON ALL ERROR RETURNS

```

ØØØ16Ø .LOC 16Ø
ØØ16Ø ØØ12ØØ UNIT
ØØ161 ØØ12Ø5 READ
ØØ162 ØØ1214 WRITE
ØØ163 ØØ1241 WEOF
ØØ164 ØØ1277 SPACE
ØØ165 ØØ1265 REWIND
ØØ166 ØØ1461 CMLTE
ØØ167 ØØ15ØØ MTINT
ØØ17Ø ØØ1347 MTHND

```

```

;*****
; UNIT *
;*****

```

```

ØØ12ØØ .LOC 12ØØ
Ø12ØØ Ø214ØØ UNIT: LDA Ø,Ø,3 ; GET PARITY AND UNIT #
Ø12Ø1 Ø4Ø536 STA Ø,PRUN ; SAVE IT
Ø12Ø2 Ø214Ø1 LDA Ø,1,3 ; GET 7 OR 9 TRACK CODE
Ø12Ø3 Ø4Ø461 STA Ø,TRAK ; SAVE IT, 1ØØ=7, Ø=9TK
Ø12Ø4 ØØ14Ø2 JMP Ø,3 ; RETURN

```

```

;*****
;   READ   *
;*****
;
01205 024532 READ:   LDA 1,PRUN           ;GET PARITY AND UNIT #
;READ=0
01206 065222           DOAC 1,MTA       ;SEND PRITY, READ, UNIT #, CLEAR
01207 060422           DIA 0,MTA        ;READ STATUS
01210 030533           LDA 2,C53442
01211 113404           AND 0,2,SR      ;SKIP IF UNIT READY
01212 001402           JMP 2,3         ;ERROR RETURN
01213 000413           JMP EXC

;*****
;   WRITE  *
;*****
;
01214 024523 WRITE:  LDA 1,PRUN           ;GET PARITY AND UNIT#
01215 020513           LDA 0,C50       ;WRITE COMMAND
01216 107000           ADD 0,1         ;ADD WRITE COMMAND
01217 065222           DOAC 1,MTA       ;SEND PARITY, WRITE, UNIT#, CLER
01220 060422           DIA 0,MTA        ;READ STATUS
01221 111203           MOVR 0,2,SNC     ;SKIP IF UNIT READY
01222 001402           JMP 2,3         ;ERROR RETURN
01223 151200           MOVR 2,2        ;SHIFT WRITE LOCK TO CARRY
01224 151202           MOVR 2,2,SEC    ;SKIP IF WRITE RING IN
01225 001402           JMP 2,3         ;ERROR RETURN
01226 044512 EXC:   STA 1,CMD          ;SAVE COMMAND
01227 025400           LDA 1,0,3       ;GET ADDRESS (ARG1)
01230 044511           STA 1,ADR        ;SAVE ADDRESS
01231 066022           DOB 1,MTA        ;SEND ADDRESS
01232 025401           LDA 1,1,3       ;GET WORD COUNT (ARG 2)
01233 044507           STA 1,WDC        ;SAVE WORD COUNT
01234 020477           LDA 0,M10       ;AC0=-8
01235 040477           STA 0,RTY        ;SET RETRY COUNTER
01236 010477           ISE BSY         ;SET BUSY FLAG
01237 067122           DOCS 1,MTA       ;SET WORD COUNTER, START
01240 001403           JMP 3,3         ;RETURN

;*****
;   WEOF   *
;*****
;
01241 024476 WEOF:   LDA 1,PRUN           ;GET PARITY, UNIT #
01242 030421           LDA 2,C7        ;3 BIT MASK
01243 147400           AND 2,1         ;SAVE UNIT #
01244 030420           LDA 2,TRAK      ;GET PARITY FOR 7 OR 9 TRACK UNIT
01245 147000           ADD 2,1         ;ADD PARITY TO UNIT #
01246 030463           LDA 2,C60       ;GET WEOF COMMAND
01247 147000           ADD 2,1         ;ADD WEOF COMMAND
01250 065222           DOAC 1,MTA       ;SEND PARITY, WEOF, UNIT, CLEAR
01251 060422           DIA 0,MTA        ;READ STATUS
01252 111203           MOVR 0,2,SNC     ;SKIP IF UNIT READY
01253 001400           JMP 0,3         ;ERFOR RETURN
01254 151200           MOVR 2,2        ;TEST WRITE LOCK
01255 151202           MOVR 2,2,SEC    ;SKIP IF WRITE RING IN
01256 001400           JMP 0,3         ;ERROR RETURN
01257 010456           ISE BSY         ;SET BUSY FLAG
01260 065122           DOAS 1,MTA       ;SEND START
01261 044457           STA 1,CMD        ;SAVE COMMAND
01262 001401           JMP 1,3         ;RETURN
01263 000007 C7:     7
01264 000000 TRAK:   0

```



```

;*****
;   REWIND   *
;*****
;
01265 024452 REWIND: LDA 1,PRUN      ;GET PARITY, UNIT #
01266 020437          LDA 0,C10    ;GET REWIND COMMAND
01267 107000          ADD 0,1      ;ADD COMMAND TO UNIT #
01270 065122          DOAS 1,MTA   ;SEND REWIND, START
01271 024405          LDA 1,Y02
01272 060422          DIA 0,MTA
01273 107404          AND 0,1,SZR
01274 000775          JMP .-3
01275 001400          JMP 0,3      ;RETURN
01276 020000 Y02: 20000
;*****
;   SPACE   *
;*****
;
01277 031400 SPACE: LDA 2,0,3      ;GET RECORD COUNT(ARG1)
01300 024437          LDA 1,PRUN   ;GET PARITY, UNIT #
01301 151113          MOVL# 2,2,SNC ;SKIP IF REVERSE
01302 000412          JMP SP1      ;DO FORWARD
01303 020424          LDA 0,C40    ;GET REVERSE COMMAND
01304 107000          ADD 0,1      ;ADD COMMAND, UNIT
01305 065222          DOAC 1,MTA   ;SEND PARITY, REVERSE, UNIT #, B
01306 060422          DIA 0,MTA   ;READ STATUS
01307 024434          LDA 1,C53442
01310 107404          AND 0,1,SZR  ;SKIP IF UNIT READY
01311 001401          JMP 1,3      ;ERROR RETURN
01312 044426          STA 1,CMD    ;SAVE COMMAND
01313 000406          JMP SP2     ;GO SEND W.C, START
;
;   FORWARD SPACE
;
01314 150400 SP1:  NEG 2,2        ;MADE RECORD COUNT (NEG)
01315 020411          LDA 0,C30    ;GET FORWARD COMMAND
01316 107000          ADD 0,1      ;ADD COMMAND, UNIT #
01317 044421          STA 1,CMD    ;SAVE COMMAND
01320 065222          DOAC 1,MTA   ;SEND COMMAND, UNIT, CLEAR
01321 010414 SP2:  ISZ BSY        ;SET BUSY
01322 050420          STA 2,WDC
01323 073122          DOCS 2,MTA   ;SEND RECORD COUNT, START
01324 001402          JMP 2,3      ;RETURN
;
;   MAG TAP COMMANDS
01325 000010 C10:  10 ;REWIND
01326 000030 C30:  30 ;SPACE FORWARD
01327 000040 C40:  40 ;SPACE REVERSE
01330 000050 C50:  50 ;WRITE
01331 000060 C60:  60 ;EOF
01332 000070 C70:  70 ;ERASE
01333 177770 H10: -10
01334 000000 RTY:  0
01335 000000 BSY:  0
01336 000000 EFG:  0
01337 000000 PRUN: 0
01340 000000 CMD:  0
01341 000000 ADR:  0
01342 000000 WDC:  0
01343 053442 C53442: 53442

