

TR79F

TR79F TAPE DIAGNOSTIC
CZTRAB0

AH-9428B-MC

COPYRIGHT © 76-78

FICHE 1 OF 1

MAR 1978

digital

MADE IN USA

B01

EOF1CZTRAB00000411 00010000 MA0022330(1046) 14-0001074112:32 PAGE00DR1CZTRABSEQ
CZTRAB.HED 14-DEC-77 12:19 SPECIAL TEST MACROS

00010000

780223
SEQ 0001

.REM ;

IDENTIFICATION

PRODUCT CODE: AC-9426B-MC
PRODUCT NAME: CZTRAB0 TR79F TAPE DIAG
DATE RELEASED: FEB 1978
MAINTAINER: DIAGNOSTIC GROUP
MODIFIED BY: BILL SCHLITZKUS

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

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

COPYRIGHT (C) 1976, 1978 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

A2. REQUIREMENTS

- A. PDP-11 COMPUTER
- B. MAGNETIC TAPE TRANSPORT CONTROL INTERFACE
- C. 12K WORDS OF MEMORY
- D. CONSOLE TELEPRINTER
- E. PAPER TAPE READER

A3. LOADING PROCEDURE

THE DIAGNOSTIC IS IN ABSOLUTE BINARY FORMAT AND IS LOADED
USING THE ABSOLUTE LOADER.

A4. STARTING PROCEDURE

- A. LOAD ADDRESS 200
- B. DEPRESS START

D01

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 4
CZTRAB.MED 14-DEC-77 12:19 SPECIAL TEST MACROS

SEQ 0003

TEST SELECTION:

AFTER THE TYPEOUT "SELECT TEST NUMBER..." THE OPERATOR
SHOULD SELECT THE DESIRED TEST BY TYPING THE NUMBER FOLLOWED BY
A CARRIAGE RETURN.

NOTE: THE NORMAL SWITCH SETTINGS ARE ALL SWITCHES DOWN

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 5
CZTRAB.HED 14-DEC-77 12:19 SPECIAL TEST MACROS

SEQ 0004

A6. TEST GROUP ABSTRACT

PRETEST: PRETEST

THIS TEST GROUP CHECKS THE RESET FUNCTION. IT IS PERFORMED
EACH TIME A TEST GROUP IS SELECTED.

NOTE: IN THE EVENT OF ERRORS IN THE PRETEST GROUP IT IS RECOMMENDED
THAT THE RESET SWITCH ON THE TAPE TRANSPORT BE DEPRESSED
AND THE TEST GROUP BE RESTARTED AT THE RESTART ADDRESS.

NOTE: WHEN TEST 000 IS SEEN IN AN ERROR TYPEOUT,
AN ERROR DURING AUTOSIZING HAS OCCURRED.
TEST 0 IS NOT A TEST GROUP.

TEST 1

TEST ONE IS A MANUAL INTERVENTION TEST REQUIRING THE OPERATOR
TO PERFORM THE TASKS AS DIRECTED BY THE TTY. IT IS SUGGESTED THAT THE
EOT STICKER BE PLACED ABOUT 50 FEET FROM THE LOAD POINT.

TEST 2

TEST GROUP TWO IS A LOGIC TEST GROUP WHICH CHECKS THE INTERFACE LOGIC.

TEST 3

TEST GROUP THREE IS A LOGIC TEST GROUP WHICH TESTS THE TRANSPORTS RESPONSE
TO ALL TYPES OF COMMANDS.

TEST 4

TEST GROUP FOUR IS A RELIABILITY TEST GROUP WHICH CHECKS THE ABILITY OF
THE TRANSPORT AND THE INTERFACE OVER AN EXTENDED PERIOD OF TIME.

TEST 5

TEST GROUP FIVE IS THE READ PART OF A COMPATIBILITY TEST. THIS
TEST GROUP READS TAPES WRITTEN BY TEST GROUP 4.

TEST 6

TEST GROUP SIX IS A MAINTENANCE AID THIS ROUTINE ALLOWS THE OPERATOR
TO LOOP ON ANY SUBTEST BY ENTERING THE STARTING ADDRESS AT THE TTY.
THE LOOP CONTROL WILL KEEP THE SELECTED TEST GROUP RUNNING IF LOOP IS NOT
SELECTED THE TEST GROUP WILL EXECUTE ONCE AND ANOTHER STARTING ADDRESS MUST
BE ENTERED.

NOTE: IF ANY ERRORS DETECTED IN BOTH TEST GROUP 3 AND 4 HAVE THE SAME
PASS NUMBER A DEFECTIVE TAPE SHOULD BE THE FIRST CONSIDERATION.

F01

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 6
CZTRAB.HED 14-DEC-77 12:19 SPECIAL TEST MACROS

SEQ 0005

IF SWITCHES #3 AND #5 ARE DOWN THEN TESTS 2 THROUGH 4 ARE LOOPED ON WHERE
NO OPERATOR INTERVENTION IS REQUIRED.
IF SWITCHES #3 AND #5 ARE UP THEN ALL TESTS ARE PERFORMED.

GO1

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 7
CZTRAB.HED 14-DEC-77 12:19 SPECIAL TEST MACROS

SEQ 0006

A7. ERRORS

THERE ARE TWO BASIC ERROR TYPEOUTS ONE OF WHICH CONTAINS THE FOLLOWING INFORMATION. THE OTHER ONLY GIVES THE PASS, PC, AND SR. WHEN THIS SHORT ERROR PRINTOUT IS TYPED THE MESSAGE WHICH FOLLOWS WILL GIVE AMPLE INFORMATION ABOUT THE ERROR.

PASS	XXX				
SUBTEST	PC	STATUS	CORRECT	ACTUAL	LOCATION
XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX	XXXXXX

PASS= XXXXXX (THIS IS THE CURRENT PASS NUMBER)
SUBTEST=XXXXXX (THIS IS THE STARTING ADDRESS OF THE
CURRENT SUBTEST BEING EXECUTED)
PC= XXXXXX (THIS IS THE PC FROM WHICH ERROR WAS CALLED)
STATUS= XXXXXX (THIS IS THE CONTENTS OF THE PROCESSOR STATUS
REGISTER AT THE TIME OF THE ERROR)
CORRECT=XXXXXX (THIS IS WHAT SHOULD HAVE BEEN IN THE
REGISTER OR LOCATION BEING TESTED)
ACTUAL= XXXXXX (THIS IS WHAT WAS IN THE REGISTER OR
LOCATION AT THE TIME IT WAS EXAMINED)
LOCATION=XXXXXX (THIS IS THE LOCATION THAT WAS IN ERROR)
COMMENT (OPTIONAL)

;

HO1

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 8
CZTRAB.HED 14-DEC-77 12:19 INTRODUCTION TO TR79 DIAGNOSTIC

SEQ 0007

```

186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228

```

```

; *AC-9426B-MC /(<377>)/TR79F CHECKOUT TESTS
; *COPYRIGHT 1975, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
; *-----

; STARTING PROCEDURE
; LOAD PROGRAM
; LOAD ADDRESS 000200
; PRESS START
; PROGRAM WILL TYPE "AC-9426B-MC /(<377>)/TR79F CHECKOUT TESTS"
; PROGRAM WILL TYPE "AUTOSIZING" TO INDICATE THAT AUTOSIZING HAS STARTED
; (SWITCH #6 MUST BE DOWN FOR AUTOSIZING)
; PROGRAM WILL TYPE "MAP OF TR79 STATUS" TO INDICATE
; CSR ADDRESS, VECTOR, AND BR LEVEL
; PROGRAM WILL TYPE "RUNNING" TO INDICATE THAT TESTING HAS STARTED
; AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
; AND THEN RESUME TESTING

; SWITCH REGISTER OPTIONS
; -----
SW15=100000      ;=1, HALT ON ERROR
SW14=40000       ;=1, LOOP ON CURRENT TEST GROUP
SW13=20000       ;=1, INHIBIT ERROR TYPEOUT
SW12=10000       ;=1, BELL ON ERROR.
SW11=4000        ;=1, ESCAPE TO NEXT TEST ON ERROR
SW10=2000        ;=1, LOOP WITH CURRENT DATA
SW09=1000        ;=1, LOOP ON ERROR
SW08=400         ;NOTE...FOR TEST 4&5 LOOPS BACK TO BEGINING.
SW07=200
SW06=100         ;=1, INHIBIT AUTOSIZING
SW05=40          ;=0, INHIBIT TEST 5
                ;=1, ENABLE TEST 5
SW04=20          ;=0, INHIBITS TESTS 1 AND 6.
SW03=10          ;(IE, NO MANUAL INTERVENTION REQUIRED)
                ;=1, ENABLES TESTS 1 AND 6.
SW02=4           ;=1, LOCK ON TEST SELECT
SW01=2           ;=1, RESTART PROGRAM AT SELECTED TEST
SW00=1           ;=1, SELECT DEVICE ADDRESS, VECTOR, ETC.

```

```

100000
040000
020000
010000
004000
002000
001000
000400
000200
000100
000040
000020
000010
000004
000002
000001

```

229
230
231 ; SWITCH #15 - WHEN SET, THE DIAGNOSTIC WILL HALT WHEN AN
232 ; ERROR IS DETECTED. THE LOCATION AT WHICH THE ERROR
233 ; WAS DETECTED WILL BE DISPLAYED IN THE CONSOLE DATA
234 ; LIGHTS.
235
236 ; IF NOT SET, THE DIAGNOSTIC WILL CONTINUE TESTING
237 ; AS PER SWITCHES #8 AND #10.
238
239
240 ; SWITCH #14 - WHEN SET THE DIAGNOSTIC WILL LOOP ON THE
241 ; CURRENT TEST. IF NOT SET, THE DIAGNOSTIC WILL FALL
242 ; THROUGH TO THE NEXT TEST UPON COMPLETION OF THE
243 ; CURRENT TEST.
244
245
246 ; SWITCH #13 - WHEN SET, ERROR REPORTS WILL NOT BE
247 ; PRINTED AT THE TTY WHEN ERRORS ARE DETECTED.
248 ; IF NOT SET, ERROR REPORTS WILL BE PRINTED.
249
250 ; MESSAGES THAT MAY FOLLOW ERROR REPORTS
251 ; ARE NOT INHIBITED BY THIS SWITCH.
252
253
254 ; SWITCH #12 - WHEN SET, IF AN ERROR IS DETECTED, THE
255 ; TTY BELL WILL RING.
256
257 ; IF NOT SET, THE TTY BELL WILL NOT RING WHEN
258 ; AN ERROR IS DETECTED.
259
260
261 ; SWITCH #10 - IF THIS SWITCH IS SET WHEN AN ERROR IS
262 ; DETECTED, AFTER THE ERROR IS REPORTED, TESTING
263 ; WILL CONTINUE, STARTING WITH THE NEXT TEST.
264
265 ; IF SWITCH #8 IS SET, THIS SWITCH HAS NO EFFECT.
266
267 ; IF SWITCH #8 IS NOT SET AND THIS SWITCH IS NOT
268 ; SET, TESTING RESUMES AT THE POINT IN THE DIAGNOSTIC
269 ; FOLLOWING THE LOCATION AT WHICH THE ERROR WAS DETECTED.
270
271
272 ; SWITCH #9 - IF SET, THE DIAGNOSTIC USES THE CURRENT PARAMETERS
273 ; TO CONTINUE TESTING.
274
275
276 ; SWITCH #8 - IF THIS SWITCH IS SET, WHEN AN ERROR IS
277 ; DETECTED, TESTING WILL RESUME AT THE BEGINNING OF
278 ; THE TEST IN WHICH THE ERROR WAS DETECTED. (I.E. LOOP
279 ; ON ERROR.) SEE SWITCH #10.
280
281
282 ; SWITCH #6 - WHEN SET, THIS SWITCH INHIBITS AUTOSIZING.
283 ; WHEN CLEAR, THE DEVICE WILL AUTOSIZE.
284

J01

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 10
CZTRAB.HED 14-DEC-77 12:19 INTRODUCTION TO TR79 DIAGNOSTIC

SEQ 0009

285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306

; SWITCH #5 - WHEN SET, TEST GROUP #5 WILL BE EXECUTED.
; WHEN CLEAR, TEST GROUP #5 WILL NOT BE EXECUTED.

; SWITCH #3 - WHEN NOT SET, THE TEST REQUIRING MANUAL
; INTERVENTION, TEST #1, WILL NOT BE EXECUTED. IF
; SET, TEST #1 WILL BE EXECUTED.

; SWITCH #2 - IF SET, THE DIAGNOSTIC WILL LOOP ON THE TEST
; SELECTED. USED WITH SWITCH #1.

; SWITCH #1 - WHEN SET, THIS SWITCH ALLOWS THE
; OPERATOR TO SELECT THE STARTING TEST, VIA A
; DIALOG AT THE TTY.

; SWITCH #0 - WHEN SET, THIS SWITCH ALLOWS THE OPERATOR
; TO SPECIFY THE DEVICE ADDRESSES AND VECTORS
; IF THEY ARE DIFFERENT FROM THE STANDARD ONES.

K01

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 11
 CZTRAB.HED 14-DEC-77 12:19 GENERAL DEFINITIONS AND EQUIVALENCES

SEQ 0010

```

307
308
309      ;REGISTER DEFINITIONS
310      ;-----
311
312      000000      R0=%0      ;GENERAL REGISTER
313      000001      R1=%1      ;GENERAL REGISTER
314      000002      R2=%2      ;GENERAL REGISTER
315      000003      R3=%3      ;GENERAL REGISTER
316      000004      R4=%4      ;GENERAL REGISTER
317      000005      R5=%5      ;GENERAL REGISTER
318      000006      SP=%6      ;PROCESSOR STACK POINTER
319      000007      PC=%7      ;PROGRAM COUNTER
320
321      ;LOCATION EQUIVALENCES
322      ;-----
323
324      177776      PS=177776    ;PROCESSOR STATUS WORD
325      001200      STACK=1200  ;START OF PROCESSOR STACK
326
327      ;INSTRUCTION DEFINITIONS
328      ;-----
329
330      005746      PUSH1SP=5746 ;DECREMENT PROCESSOR STACK 1 WORD
331      005726      POP1SP=5726  ;INCREMENT PROCESSOR STACK 1 WORD
332      010046      PUSHRO=10046 ;SAVE R0 ON STACK
333      012600      POPRO=12600  ;RESTORE R0 FROM STACK
334      024646      PUSH2SP=24646;DECREMENT STACK TWICE
335      022626      POP2SP=22626 ;INCREMENT STACK TWICE
336      .EQUIV EMT,ERROR ;BASIC DEFINITION OF ERROR CALL
337
338      ;BIT DEFINITIONS
339      ;-----
340
341      100000      BIT15=100000
342      040000      BIT14=40000
343      020000      BIT13=20000
344      010000      BIT12=10000
345      004000      BIT11=4000
346      002000      BIT10=2000
347      001000      BIT9=1000
348      000400      BIT8=400
349      000200      BIT7=200
350      000100      BIT6=100
351      000040      BIT5=40
352      000020      BIT4=20
353      000010      BIT3=10
354      000004      BIT2=4
355      000002      BIT1=2
356      000001      BIT0=1
357
358      ;PROCESSOR PRIORITY LEVELS
359      ;-----
360
361      000340      LEVEL7=340
362      000300      LEVEL6=300

```

LO1

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 12
 CZTRAB.HED 14-DEC-77 12:19 GENERAL DEFINITIONS AND EQUIVALENCES

SEQ 0011

363	000240	LEVEL5=240
364	000200	LEVEL4=200
365	000140	LEVEL3=140
366	000100	LEVEL2=100
367	000040	LEVEL1=040
368	000000	LEVEL0=000

; TR79 CONTROL AND STATUS REGISTER DEFINITIONS
 ; (TRCR) BIT DEFINITIONS
 ; -----

374		; TAPE TEST EQUALS	
375			; UNIT
376	000000	ILC00 =0	0 ILLEGAL COMMAND
377	000402	WRITE =402	1 WRITE FORWARD
378	000004	READ =4	0 READ FORWARD
379	000406	ILC03 =406	1 ILLEGAL COMMAND
380	001410	SPACER =1410	3 SPACE REVERSE 1 RECORD
381	001012	ILC05 =1012	2 ILLEGAL COMMAND
382	001414	ILC06 =1414	3 ILLEGAL COMMAND
383	000016	ERASE =16	0 ERASE WORD COUNT FORWARD
384	001020	REWIND =1020	1 REWIND TAPE
385	000022	ILC11 =22	0 ILLEGAL COMMAND
386	000424	ILC12 =424	1 ILLEGAL COMMAND
387	004000	PWRCLR=BIT11	2 DEFINITION OF POWER CLEAR BIT
388	001026	GOEOT =1026	2 FAST FORWARD TO EOT
389	001030	ILC14 =1030	2 ILLEGAL COMMAND
390	000432	WIDB =432	1 WRITE I.D. BLOCK
391	001034	WEOF =1034	2 WRITE END OF FILE
392	001436	OFLINE =1436	3 TAKE UNIT OFF UNIT