```

```

01344 051202 C51202: 51202
01345 053242 C53242: 53242
01346 000400 C400: 400
;*****
; MAG TAP HANDLER *
;*****
;
; ENTRY POINT FROM INTERRUPT SERVICE ROUTINE
01347 054504 MTHND: STA 3,MTRTN ;SAVE RETURN AND LOOP (SAME)
01350 054504 STA 3,MTRT1
01351 000406 JMP MT1+2
;
; ENTRY POINT FROM COMPLETE IF INTERRUPTS DISABLED
;
01352 054501 MTHD1: STA 3,MTRTN ;SAVE RETURN ADDRESS
01353 034502 LDA 3,MT7 ;GET LOOP ADDRESS
01354 054500 STA 3,MTRT1 ;SET LOOP ADDRESS IF ERROR
;
; LOOP BACK HERE FOR RETRY (NO-INT)
01355 063622 MT1: SKPDN MTA ;SKIP IF MTA DONE
01356 000777 JMP .-1 ;WAIT FOR DONE
01357 024757 LDA 1,EFG ;GET ERROR FLAG
01360 125112 MOVL# 1,1,SEC ;SKIP IF NO ERROR
01361 000443 JMP MT5 ;GO TRY RECOVERY
01362 060622 DIAC 0,MTA ;READ STATUS. CLEAR
01363 101112 MOVL# 0,0,SEC ;SKIP IF NO ERROR
01364 000406 JMP MT3 ;GO TEST ERROR TYPE
01365 102400 SUB 0,0 ;CLEAR ERROR FLAG
01366 040750 MT2: STA 0,EFG ;STORE ERROR FLAG
01367 102400 SUB 0,0 ;CLEAR BUSY FLAG
01370 040745 STA 0,BSY
01371 002462 JMP 0MTRTN
;
; TEST IF ERROR IS EOF ONLY
01372 024753 MT3: LDA 1,C53242 ;MASK BITS (1,3,5,6,8,10,14)
01373 107405 AND 0,1,SNR ;SKIP IF OTHER THAN EOF
01374 000772 JMP MT2 ;GO SET ERROR FLAG, ZERO BUSY
01375 040741 STA 0,EFG ;SAVE ERROR FLAG
01376 030742 LDA 2,CMD ;GET LAST COMMAND
01377 024733 LDA 1,C70 ;MASK COMMAND FIELD
01400 133405 AND 1,2,SNR ;SKIP IF NOT READ
01401 000404 JMP MT4 ;YES READ
01402 024726 LDA 1,C50 ;CHECK WRITE COMMAND
01403 132414 SUB# 1,2,SER ;SKIP IF WRITE COMMAND
01404 000763 JMP MT2+1 ;NOT READ OR WRITE
;
; TEST FOR PARITY OR BAD TAPE ERROR
01405 050451 MT4: STA 2,CM2 ;SAVE LAST COMMAND
01406 010726 ISE RTY ;RETRY = ZERO
01407 000402 JMP .+2 ;NO
01410 000757 JMP MT2+1 ;UNABLE TO RECOVER (EXIT)
01411 024733 LDA 1,C51202 ;MASK BITS (1,3,6,8,14)
01412 107404 AND 0,1,SER ;SKIP IF PARITY OR BAD TAPE
01413 000754 JMP MT2+1 ;NO RETRIES
01414 024723 LDA 1,PRUN ;GET UNIT #
01415 030712 LDA 2,C40 ;SPACE REVERSE COMMAND
01416 050441 STA 2,CM3 ;SAVE SPACE REVERSE
01417 147000 ADD 2,1 ;ADD COMMAND, UNIT #
01420 065222 DOAC 1,MTA ;SEND COMMAND, UNIT #
01421 024437 LDA 1,M1 ;AC1=-1

```

```

01422 067122      DOCS 1,MTA      ; SEND WORD COUNT, START
01423 002431      JMP 0MTRT1      ; RETURN (LOOP IF NO INT)
;
; TRY RECOVERY FROM BAD OPERATION
01424 024433 MT5:   LDA 1,CM3      ; LAST COMMAND
01425 030702      LDA 2,C40      ; SPACE REVERSE
01426 132404      SUB 1,2,SZR      ; SKIP IF LAST OP SP
01427 000413      JMP MT6        ; GO TO RETRY
01430 024426      LDA 1,CM2      ; GET ORIGINAL COMMAND
01431 030677      LDA 2,C50      ; WRITE COMMAND
01432 132404      SUB 1,2,SZR      ; SKIP IF ORIGINAL COM WRITE
01433 000407      JMP MT6        ; GO TO RETRY
01434 024703      LDA 1,FRUN      ; GET UNIT #
01435 030675      LDA 2,C70      ; ERASE COMMAND
01436 050421      STA 2,CM3      ; SAVE IT
01437 147000      ADD 2,1      ; ADD COMMAND, UNIT
01440 065122      DOAS 1,MTA      ; ERASE TAPE
01441 002413      JMP 0MTRT1      ; RETURN
;
; RETRY ORIGINAL COMMAND
;
01442 024676 MT6:   LDA 1,CMD      ; GET COMMAND
01443 065222      DOAC 1,MTA      ; SEND COMMAND, UNIT
01444 024675      LDA 1,ADR      ;
01445 102400      SUB 0,0      ; CLEAR ACP
01446 040670      STA 0,EFG      ; ZERO ERROR FLAG
01447 066222      DOB 1,MTA      ; SEND ADDRESS
01450 024672      LDA 1,WDC      ;
01451 067122      DOCS 1,MTA      ; SEND WORD COUNTER, START
01452 002402      JMP 0MTRT1      ; RETURN
01453 000000 MTRTN: 0
01454 000000 MTRT1: 0
01455 001355 MT7:   MT1
01456 000000 CM2:   0
01457 000000 CM3:   0
01460 177777 M1:   -1
;*****
; COMPLETE *
;*****
;
01461 054416 CMPLTE: STA 3,CMPR      ; SAVE LINK
01462 004670 CM1:   JSR MTRHD1      ; SENSE MTA DONE (NOP IF INTERRUPT
01463 034414      LDA 3,CMPR      ; GET LINK ADDRESS
01464 024651      LDA 1,PSY      ; GET BUSY FLAG
01465 125004      MOV 1,1,SZR      ; SKIP IF DONE
01466 000776      JMP .-2      ; WAIT FOR DONE
01467 020647      LDA 0,EFG      ; GET ERROR FLAG
01470 044646      STA 1,EFG      ; ZERO ERROR FLAG
01471 101113      MOVL# 0,0,SNC      ; SKIP IF ERROR
01472 001402      JMP 2,3      ; RETURN
01473 024653      LDA 1,C400      ; BIT 7 MASK
01474 107405      AND 0,1,SNR      ; SKIP IF EOF SET
01475 001400      JMP 0,3      ; ERROR RETURN
01476 001401      JMP 1,3      ; EOF RETURN
01477 000000 CMPP:  0
;
; INITIALIZE IF MAG TAP INTERRUPTS
01500 020403 MTINT: LDA 0,NOP      ; GET NOP (JMP .+1)
01501 040761      STA 0,CM1      ; NOP JSR TO MT HANDLERS
01502 001400      JMP 0,3      ; RETURN

```

 ADR 001341
 BSY 001335
 C10 001325
 C30 001326
 C40 001327
 C400 001346
 C50 001330
 C5120 001344
 C5324 001345
 C5344 001343
 C60 001331
 C7 001263
 C70 001332
 CM1 001462
 CM2 001456
 CM3 001457
 CMD 001340
 CMFLT 001461
 CMFR 001477
 EFG 001336
 EXC 001226
 M1 001460
 M10 001333
 MT1 001355
 MT2 001366
 MT3 001372
 MT4 001405
 MT5 001424
 MT6 001442
 MT7 001455
 MTHD1 001352
 MTHND 001347
 MTINT 001500
 MTRT1 001454
 MTRTN 001453
 NOP 001503
 PRUN 001337
 READ 001205
 REWIN 001265
 RTY 001334
 SP1 001314
 SP2 001321
 SPACE 001277
 TRAK 001264
 UNIT 001200
 WDC 001342
 WEOF 001241
 WRITE 001214
 Y02 001276

01503 000401 NOP: JMP .+1
 .END

SUPPLEMENT TO TC-120

SOFTWARE MANUAL

4. A 33 word buffer register provides buffering of data between tape and data channel. Therefore, the channel has almost 34 character-times in which to respond to requests by the Controller.

4.1 TO RUN THE TAPE

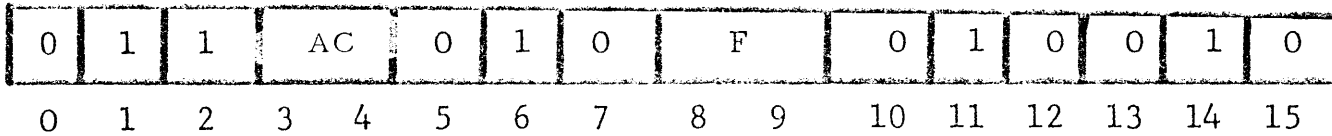
To run the tape, the program must select a tape transport and a command. Most of the commands also require specification of parity, an initial address to the 15-bit address counter for data channel access, and the negative of a word count in two's-complement form.

4.2 I/O TRANSFER INSTRUCTIONS

Five IOT instructions are used. Busy and Done are sensed or controlled by bits 8 and 9 in all I/O instructions with Device Code 22, mnemonic MTA. Interrupt Disable is controlled by interrupt priority mask bit 10. A second tape system connected to the bus would have Device Code 62.

4.2.1 DOA-, MTA (DATA OUT A, MAGNETIC TAPE)

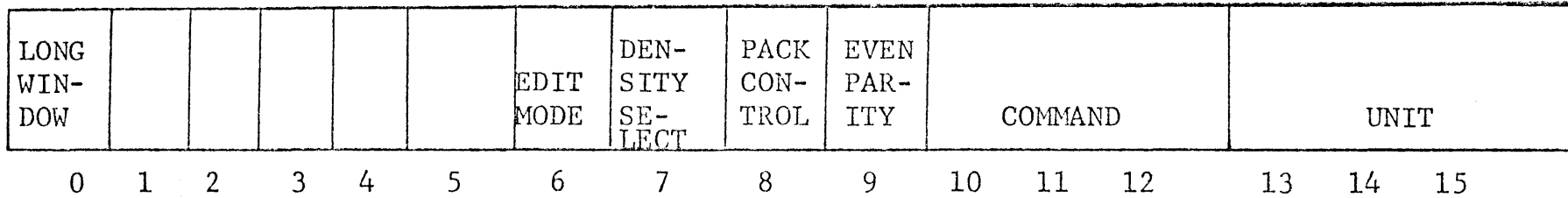
0	1	1	AC	0	1	0	F	0	1	0	0	1	0		
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15



- Bit 0 -- "0" specifies operation is neither arithmetic nor logical.
- Bits 1, 2 -- Indicate that this is an "in-out" instruction.
- Bits 3, 4 -- Address one of four computer accumulator/index registers.
- Bits 5, 6, 7 -- Indicate direction of transfer, select one of three registers (in this case, the tape command register).
- Bits 8, 9 -- Specify action to be performed, such as, clear "Busy" and "Done."
- Bits 10-15 -- Device code.

*Transfer takes place between the accumulator addressed by Bits 3 and 4 and the device that is selected by Bits 10-15.

FIGURE 4-1. INSTRUCTION FORMAT, "IN-OUT" OPERATION*



- Bit 0 -- Selects Long Window
- Bit 7 -- "0" Selects NRZ "1" Selects PE
- Bit 8 -- "0" Selects 8, 8 or 6, 6 Packing "1" Selects 4, 6, 6, Packing
- Bit 9 -- "0" Selects Odd, "1" Selects even parity
- Bits 10-12 -- Select the command as follows:
- | | |
|-------------------|-----------------------|
| 0 = Read | 4 = Space Reverse |
| 1 = Rewind | 5 = Write |
| 2 = | 6 = Write End-of-File |
| 3 = Space Forward | 7 = Erase |
- Bits 13-14-15 -- Selects one of the eight tape units.

Load the contents of AC Bits 9-15 into the tape command register as shown and perform the function specified by F.

4.2.2 DOB-, MTA (DATA OUT B, MAGNETIC TAPE)

0	1	1	AC	0	1	0	F	0	1	0	0	1	0		
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Load the contents of AC Bits 1-15 into the address counter (AC Bit 0 should be 0), and perform the function specified by F.

NOTE

If this instruction is given with a "1" in AC bit 0, and if the Controller then executes a Read Command in which the word counter does not overflow, the Controller reads the CRC in nine-track and the LPCC in seven-track at the end of the record and sends it to the next memory location specified by the address counter. This is primarily for maintenance, for the program to check whether the CRC or LPCC is being generated properly.

4.2.3 DOC-, MTA (DATA OUT C, MAGNETIC TAPE)

0	1	1	AC	1	1	0	F	0	1	0	0	1	0		
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Load the contents of AC Bits 4-15 into the word counter and perform the function specified by F.

4.2.4 DIA-, MTA (DATA IN A, MAGNETIC TAPE)

0	1	1	AC	0	0	1	F	0	1	0	0	1	0		
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Read the status of the tape system into AC as shown, and perform the function specified by F.

STATUS WORD FORMAT

ERROR FLAG	DATA LATE	RE-WINDING	IL-LEGAL	SEE TABLE 4-1	PAR-ITY ERROR	END OF TAPE	END OF FILE	LOAD POINT	SEE TABLE 4-1	BAD TAPE	IDENT. STATUS	CORRECTED PARITY ERROR	WRITE LOCK	ODD # CHAR/RECORD	TAPE UNIT READY
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

TABLE 4-1

STATUS TABLE

	Bit 4 -- Density	Bit 9 -- 9-Track
7-Track Low Density	0	0
7-Track High Density	1	0
9-Track P.E.	1	1
9-Track NRZ	1	1

START (F01) clears Error, Data Late, Parity Error, End-of-File, and Bad Tape. CLEAR (F10) does the same, plus Illegal. The remaining flags are supplied by the addressed transport (which is automatically Unit 0 after CLEAR is given).

TABLE 4-2
STATUS DESCRIPTION

A One
In Bit

INDICATES THAT:

- 0 Bit 1, 3, 5, 6, 7, 8, 10, or 14 is "One."
- 1 Servicing of data channel is not fast enough.
- 2 The addressed transport is now rewinding.
- 3 The program gives START in any of the following:
 - a. The command is Write, Erase, or Write-End-of-File, and Write Lock is "One."
 - b. The command is Space Reverse and Load Point (Bit 8) is "One."
 - c. Busy is zero, but Unit Ready (Bit 15) is also zero.

The setting of Illegal prevents the Tape Controller from going into operation and sets Done, requesting an interrupt if interrupt Disable is clear. The program must

TABLE 4-2. STATUS DESCRIPTION (Continued)

A One In Bit	give Clear before proceeding (Start does not clear Illegal).
4	The addressed transport is set to High Density (0 indicates Low Density).
5	In Read or Write, the Controller meets a data character the lateral parity of which differs from that specified with the command, or has discovered a track with odd parity in a CRC or LRCC does not set this bit, but specifying the wrong parity when reading a file mark does.
6	The addressed tape is beyond the EOT marker. (Reverse motion clears this bit.)
7	The Controller has written a file mark or has encountered one in reading or spacing. If there is an error in a file mark, it is not recognized as such; i.e., the Controller interprets it as a very short data record.
8	The addressed tape is at Load Point.
9	The addressed transport handles 9-track tape ("0" indicates 7-track).