394			
395	164000	TCR= 164000	; DEFAULT CONTROL REGISTER ADDRESS
396	164002	TSR= 164002	; DEFAULT STATUS REGISTER ADDRESS
397	164004	TWC= 164004	; DEFAULT WORD COUNT ADDRESS
398	164006	TBA= 164006	; DEFAULT BUS ADDRESS ADDRESS
399	000170	TVA= 170	; DEFAULT TAPE VECTOR ADDRESS
400	000172	TSA= 172	; DEFAULT TAPE STATUS ADDRESS

MO1

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 13
CZTRAB.MED 14-DEC-77 12:19

TRAPCATCHER FOR UNEXPECTED INTERRUPTS

SEQ 0012

```

401 ;*****
402 ;-----
403 ; TRAPCATCHER FOR ILLEGAL INTERRUPTS
404 ; THE STANDARD "TRAP CATCHER" IS PLACED
405 ; BETWEEN ADDRESS 0 TO ADDRESS 776.
406 ; IT LOOKS LIKE "PC+2 HALT".
407 ;-----
408 ;*****
409
410 000000 . =0
411 ; STANDARD INTERRUPT VECTORS
412 ;-----
413
414 . =24
415 000024 000024 . PFAIL ; POWER FAIL HANDLER
416 000026 000340 340 ; SERVICE AT LEVEL 7
417 000030 004642 . EROR ; ERROR HANDLER
418 000032 000340 340 ; SERVICE AT LEVEL 7
419 000034 004576 . TRPSRV ; GENERAL HANDLER DISPATCH SERVICE
420 000036 000340 340 ; SERVICE AT LEVEL 7
421
422 . =40
423 000040 000000 0 ; SAVE FOR ACT-11 OR XXDP
424 000042 000000 0 ; RETURN ADDRESS IF UNDER ACT-11 OR XXDP
425 000044 000000 0 ; SAVE FOR ACT-11 OR XXDP
426 000046 003470 . SENDAD ; FOR USE WITH ACT-11 OR XXDP
427 000052 000052 . =52
428 000000 0 ; ACT-11 PROGRAM CHARACTERISTICS
429
430 . =174
431 000174 000000 DISPREG:0 ; SOFTWARE DISPLAY REGISTER FOR SMALL 115
432 000176 000000 SWREG: 0 ; SOWTARE SWITCH REGISTER FOR SMALL 115
433 000200 000137 . =200
434 001510 JMP .START ; GO TO START OF PROGRAM
435
436 . =1000
437 001000 005377 041501 034455 MTITLE: .ASCIZ <377><12>/AC-9426B-MC /<377>/TR79F CHECKOUT TESTS/<377>
438 (2)
439 001200 001200 . =1200
440 001202 177570 DISPLAY: 177570
441 177570 SWR: 177570
442 ; INDIRECT POINTERS TO TELETYPE VECTORS AND REGISTERS
443 ;-----
444 001204 177560 TKCSR: 177560 ; TELETYPE KEYBOARD CONTROL REGISTER
445 001206 177562 TKDBR: 177562 ; TELETYPE KEYBOARD DATA BUFFER
446 001210 177564 TPCSR: 177564 ; TELEPRINTER CONTROL REGISTER
447 001212 177566 TPDBR: 177566 ; TELEPRINTER DATA BUFFER
448
449 ; PROGRAM CONTROL PARAMETERS
450 ;-----
451
452 001214 000000 RETURN: 0 ; SCOPE ADDRESS FOR LOOP ON TEST
453 001216 000000 NEXT: 0 ; ADDRESS OF NEXT TEST TO BE EXECUTED
454 001220 000000 LOCK: 0 ; ADDRESS FOR LOCK ON CURRENT DATA
455 001222 000001 ICOUNT: 1 ; NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED

```

NO1

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 14
CZTRAB.HED 14-DEC-77 12:19

PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

SEQ 0013

456 001224 000000
457 001226 000000
458 001230 000000
459 001232 000000
460 001234 000000
461 001236 000000
462
463
464
465

LPCNT: 0 ; NUMBER OF ITERATIONS COMPLETED
TSTNO: 0 ; NUMBER OF TEST IN PROGRESS
PASCNT: 0 ; NUMBER OF PASSES COMPLETED
ERRCNT: 0 ; TOTAL NUMBER OF ERRORS
LSTERR: 0 ; PC OF LAST ERROR CALL
SAVLIN: .WORD ; PROGRAM PARAMETER

; PROGRAM VARIABLES

466 001240 000010
467 001242 000000
468 001244 000000
469 001246 000000
470 001250 000000
471 001252 000000
472 001254 000000
473 001256 000000
474 001260 000000
475 001262 000000
476 001264 000000
477 001266 000000
478 001270 000215
479 001272 000212
480 001274 000377
481 001276 000000
482 001278 000000
483 001302 000000
484 001304 000000
485 001306 000000
486 001310 000000
487 001312 000000
488 001314 000000
489 001316 000000
490 001320 000000
491 001322 000000
492 001324 000000
493 001326 000000
494 001330 000000
495 001332 000000
496 001334 000000
497 001336 000000
498 001340 000000
499
500 001342 001500

HOLD: 10 ; DEFAULT DELAY TIME FOR 11/70 PROCESSOR
TRBASE: 0 ; BASE ADDRESS OF A GIVEN TR79 UNDER TEST
REGIST: 0 ; DEVICE ADDRESS STORAGE LOCATION
STAT: 0 ; TR STATUS WORD STORAGE
TSTPTR: 0
ERR: 0
XPC: 0
XSR: 0
COR: 0
ACT: 0
WRTFLG: 0
LOC: 0
CR: 215 ; TTY CARRIAGE RETURN
LF: 212 ; TTY LINE FEED
RO: 377 ; TTY RUBOUT
XTEST: 0
ZERO: 0
TC: 0 ; CONTROL STORAGE
TEMP1: 0 ; TEMPORARY STORAGE
TEMP2: 0 ; TEMPORARY STORAGE
TEMP3: 0 ; TEMPORARY STORAGE
TEMP4: 0 ; TEMPORARY STORAGE
TEMP5: 0 ; TEMPORARY STORAGE
SAVR0: 0 ; R0 STORAGE
SAVR1: 0 ; R1 STORAGE
SAVR2: 0 ; R2 STORAGE
SAVR3: 0 ; R3 STORAGE
SAVR4: 0 ; R4 STORAGE
SAVR5: 0 ; R5 STORAGE
SAVSP: 0 ; STACK POINTER STORAGE
SAVPC: 0 ; PROGRAM COUNTER STORAGE
SAVE0: 0
SAVE1: 0
.EVEN
ACTIVE: TR.MAP ; TABLE POINTER.

```

501
502           ;PROGRAM CONTROL FLAGS
503           ;-----
504
505 001344      000      INIFLG: .BYTE      0      ;PROGRAM INITIALIZATION FLAG
506 001345      000      ERRFLG: .BYTE      0      ;ERROR OCCURRED FLAG
507 001346      000      LOKFLG: .BYTE      0      ;LOCK ON CURRENT TEST FLAG
508 001347      000      QV.FLG: .BYTE      0      ;QUICK VERIFY FLAG
509 001350      000      MNTFLG: .BYTE      0      ;MAINTENANCE BIT SET FLAG
510                                     ;ON FIRST PASS OF EACH TR79 ITERATIONS WILL BE SUPPRESSE
511           001352      .EVEN
512
513           ;DEFINITIONS FOR TRAP SUBROUTINE CALLS
514           ;POINTERS TO SUBROUTINES CAN BE FOUND
515           ;IN THE TABLE IMMEDIATELY FOLLOWING THE DEFINITIONS
516
517           ;:*****
518           ;-----
519 001352      104400      TRPTAB:
520                                     SCOPE=TRAP+0      ;CALL TO SCOPE LOOP AND ITERATION HANDLER
521 001352      003544      .SCOPE
522                                     SCOP1=TRAP+1      ;CALL TO LOOP ON CURRENT DATA HANDLER
523 001354      003662      .SCOP1
524                                     TYPE=TRAP+2      ;CALL TO TELETYPE OUTPUT ROUTINE
525 001356      003706      .TYPE
526                                     TYPEF=TRAP+3      ;CALL TO FAILURE MESSAGE OUTPUT ROUTINE
527 001360      003752      .TYPEF
528                                     TYPEL=TRAP+4      ;CALL TO REPEAT MESSAGE OUTPUT ROUTINE
529 001362      003764      .TYPEL
530                                     INSTR=TRAP+5      ;CALL TO ASCII STRING INPUT ROUTINE
531 001364      003776      .INSTR
532                                     INSTER=TRAP+6      ;CALL TO INPUT ERROR HANDLER
533 001366      004102      .INSTER
534                                     PARAM=TRAP+7      ;CALL TO NUMERICAL DATA INPUT ROUTINE
535 001370      004122      .PARAM
536                                     SAVOS=TRAP+10      ;CALL TO REGISTER SAVE ROUTINE
537 001372      004322      .SAVOS
538                                     RESOS=TRAP+11      ;CALL TO REGISTER RESTORE ROUTINE
539 001374      004362      .RESOS
540                                     CONVRT=TRAP+12      ;CALL TO DATA OUTPUT ROUTINE
541 001376      004414      .CONVRT
542                                     CNVRT=TRAP+13      ;CALL TO DATA OUTPUT ROUTINE WITHOUT CR/LF.
543 001400      004420      .CNVRT
544
545           ;:*****
546           ;-----

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 16

CZTRAB.HED 14-DEC-77 12:19

PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

SEQ 0015

```
547                                     ;TR79 VECTOR AND REGISTER INDIRECT POINTERS
548                                     ;WORKING AREA
549
550 001402 164000      TRCR: 164000 ;R/W
551 001404 164001      HTRCR: 164001 ;R/W
552 001406 164002      TRSR: 164002 ;READ ONLY
553 001410 164003      HTRSR: 164003 ;READ ONLY
554 001412 164004      TRWC: 164004 ;R/W
555 001414 164005      HTRWC: 164005 ;R/W
556 001416 164006      TRBA: 164006 ;READ ONLY
557                                     ;DEFAULT TR VECTORS
558 001420 000170      TRVCT: 170 ;REC INTR VECTOR
559 001422 000172      TRRIS: 172 ;REC INTR STATUS
560
561
```

CZTRAB0 MACY11 30(1046) 14-DEC-77 12:32 PAGE 17
CZTRAB.HED 14-DEC-77 12:19

PROGRAM PARAMETERS, VARIABLES, AND TRAP CALLS.

SEQ 0016

```
562                                     ; TR79 STATUS TABLE AND ADDRESS ASSIGNMENTS
563                                     ; -----
564
565                                     ;=1500
566 001500 001500 TR.MAP:
567
568 001500 000001 TACRO: .BLKW 1 ; CONTROL REGISTER FOR TR79 NUMBER 0
569 001502 000001 TRVCO: .BLKW 1 ; BASE VECTOR FOR TR79 NUMBER 0
570 001504 000001 TRLVO: .BLKW 1 ; PRIORITY LEVEL
571
572 001506 000000 TR.END: 000000
```


F02

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 19
 CZTRAB.HED 14-DEC-77 12:19 PROGRAM INITIALIZATION AND START UP.

SEQ 0018

```

629 001752 002026          69$          ; POINTER TO MESSAGE TO BE PRINTED
630 001754 104407          PARAM        ; CALL THE OCTAL TO ASCII CONVERT ROUTINE
631 001756 160000          160000       ; LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
632 001760 175500          175500       ; HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
633 001762 001500          TRCRO        ; POINTER TO MAP LOCATION TO BE FILLED
634 001764 007            .BYTE 7      ; MASK OF INVALID BITS FOR THIS PARAMETER
635 001765 001            .BYTE 1      ; NUMBER OF PARAMETERS TO STORE
636
637          ; GET THE BASE VECTOR ADDRESS
638
639          34$:
640 001766          INSTR              ; CALL THE STRING INPUT ROUTINE
641 001770 104405       70$          ; POINTER TO MESSAGE TO BE PRINTED
642 001772 002054       PARAM        ; CALL THE OCTAL TO ASCII CONVERT ROUTINE
643 001774 104407       170         ; LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
644 001776 000170       776         ; HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
645 001778 000776       TRVCO        ; POINTER TO MAP LOCATION TO BE FILLED
646 002000 001502       .BYTE 3      ; MASK OF INVALID BITS FOR THIS PARAMETER
647 002002 003          .BYTE 1      ; NUMBER OF PARAMETERS TO STORE
648
649          ; GET THE BUS REQUEST LEVEL
650
651 002004 104405       INSTR              ; CALL THE STRING INPUT ROUTINE
652 002006 002102       72$          ; POINTER TO MESSAGE TO BE PRINTED
653 002010 104407       PARAM        ; CALL THE OCTAL TO ASCII CONVERT ROUTINE
654 002012 000004       4           ; LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
655 002014 000007       7           ; HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
656 002016 001504       TRLVO        ; POINTER TO MAP LOCATION TO BE FILLED
657 002020 000          .BYTE 0      ; MASK OF INVALID BITS FOR THIS PARAMETER
658 002021 001          .BYTE 1      ; NUMBER OF PARAMETERS TO STORE
659 002022 000137 002146  JMP 66$      ; GO CONTINUE PROCESSING
660
661 002026 041777 047117 051124 69$: .ASCIZ <377>/CONTROLLER ADDRESS /
(2) 002054 030777 052123 053040 70$: .ASCIZ <377>/1ST VECTOR ADDRESS /
(2) 002102 047377 052117 035105 72$: .ASCII <377>/NOTE: ONE TR79 ONLY /<377>
(2) 002131 377 051102 046040 71$: .ASCIZ <377>/BR LEVEL /
(2) 002146 002146 .EVEN
(2) 002146 032777 000100 177026 66$: BIT #BIT6, @SWR ; INHIBIT AUTOSIZING?
662 002154 001402 1$ BEQ 1$ ; NO
663 002156 000137 003140 JMP HEADER ; YES-GO PRINTOUT MAP TABLE
664 002162 104402 005320 1$: TYPE ,MCRLF ;
665 002166 104402 010401 TYPE ,SIZE ; TYPE "AUTOSIZING"
666 002172 104402 005320 TYPE ,MCRLF ;
667 002176 013737 001500 001336 MOV @TR.MAP, SAVED ; SAVE PRE-AUTOSIZE CSR ADDRESS
668 002204 012706 001200 CSRSIZ: MOV #STACK, SP ; RESET STACK
669 002210 005037 001226 CLR TSTNO ; CALL THIS "TEST 0"
670 002214 012737 002204 001214 MOV #CSRSIZ, RETURN ; SETUP FOR LOOP ON ERROR
671 002222 012737 002376 001216 MOV #22$, NEXT ; SETUP FOR SKIP ON ERROR
672 002230 004737 013146 JSR PC, TESTE ; CHECK INT ENB BIT SETS
673 002234 004737 013202 JSR PC, TESTF ; CHECK INT ENB BIT CLEARS
674 002240 004737 013234 JSR PC, TESTG ; CHECK PWR CLR WORKS
675 002244 012737 002310 000004 MOV #99$, @#4 ; SETUP FOR TIMEOUT
676 002252 012737 000340 000006 MOV #340, @#6
677 002260 012700 160000 MOV #160000, RD ; FIRST VALUE IN RANGE
678 002264 005710 1$ TST (RD) ; IS THIS THE ADDRESS?
679 002266 012737 000006 000004 MOV #6, @#4 ; YES-RESTORE TRAP CATCHER

```

G02

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 20
 CZTRAB.HED 14-DEC-77 12:19 PROGRAM INITIALIZATION AND START UP.

SEQ 0019

680	002274	012737	000000	000006	MOV	#0,	2#6	
681	002302	010037	001340		MOV	RO,	SAVE1	;STORE AUTOSIZED VALUE
682	002306	000421			BR	20\$		
683	002310	022626			99\$: POP2SP			;TIMEOUT-RESTORE STACK
684	002312	062700	000002		ADD	#2,	RO	;SETUP FOR NEXT ADDRESS
685	002316	022700	175502		CMP	#175502,	RO	;END OF RANGE
686	002322	001401			BEQ	2\$		;YES
687	002324	000757			BR	1\$		;NO
688	002326	012737	000006	000004	2\$: MOV	#6,	2#4	;RESTORE TRAP CATCHER
689	002334	012737	000000	000006	MOV	#0,	2#6	
690	002342	104014			ERROR	14		;CSR NEVER SIZED-ERROR MSG
691	002344	104402	010323		TYPE	,PRHLT		
692	002350	000000			HALT			
693	002352	023737	001336	001340	20\$: CMP	SAVED,	SAVE1	;EXPECTED CSR ADDRESS?
694	002360	001406			BEQ	22\$		;YES
695	002362	013737	001340	001500	MOV	SAVE1,	2#TR.MAP	;NO-PUT SIZED VALUE INTO TABLE
696	002370	104012			ERROR	12		;ERROR MSG
697	002372	104403	010415		TYPEF	CSRER2		
698	002376	013737	001502	001336	22\$: MOV	2#TR.MAP+2,	SAVED	;SAVE PRE-SIZE VECTOR
699	002404	012737	002404	001214	VECSIZ: MOV	#VECSIZ,	RETURN	;SETUP FOR LOOP ON ERROR
700	002412	012737	002710	001216	MOV	#23\$,	NEXT	;SETUP FOR SKIP ON ERROR
701	002420	012706	001200		MOV	#STACK,	SP	;RESET STACK
702	002424	012737	000340	000022	MOV	#340,	2#22	
703	002432	012737	002632	000020	MOV	#4\$,	2#20	;SETUP FOR IOT TRAP
704	002440	012737	000340	000006	MOV	#340,	2#6	
705	002446	012737	002604	000004	MOV	#8\$,	2#4	;SETUP FOR TIMEOUT
706	002454	012737	000172	000170	MOV	#172,	2#170	;SETUP FOR INT TO THIS ADDRESS
707	002462	012737	000004	000172	MOV	#4,	2#172	
708	002470	012700	000210		MOV	#210,	RO	;SETUP FOR INT TO ALL POSSIBLE ADDRESSES
709	002474	012701	000212		MOV	#212,	R1	
710	002500	010120			10\$: MOV	R1,	(RO)+	
711	002502	012721	000004		MOV	#4,	(R1)+	
712	002506	022021			CMP	(RO)+,	(R1)+	
713	002510	020127	001000		CMP	R1,	#1000	
714	002514	101771			BLOS	10\$		
715	002516	005037	177776		CLR	PS		;MAKE CPU PRIORITY 0
716	002522	005037	001340		CLR	SAVE1		
717	002526	052777	004100	176744	BIS	#4100,	2#TR.MAP	;ENABLE INTERRUPT
718	002534	005000			CLR	RO		;WAIT 900 MSEC
719	002536	012701	177766		MOV	#-12,	R1	
720	002542	005200			1\$: INC	RO		
721	002544	001376			BNE	1\$		
722	002546	005201			INC	R1		
723	002550	001374			BNE	1\$		
724	002552	042777	004100	176720	BIC	#4100,	2#TR.MAP	;NO INTERRUPT-DISABLE INTERRUPT
725	002560	012737	000006	000004	MOV	#6,	2#4	;RESTORE TRAP CATCHER
726	002566	012737	000000	000006	MOV	#0,	2#6	
727	002574	104013			ERROR	13		;ERROR MSG
728	002576	104402	010323		TYPE	,PRHLT		
729	002602	000000			HALT			
730	002604	012737	000006	000004	8\$: MOV	#6,	2#4	;TIMEOUT-RESTORE TRAP CATCHER
731	002612	012737	000000	000006	MOV	#0,	2#6	
732	002620	022626			POP2SP			;RESTORE STACK
733	002622	104015			ERROR	15		;ERROR MSG
734	002624	104402	010323		TYPE	,PRHLT		
735	002630	000000			HALT			

H02

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 21

CZTRAB.HED 14-DEC-77 12:19

PROGRAM INITIALIZATION AND START UP.

SEQ 0020