TABLE 4-2. STATUS DESCRIPTION (Continued)

A One
In Bit

- 10 The Controller has encountered either data in a record gap or a false end-of-record (two or more contiguous blank characters). Spacing reverse over an unrecognized file mark also sets Bad Tape.
- 11 Identification status for phase-encoded tapes if tape is written in phase-encoded format. This status is true when first record is read off of Load Point.
- 12 Corrected parity error indicates single dead track error. Sets parity error (bits) if Write operation. Does not set parity error if Read operation.
- 13 The Write-Enable ring is not on the supply reel of the addressed transport.
- 14 An odd number of characters were detected while Reading or Writing.
- 15 The addressed transport is ready for operation by the program. This can be an early ready for on-the-fly operation if this option is selected on the computer adapter board (see Paragraph 2.5).

4.2.5 DIB-, MIA (DATA IN B, MAGNETIC TAPE)

0	1	1	AC		0	1	1	F		0	1	0	0	1	0
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Read the present contents of the address counter into AC bits 1-15, and perform the function specified by F. Clear AC Bit 0.

4.3 TAPE COMMANDS

To perform any operation, the program must select the tape unit while giving a command. All commands are initiated by giving Start (F01). The two Rewind commands do not actually place the Controller in operation, but for all other commands, Start clears Done and sets Busy, and at the termination of the command the Controller clears Busy and sets Done, requesting an interrupt if Interrupt Disable is clear.

The flow charts that follow this section show the actual procedures for programming the tape commands properly. The timing in all cases is dependent upon the transport speed, tape handling characteristics, and density.

4.3.1 WRITE (SEE FIGURE 4-2)

The program must specify parity, a (negative) word count, and an initial address. If Write Lock is 1, Start sets Illegal and Done, and the Controller does not go into operation. Otherwise, the Controller makes an immediate data request for the first word, and it writes the words it receives via the data channel from the locations specified by the address counter until either the word counter overflows or Data Late sets, at which time the Controller terminates the record and sets Done.

4.3.2 WRITE END OF FILE (SEE FIGURE 4-3)

The program must specify even parity for a 7-track tape, odd parity for a 9-track tape, or the Controller will not write a file mark properly. If Write Lock is 1, Start sets Illegal and Done, and the Controller does not go into operation. Otherwise, the Controller erases 2- $\frac{1}{2}$ inches of tape (i.e., it extends the present record gap to three inches, writes a file mark, and then sets Done).

4.3.3 ERASE

If Write Lock is 1, Start sets Illegal and Done, and the Controller does not go into operation. Otherwise, the

4.3.3 ERASE (Continued)

Controller erases 2- $\frac{1}{2}$ inches of tape and then sets Done.

This command is used primarily to skip sections of tape on which the program has found it impossible to write data correctly; i.e., with parity errors or a bad tape indication.

4.3.4 READ (SEE FIGURE 4-4)

The program must specify parity, a (negative) word count, and an initial address. The Controller reads a single record from tape, and sends the data via the data channel to the locations specified by the address counter until it encounters the EOR gap or the word counter overflows, whichever occurs first. Giving a large word count (e.g. giving zero) ensures that the entire record will be read even if its length is unknown. If the record contains an odd number of data characters, the final one is sent to memory in a separate word right justified. The setting of Data Late during the record indicates that information has been lost, but data transfers continue until overflow or the record ends. After completing the record, the Controller sets Done.

4.3.4 READ (Continued)

If the record read is a file mark, its single "data" character is sent to memory via the data channel. The length of a record of unknown size can be determined after it is read by giving a DIB to check the contents of the address counter, which will be one greater than the address to which the last word in the record was sent (provided, of course, that the word count was large enough).

4.3.5 SPACE FORWARD (SEE FIGURE 4-5)

The program should give a (negative) word count equal to the number of records to be spaced. The Controller spaces forward over the given number of records unless it encounters a file mark or the end of tape, in which case it stops at the mark or at the end of the record in which the EOT marker is encountered. To space a file, the program can simply give a zero word count.

4.3.6 SPACE REVERSE (SEE FIGURE 4-6)

The program should give a (negative) word count equal to the number of records to be spaced. If Load Point is 1, Start sets Illegal and Done, and the Controller

4.3.6 SPACE REVERSE (Continued)

does not go into operation. Otherwise, the Controller spaces reverse over the given number of records, but it stops the tape automatically upon encountering a file mark or the Load Point. To space a file, the program can simply give a zero word count.

4.3.7 REWIND (SEE FIGURE 4-7)

Start does not affect the Controller, but simply initiates the rewind in the addressed transport and the Controller is free for further use by the program. The addressed transport rewinds the tape at high speed onto the supply reel and stops at Load Point.

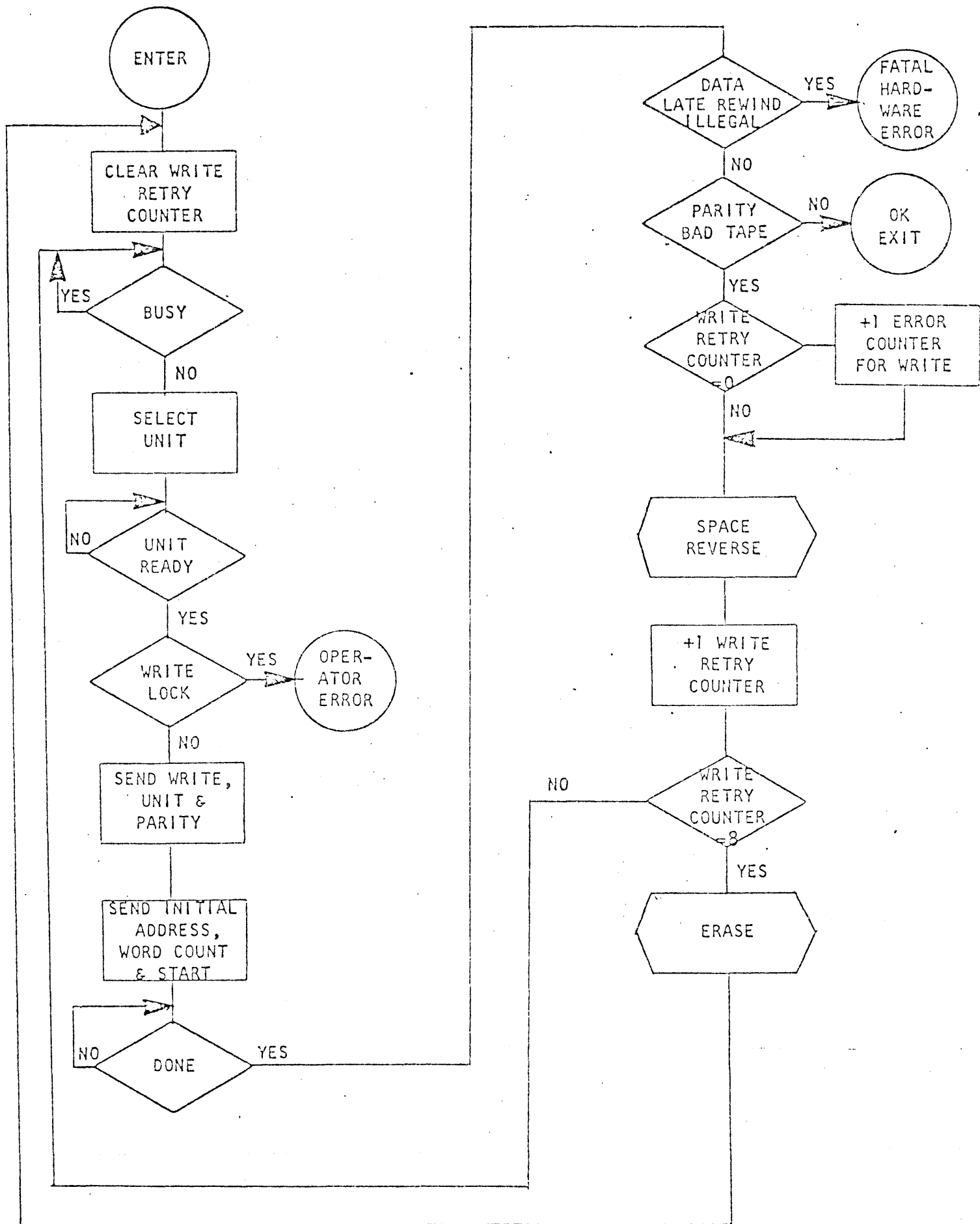


Figure 4-2. "WRITE" Flow Chart

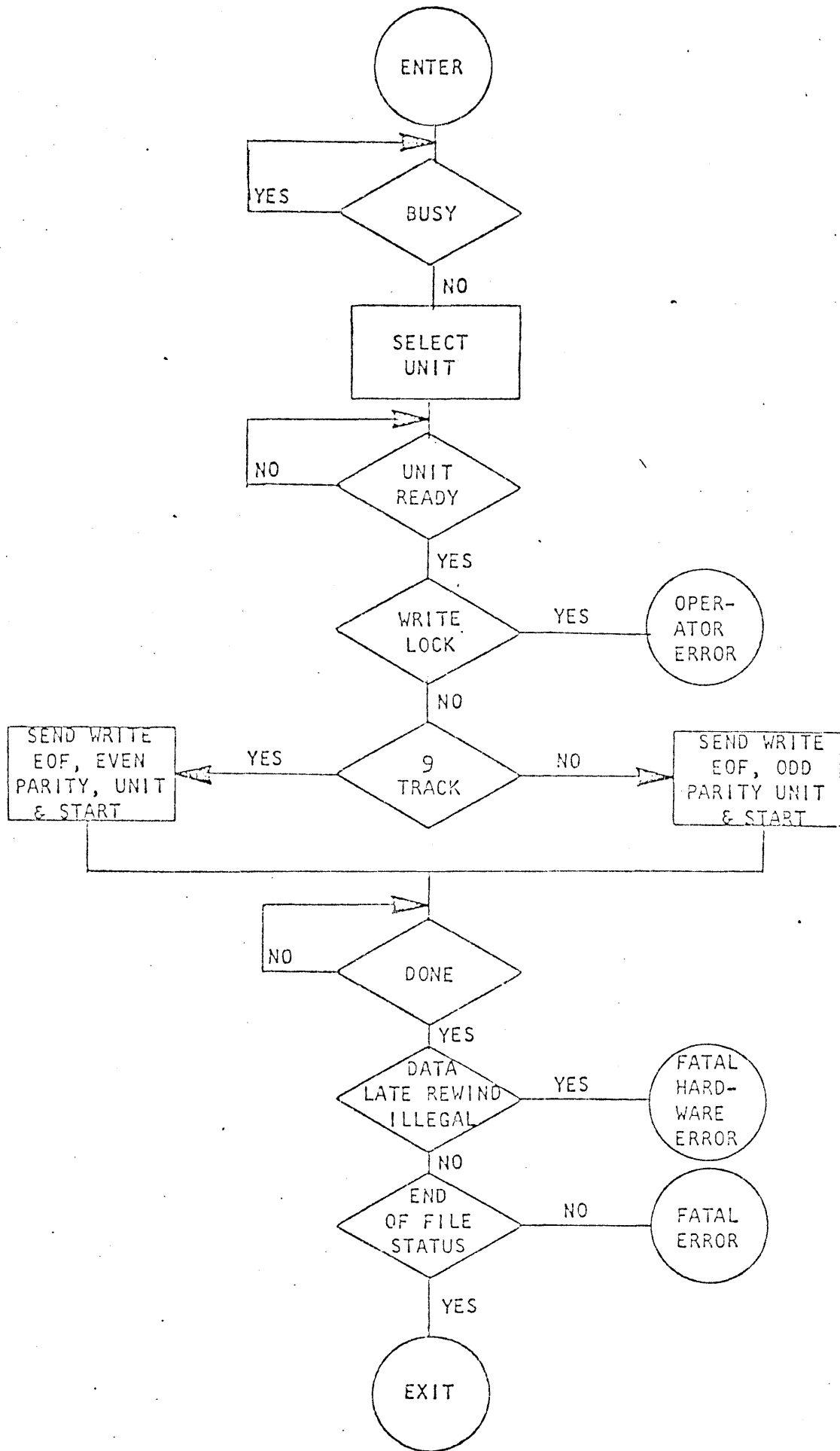


Figure 4-3. "WRITE END OF FILE" Flow Chart

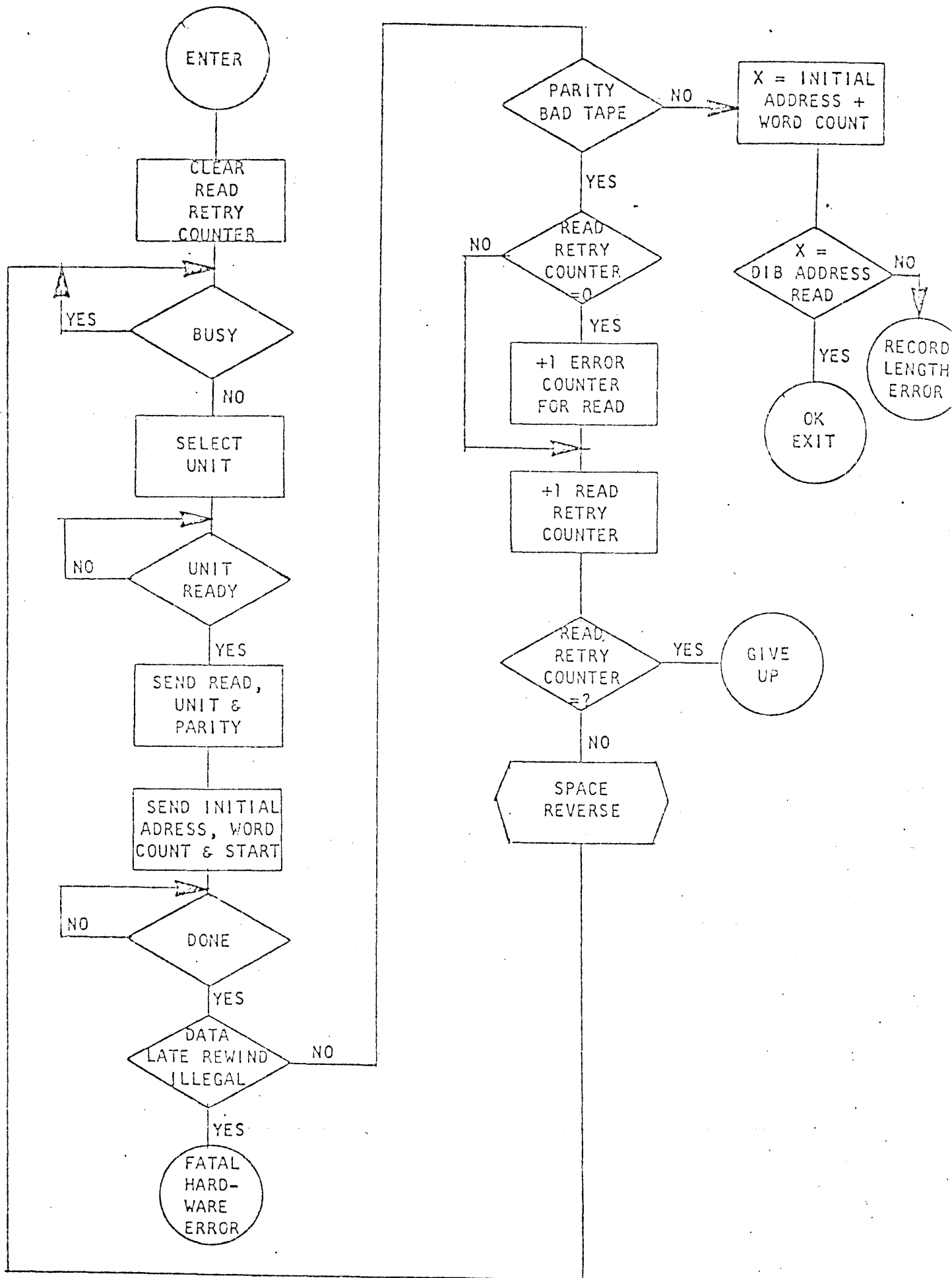


Figure 4-4. "READ" Flow Chart

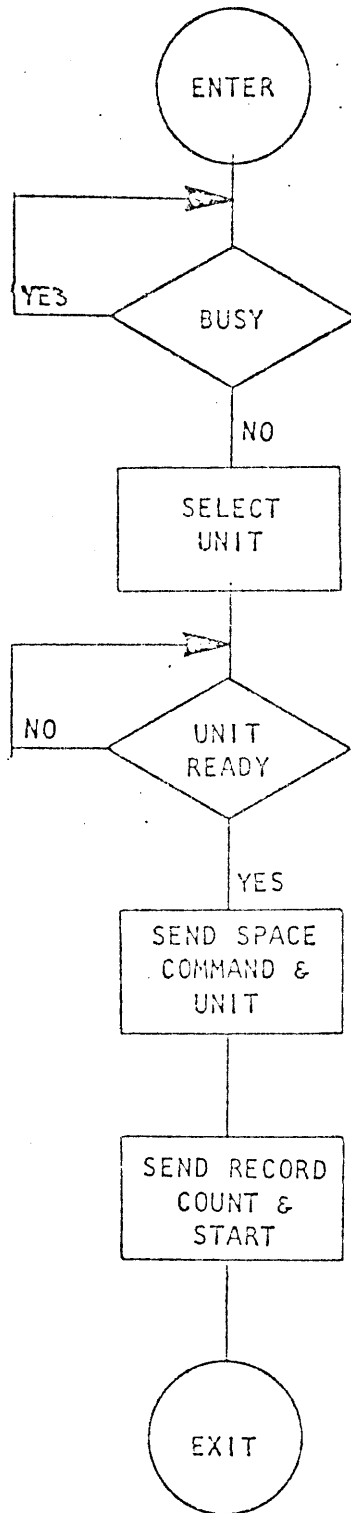


Figure 4-5. "SPACE FORWARD/REVERSE" Flow Chart

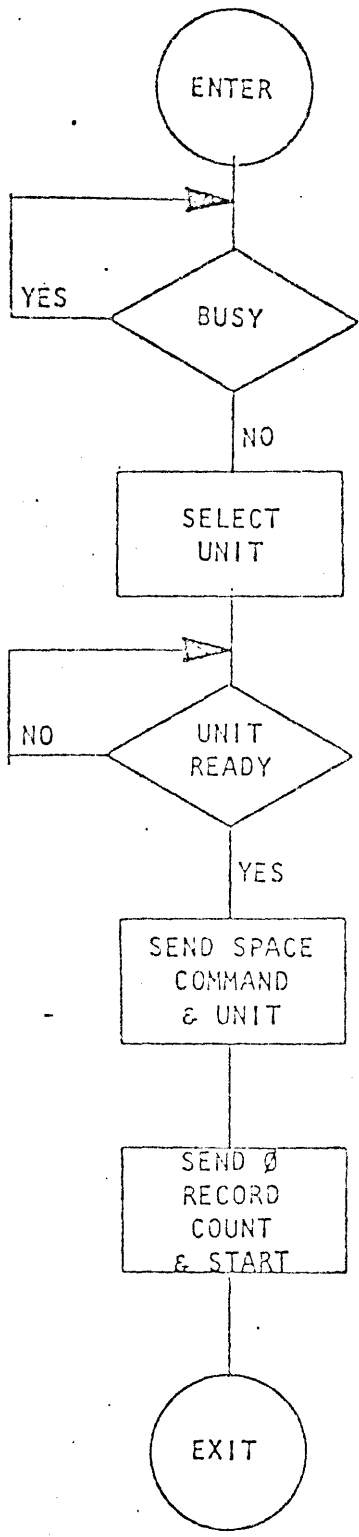


Figure 4-6. "SPACE FILE FORWARD/REVERSE" Flow Chart

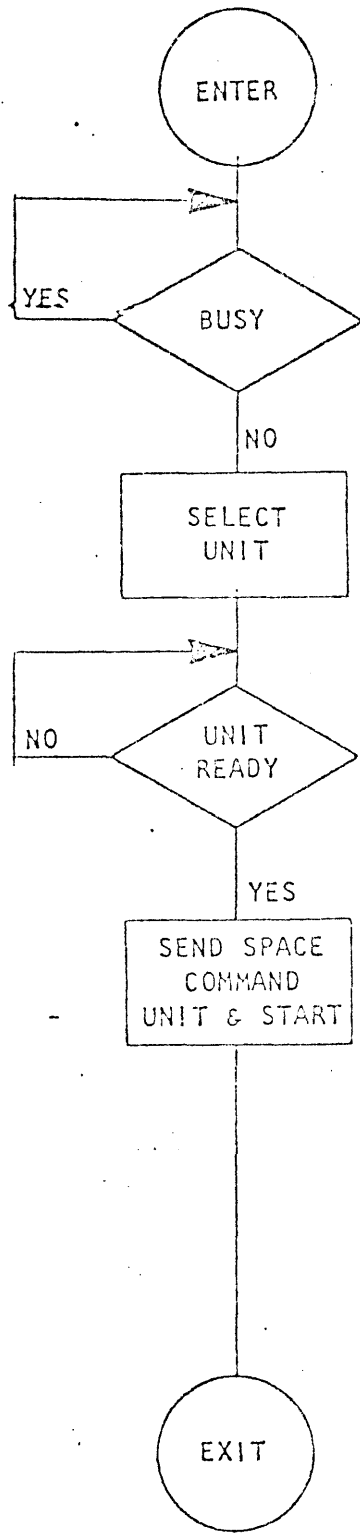
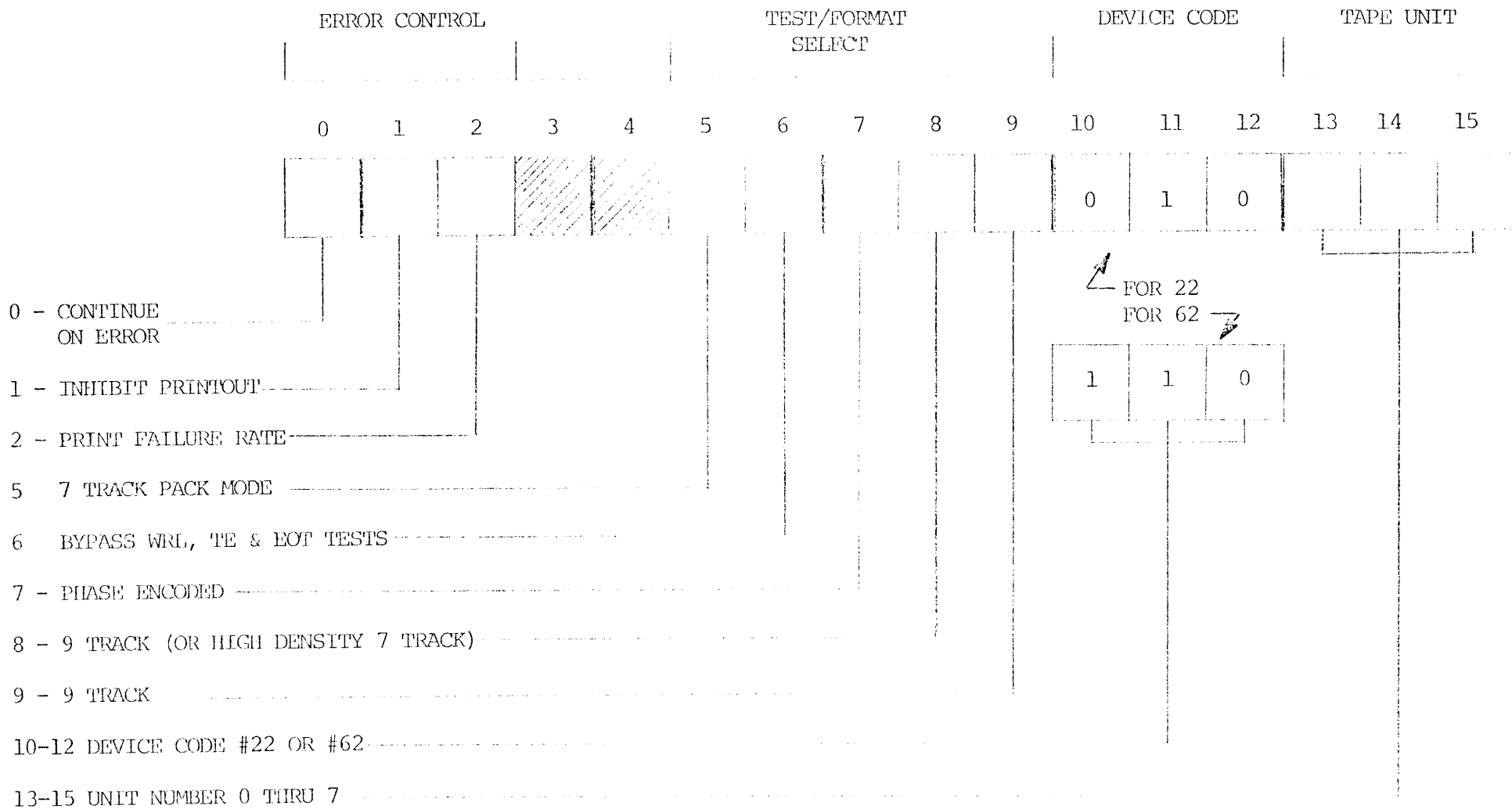


Figure 4-7. "REWIND" Flow Chart



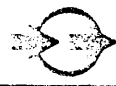
DIAGNOSTIC SWITCH REGISTER CONTROL

REVISIONS			
LTR	DESCRIPTION	DATE	APPROVED
A	PROD. REL.	8-22-78	<i>[Signature]</i>

MASTER DIAGNOSTIC TAPE
 BOT #1 NRZ REV. C
 BOT #2 PE 30-JUN-78

PDP-11 BOOTSTRAP LOADER

	LOC	INST	
1. TC130 DIAGNOSTIC	37000	12737	MOV NXT ADD INTO
2. TC130 RELIABILITY			CMD REG
3. TC120 DIAGNOSTIC	2	10000	POWER CLEAR
4. TC120 RELIABILITY	4	172522	CMD REG
5. DC220 DIAGNOSTIC	6	12737	MOV NXT ADD INTO
6. DC220 RELIABILITY			BYTE CTR
7. DC220-10 FORMATTER			
8. DC220-10 RELIABILITY	10	160000	(4K)
9. DC220-10 DIAGNOSTIC, 1ST 4K	12	172524	BYTE CNT REG
10. DC220-10 DIAGNOSTIC, 2ND 4K	14	12737	MOV NXT ADD INTO
11. DC230 DIAGNOSTIC			CMD REG