```

736 002632 012737 000006 000004 4$: MOV #6, @#4 ;RESTORE TRAP CATCHER
737 002640 012737 000000 000006 MOV #0, @#6
738 002646 011637 001340 MOV (SP), SAVE1 ;STACK POINTS TO VECTOR ADDRESS PLUS 4
739 002652 162737 000004 001340 SUB #4, SAVE1
740 002660 022626 POP2SP ;RESTORE STACK
741 002662 022626 POP2SP
742 002664 023737 001336 001340 CMP SAVED, SAVE1 ;EXPECTED VECTOR ADDRESS?
743 002672 001406 BEQ 23$ ;YES
744 002674 013737 001340 001502 MOV SAVE1, @#TR.MAP+2 ;NO-PUT SIZED VALUE INTO TABLE
745 002702 104012 ERROR 12 ;ERROR MSG
746 002704 104403 TYPEF ;VECE3
747 002710 013737 001504 001336 23$: MOV @#TR.MAP+4, SAVED ;SAVE PRE-SIZE BR LEVEL
748 002716 012737 000300 001304 MOV #300, TEMP1 ;STORE CPU PRIORITY TO START BR SIZING
749 002724 012737 000006 001340 MOV #6, SAVE1 ;#300 IS PRIORITY 6
750 002732 012737 002732 001214 BRSIZ: MOV #BRSIZ, RETURN ;SETUP FOR LOOP ON ERROR
751 002740 012737 003140 001216 MOV #HEADER, NEXT ;SETUP FOR SKIP ON ERROR
752 002746 012706 001200 MOV #STACK, SP ;RESET STACK
753 002752 013737 001334 177776 MOV TEMP1, PS ;SET CPU PRIORITY
754 002760 013702 001502 MOV @#TR.MAP+2, R2 ;GET VECTOR ADDRESS
755 002764 012722 003100 MOV #34$, (R2)+ ;SETUP FOR INT
756 002770 012712 000340 MOV #340, (R2)
757 002774 052777 004100 176476 BIS #4100, @#TR.MAP ;ENABLE INT
758 003002 005000 CLR RO ;WAIT 900 MSEC
759 003004 012701 177766 MOV #-12, R1
760 003010 005200 1$: INC RO
761 003012 001376 BNE 1$
762 003014 005201 INC R1
763 003016 001374 BNE 1$
764 003020 042777 004100 176452 BIC #4100, @#TR.MAP ;NO INT OCCURRED
765 003026 023737 001340 001336 CMP SAVE1, SAVED ;EXPECTED INT?
766 003034 002003 BGE 2$ ;NO
767 003036 104013 ERROR 13 ;YES-ERROR MSG
768 003040 104403 TYPEF ;BRER1
769 003044 162737 000040 001304 2$: SUB #40, TEMP1 ;DECREMENT CPU PRIORITY
770 003052 005337 001340 DEC SAVE1
771 003056 022737 000002 001340 CMP #2, SAVE1 ;IS CPU AT LEVEL 2?
772 003064 001402 BEQ 11$ ;YES
773 003066 000137 002732 JMP @#BRSIZ ;NO-CONTINUE
774 003072 104402 010323 11$: TYPE ,PRHLT ;IF NO INT AT CPU LEVEL 3-
775 003076 000000 HALT ;HALT
776 003100 042777 004100 176372 34$: BIC #4100, @#TR.MAP ;INT OCCURRED
777 003106 022626 POP2SP ;RESTORE STACK
778 003110 013737 001340 001504 MOV SAVE1, @#TR.MAP+4 ;DEVICE PRIORITY-
779 003116 005237 001504 INC @#TR.MAP+4 ;IS CPU'S AT INT PLUS 1
780 003122 023737 001340 001336 CMP SAVE1, SAVED ;EXPECTED INT?
781 003130 002403 BLT HEADER ;YES
782 003132 104016 ERROR 16 ;NO-ERROR MSG
783 003134 104403 TYPEF ;BRER1
784 003140 012737 000001 001226 HEADER: MOV #1, TSTNO
785 003146 104402 005320 TYPE ,MCRLF
786 003152 104402 005631 16$: TYPE ,XHEAD ;TYPE MAP HEADER
787 003156 012737 001500 001304 MOV #1500, TEMP1
788 003164 012700 001500 MOV @#TR.MAP, RO ;SET POINTER
789 003170 012037 001306 5$: MOV (RO)+, TEMP2 ;SET DATA
790 003174 001406 BEQ MON ;ALL DONE WITH DATA
791 003176 104412 CONVRT ;CALL THE OCTAL TO ASCII CONVERSION ROUTINE

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 22
 CZTRAB.HED 14-DEC-77 12:19 PROGRAM INITIALIZATION AND START UP.

SEQ 0021

```

792 003200 010502          XSTATQ          ;CONVERT THE DATA AT THIS ADDRESS
793 003202 062737 000002 001304      ADD      #2,      TEMP1
794 003210 000767          BR      5$
795 003212 005037 001304      MON:      CLR      TEMP1
796 003216 005737 000042          TST      @#42          ; IS PROGRAM RUNNING UNDER MONITOR
797 003222 001000          BNE      3$          ; YES
798 003224 012700 000210      3$:      MOV      #210,RO      ; PREPARE TO CLEAR THE REMAINING
799 003230 012701 000212          MOV      #212,R1          ; VECTOR AREA. 210-776
800 003234 010120      4$:      MOV      R1,(R0)+          ; START PUTTING "PC+2 - HALT"
801 003236 005021          CLR      (R1)+          ; IN VECTOR AREA.
802 003240 022021          CMP      (R0)+,(R1)+          ; POP POINTERS
803 003242 022700 001000          CMP      #1000,RO          ; ALL DONE??
804 003246 001372          BNE      4$          ; BR IF NO.
805 003250 005037 000172          CLR      @#172
806
807          ;TEST START AND RESTART
808          ;-----
809
810 003254 012737 000340 177776 .BEGIN: MOV      #340,PS          ; LOCK OUT INTERRUPTS
811 003262 012706 001200          MOV      #STACK,SP          ; SET UP STACK
812 003266 005737 000042          TST      @#42          ; IS PROGRAM UNDER MONITOR CONTROL
813 003272 001023          BNE      2$          ; BR IF YES
814 003274 032777 000004 175700      BIT      #BIT2,@SWR          ; CHECK FOR LOCK ON TEST
815 003302 001411          BEQ      1$          ; BR IF NO LOCK DESIRED.
816 003304 104402 005523          TYPE      ,MLOCK          ; TYPE LOCK SELECTED.
817 003310 012737 000240 003556          MOV      #NOP,TTST          ; ADJUST SCOPE ROUTINE.
818 003316 012737 000240 003560          MOV      #NOP,TTST+2          ; SET UP TO LOCK
819 003324 000406          BR      2$          ; CONTINUE ALONG.
820 003326 013737 003656 003556 1$:      MOV      BRW,TTST          ; PREPARE NORMAL SCOPE ROUTINE
821 003334 013737 003660 003560          MOV      BRX,TTST+2          ; LOCK NOT SELECTED, SET UP FOR NORMAL SCOPE LOOP
822 003342 012737 010664 001214 2$:      MOV      #CYCLE,RETURN          ; START AT "CYCLE" FIND WHICH DEVICE TO TEST
823 003350 104402 005320          TYPE      ,MCRLF
824 003354 104402 005404          TYPE      ,MR
825 003360 000177 175630          JMP      @RETURN          ; TYPE "RUNNING"
                                     ; START TESTING

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 23
 CZTRAB.HED 14-DEC-77 12:19 END OF PASS ROUTINE

SEQ 0022

```

826 ;END OF PASS
827 ;TYPE NAME OF TEST
828 ;UPDATE PASS COUNT
829 ;CHECK FOR EXIT TO ACT-11
830 ;RESTART TEST
831
832 .EOP:
833 003364 005037 001234 CLR LSTERR ;CLEAR LAST ERROR PC
834 003370 105037 001345 CLR ERRFLG ;CLEAR ERROR FLAG
835 003374 005237 001230 INC PASCNT ;UPDATE PASS COUNT
836 003400 013777 001230 175572 MOV PASCNT, @DISPLAY ;DISPLAY PASS COUNT
837 003406 104402 005361 TYPE ,MEPASS ;TYPE END PASS
838 003412 104402 005552 TYPE ,MCSR ;TYPE CSR
839 003416 104413 003514 CNVRT ,XCSR ;SHOW IT
840 003422 104402 005560 TYPE ,MVECX ;TYPE VECTOR
841 003426 104413 003522 CNVRT ,XVEC ;SHOW IT
842 003432 104402 005566 TYPE ,MPASSX ;TYPE PASSES
843 003436 104413 003530 CNVRT ,XPASS ;SHOW IT
844 003442 104402 005577 TYPE ,MERRX ;TYPE ERRORS
845 003446 104413 003536 CNVRT ,XERR ;SHOW IT
846 003452 112737 000377 001347 MOVB #377, @V.FLG ;SET THE QUICK VERIFY FLAG.
847 003460 013701 000042 MOV @#42, R1 ;CHECK FOR ACT-11 OR DDP
848 003464 001406 BEQ RESTART ;IF NOT, CONTINUE TESTING
849 003466 000005 RESET ;STOP THE SHOW--CLEAR THE WORLD
850
851 $ENDAD:
852 003470 004711 JSR PC, (R1)
853 003472 000240 NOP
854 003474 000240 NOP
855 003476 000240 NOP
856 003500 000240 NOP
857 003502 012737 010664 001214 RESTART: MOV #CYCLE, RETURN
858 003510 000137 010664 JMP CYCLE
859 003514 000001 XCSR: 1
860 003516 006 002 .BYTE 6,2
861 003520 001402 TRCR
862 003522 000001 XVEC: 1
863 003524 003 002 .BYTE 3,2
864 003526 001420 TRVCT
865 003530 000001 XPASS: 1
866 003532 006 002 .BYTE 6,2
867 003534 001230 PASCNT
868 003536 000001 XERR: 1
869 003540 006 002 .BYTE 6,2
870 003542 001232 ERRCNT
871
872 ;SCOPE LOOP AND ITERATION HANDLER
873 ;-----
874 .SCOPE: CLR LSTERR ;CLEAR LAST ERROR PC.
875 003550 032777 040000 175424 BIT #BIT14, @SWR ;"LOOP ON THIS TEST"?
876 003556 001407 TTST: BEQ 1$ ;BR IF NO. (IF LOCK SW01=1; THIS LOC =240)
877 003560 000433 BR 3$ ;GOTO 3$ (IF LOCK SW01=1; THIS LOC =240)
878 003562 105777 175416 TSTB @TKCSR ;KEYBOARD DONE?
879 003566 100030 BPL 3$ ;BR IF NO. (LOCK: HIT KEY TO GOTO NEXT TEST)
880 003570 017716 175412 MOV @TKDBR, (SP) ;CLEAR DONE BIT
881 003574 000411 BR 2$ ;CONTINUE

```

K02

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 24
 CZTRAB.HED 14-DEC-77 12:19 GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

SEQ 0023

```

882 003576 105737 001347 1$: TSTB QV.FLG ; HAVE PASSES BEEN COMPLETED?
883 003602 001406 BEQ 2$ ; BR IF QUICK PASS.
884 003604 005237 001224 INC LPCNT ; UPDATE ITERATION COUNTER
885 003610 023737 001224 001222 CMP LPCNT,ICOUNT ; ARE ALL ITERATIONS DONE??
886 003616 101414 BLOS 3$ ; BR IF NOT YET
887 003620 105037 001345 2$: CLRB EPRFLG ; PREPARE FOR NEW TEST
888 003624 005037 001224 CLR LPCNT ; START ICOUNTER AT 0
889 003630 005037 001220 CLR LOCK
890 003634 012737 000000 001222 MOV #0,ICOUNT ; RESET ITERATIONS
891 003642 013737 001216 001214 MOV NEXT,RETURN ; GET NEXT TEST
892 003650 022626 3$: POP2SP ; FAKE AN "RTI"
893 003652 000177 175336 4$: JMP @RETURN ; GO DO THE TEST
894 003656 001407 BRW: 1407
895 003660 000437 BRX: 437

; CHECK FOR FREEZE ON CURRENT DATA
;-----
900 003662 032777 001000 175312 .SCOPI: BIT #SW09,@SWR ; IS SW09=1(SET)?
901 003670 001405 BEQ 1$ ; BR IF NOT SET.
902 003672 005737 001220 TST LOCK ; IS THER A TIGHT LOOP SPECIFIED?
903 003676 001402 BEQ 1$ ; I NO, RETURN
904 003700 013716 001220 MOV LOCK,(SP) ; IF YES, GOTO THE ADDRESS IN LOCK.
905 003704 000002 1$: RTI ; GO BACK.

; TELETYPE OUTPUT ROUTINE
;-----
910 003706 010546 .TYPE: MOV R5,-(SP) ; SAVE R5 ON THE STACK.
911 003710 017605 000002 MOV @2(SP),R5 ; GET ADDRESS OF MESSAGE.
912 003714 062766 000002 000002 ADD #2,2(SP) ; POP OVER ADDRESS.
913 003722 105715 1$: TSTB (R5)
914 003724 100002 BPL 2$
915 003726 104402 005320 TYPE MCRLF
916 003732 105777 175252 2$: TSTB @TPCSR ; TTY READY?
917 003736 100375 BPL 2$ ; BR IF NO.
918 003740 112577 175246 MOVB (R5)+,@TPDBR ; PRINT CURRENT CHAR.
919 003744 001366 BNE 1$ ; IF NOT ZERO KEEP PRINTING!
920 003746 012605 3$: MOV (SP)+,R5 ; END OF OUTPUT. RESTORE R5
921 003750 000002 RTI ; GO HOME

; AUXILLIARY TELETYPE OUTPUT ROUTINES
;-----
926 003752 017637 000000 003766 .TYPEF: MOV @2(SP),LSTMSG ; GET THE ADDRESS OF THE MESSAGE TO PRINT
927 003760 062716 000002 ADD #2,(SP) ; POINT TO THE NEXT LOCATION IN THE MAIN PROCEDURE
928 003764 104402 .TYPEL: TYPE ; BOTH ROUTINES TYPE HERE
929 003766 000000 LSTMSG: .WORD 0 ; POINTER TO MESSAGE TO PRINT
930 003770 104402 TYPE ; THIS PART PRINTS THE REMAINDER OF THE MESSAGE
931 003772 010312 MFAIL ; THIS IS THE 'FAILURE' PART OF IT
932 003774 000002 1$: RTI ; RETURN TO THE MAIN PROCEDURE

; STRING INPUT ROUTINE
;-----
937 003776 010346 .INSTR: MOV R3,-(SP) ; SAVE R3 ON STACK

```

```

938 004000 010446          MOV      R4, -(SP)      ;SAVE R4 ON STACK
939 004002 017637 000004 004020  MOV      R4, -(SP)      ;GET THE ADDRESS OF THE MESSAGE TO BE PRINTED
940 004010 062766 000002 000004  ADD      #2, 4(SP)      ;POINT TO INSTRUCTION AFTER THE ADDRESS POINTER
941 004016 104402          .INST1: TYPE      ;PRINT THE MESSAGE
942 004020 000000          .MSG: 0          ;MESSAGE IS POINTED TO FROM HERE
943 004022 012704 010514          MOV      #INBUF, R4      ;POINT R4 TO THE INPUT BUFFER
944 004026 012703 000007          MOV      #7, R3          ;SET THE MAXIMUM NUMBER OF CHARACTERS ALLOWED
945 004032 105777 175146          1$: TSTB     @TKCSR      ;HAS A CHARACTER BEEN RECEIVED?
946 004036 100375          BPL          1$              ;IF NO, KEEP WAITING FOR IT
947 004040 117714 175142          MOV      @TKDBR, (R4)      ;IF YES, SAVE IT IN THE INPUT BUFFER
948 004044 142714 000200          BICB     #200, (R4)      ;KEEP ONLY THE 7-BIT ASCII INFORMATION
949 004050 122427 000015          CMPB     (R4), #15      ;IS THIS CHARACTER A LINE FEED?
950 004054 001417          BEQ          INSTR2          ;IF SO, TERMINATE THE INPUT SEQUENCE
951 004056 105777 175126          2$: TSTB     @TPCSR      ;IF NOT, CHECK TO SEE IF THE CHARACTER CAN PRINT
952 004062 100375          BPL          2$              ;IF WE CAN'T, WAIT UNTIL WE CAN
953 004064 017777 175116 175120  MOV      @TKDBR, @TPDBR      ;ECHO THE CHARACTER BACK
954 004072 005303          DEC          R3              ;REDUCE THE NUMBER OF CHARACTERS RECEIVED
955 004074 001356          BNE          1$              ;IF WE DON'T HAVE 7, GO GET SOME MORE
956 004076 012604          MOV      (SP)+, R4          ;IF WE HAVE 7, RESTORE R4
957 004100 012603          MOV      (SP)+, R3          ;RESTORE R3
958 004102 010346          .INSTE: MOV      R3, -(SP)      ;SAVE R3 ON THE STACK
959 004104 010446          MOV      R4, -(SP)      ;SAVE R4 ON THE STACK
960 004106 104402 005314          TYPE      ,MOM          ;PRINT A QUESTION MARK... WHAT'S GOING ON?
961 004112 000741          BR          .INST1          ;GO PRINT THE MESSAGE AGAIN
962 004114 012604          INSTR2: MOV      (SP)+, R4      ;RESTORE R4
963 004116 012603          MOV      (SP)+, R3      ;RESTORE R3
964 004120 000002          RTI          ;RETURN TO THE MAIN PROCEDURE
965
966          ;CONVERT ASCII STRING TO OCTAL
967          ;-----
968
969 004122 010546          .PARAM: MOV      R5, -(SP)      ;SAVE R5 ON THE STACK
970 004124 010446          MOV      R4, -(SP)      ;SAVE R4 ON THE STACK
971 004126 016605 000004          MOV      4(SP), R5      ;GET THE SETUP INFORMATION POINTER
972 004132 012537 004312          MOV      (R5)+, LOLIM      ;SET THE LOW LIMIT FOR THE INPUT
973 004136 012537 004314          MOV      (R5)+, HILIM      ;SET THE HIGH LIMIT FOR THE INPUT
974 004142 012537 004316          MOV      (R5)+, DEVADR      ;SAVE THE ADDRESS WHERE THE RESULT WILL BE STORED
975 004146 112537 004320          MOV      (R5)+, LOBITS      ;GET THE MASK OF THE INCORRECT BITS
976 004152 112537 004321          MOV      (R5)+, ADRCNT      ;GET THE COUNT OF ITEMS TO BE STORED
977 004156 010566 000004          MOV      R5, 4(SP)      ;POINT TO WHERE THE MAIN PROGRAM WILL RESUME
978 004162 005005          PARAM1: CLR      R5          ;INITIALIZE THE ASCII TO OCTAL RESULT WORD
979 004164 012704 010514          MOV      #INBUF, R4      ;POINT TO THE INPUT BUFFER
980 004170 122714 000015          CMPB     #15, (R4)      ;IS THIS CHARACTER A CARRIAGE RETURN?
981 004174 001420          BEQ          PARERR      ;IF SO, PRINT THE MESSAGE AGAIN
982 004176 121427 000060          1$: CMPB     (R4), #60      ;IS THIS CHARACTER BELOW THE NUMERIC RANGE?
983 004202 002415          BLT          PARERR      ;IF SO, GO PRINT THE MESSAGE AGAIN
984 004204 121427 000067          CMPB     (R4), #67      ;IS THIS CHARACTER ABOVE THE NUMERIC RANGE?
985 004210 003012          BGT          PARERR      ;IF SO, GO PRINT THE MESSAGE AGAIN
986 004212 142714 000060          BICB     #60, (R4)      ;ISOLATE THE NUMBER THE CHARACTER REPRESENTS
987 004216 152405          BISB     (R4)+, R5      ;CONCATENATE THESE BITS TO THE EXISTING STRING
988 004220 122714 000015          CMPB     #15, (R4)      ;IS THE NEXT CHARACTER A CARRIAGE RETURN?
989 004224 001406          BEQ          LIMITS      ;IF SO, GO SEE IF THE NUMBER IS WITHIN LIMITS
990 004226 006305          ASL          R5          ;CLEAR BIT POSITION 0, MOVE THE EXISTING STRING TO THE L
991 004230 006305          ASL          R5          ;CLEAR POSITION 1, MOVE THE STRING TO THE LEFT AGAIN
992 004232 006305          ASL          R5          ;MOVE THE STRING ONE MORE TIME TO MAKE ROOM FOR
993          ;NEXT THREE BITS

```

M02

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 26

CZTRAB.HED 14-DEC-77 12:19

GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

SEQ 0025

```

994 004234 000760
995 004236 104406
996 004240 000750
997
998
999
1000
1001 004242 020537 004314
1002 004246 101373
1003 004250 020537 004312
1004 004254 103700
1005 004256 133705 004320
1006 004262 001365
1007
1008
1009
1010 004264 013704 004316
1011 004270 010524
1012 004272 062705 000002
1013 004276 105337 004321
1014 004302 001372
1015 004304 012604
1016 004306 012605
1017 004310 000002
1018
1019 004312 000000
1020 004314 000000
1021 004316 000000
1022 004320 000
1023 004321 000
1024
1025
1026
1027
1028 004322 016637 000004 001334 .SAV05: MOV 4(SP),SAVPC ;SAVE R7 (PC)
1029
1030
1031
1032 004330 010537 001330
1033 004334 010437 001326
1034 004340 010337 001324
1035 004344 010237 001322
1036 004350 010137 001320
1037 004354 010037 001316
1038 004360 000002
1039
1040
1041
1042 004362 013700 001316
1043 004366 013701 001320
1044 004372 013702 001322
1045 004376 013703 001324
1046 004402 013704 001326
1047 004406 013705 001330
1048 004412 000002
1049

```

```

PARERR: BR 1$ ;GO GET THE NEXT CHARACTER
;THERE WAS AN ERROR- GO PRINT THE MESSAGE AGAIN
BR PARAM1 ;TRY GETTING THE PARAMETERS AGAIN

;TEST TO SEE IF NUMBER IS WITHIN LIMITS
-----
LIMITS: CMP R5,HILIM ;DOES RESULT EXCEED ITS MAXIMUM CORRECT VALUE?
BHI PARERR ;IF YES, GO PRINT THE MESSAGE AGAIN
CMP R5,LOLIM ;IS THE RESULT LOWER THAN ALLOWED?
BLO PARERR ;IF YES, GO PRINT THE MESSAGE AGAIN
BITB LOBITS,R5 ;ARE ANY INCORRECT BITS SET IN THE RESULT?
BNE PARERR ;IF SO, GO PRINT THE MESSAGE AGAIN

;STORE NUMBER AT SPECIFIED ADDRESS
1$: MOV DEVADR,R4 ;POINT TO THE LOCATION WHERE THE RESULT WILL BE STORED
MOV R5,(R4)+ ;STORE THE RESULT
ADD #2,R5 ;CALCULATE THE NEXT DATUM
DECB ADRCNT ;REDUCE COUNT OF STORED RESULTS. IS IT EXCEEDED?
BNE 1$ ;IF NOT, GO STORE THE NEXT DATUM
MOV (SP)+,R4 ;RESTORE R4
MOV (SP)+,R5 ;RESTORE R5
RTI ;RETURN TO THE MAIN PROGRAM

LOLIM: 0 ;LOWEST ACCEPTABLE VALUE
HILIM: 0 ;HIGHEST ACCEPTABLE
DEVADR: 0 ;LOCATION WHERE RESULT WILL BE STORED
LOBITS: .BYTE 0 ;INCORRECT BITS MASK
ADRCNT: .BYTE 0 ;COUNT OF ITEMS TO BE STORED

;SAVE PC OF TEST THAT FAILED AND R0-R5
-----
.SAV05: MOV 4(SP),SAVPC ;SAVE R7 (PC)
;SAVE R0-R5
SV05: MOV R5,SAVR5 ;SAVE R5
MOV R4,SAVR4 ;SAVE R4
MOV R3,SAVR3 ;SAVE R3
MOV R2,SAVR2 ;SAVE R2
MOV R1,SAVR1 ;SAVE R1
MOV R0,SAVR0 ;SAVE R0
RTI ;LEAVE.

;RESTORE R0-R5
.RES05: MOV SAVR0,R0 ;RESTORE R0
MOV SAVR1,R1 ;RESTORE R1
MOV SAVR2,R2 ;RESTORE R2
MOV SAVR3,R3 ;RESTORE R3
MOV SAVR4,R4 ;RESTORE R4
MOV SAVR5,R5 ;RESTORE R5
RTI ;LEAVE

```

```

1050 ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
1051 ;-----
1052
1053 004414 104402 005320 .CONVR: TYPE MCRLF ;PRINT A CARRIAGE RETURN
1054 004420 010046 .CNVRT: MOV R0,-(SP) ;SAVE R0
1055 004422 010146 MOV R1,-(SP) ;SAVE R1
1056 004424 010346 MOV R3,-(SP) ;SAVE R3
1057 004426 010446 MOV R4,-(SP) ;SAVE R4
1058 004430 010546 MOV R5,-(SP) ;SAVE R5
1059 004432 017601 000012 MOV @12(SP),R1 ;PLACE THE ADDRESS OF THE ARGUMENTS IN R1
1060 004436 062766 000002 000012 ADD #2,12(SP) ;POINT TO WHERE THE MAIN PROGRAM WILL RESUME
1061 004444 012137 004570 MOV (R1)+,WORDCNT ;GET THE NUMBER OF WORDS TO BE PRINTED
1062 004450 112105 1S: MOV (R1)+,R5 ;GET THE NUMBER OF CHARACTERS TO BE PRINTED
1063 004452 112100 MOV (R1)+,R0 ;GET THE NUMBER OF SPACES TO PRINT
1064 004454 013104 MOV @12(SP)+,R4 ;COPY THE WORD TO BE CONVERTED
1065 004456 110537 004572 MOV R5,CHRCNT ;COPY THE CHARACTER COUNT
1066 004462 010403 3S: MOV R4,R3 ;COPY THE ARGUMENT WORD AGAIN
1067 004464 042703 177770 BIC #177770,R3 ;ISOLATE THREE BITS TO BE TREATED AS A CHARACTER
1068 004470 062703 000060 ADD #060,R3 ;MAKE AN ASCII CHARACTER OUT OF THEM
1069 004474 110346 MOV R3,-(SP) ;SAVE THAT CHARACTER
1070 004476 006004 ROR R4 ;MOVE THE NEXT THREE BITS INTO PLACE
1071 004500 006204 SR R4 ;MOVE THEM AGAIN
1072 004502 006204 ROR R4 ;AND FINALLY A THIRD TIME
1073 004504 005305 D R5 ;REDUCE THE CHARACTER COUNT. ARE ALL CHARACTERS
1074 ;BUILT?
1075 004506 001365 BNE 3S ;IF NO, GO BUILD THE NEXT ONE.
1076 004510 012703 010622 MOV #MDATA,R3 ;NOW POINT TO WHERE NUMBER WILL BE PRINTED FROM
1077 004514 112623 4S: MOV (SP)+,(R3)+ ;STORE THE CHARACTER, STARTING WITH THE MOST
1078 004516 105337 004572 DECB CHRCNT ;REDUCE COUNT. ARE ALL CHARACTERS TRANSFERRED?
1079 004522 001374 BNE 4S ;IF NO, GO TRANSFER ANOTHER
1080 004524 105700 TSTB R0 ;ARE ANY SPACES TO BE PRINTED?
1081 004526 001404 BEQ 6S ;IF NO, DON'T SET UP ANY
1082 004530 112723 000040 5S: MOV #040,(R3)+ ;ADD A SPACE TO THE OUTPUT BUFFER
1083 004534 105300 DECB R0 ;REDUCE THE COUNT. SHOULD WE PRINT MORE?
1084 004536 001374 BNE 5S ;IF YES, GO ADD ANOTHER SPACE
1085 004540 105013 6S: CLRB (R3) ;TERMINATE THE OUTPUT BUFFER WITH A ZERO
1086 004542 104402 010622 TYPE ,MDATA ;PRINT THE STRING WE JUST BUILT
1087 004546 005337 004570 DEC WORDCNT ;REDUCE THE WORD COUNT. ARE ANY MORE WORDS LEFT?
1088 004552 001336 BNE 1S ;IF YES, GO CONVERT THEM
1089 004554 012605 MOV (SP)+,R5 ;RESTORE R5
1090 004556 012604 MOV (SP)+,R4 ;RESTORE R4
1091 004560 012603 MOV (SP)+,R3 ;RESTORE R3
1092 004562 012601 MOV (SP)+,R1 ;RESTORE R1
1093 004564 012600 MOV (SP)+,R0 ;RESTORE R0
1094 004566 000002 RTI ;RETURN TO THE MAIN PROGRAM
1095 004570 000000 WORDCNT: 0
1096 004572 000 CHRCNT: .BYTE ;NUMBER OF CHARACTERS TO PRINT
1097 004573 000 SPACNT: .BYTE 0 ;NUMBER OF SPACES TO PRINT
1098
1099 004574 000000 BINWRD: 0
1100
1101 ;TRAP DISPATCH SERVICE
1102 ;ARGUMENT OF TRAP IS EXTRACTED
1103 ;AND USED AS OFFSET TO OBTAIN POINTER
1104 ;TO SELECTED SUBROUTINE
1105

```