12. DC230 RELIABILITY			
13. TC130 DIAGNOSTIC 11/34	16	60003	
14. TC130 RELIABILITY 11/34	20	172522	CMD REG
	22	12700	MOV 0 INTO REG 0
	24	00000	
	26	5200	INC R0
	30	1376	BR=0
	32	5200	INC R0
	34	1316	BR=0
	36	00000	HLT

TOLERANCES UNLESS OTHERWISE SPECIFIED		 western peripherals ANAHEIM, CALIFORNIA	
FRACTIONS	DEC.		
=	=	=	
APPROVALS		DATE	
DRAWN		<i>[Signature]</i> 8-22-78	
CHECKED		SCALE	SIZE
			DRAWING NO.
			A 66000001
DO NOT SCALE DRAWING			SHEET 1 OF 2

REVISIONS			
LTR	DESCRIPTION	DATE	APPROVED
A	PROD. REL.	8-23-73	<i>[Signature]</i>

Operation Instructions:

Clear Accumulators to 0. Load first test by starting Read routine. Load succeeding tests by pressing Continue.

To load 220-10 Diagnostic Part 2, change Accumulators 1 and 2, press Continue.

Nova Read Routine - Load in top 8 locations of memory.

67022	DOC 1
72022	DCB 2
61122	DOAS 0
63622	SKPDM
777	JMP-1
74422	DIA 3
63077	HALT
771	JMP-7

REGISTERS

Normal	220-10 Part 2	ACC
0000	0	COMMAND
0000	10	WD CNT
0000	10000	STRT ADDR (DIRECT)
0000		STATUS

- 0 = Read
- 1 = Rewind
- 2 = Not Used
- 3 = Space Forward
- 4 = Space Rev.
- 5 = Write
- 6 = WEOF
- 7 = Erase



Western Peripherals
ANAHEIM, CALIFORNIA

DIRECTORY AND LOADING PROCEDURE
FOR WP MASTER DIAGNOSTIC M.T.

TOLERANCES UNLESS