```

1106
1107 004576 011646 .TRPSR: MOV (SP),-(SP) ;GET PC OF RETURN
1108 004600 162716 000002 SUB #2,(SP) ;=PC OF TRAP
1109 004604 017616 000000 MOV @2(SP), (SP) ;GET TRP
1110 004610 006316 TRPOK: ASL (SP) ;MULTIPLY TRAP ARG BY 2
1111 004612 042716 117001 BIC #117001,(SP) ;CLEAR UNWANTED BITS
1112 004616 062716 001352 ADD #.TRPTAB,(SP) ;POINTER TO SUBROUTINE ADDRESS
1113 004622 017616 000000 MOV @2(SP), (SP) ;SUBROUTINE ADDRESS
1114 004626 000136 JMP @2(SP)+ ;GO TO SUBROUTINE
1115
1116 ;SOFTWARE TIMER FOR RELATIVE TIMING TESTS
1117 -----
1118
1119
1120 004630 005203 CLOCK: INC R3 ;COUNT A TIME TICK
1121 004632 001376 BNE CLOCK ;KEEP GOING UNTIL WE REACH 0
1122 004634 005237 001304 INC TEMP1 ;NOW ADD TO THE OVERALL TIMER
1123 004640 000773 BR CLOCK ;KEEP GOING
1124 ;ERROR HANDLER
1125 -----
1126
1127 004642 032777 010000 174332 .ERROR: BIT #SW12,@SWR ;BELL ON ERROR?
1128 004650 001406 BEQ XBK ;BR IF NO BELL
1129 004652 105777 174332 TSTB @TPCSR ;TTY READY.
1130 004656 100003 BPL XBK ;DON'T WAIT IF TTY NOT READY.
1131 004660 112777 000207 174324 MOVB #207,@TPDBR ;PUSH A BELL AT THE TTY.
1132 004666 032777 020000 174306 XBK: BIT #SW13,@SWR ;DELETE ERROR PRINT OUT?
1133 004674 001111 BNE HALTS ;BR IF NO PRINT OUT WANTED.
1134 004676 021637 001234 CMP (SP),LSTERR ;WAS THIS ERROR FOUND LAST TIME?
1135 004702 001404 BEQ 1$ ;BR IF YES
1136 004704 011637 001234 MOV (SP),LSTERR ;RECORD BEING HERE
1137 004710 105037 001345 CLRB ERRFLG ;PREPARE HEADER
1138 004714 104410 1$: SAVD5 ;SAVE ALL PROC REGISTERS
1139 004716 011605 MOV (SP),R5 ;GET THE PC OF ERROR
1140 004720 162705 000002 SUB #2,R5 ;GET ADDRESS OF TRAP CALL
1141 004724 011504 MOV (R5),R4 ;GET ERROR INSTRUCTION
1142 004726 006304 ASL R4 ;MULT BY TWO
1143 004730 061504 ADD (R5),R4 ;DOUBLE IT
1144 004732 006304 ASL R4 ;MULT AGAIN
1145 004734 042704 137001 BIC #137001,R4 ;CLEAR JUNK
1146 004740 062704 042126 ADD #.ERRTAB,R4 ;GET POINTER
1147 004744 012437 005070 MOV (R4)+,ERRMSG ;GET ERROR MESSAGE
1148 004750 012437 005102 MOV (R4)+,DATAHD ;GET DATA HEADER
1149 004754 011437 005114 MOV (R4),DATABP ;GET DATA TABLE
1150 004760 105737 001345 TSTB ERRFLG ;TYPE HEADER
1151 004764 001403 BEQ TYPMSG ;BR IF YES
1152 004766 005737 005114 TST DATABP ;DOES DATA TABLE EXIST?
1153 004772 001044 BNE TYPDAT ;BR IF YES.
1154 004774 104402 005320 TYPMSG: TYPE ,MCRLF ;TYPE A CARRIAGE RETURN
1155 005000 104402 005320 TYPE ,MCRLF ;AND TYPE ANOTHER
1156 005004 005737 001220 TST LOCK
1157 005010 001402 BEQ 1$
1158 005012 104402 005622 TYPE ,MASTEK ;TYPE TEST NO
1159 005016 104402 005610 TYPE ,MTSTN ;SHOW IT
1160 005022 104413 005222 CNVRT ,XTSTN ;TYPE PC.
1161 005026 104402 005624 TYPE ,MERRPC

```

```

1162 005032 104413 005214 CNVRT ,ERTABO ;SHOW IT
1163 005036 104402 005566 TYPE ,MPASSX ;TYPE PASS NO
1164 005042 104413 003530 CNVRT ,XPASS ;SHOW IT
1165 005046 104402 005320 TYPE ,MCRLF ;GIVE A CR/LF
1166 005052 112737 177777 001345 MOVB #-1,ERRFLG ;NO MORE HEADER UNLESS NO DATA TABLE.
1167 005060 005737 005070 TST EPRMSG ;IS THERE AN ERROR MESSAGE?
1168 005064 001402 BEQ WTBS.FM ;BR IF NO.
1169 005066 104402 TYPE ;TYPE
1170 005070 000000 ERRMSG: 0 ;ERROR MESSAGE
1171 005072 WTBS.FM: TST DATAHD ;DATA HEADER?
1172 005072 005737 005102 BEQ TYPDAT ;BR IF NO
1173 005076 001402 TYPE ;TYPE
1174 005100 104402 DATAHD: 0 ;DATA HEADER
1175 005102 000000 TYPDAT: TST DATABP ;DATA TABLE?
1176 005104 005737 005114 BEQ RESREG ;BR IF NO.
1177 005110 001402 CONVRT ;SHOW
1178 005112 104412 DATABP: 0 ;DATA TABLE
1179 005114 000000 RESREG: RESOS ;RESTORE PROC REGISTERS
1180 005116 104411 HALTS: CMP #SENDAD,2#42 ;CHECK TO SEE IF IN ACT-11 MODE
1181 005120 022737 003470 000042 BEQ 1$ ;IF SO, HANDLE ACCORDINGLY
1182 005126 001403 TST 2$SWR ;HALT ON ERROR?
1183 005130 005777 174046 BPL EXITER ;BR IF NO HALT ON ERROR
1184 005134 100005 1$: PUSHRO ;SAVE RO
1185 005136 010046 MOV 2(SP),RO ;SHOW ERROR PC IN DATA DISPLAY
1186 005140 016600 000002 HALT ;HALT
1187 005144 000000 POPRO ;GET RO
1188 005146 012600 EXITER: INC ERRCNT ;UPDATE ERROR COUNT
1189 005150 005237 001232 BIT #SW08,2$SWR ;GOTO TOP OF TEST?
1190 005154 032777 000400 174020 BNE 1$ ;BR IF YES
1191 005162 001007 BIT #SW10,2$SWR ;GOTO NEXT TEST?
1192 005164 032777 002000 174010 BEQ 2$ ;BR IF NO
1193 005172 001407 MOV NEXT,RETURN ;SET FOR NEXT TEST
1194 005174 013737 001216 001214 1$: MOV #STACK,SP ;RESET SP
1195 005202 012706 001200 JMP 2$RETURN ;GOTO SPECIFIED TEST
1196 005206 000177 174002 2$: RTI ;RETURN
1197 005212 000002 ERTABO: 1
1198 005214 000001 .BYTE 6,2
1199 005216 006 002 SAVPC
1200 005220 001334 XTSTN: 1
1201 005222 000001 .BYTE 3,2
1202 005224 003 002 TSTNO
1203 005226 001226 ;ENTER HERE ON POWER FAILURE
1204 ;-----
1205
1206
1207
1208 005230 .PFAIL: MOV #RESTART,24 ;SET UP FOR POWER UP TRAP
1209 005230 012737 005242 000024 HALT ;HALT ON POWER DOWN NORMAL
1210 005236 000000 BR ;PREVENT ANY FURTHER PROGRESS
1211 005240 000777 ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
1212
1213
1214
1215 005242 RESTART: MOV #.PFAIL,24 ;SET UP FOR POWER FAILURE
1216 005242 012737 005230 000024 MOV #STACK,SP ;RESET THE STACK POINTER
1217 005250 012706 001200

```

D03

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 30

CZTRAB.HED 14-DEC-77 12:19

GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

SEQ 0029

```

1218 005254 005000
1219 005256 005200
1220 005260 001376
1221 005262 104402 005323
1222 005266 104413 005306
1223 005272 105037 001345
1224 005276 005037 001234
1225 005302 000177 173706
1226 005306 000001
1227 005310 003 002
1228 005312 001226
1229 005314 020040 000077
(2) 005320 005015 000
(2) 005323 377 053520 020122
(2) 005361 377 047105 020104
(2) 005404 051377 047125 044516
(2) 005416 050377 047522 051107
(2) 005465 377 047111 052523
(2) 005511 377 042524 052123
(2) 005523 377 047514 045503
(2) 005552 051503 035122 000040
(2) 005560 042526 035103 000040
(2) 005566 040520 051523 051505
(2) 005577 105 051122 051117
(2) 005610 042524 052123 047040
(2) 005622 000052
(2) 005624 041520 020072 000
(2) 005631 377 040515 020120
(2) 005656 051377 051505 052105
(2) 005667 377 051127 052111
(2) 005700 051377 040505 020104
(2) 005710 051777 040520 042503
(2) 005731 377 051105 051501
(2) 005742 020377 042522 047515
(2) 006022 044777 051516 051105
(2) 006100 044777 052116 051105
(2) 006124 044777 052116 051105
(2) 006151 377 047520 042527
(2) 006170 053777 044522 042524
(2) 006232 051377 053505 047111
(2) 006264 044777 046114 043505
(2) 006310 053777 044522 042524
(2) 006342 053777 044522 042524
(2) 006367 377 051127 052111
(2) 006435 377 046120 041501
(2) 006520 047777 020116 044514
(2) 006541 377 051127 052111
(2) 006603 377 047125 052111
(2) 006637 377 043117 020106
(2) 006661 377 042522 044527
(2) 006704 051377 040505 020104
(2) 006727 377 050123 041501
(2) 006771 377 042522 042101
(2) 007015 377 051127 052111
(2) 007037 377 042522 042101
(2) 007060 053777 044522 042524

1S: CLR RO ;READY FOR TIMER
INC RO ;PLUS ONE TO THE TIMER!
BNE 1S ;BR IF MORE TO GO
TYPE ,MPFAIL ;TYPE THE MESSAGE
CNVRT ,PFTAB ;TELL WHAT TEST TO RETURN TO.
CLRB ERRFLG ;START CLEAN
CLR LSTERR
JMP RETURN ;START DOING THAT TEST AGAIN.

PFTAB: 1
.BYTE 3,2
TSTNO
MOM: .ASCIZ / ? /
MCRFL: .ASCIZ <15><12>
MPFAIL: .ASCIZ <377>/PWR FAILED. RESTART AT TEST /
MEPASS: .ASCIZ <377>/END PASS CZTRAB /
MR: .ASCIZ <377>/RUNNING /
MERR2: .ASCIZ <377>/PROGRAM INDICATES NO DEVICES PRESENT./
MERR3: .ASCIZ <377>/INSUFFICIENT DATA!/
MTSTPC: .ASCIZ <377>/TEST PC-/
MLOCK: .ASCIZ <377>/LOCK ON SELECTED TEST/
MCSR: .ASCIZ /CSR: /
MVECX: .ASCIZ /VEC: /
MPASSX: .ASCIZ /PASSES: /
MERRX: .ASCIZ /ERRORS: /
MTSTN: .ASCIZ /TEST NO: /
MASTEK: .ASCIZ /*/
MERRPC: .ASCIZ /PC: /
XHEAD: .ASCIZ <377>/MAP OF TR79 STATUS/<377>
MRSTFL: .ASCIZ <377>/RESET /
MWRTFL: .ASCIZ <377>/WRITE /
MRDFL: .ASCIZ <377>/READ /
MSRVFL: .ASCIZ <377>/SPACE REVERSE /
MERFL: .ASCIZ <377>/ERASE /
MRMVRG: .ASCIZ <377>/ REMOVE WRITE ENABLE RING LOAD & PLACE ONLINE /
MINSRG: .ASCIZ <377>/INSERT WRITE ENABLE RING LOAD & PLACE ONLINE/
MIENFL: .ASCIZ <377>/INTERRUPT ENABLE /
MIDSF: .ASCIZ <377>/INTERRUPT DISABLE /
MPCFL: .ASCIZ <377>/POWER CLEAR /
MWIPLFL: .ASCIZ <377>/WRITE ID BLOCK PAST LOAD POINT /
MRWLPFL: .ASCIZ <377>/REWIND FROM LOAD POINT /
MILFFL: .ASCIZ <377>/ILLEGAL FUNCTION /
MWLPFL: .ASCIZ <377>/WRITE FROM LOAD POINT /
MWEFFL: .ASCIZ <377>/WRITE END OF FILE /
MWELPFL: .ASCIZ <377>/WRITE END OF FILE FROM LOAD POINT /
MUOLN: .ASCIZ <377>/PLACE UNIT ON LINE AFTER TAPE MOTION HAS STOPPED /
MOLEFL: .ASCIZ <377>/ON LINE ERROR /
MWILPFL: .ASCIZ <377>/WRITE ID BLOCK FROM LOAD POINT /
MUNOFLN: .ASCIZ <377>/UNIT DID NOT GO OFF LINE /
MOFLEFL: .ASCIZ <377>/OFF LINE ERROR /
MRWOFL: .ASCIZ <377>/REWIND OFF LINE /
MRDERR: .ASCIZ <377>/READ DATA ERROR /
MSREFFL: .ASCIZ <377>/SPACE REVERSE OVER END OF FILE /
MREFFL: .ASCIZ <377>/READ END OF FILE /
MWTOFL: .ASCIZ <377>/WRITE TIME OUT /
MRTOFL: .ASCIZ <377>/READ TIME OUT /
MWNXFL: .ASCIZ <377>/WRITE NON-EXISTENT MEMORY /

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 31
 CZTRAB.HED 14-DEC-77 12:19

GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

SEQ 0030

(2)	007115	377	040505	041040	MBNXFL: .ASCIZ	<377>/EA BIT 12- NON EXISTENT MEMORY /
(2)	007157	377	054105	042524	MNXBFL: .ASCIZ	<377>/EXTENSION BIT 13 NON EXISTENT MEMORY /
(2)	007227	377	040505	041040	MNXMFL: .ASCIZ	<377>/EA BITS 12 AND 13- NON EXISTENT MEMORY /
(2)	007301	377	042522	042101	MRCFL: .ASCIZ	<377>/READ COUNT /
(2)	007317	377	047105	020104	METFL: .ASCIZ	<377>/END OF TAPE /
(2)	007336	051377	051505	052105	MRSTU: .ASCIZ	<377>/RESET TAPE UNIT /
(2)	007361	377	040515	052516	MMRSFL: .ASCIZ	<377>/MANUAL RESET /
(2)	007401	377	051127	052111	MWPETFL: .ASCIZ	<377>/WRITE PAST END OF TAPE /
(2)	007434	051777	040520	042503	MSRETFL: .ASCIZ	<377>/SPACE REVERSE PAST END OF TAPE /
(2)	007476	051377	040505	020104	MRPETFL: .ASCIZ	<377>/READ PAST END OF TAPE /
(2)	007527	377	051127	052111	MWETFETFL: .ASCIZ	<377>/WRITE END OF FILE PAST END OF TAPE /
(2)	007600	051377	040505	020104	MREFETFL: .ASCIZ	<377>/READ END OF FILE PAST END OF TAPE /
(2)	007650	053777	044522	042524	MWENFL: .ASCIZ	<377>/WRITE ENABLE /
(2)	007670	051777	040520	042503	MSRLPFL: .ASCIZ	<377>/SPACE REVERSE AT LOAD POINT /
(2)	007727	377	050123	041501	MSRIFL: .ASCIZ	<377>/SPACE REVERSE AT ID BLOCK /
(2)	007764	051377	040505	020104	MRIBFL: .ASCIZ	<377>/READ ID BLOCK /
(2)	010005	377	053105	047105	MPARFL: .ASCIZ	<377>/EVEN PARITY /
(2)	010024	042777	042526	020116	MEPSRFL: .ASCIZ	<377>/EVEN PARITY SPACE REVERSE /
(2)	010061	377	042522	042101	MREPFL: .ASCIZ	<377>/READ EVEN PARITY /
(2)	010105	377	047527	042122	MWCF: .ASCIZ	<377>/WORD COUNT /
(2)	010123	377	047125	052111	MUNERR: .ASCIZ	<377>/UNIT HAS ERROR FLAG SET /
(2)	010155	377	040524	042520	MUNRDY: .ASCIZ	<377>/TAPE UNIT NOT READY /
(2)	010203	377	177777	042607	MEOT: .ASCIZ	<377><377><377><207>/END OF TAPE... TAPE REWOUND /
(2)	010244	052377	054524	041040	MTBEX: .ASCIZ	<377>/TTY BUFFER EXCEEDED. RESTART INPUT. /
(2)	010312	040506	046111	051125	MFAIL: .ASCIZ	/FAILURE /
(2)	010323	015	050012	047522	PRHLT: .ASCIZ	<15><12>/PROGRAM HALT - DEVICE FAILED TO AUTOSIZE /<15><12>
(2)	010401	377	052501	047524	SIZE: .ASCIZ	<377>/AUTOSIZING /
(2)	010415	377	051503	020122	CSRER2: .ASCIZ	<377>/CSR ADDRESS /
(2)	010433	377	042526	052103	VECER3: .ASCIZ	<377>/VECTOR ADDRESS /
(2)	010454	041377	020122	042514	BRER1: .ASCIZ	<377>/BR LEVEL INTERRUPT /
(2)	010502	010502			.EVEN	
(2)	010502	000002			XSTATQ: 2	
1230	010504	006	003		.BYTE	6,3
1231	010506	001304			TEMP1	
1232	010510	006	002		.BYTE	6,2
1233	010512	001306			TEMP2	
1234					.EVEN	
1235						
1236					;BUFFERS FOR INPUT-OUTPUT	
1237						
1238	010514	000000			INBUF: 0	
1239		010556			.=. +40	
1240	010556	000000			TTYEND: 0	
1241	010560	000000			TEMP: 0	
1242		010622			.=. +40	
1243	010622	000000			MDATA: 0	
1244		010664			.=. +40	

1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291

```

; THIS ROUTINE SETS UP THE CONTROL ADDRESS FOR THE DIAGNOSTIC
; THIS ROUTINE *MUST*
; BE RUN FIRST BEFORE ENTERING THE DIAGNOSTIC FOR THE
; SETUP NECESSARY.
;
CYCLE: MOV     ACTIVE,RO      ;GET ADDRESS POINTER.
        MOV     (RO)+,TRBASE ;LOAD SYSTEM CTRL. REG
        MOV     (RO)+,TRVCT  ;LOAD VECTOR
        MOV     (RO)+,TRPRT  ;LOAD PRIORITY
        JSR     PC,TRLEV     ;SET UP
        TST     @#42         ;ARE WE UNDER MONITOR CONTROL?
        BNE     4$           ;IF YES, SKIP THIS SETUP
        BIT     #SW01,@SWR   ;IF SW01=1, GET STARTING TEST #
        BEQ     4$           ;BR IF NO TEST IS TO BE INPUTTED
7$:     TYPE     ,MCRLF      ;GET THE STARTING TEST NUMBER
        INSTR   ;CALL THE STRING INPUT ROUTINE
        MTSTN   ;POINTER TO MESSAGE TO BE PRINTED
        PARAM   ;CALL THE OCTAL TO ASCII CONVERT ROUTINE
        1       ;LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
        1000    ;HIGHEST LEGITIMATE VALUE OF EXPECTED RESPONSE
        TSTNO   ;POINTER TO MAP LOCATION TO BE FILLED
        .BYTE   0           ;MASK OF INVALID BITS FOR THIS PARAMETER
        .BYTE   1           ;NUMBER OF PARAMETERS TO STORE
        MOV     #TST1,RO
        CMP     #12737,(RO)
        BNE     6$
        CMP     TSTNO,2(RO)
        BNE     6$
        CMP     #TSTNO,4(RO)
        BNE     6$
        MOV     RO,RETURN    ;SAVE PC
        TYPE     ,MCRLF
        BR      8$
6$:     TST     (RO)+
        CMP     RO,#TLAST+10
        BNE     5$
        TYPE     ,MQM
        BR      7$
4$:     MOV     #TST1,RETURN  ;PREPARE RETURN ADDRESS
8$:     JMP     @RETURN      ;GO START TESTING.

```

G03

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 33
CZTRAB.MED 14-DEC-77 12:19 TR79 DEVICE DIAGNOSTICS.

COPYRIGHT 1977 DIGITAL EQUIP. CORP.

SEQ 0032

```

1292
1293      000062      MAXERR=50.      ;MAXIMUM NUMBER OF DEROR PRINTOUTS
1294      000006      MAXTST=6.      ;MAXIMUM NUMBER OF TESTS
1295      000012      PASRAT=10.     ;PASS RATE PRINT OUT
1296      040000      CORSIZ=40000   ;CORE REQUIRED
1297
1298      ;-----*
1299      ;SBTTL  PRETEST ROUTINE
1300
1301
1302
1303      011044      012737      011044      001250      PRETST:  MOV      #PRETST,TSTPTR      ;SET TEST ADDRESS
1304      011052      004737      014664      JSR      PC,TESTN
1305      011056      013727      001240      MOV      HOLD,(PC)+      ;PICK UP TIME PARAMETER
1306      011062      000000      64$:      .WORD      0      ;USE THIS WORD AS A TIME COUNTER
1307      011064      66$:
1308      011064      005227      000000      INC      #0      ;IF NO,COUNT 1 OF 65535 TICKS
1309      011070      001375      BNE      66$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
1310      011072      005337      011062      DEC      64$      ;HAS THE TOTAL TIME ELAPSED?
1311      011076      001372      BNE      66$      ;IF NO,GO WAIT A LITTLE LONGER
1312
1313      011100      052777      004000      170274      65$:      BIS      #PWRCLR,ATRCR      ;DEVICE MASTER RESET
1314      011106      032777      004000      170266      67$:      BIT      #PWRCLR,ATRCR      ;INSTRUCTION CLEAR ?
1315      011114      001374      BNE      67$      ;IF NO, WAIT FOR IT TO CLEAR
1316      011116      000005      RESET
1317      011120      022777      006026      170254      CMP      #6026,ATRCR      ;CHECK COMMAND REGISTER
1318      011126      001412      BEQ      1$      ;SKIP IF OK
1319      011130      012705      006026      MOV      #6026,R5
1320      011134      017704      170242      MOV      ATRCR,R4
1321      011140      013737      001402      001244      MOV      TRCR,REGIST
1322      011146      104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
1323      011150      104403      005656      TYPEF      ,MRSTFL
1324
1325      011154      1$:
1326      011154      012737      000050      001240      MOV      #50,HOLD      ;SET UP FOR A MASSIVE DELAY
1327      011162      013727      001240      MOV      HOLD,(PC)+      ;PICK UP TIME PARAMETER
1328      011166      000000      68$:      .WORD      0      ;USE THIS WORD AS A TIME COUNTER
1329      011170      70$:
1330      011170      005227      000000      INC      #0      ;IF NO,COUNT 1 OF 65535 TICKS
1331      011174      001375      BNE      70$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
1332      011176      005337      011166      DEC      68$      ;HAS THE TOTAL TIME ELAPSED?
1333      011202      001372      BNE      70$      ;IF NO,GO WAIT A LITTLE LONGER
1334
1335      011204      012737      000010      001240      69$:      MOV      #10,HOLD      ;RESTORE THE NORMAL DELAY FACTOR
1336      011212      022777      002226      170162      CMP      #2226,ATRCR      ;TEST COMMAND REGISTER
1337      011220      001412      BEQ      2$      ;SKIP IF OK
1338      011222      012705      002226      MOV      #2226,R5
1339      011226      017704      170150      MOV      ATRCR,R4
1340      011232      013737      001402      001244      MOV      TRCR,REGIST
1341      011240      104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
1342      011242      104403      005656      TYPEF      ,MRSTFL
1343
1344      011246      2$:
1345      011246      017701      170134      MOV      ATRSR,R1      ;GET STATUS
1346      011252      042701      002247      BIC      #2247,R1      ;CLEAR UNWANTED BITS
1347      011256      005701      TST      R1      ;TEST THE STATUS REGISTER

```

```

1348 011260 001410      BEQ      3$      ;SKIP IF OK
1349 011262 012705 000000      MOV      #0,R5
1350 011266 010104      MOV      R1,R4
1351 011270 013737 001406 001244      MOV      TRSR,REGIST
1352 011276 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
1353 011300 104404      TYPEL
1354
1355 011302      3$:      TST      ERR      ;TEST THE DEROR FLAG
1356 011302 005737 001252      BEQ      71$      ;SKIP IF NO DEROR
1357 C.1306 001406      BIT      #BIT14,DSWR      ;TEST LOOP BIT
1358 011310 032777 040000 167664      BEQ      71$      ;BRANCH IF CLEAR
1359 011316 001402      JMP      PRETST      ;JUMP IF SET
1360 011320 000137 011044      CLR      ERR      ;CLEAR THE DEROR FLAG
1361 011324 005037 001252      JSR      PC,FIXBUF
1362 011330 004737 030232
1363
1364 011334 000207      RTS      PC
1365
1366      ;***** TEST 1 *****
1367      ;* TEST GROUP 1
1368      ;*****
1369      ;*****
1370      ;* TEST 1
1371      ;*
1372      ;*****
1373
1374 011336 012737 000001 001226      TST1: MOV      #1,DSWSTNO      ;LOAD THE NUMBER OF THIS TEST IN THE INDICATOR
1375 011344 012737 011516 001216      MOV      #TST2,NEXT      ;POINT TO THE START OF THE NEXT TEST
1376 011352 032777 000010 167622      BIT      #10,DSWR      ;SKIP??
1377 011360 001002      BNE      2$      ;NO
1378 011362 000137 011456      JMP      END1
1379 011366 012737 011460 001302      2$: MOV      #TABLE1,TC      ;SET TEST CONTROL INDEX
1380 011374 012706 001200      MOV      #STACK,SP      ;RESET THE STACK
1381 C.1400
1382 011400 013737 001302 001250      1$: MOV      TC,TSTPTR
1383 011406 013705 001302      MOV      TC,R5      ;SET DEROR SUBTEST ADDRESS
1384 011412 062737 000002 001302      ADD      #2,TC      ;POINT TO NEXT TEST
1385 011420 004775 000000      JSR      PC,@(R5)      ;PERFORM THE TEST
1386 011424 000240      NOP      ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
1387      ;EQUIPMENT STATUS OR PROGRAM OPERATION
1388 011426 005737 001252      TST      ERR      ;TEST THE DEROR FLAG
1389 011432 001406      BEQ      64$      ;SKIP IF NO DEROR
1390 011434 032777 040000 167540      BIT      #BIT14,DSWR      ;TEST LOOP BIT
1391 011442 001402      BEQ      64$      ;BRANCH IF CLEAR
1392 011444 000137 011400      JMP      1$      ;JUMP IF SET
1393 011450 005037 001252      64$: CLR      ERR      ;CLEAR THE DEROR FLAG
1394 011454 000751      BR      1$      ;DO NEXT TEST
1395 011456 104400      END1: SCOPE      ;PASS COMPLETED
1396
1397
1398
1399
1400      ;THIS TABLE CONTAINS THE SUBTESTS TO BE PERFORMED BY TEST 1
1401 011460 012564      TABLE1: TESTB      ;REMOVE WRITE RING
1402 011462 024014      RUNAS      ;TRY TO WRITE WITHOUT RING
1403 011464 012646      TESTC      ;INSERT WRITE RING

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 35
 CZTRAB.HED 14-DEC-77 12:19 PRETEST ROUTINE

SEQ 0034

1404	011466	021752	TESTAJ	: DELAY 10 SECONDS
1405	011470	015072	TESTP	: TAKE UNIT OFF LINE
1406	011472	022024	TESTAK	: TELL OPERATOR TO RESET
1407	011474	013626	TESTI	: REWIND THE TAPE
1408	011476	022260	TESTAL	: WRITE PAST EOT
1409	011500	022616	TESTAM	: SPACE REVERSE PAST EOT
1410	011502	023024	TESTAN	: READ PAST EOT
1411	011504	023172	TESTAO	: WRITE EOF PAST EOT
1412	011506	023400	TESTAP	: SPACE REVERSE OVER EOF PAST EOT
1413	011510	023606	TESTAQ	: READ EOF PAST EOT
1414	011512	015214	TESTQ	: REWIND AND GO OFF LINE
1415	011514	011456	END1	

***** TEST 2 *****
 * TEST GROUP 2 *

1420			*****	
1421			*****	
1422			TEST 2	
1423			*****	
1424			*****	
1425	011516	012737	000002	001226
1426	011524	012737	011656	001216
1427	011532	012737	011624	001302
1428	011540	012706	001200	
1429	011544			
1430	011544	013737	001302	001250
1431	011552	013705	001302	
1432	011556	062737	000002	001302
1433	011564	004775	000000	
1434	011570	000240		
1435				
1436	011572	005737	001252	
1437	011576	001406		
1438	011600	032777	04000C	167374
1439	011606	001402		
1440	011610	000137	011544	
1441	011614	005037	001252	
1442	011620	000751		
1443	011622	104400		
1444				
1445				
1446				
1447				
1448				
1449	011624	013626		
1450	011626	012646		
1451	011630	012760		
1452	011632	013146		
1453	011634	013202		
1454	011636	014034		
1455	011640	014270		
1456	011642	014550		
1457	011644	024314		
1458	011646	020070		
1459	011650	024522		

; THIS TABLE CONTAINS THE SUBTESTS TO BE PERFORMED BY TEST 2

TABLE2:	TESTI	: REWIND THE TAPE
	TESTC	: INSURE WRITE RING
	TESTD	: UNIBUS INTENTION BIT TEST
	TESTE	: INTERRUPT BIT SET TEST
	TESTF	: INTERRUPT BIT CLEAR TEST
	TESTJ	: ILLEGAL COMMAND TEST
	TESTK	: WRITE FROM LOAD POINT
	TESTM	: WRITE EOF FROM LOAD POINT
	TESTAT	: SPACE REVERSE AT LOAD POINT
	TESTAB	: NXM TEST
	TESTAU	: BUSS ADDRESS BIT TEST

```

1460 011652 026144 TESTBA ;WORD COUNT BIT TEST
1461 011654 011622 END2
1462
1463
1464 ;***** TEST 3 *****
1465 ;* TEST GROUP 3
1466 ;*****
1467 ;*****
1468 ;* TEST 3
1469 ;*
1470 ;*****
1471 ;*****
1472 011656 012737 000203 001226 TST3: MOV #3,2#TSTNO ;LOAD THE NUMBER OF THIS TEST IN THE INDICATOR
1473 011664 012737 012030 001216 MOV #TST4,NEXT ;POINT TO THE START OF THE NEXT TEST
1474 011672 012737 011764 001302 MOV #TABLE3,TC ;SET TEST CONTROL INDEX
1475 011700 012706 001200 MOV #STACK,SP ;RESET THE STACK
1476 011704
1477 011704 013737 001302 001250 15: MOV TC,TSTPTR ;SET DEROR SUBTEST ADDRESS
1478 011712 013705 001302 MOV TC,R5 ;POINT TO NEXT TEST
1479 011716 062737 000002 001302 ADD #2,TC ;PERFORM THE TEST
1480 011724 004775 000000 JSR PC,2(R5) ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
1481 011730 000240 NOP ;EQUIPMENT STATUS OR PROGRAM OPERATION
1482
1483 011732 005737 001252 TST ERR ;TEST THE DEROR FLAG
1484 011736 001406 BEQ 64$ ;SKIP IF NO DEROR
1485 011740 032777 040000 167234 BIT #BIT14,2SWR ;TEST LOOP BIT
1486 011746 001402 BEQ 64$ ;BRANCH IF CLEAR
1487 011750 000137 011704 JMP 1$ ;JUMP IF SET
1488 011754 005037 001252 64$: CLR ERR ;CLEAR THE DEROR FLAG
1489 011760 000751 BR 1$ ;DO NEXT TEST
1490 011762 104400 END3: SCOPE ;PASS COMPLETED
1491
1492
1493
1494 ;THIS TABLE CONTAINS THE SUBTESTS TO BE PERFORMED BY TEST 3
1495
1496 011764 012646 TABLE3: TESTC ;INSURE WRITE ENABLE RING
1497 011766 013626 TESTI ;REWIND THE TAPE
1498 011770 014764 TESTO ;WRITE ID BLOCK
1499 011772 013500 TESTH ;ILLEGAL ID BLOCK TEST
1500 011774 024562 TESTAV ;SPACE REVERSE OVER ID BLOCK
1501 011776 024772 TESTAW ;READ ID BLOCK
1502 012000 014414 RUNL ;WRITE EOF MARKS
1503 012002 015706 RUNU ;SPACE REVERSE OVER EOF MARKS
1504 012004 016116 RUNV ;READ EOF MARKS
1505 012006 016322 TESTW ;TIME OUT TEST
1506 012010 020350 TESTAC ;READ COUNT
1507 012012 020610 TESTAD ;READ COUNT
1508 012014 021050 TESTAE ;READ COUNT
1509 012016 021242 TESTAF ;SPACE COUNT
1510 012020 025236 TESTAX ;WRITE EVEN PARITY
1511 012022 025500 TESTAY ;SPACE REVERSE OVER EVEN PARITY
1512 012024 025732 TESTAZ ;READ EVEN PARITY
1513 012026 011762 END3
1514
1515 ;***** TEST 4 *****

```

```

1516 ;* TEST GROUP 4
1517 ;*****
1518 ;*****
1519 ;*
1520 ;* TEST 4
1521 ;*
1522 ;*****
1523 012030 012737 000004 001226 TST4: MOV #4,2#TSTNO ;LOAD THE NUMBER OF THIS TEST IN THE INDICATOR
1524 012036 012737 012146 001216 MOV #TST5,NEXT ;POINT TO THE START OF THE NEXT TEST
1525 012044 012737 012136 001302 MOV #TABLE4,TC ;SET TEST CONTROL INDEX
1526 012052 012706 001200 MOV #STACK,SP ;RESET THE STACK
1527 012056
1528 012056 013737 001302 001250 AAA: MOV TC,TSTPTR ;SET DEROR SUBTEST ADDRESS
1529 012064 013705 001302 MOV TC,R5 ;POINT TO NEXT TEST
1530 012070 062737 000002 001302 ADD #2,TC ;PERFORM THE TEST
1531 012076 004775 000000 JSR PC,2(R5)
1532 012102
1533 012102 000240 RTA: NOP ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
1534 ;EQUIPMENT STATUS OR PROGRAM OPERATION
1535 012104 005737 001252 TST ERR ;TEST THE DEROR FLAG
1536 012110 001406 BEQ 64$ ;SKIP IF NO DEROR
1537 012112 032777 040000 167062 BIT #BIT14,2SWR ;TEST LOOP BIT
1538 012120 001402 BEQ 64$ ;BRANCH IF CLEAR
1539 012122 000137 012056 JMP AAA ;JUMP IF SET
1540 012126 005037 001252 64$: CLR ERR ;CLEAR THE DEROR FLAG
1541 012132 000751 BR AAA ;DO NEXT TEST
1542 012134 104400 END4: SCOPE ;PASS COMPLETED
1543
1544
1545 ;THIS TABLE CONTAINS THE SUBTESTS TO BE PERFORMED BY TEST 4
1546
1547 TABLE4: TESTC ;INSURE WRITE RING
1548 012136 012646 TESTO ;REWIND AND WRITE ID BLOCK
1549 012140 014764 TESTXX ;WRITE, ERASE,SPACE REV, READ AND TEST DATA
1550 012142 027210
1551 012144 012134
1552
1553 ;***** TEST 5 *****
1554 ;* TEST GROUP 5
1555 ;*****
1556 ;*****
1557 ;*
1558 ;* TEST 5
1559 ;*
1560 ;*****
1561 012146 012737 000005 001226 TST5: MOV #5,2#TSTNO ;LOAD THE NUMBER OF THIS TEST IN THE INDICATOR
1562 012154 012737 012316 001216 MOV #TST6,NEXT ;POINT TO THE START OF THE NEXT TEST
1563 012162 032777 000040 167012 BIT #BIT5,2SWR ;SKIP THIS TEST?
1564 012170 001002 BNE 1$ ;NO
1565 012172 000137 012266 JMP ENDS ;YES
1566 012176 012737 012306 001302 1$: MOV #TABLE5,TC ;SET TEST CONTROL INDEX
1567 012204 012706 001200 MOV #STACK,SP ;RESET THE STACK
1568 012210
1569 012210 013737 001302 001250 BBB: MOV TC,TSTPTR ;SET DEROR SUBTEST ADDRESS
1570 012216 013705 001302 MOV TC,R5 ;POINT TO NEXT TEST
1571 012222 062737 000002 001302 ADD #2,TC

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 38
 CZTRAB.HED 14-DEC-77 12:19 PRETEST ROUTINE

SEQ 0037

```

1572 012230 004775 000000      JSR      PC,2(R5)      ;PERFORM THE TEST
1573 012234      RTB:      NOP                      ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
1574 012234 000240                      ;EQUIPMENT STATUS OR PROGRAM OPERATION
1575      TST      ERR                      ;TEST THE DEROR FLAG
1576 012236 005737 001252      BEQ      64$          ;SKIP IF NO DEROR
1577 012242 001406      BIT      #BIT14,2SWR        ;TEST LOOP BIT
1578 012244 032777 040000 166730      BEQ      64$          ;BRANCH IF CLEAR
1579 012252 001402      JMP      BBB              ;JUMP IF SET
1580 012254 000137 012210      CLR      ERR          ;CLEAR THE DEROR FLAG
1581 012260 005037 001252      BR       BBB          ;DO NEXT TEST
1582 012264 000751      ENDS:
1583 012266      BIT      #10,2SWR              ;SKIP NEXT TEST?
1584 012266 032777 000010 166706      BNE      1$          ;NO
1585 012274 001003      MOV      #.EOP,NEXT        ;SET EOP
1586 012276 012737 003364 001216      SCOPE      ;PASS COMPLETED
1587 012304 104400
1588
1589      ;THIS TABLE CONTAINS THE SUBTESTS TO BE PERFORMED BY TEST 5
1590
1591 012306 013626      TABLES: TESTI      ;REWIND THE TAPE
1592 012310 024772      TESTAW      ;READ THE ID BLOCK
1593 012312 027436      TESTXY      ;READ THE TAPE
1594 012314 012266      ENDS
1595
1596      ;***** TEST 6 *****
1597      ;* TEST GROUP 6
1598      ;*****
1599
1600      ;*****
1601      ;* TEST 6
1602      ;*****
1603
1604 012316 012737 000006 001226      TST6: MOV      #6,2TSTNO      ;LOAD THE NUMBER OF THIS TEST IN THE INDICATOR
1605 012324 012737 003364 001216      MOV      #.EOP,NEXT        ;POINT TO THE END-OF-PASS HANDLER
1606 012332 012737 012520 001302      MOV      #TABLE6,TC      ;SET TEST CONTROL INDEX
1607 012340 012706 001200      MOV      #STACK,SP      ;RESET THE STACK
1608 012344
1609 012344 104405      1$:      INSTR                      ;CALL THE STRING INPUT ROUTINE
1610 012346 012456      MESTADD          ;POINTER TO MESSAGE TO BE PRINTED
1611 012350 104407      PARAM                      ;CALL THE OCTAL TO ASCII CONVERT ROUTINE
1612 012352 011336      TST1                      ;LOWEST LEGITIMATE VALUE OF EXPECTED RESPONSE
1613 012354 027210      TESTXX          ;HIGHEST LEGITIMATE VALUE OF EXPECTED REPOSENSE
1614 012356 012520      TABLE6          ;POINTER TO MAP LOCATION TO BE FILLED
1615 012360 001      .BYTE      1              ;MASK OF INVALID BITS FOR THIS PARAMETER
1616 012361 001      .BYTE      1              ;NUMBER OF PARAMETERS TO STORE
1617 012362 012737 012520 001302      2$: MOV      #TABLE6,TC      ;SET TEST CONTROL INDEX
1618 012370 017737 166706 001250      MOV      2TC,TSTPTR        ;SET DEROR SUBTEST ADDRESS
1619 012376 013705 001302      MOV      TC,R5
1620 012402 062737 000002 001302      ADD      #2,TC
1621 012410 004775 000000      JSR      PC,2(R5)      ;PERFORM THE TEST
1622 012414 000240      NOP                      ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
1623
1624 012416 012737 177777 001252      MOV      #-1,ERR          ;EQUIPMENT STATUS OR PROGRAM OPERATION
1625 012424 005737 001252      TST      ERR          ;SET DEROR FLAG
1626 012430 001406      BEQ      64$          ;TEST THE DEROR FLAG
1627 012432 032777 040000 166542      BIT      #BIT14,2SWR        ;SKIP IF NO DEROR
                                ;TEST LOOP BIT

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 39
 CZTRAB.HED 14-DEC-77 12:19 PRETEST ROUTINE

SEQ 0038

```

1629 012440 001402          BEQ 64$          ;BRANCH IF CLEAR
1629 012442 000137 012362    JMP 2$          ;JUMP IF SET
1630 012446 005037 001252    64$: CLR ERR      ;CLEAR THE DEROR FLAG
1631 012452 000734          BR 1$           ;DO NEXT TEST
1632 012454 104400          END6: SCOPE      ;PASS COMPLETED
1633
1634 012456 042777 052116 051105 MESTADD: .ASCIZ <377>/ENTER STARTING ADDRESS OF TEST /
1635 012464 051440 040524 052122
1636 012472 047111 020107 042101
1637 012500 05_104 051505 020123
1638 012506 043117 052040 051505
1639 012514 020124 000
1640 012520
1641 .EVEN
1642
1643 ;THIS TABLE CONTAINS THE SUBTESTS TO BE PERFORMED BY TEST 6
1644
1645 012520 000000 TABLE6: 0
1646 012522 012454 END6
1647
1648 ;-----*
1649
1650 .SBTTL SUB-TEST ROUTINES
1651
1652 ;THIS ROUTINE WAITS FOR TAPE UNIT TO GET READY OR DEROR
1653 ;OR TIMES OUT AND EXITS WHEN IT DOES
1654
1655 012524 005001 TESTA: CLR R1 ;CLEAR TIMER
1656 012526 1$: INC R1 ;COUNT UP
1657 012526 005201 BEQ 2$ ;SKIP IF ZERO
1658 012530 001414 BIT #BIT15,@TRCR ;SKIP IF SET
1659 012532 032777 100000 166642 BNE 2$
1660 012540 001010 BIT #BIT10,@TRCR ;NO-BRANCH
1661 012542 032777 002000 166632 BEQ 1$
1662 012550 001766 BIT #BIT7,@TRCR ;NO-BRANCH
1663 012552 032777 000200 166622 BEQ 1$
1664 012560 001762 2$: RTS PC
1665 012562
1666 012562 000207
1667
1668 ;-----*
1669 ;THIS ROUTINE INSURES THE WRITE ENABLE RING
1670 ;REMOVED AND EXITS WHEN IT IS
1671
1672 012564 TESTB: BIT #BIT2,@TRSR
1673 012564 032777 000004 166614 BNE 2$ ;SKIP IF ONE
1674 012572 001015 TYPE ,MRMVRG
1675 012574 104402 005742 1$: BIT #BIT2,@TRSR ;SKIP IF ONE
1676 012600 BNE 2$
1677 012600 032777 000004 166600 INC TEMP
1678 012606 001007 010560 MOV #207,@TPDBR ;RING THE BELL
1679 012610 005237 64$: BR 1$
1680 012614 001003
1681 012616 012777 000207 166366
1682 012624
1683 012624 000765

```

```

1684 012626 052777 004000 166546 2$: BIS #PWRCLR,ATCR ;DEVICE MASTER RESET
1685 012626 032777 004000 166540 6$: BIT #PWRCLR,ATCR ;INSTRUCTION CLEAR ?
1686 012634 001374 000207 65$: BNE 65$ ;IF NO, WAIT FOR IT TO CLEAR
1687 012642 000207 65$: RTS PC
1688 012644 000207
1689
1690

```

```

;-----*
; THIS ROUTINE INSURES THE WRITE ENABLE RING
; IS INSERTED AND EXITS WHEN IT IS

```

```

1694 012646 032777 000004 166532 TESTC: BIT #BIT2,ATRSR
1695 012646 001420 BEQ 2$ ;SKIP IF ZERO
1696 012654 104402 006022 TYPE ,MINSRG
1697 012662 005237 010560 1$: INC TEMP
1698 012662 001003 000207 166314 64$: MOV #207,ATPDBR ;RING THE BELL
1700 012670 032777 000004 166502 BIT #BIT2,ATRSR
1701 012670 001366 002000 166466 1$: BNE 1$ ;LOOP TILL ZERO
1702 012676 032777 000004 166502 BIT #BIT10,ATCR
1703 012704 001366 002000 166466 1$: BEQ 1$ ;BRANCH IF CLEAR
1704 012706 032777 000004 166502 2$: BIS #PWRCLR,ATCR ;DEVICE MASTER RESET
1705 012714 001762 004000 166456 65$: BIT #PWRCLR,ATCR ;INSTRUCTION CLEAR ?
1706 012716 052777 004000 166450 65$: BNE 65$ ;IF NO, WAIT FOR IT TO CLEAR
1707 012724 001374 014664 JSR PC,TESTN
1708 012732 004737 004000 166434 66$: BIS #PWRCLR,ATCR ;DEVICE MASTER RESET
1709 012740 052777 004000 166426 66$: BIT #PWRCLR,ATCR ;INSTRUCTION CLEAR ?
1710 012746 001374 000207 66$: BNE 66$ ;IF NO, WAIT FOR IT TO CLEAR
1711 012754 000207 66$: RTS PC
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721

```

```

;-----*
; THIS ROUTINE TESTS THE READ WRITE ABILITY OF THE
; UNIBUS EXTENSION ADDRESS BITS

```

```

1722 012760 052777 004000 166414 TESTD: BIS #PWRCLR,ATCR ;DEVICE MASTER RESET
1723 012760 032777 004000 166406 64$: BIT #PWRCLR,ATCR ;INSTRUCTION CLEAR ?
1724 012766 001374 001402 001244 64$: BNE 64$ ;IF NO, WAIT FOR IT TO CLEAR
1725 012774 013737 010000 166370 MOV TRCR,REGIST ;INDICATE THE ERROR REGIS - ?
1726 012776 052777 010000 166362 BIS #010000,ATCR ;SET UNIBUS EXTENSION BIT 16
1727 013004 032777 010000 166362 BIT #010000,ATCR ;TEST UNIBUS EXTENSION BIT 16
1728 013012 001C01 1$ ;SKIP IF OK
1729 013020 104000 ERROR ;ERROR-BIT WAS NOT CORRECT
1730 013022
1731
1732 013024 1$:
1733
1734 013024 042777 010000 166350 BIC #010000,ATCR ;CLEAR THE BIT
1735 013032 032777 010000 166342 BIT #010000,ATCR ;TEST THE BIT
1736 013040 001401 2$ ;SKIP IF CLEAR
1737 013042 104000 ERROR ;ERROR-BIT WAS NOT CORRECT
1738
1739 013044 2$:

```

```

1740
1741
1742 013044 052777 020000 166330      BIS      #020000,2TRCR    ;SET UNIBUS EXTENSION BIT 17
1743 013052 032777 020000 166322      BIT      #020000,2TRCR    ;TEST UNIBUS EXTENSION BIT 17
1744 013060 001001                      BNE      3$              ;SKIP IF OK
1745 013062 104000                      ERROR                     ;ERROR-BIT WAS NOT CORRECT
1746
1747 013064                      3$:
1748
1749
1750 013064 042777 020000 166310      BIC      #020000,2TRCR    ;CLEAR THE BIT
1751 013072 032777 020000 166302      BIT      #20000,2TRCR    ;TEST THE BIT
1752 013100 001401                      BEQ      4$              ;SKIP IF OK
1753 013102 104000                      ERROR                     ;ERROR-BIT WAS NOT CORRECT
1754
1755 013104                      4$:
1756
1757
1758 013104 052777 030000 166270      BIS      #30000,2TRCR    ;SET BOTH EA BITS
1759 013112 032777 030000 166262      BIT      #30000,2TRCR    ;TEST BITS ON
1760 013120 001001                      BNE      5$              ;
1761 013122 104000                      ERROR                     ;
1762
1763 013124                      5$:
1764
1765 013124 042777 030000 166250      BIC      #30000,2TRCR    ;CLEAR EA BITS
1766 013132 032777 030000 166242      BIT      #30000,2TRCR    ;MAKE SURE
1767 013140 001401                      BEQ      6$              ;
1768 013142 104000                      ERROR                     ;
1769
1770 013144                      6$:
1771 013144 000207                      RTS      PC
1772
1773 ;+-----+
1774
1775 ;THIS ROUTINE SETS THE INTERRUPT BIT AND TESTS IT
1776 ;TO INSURE IT IS SET
1777
1778 013146 012737 000340 177776  TESTE:  MOV      #LEVEL7,PS    ;SET PRIORITY TO 7
1779 013154 052777 000100 166220      BIS      #BIT6,2TRCR
1780 013162 032777 000100 166212      BIT      #BIT6,2TRCR
1781 013170 001003                      BNE      1$              ;SKIP IF SET
1782 013172 104000                      ERROR                     ;ERROR-BIT WAS NOT CORRECT
1783 013174 104403 006100                      TYPEF      ,MIENFL
1784
1785 013200                      1$:
1786 013200 000207                      RTS      PC
1787
1788 ;+-----+
1789
1790 ;THIS ROUTINE CLEARS THE INTERRUPT ENABLE AND TESTS
1791 ;TO INSURE IT IS CLEAR
1792
1793 013202                      TESTF:
1794 013202 042777 000100 166172      BIC      #BIT6,2TRCR
1795 013210 032777 000100 166164      BIT      #BIT6,2TRCR