OTHERWISE SPECIFIED
FRACTIONS DEC. ANGLES

± ± ±

APPROVALS DATE

DRAWN *[Signature]* 8-22-73

CHECKED

SCALE

SIZE **A**


DRAWING NO.
66000001

DO NOT SCALE DRAWING

SHEET 2 OF 2

REVISIONS			
LTR	DESCRIPTION	DATE	APPROVED
A	RELEASE	9/3/81	<i>K. P. ...</i>

APR 8 1981

TOLERANCES UNLESS OTHERWISE SPECIFIED		 western peripherals ™ TUSTIN, CALIFORNIA	
FRACTIONS	DEC.		
=	=	=	
APPROVALS		DATE	
DRAWN <i>L. Crawford</i>		4-02-81	
CHECKED <i>[Signature]</i>		4-02-81	
		SCALE	SIZE A
		DRAWING NO. 66000068	
DO NOT SCALE DRAWING			SHEET 1 OF 3

REVISIONS

LTR	DESCRIPTION	DATE	APPROVED
-----	-------------	------	----------

1. MASTER DIAGNOSTIC TAPE CARTRIDGE

<u>Record</u>	<u>Program</u>
1	TC160 DIAGNOSTIC
2	TC160 RELIABILITY
3	TC170 DIAGNOSTIC
4	TC170 RELIABILITY
5	TC180 DIAGNOSTIC
6	TC180 RELIABILITY

2. DEC BOOTSTRAP LOADER

<u>Loc</u>	<u>Inst</u>	
37000	12737	MOV
2	10000	POWER CLEAR
4	172522	CMD REG
6	12737	MOV
10	160000	(4K)
12	172524	BYTE CNT REG
14	12737	MOV
16	3	READ
20	172522	CMD REG
22	32737	BIT TEST
24	1	FOR TUR BIT
26	172520	IN STATUS REG
30	1774	BRANCH IF NOT READY
32	0	HALT

3. DATA GENERAL BCOTSTRAP LOADING PROCEDURE

Clear Accumulators to 0. Load first test by starting Read routine. Load succeeding tests by pressing Continue.

NOVA Read Routine - Load in top 8 locations of memory.

67022	DOC 1
72022	DOB 2
61122	DOAS 0
63622	SKPDN
777	JMP-1
74422	DIA 3
63077	HALT
771	JMP-7

APR 2 1968

SCALE	SIZE	DRAWING NO.
	A	66000068
DO NOT SCALE DRAWING		SHEET 2 OF 3

REVISIONS

LTR	DESCRIPTION	DATE	APPROVED
-----	-------------	------	----------

NOVA Read Routine (Cont'd)

<u>Accumulator</u>	<u>No.</u>	<u>Commands</u>
COMMAND	0	0 = Read
WD CNT	1	1 = Rewind
STRT ADDR (DIRECT)	2	2 = Not Used
STATUS	3	3 = Space Forward
		4 = Space Reverse
		5 = Write
		6 = WEOF
		7 = Erase

4. REMOVE THE PROGRAM TAPE FROM THE DRIVE AND INSERT A SCRATCH TAPE.
5. REFER TO THE APPROPRIATE DIAGNOSTIC MANUAL FOR REQUIRED PATCHES AND FOR OPERATING INSTRUCTIONS.

SEP 3 1961

SCALE	SIZE A	DRAWING NO. 66000068
DO NOT SCALE DRAWING		SHEET 3 OF 3

NOTES

NOVA WRITE-READ/BOOTSTRAP ROUTINE

(Use for program load or troubleshooting)

<u>LOCATION</u>	<u>CONTENTS</u>	<u>INSTRUCTIONS</u>
17770	67022	DOC WD CNTR
17771	72022	DOB ADDR REG
17772	61122	DOAS CMD REG
17773	63622	SKP DN
17774	777	JMP - 1
17775	74422	STATUS TO AC3
17776	63077	HLT
17777	771	JMP - 7

(ZERO THE ACCUMULATORS FOR PROGRAM BOOT)

AC0	50 00000	WRITE CMD READ CMD
AC1	177770	WORD COUNT (2'S COMP)
AC2	17700	DATA BUFF ADDR
AC3	-----	STATUS

17700 (Variable) DATA BUFF
TO
17710

Press CONTINUE for each record on tape

NOTES

SOFTWARE PATCHES

FOR COMPUTERS WITHOUT PANEL SWITCH REGISTER

DIAGNOSTIC PATCHES

<u>LOCATION</u>	<u>WAS</u>	<u>SHOULD BE</u>
3175	74477	34303 (LDA, 3)
303	(not used)	Parameters*

*Set parameters as shown in diagnostic operating instructions. Example: 1320 for 9 track, NRZI, Unit Ø, Device Code 22; 1720 changes mode to PE.

RELIABILITY PATCHES

Changes mode from NRZI to PE

<u>LOCATION</u>	<u>WAS</u>	<u>SHOULD BE</u>
2041	60477	20241
	(read SW)	(LDA, Ø PKFLG (=100 000))

NOTES

LOAD PROGRAM FOR NOVA BOOTSTRAP LOADER PAPER TAPE (24K CPU)

57757	126440
60	63612
61	777
62	60512
63	127100
64	127100
65	107003
66	772
67	1400
70	60112
71	4766
72	44402
73	4764

1. Put Nova Binary loader in Reader.
2. Start at location 57770
3. Put TC-120/170 Diagnostic Paper Tape in Reader.
4. Start at location 57777 with switch "Ø" on.
5. Set Console Switches to desired starting address (400).

NOTES

NOVA WRITE ROUTINE (Variable Delay/Reset)

<u>Location</u>	<u>Contents</u>	<u>Instructions</u>
17770	67022	DOC WD CNTR
17771	72022	DOB ADDR CNTR
17772	61122	DOAS CMD REG
17773	54377	ST. AC3 @ 377
17774	10377	INC 377 SKZ
17775	777	JMP - 1
17776	62677	I/O RST
17777	771	LOOP

ACC	000050	CMD
AC1	177770	WD CNT
AC2	17700	DATA BUFF ADDR
AC3	(Variable)	DELAY CONSTANT

17700	(Variable)	DATA BUFF
TO		
17710		

NOTES

NOVA WRITE ROUTINE
(With Variable Delay)

<u>Location</u>	<u>Contents</u>	<u>Instructions</u>
17770	67022	DOC WD CNTR
17771	72022	DOB ADDR CNTR
17772	61122	DOAS CMD REG
17773	63622	SKPDN
17774	777	JMP - 1
17775	54377	ST. AC3 @ 377
17776	10377	INC 377 SKZ
17777	777	JMP - 1
17710	770	LOOP

AC0	000050	CMD
AC1	177770	WD CNT
AC2	17700	DATA BUFF ADDR
AC3	(Variable)	DELAY CONSTANT

17700	(Variable)	DATA BUFF
TO		
17710		

NOTES

RELIABILITY PROGRAM

DEVICE CODE - This program normally operates with device code 22. If you must operate the program with device code 62, modify the 58 locations identified "MTA" listed on page 63.

NOTES

DIAGNOSTIC PATCHES

CHANGE DESCRIPTION

Enter the following changes when running the TC-120 Diagnostics
(Paper Tape, Diagnostic, PN 120009)
(Paper Tape, Reliability, PN 120010)

FUNCTIONAL DIAGNOSTIC

<u>Location</u>	<u>Was</u>	<u>Should Be</u>	<u>Description</u>
73	1	2	Timer, Real Time Clock
216	0	177220	TTY Speed
453	3242	420 or 406	Mapped Processor
3276	44216	401	TTY Speed

RELIABILITY PROGRAM

<u>Location</u>	<u>Was</u>	<u>Should Be</u>	<u>Description</u>
1021	101010	401	Error Trap (Nova 3)
1207	101010	401	Error Trap (Nova 3)



TITLE/DESCRIPTION Pertec T 1000 Diagnostic Compatibility			NUMBER 12005
PRODUCT LINE Tape Controller	MODEL AFFECTED TC-120/128	EFFECTIVE DATE 5-10-78	REFERENCE ECO # 349
DISPOSITION <input checked="" type="checkbox"/> IMPROVEMENT <input type="checkbox"/> RETROFIT ON FAILURE <input type="checkbox"/> RECOMMEND MANDATORY <input type="checkbox"/> INFORMATION ONLY		EFFECTIVITY Diagnostic tape 120009 (no revision) requires this change when using T 1000	

CHANGE PROCEDURE

This program must be manually altered for operation of the diagnostic with a Pertec T 1000 tape drive. After loading diagnostic program tape 120009, and before executing the program, one location must be altered as shown below:

Location	Was	Should Be
000577	006126 JSR	000430 JMP

SHOULD ADDITIONAL INFORMATION BE REQUIRED - CONTACT

WESTERN PERIPHERALS, INC.
1100 CLAUDINA PLACE
ANAHEIM, CALIF. 92805
PHONE: (714) 991-8700
ATTN: MANAGER, FIELD SERVICE

NOTES