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 42
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0041

```

1796 013216 001403      BEQ      1$      ;SKIP IF ZERO
1797 013220 104000      ERROR      ;ERROR-BIT WAS NOT CORRECT
1798 013222 104403 006124 TYPEF    ,MIDSFL
1799
1800 013226      1$:      CLR      PS      ;SET PROCESSOR PRIORITY TO 00
1801 013226 005037 177776      PC
1802 013232 000207
1803
1804 ;-----
1805 ; THIS ROUTINE TESTS THE FUNCTION OF THE POWER CLEAR
1806
1807
1808 013234 000005      TESTG: RESET
1809 013236 012777 000026 166136 MOV      #26,2TRCR      ;SET COMMAND REGISTER
1810 013244 052777 004000 166130 BIS      #BIT11,2TRCR
1811 013252 012701 177300 MOV      #-500,R1      ;SET DELAY
1812 013256      1$:      INC      R1      ;DELAY
1813 013256 005201      BNE      1$      ;CONTINUE
1814 013260 001376
1815
1816 013262 032777 004000 166112 BIT      #BIT11,2TRCR
1817 013270 001001      BNE      2$      ;SKIP IF SET
1818 013272 104000      ERROR      ;ERROR-BIT WAS NOT CORRECT
1819
1820 013274      2$:
1821
1822 013274 013727 001240      MOV      HOLD,(PC)+      ;PICK UP TIME PARAMETER
1823 013300 000000      64$:      .WORD      0      ;USE THIS WORD AS A TIME COUNTER
1824 013302      66$:
1825 013302 005227 000000      INC      #0      ;IF NO,COUNT 1 OF 65535 TICKS
1826 013306 001375      BNE      66$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
1827 013310 005337 013300      DEC      64$      ;HAS THE TOTAL TIME ELAPSED?
1828 013314 001372      BNE      66$      ;IF NO,GO WAIT A LITTLE LONGER
1829 013316      65$:
1830 013316 013727 001240      MOV      HOLD,(PC)+      ;PICK UP TIME PARAMETER
1831 013322 000000      67$:      .WORD      0      ;USE THIS WORD AS A TIME COUNTER
1832 013324      69$:
1833 013324 005227 000000      INC      #0      ;IF NO,COUNT 1 OF 65535 TICKS
1834 013330 001375      BNE      69$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
1835 013332 005337 013322      DEC      67$      ;HAS THE TOTAL TIME ELAPSED?
1836 013336 001372      BNE      69$      ;IF NO,GO WAIT A LITTLE LONGER
1837 013340      68$:
1838 013340      3$:
1839 013340 032777 004000 166034 BIT      #BIT11,2TRCR
1840 013346 001403      BEQ      4$      ;SKIP IF CLEAR
1841 013350 005237 010560      INC      TEMP
1842 013354 001371      BNE      3$      ;DELAY
1843
1844 013356      4$:
1845
1846 013356 032777 004000 166016 BIT      #BIT11,2TRCR
1847 013364 001401      BEQ      5$      ;SKIP IF CLEAR
1848 013366 104000      ERROR      ;ERROR-BIT WAS NOT CORRECT
1849
1850 013370      5$:
1851

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 43
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0042

```

1852 013370 032777 002000 166010      BIT      #BIT10,ATRSR
1853 013376 001001      BNE      6$      ;SKIP IF ON LINE
1854 013400 000207      RTS      PC
1855 013402      6$:
1856
1857 013402 017701 165774      MOV      ATCR,R1      ;GET COMMAND REGISTER
1858 013406 042701 000036      BIC      #00036,R1      ;CLEAR UNWANTED BITS
1859 013412 022701 002200      CMP      #2200,R1      ;TEST COMMAND REGISTER
1860 013416 001411      BEQ      7$      ;SKIP IF OK
1861 013420 012705 002200      MOV      #2200,R5
1862 013424 010104      MOV      R1,R4
1863 013426 013737 001402 001244      MOV      ATCR,REGIST
1864 013434 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
1865 013436 104403 006151      TYPEF      ,MPCFL
1866
1867 013442      7$:
1868
1869 013442 017701 165740      MOV      ATRSR,R1      ;GET STATUS REGISTER
1870 013446 042701 002247      BIC      #2247,R1      ;CLEAR UNWANTED BITS
1871 013452 005701      TST      R1      ;TEST STATUS REGISTER
1872 013454 001410      BEQ      8$      ;SKIP IF OK
1873 013456 012705 002247      MOV      #2247,R5
1874 013462 010104      MOV      R1,R4
1875 013464 013737 001406 001244      MOV      TRSR,REGIST
1876 013472 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
1877 013474 104404      TYPEL
1878
1879 013476      8$:
1880
1881 013476 000207      RTS      PC
1882
1883
1884
1885
1886
1887 013500
1888 013500 004737 014764      TESTH:  JSR      PC,TESTO
1889 013504 004737 026610      JSR      PC,TESTCK
1890
1891 013510 004537 027546      JSR      R5,TRYIT
1892 013514 000432      .WORD      WIDB      ;FUNCTION
1893 013516 000000      .WORD      -0      ;WORD COUNT
1894 013520 000000      .WORD      0      ;BUS ADDRESS
1895
1896 013522 022777 142632 165652      CMP      #142632,ATCR      ;TEST COMMAND REGISTER
1897 013530 001412      BEQ      1$      ;SKIP IF OK
1898 013532 012705 142632      MOV      #142632,R5
1899 013536 017704 165640      MOV      ATCR,R4
1900 013542 013737 001402 001244      MOV      ATCR,REGIST
1901 013550 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
1902 013552 104403 006170      TYPEF      ,MWIPLFL
1903
1904 013556      1$:
1905
1906 013556 022777 002000 165622      CMP      #2000,ATRSR      ;TEST THE STATUS REGISTER
1907 013564 001410      BEQ      2$      ;SKIP IF OK

```

```

1908 013566 012705 002000      MOV    #2000,R5
1909 013572 017704 165610      MOV    @TRSR,R4
1910 013576 013737 001406 001244  MOV    TRSR,REGIST
1911 013604 104001      ERROR    1      ; INCORRECT REGISTER MATCHUP
1912
1913 013606      2$:
1914
1915 013606 052777 004000 165566      BIS    #PWRCLR,@TRCR      ; DEVICE MASTER RESET
1916 013614 032777 004000 165560 64$:      BIT    #PWRCLR,@TRCR      ; INSTRUCTION CLEAR ?
1917 013622 001374      BNE      64$      ; IF NO, WAIT FOR IT TO CLEAR
1918 013624 000207      RTS      PC
1919
1920      ; +-----+
1921      ; THIS ROUTINE TESTS THE FUNCTION OF REWINDING
1922      ; WHILE AT LOAD POINT THIS IS THE MAIN REWIND ROUTINE
1923
1924 013626      TESTI:
1925 013626 052777 004000 165546      BIS    #PWRCLR,@TRCR      ; DEVICE MASTER RESET
1926 013634 032777 004000 165540 64$:      BIT    #PWRCLR,@TRCR      ; INSTRUCTION CLEAR ?
1927 013642 001374      BNE      64$      ; IF NO, WAIT FOR IT TO CLEAR
1928 013644 004537 027546      JSR     R5,TRYIT
1929 013650 001020      .WORD    REWIND      ; FUNCTION
1930 013652 000000      .WORD    -0      ; WORD COUNT
1931 013654 000000      .WORD    0      ; BUS ADDRESS
1932 013656      A=:
1933 013656 032777 000002 165522      BIT    #BIT1,@TRSR
1934 013664 001374      BNE      A      ; BRANCH IF SET
1935 013666 032777 000040 165512      BIT    #BITS,@TRSR
1936 013674 001770      BEQ     A      ; BRANCH IF CLEAR
1937 013676 052777 004000 165476      BIS    #PWRCLR,@TRCR      ; DEVICE MASTER RESET
1938 013704 032777 004000 165470 65$:      BIT    #PWRCLR,@TRCR      ; INSTRUCTION CLEAR ?
1939 013712 001374      BNE      65$      ; IF NO, WAIT FOR IT TO CLEAR
1940 013714 004537 027546      JSR     R5,TRYIT
1941 013720 001020      .WORD    REWIND      ; FUNCTION
1942 013722 000000      .WORD    -0      ; WORD COUNT
1943 013724 000000      .WORD    0      ; BUS ADDRESS
1944
1945 013726 022777 003220 165446      CMP     #3220,@TRCR      ; TEST COMMAND REGISTER
1946 013734 001412      BEQ     A$0
1947 013736 012705 003220      MOV     #3220,R5
1948 013742 017704 165434      MOV     @TRSR,R4
1949 013746 013737 001402 001244      MOV     TRSR,REGIST
1950 013754 104001      ERROR    1      ; INCORRECT REGISTER MATCHUP
1951 013756 104403 006232      TYPEF    ,MRWLPFL
1952
1953 013762 000001      A$0:      STAGAS=STAGAS+1
1954
1955 013762 022777 002040 165416      CMP     #2040,@TRSR      ; TEST STATUS REGISTER
1956 013770 001411      BEQ     A$1
1957 013772 012705 002040      MOV     #2040,R5
1958 013776 017704 165404      MOV     @TRSR,R4
1959 014002 013737 001406 001244      MOV     TRSR,REGIST
1960 014010 104001      ERROR    1      ; INCORRECT REGISTER MATCHUP
1961 014012 104404      TYPEL
1962
1963 014014 000002      A$1:      STAGAS=STAGAS+1

```

```

1964
1965 014014 052777 004000 165360      BIS      #PWRCLR, @TRCR ; DEVICE MASTER RESET
1966 014022 032777 004000 165352 64$: BIT      #PWRCLR, @TRCR ; INSTRUCTION CLEAR ?
1967 014030 001374      BNE      64$ ; IF NO, WAIT FOR IT TO CLEAR
1968 014032 000207      RTS      PC
1969
1970 ; +-----+
1971 ; THIS ROUTINE TESTS THE FUNCTIONS OF ILLEGAL COMMANDS
1972
1973
1974 014034      TESTJ: JSR      PC, TESTI
1975 014034 004737 013626      JSR      PC, ENAINT
1976      MOV      #ILCTAB, R3 ; POINT TO THE ILLEGAL INSTRUCTION CODE TABLE
1977 014040 004737 031212      MOV      #6, R4 ; COUNT THE NUMBER OF ILLEGAL INSTRUCTIONS
1978 014044 012703 014254      MOV      (R3)+, 2$ ; INSERT AN ILLEGAL INSTRUCTION CODE
1979 014050 012704 000006      JSR      R5, TRYIT ; GO DO THE FUNCTION
1980 014054 012337 014064 1$:      .WORD 0, 0 ; LOCATION OF FUNCTION TO DO
1981 014060 004537 027546      JSR      PC, TESTJ1 ; PARAMETERS TO TRYIT
1982 014064 000000      DEC      R4 ; CHECK IT OUT
1983 014066 000000 000000      BNE      1$ ; REDUCE THE COUNT. ARE WE DONE?
1984 014072 004737 014104      RTS      PC ; IF NOT, CONTINUE
1985 014076 005304
1986 014100 001365
1987 014102 000207
1988
1989 014104 012737 000030 001240 TESTJ1: MOV      #30, HOLD ; DELAY PAST TIME OUT
1990 014112 013727 001240      MOV      HOLD, (PC)+ ; PICK UP TIME PARAMETER
1991 014116 000000 64$:      .WORD 0 ; USE THIS WORD AS A TIME COUNTER
1992 014120 66$:
1993 014120 005227 000000      INC      #0 ; IF NO, COUNT 1 OF 65535 TICKS
1994 014124 001375      BNE      66$ ; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
1995 014126 005337 014116      DEC      64$ ; HAS THE TOTAL TIME ELAPSED?
1996 014132 001372      BNE      66$ ; IF NO, GO WAIT A LITTLE LONGER
1997 014134 65$:
1998 014134 012737 000010 001240      MOV      #10, HOLD ; RESTORE NORMAL DELAY FACTOR
1999 014142 017701 165234      MOV      @TRCR, R1 ; GET COMMAND REGISTER
2000 014146 042701 001436      BIC      #1436, R1 ; CLEAR UNWANTED BITS
2001 014152 022701 142200      CMP      #142200, R1 ; TEST COMMAND REGISTER
2002 014156 001411      BEQ      A$2
2003 014160 012705 142200      MOV      #142200, R5
2004 014164 010104      MOV      R1, R4
2005 014166 013737 001402 001244      MOV      TRCR, REGIST
2006 014174 104001      ERROR 1 ; INCORRECT REGISTER MATCHUP
2007 014176 104403 006264      TYPEF ,MILFFL
2008
2009
2010 014202 000003      A$2: STAGAS=STAGAS+1
2011 014202 017701 165200      MOV      @TRSR, R1 ; GET STATUS REGISTER
2012 014206 022701 002040      CMP      #2040, R1 ; TEST STATUS REGISTER
2013 014212 001410      BEQ      A$3
2014 014214 012705 002040      MOV      #2040, R5
2015 014220 010104      MOV      R1, R4
2016 014222 013737 001406 001244      MOV      TRSR, REGIST
2017 014230 104001      ERROR 1 ; INCORRECT REGISTER MATCHUP
2018 014232 104404      TYPEL
2019

```

```

2020
2021 014234 000004
2022 014234 052777 004000 165140
2023 014242 032777 004000 165132
2024 014250 001374
2025 014252 000207
2026 014254 000406 001012 001414
2027 014262 000022 000424 001030
2028
2029
2030
2031
2032
2033
2034 014270
2035 014270 004737 013626
2036 014274 004537 027546
2037 014300 000402
2038 014302 177700
2039 014304 031600
2040
2041 014306 022777 142602 165066
2042 014314 001412
2043 014316 012705 142602
2044 014322 017704 165054
2045 014326 013737 001402 001244
2046 014334 104001
2047 014336 104403 006310
2048
2049 014342 000005
2050
2051 014342 022777 002040 165036
2052 014350 001411
2053 014352 012705 002040
2054 014356 017704 165024
2055 014362 013737 001406 001244
2056 014370 104001
2057 014372 104404
2058
2059 014374 000006
2060
2061 014374 052777 004000 165000
2062 014402 032777 004000 164772
2063 014410 001374
2064 014412 000207
2065
2066
2067
2068
2069
2070 014414
2071 014414 004737 021712
2072 014420 012737 000062 014444
2073 014426 005737 014444
2074 014432 001001
2075 014434 000207

```

AS3: STAGAS=STAGAS+1
 BIS #PWRCLR, @TRCR ; DEVICE MASTER RESET
 64\$: BIT #PWRCLR, @TRCR ; INSTRUCTION CLEAR ?
 BNE 64\$; IF NO, WAIT FOR IT TO CLEAR
 RTS PC
 ILCTAB: .WORD ILC03, ILC05, ILC06, ILC11, ILC12, ILC14

; +-----
 ; THIS ROUTINE TESTS THE DEROR CONDITIONS OF WRITING
 ; FROM LOAD POINT

TESTK: JSR PC, TESTI
 JSR RS, TRYIT
 .WORD WRITE ; FUNCTION
 .WORD -100 ; WORD COUNT
 .WORD OUTPUT ; BUS ADDRESS

CMP #142602, @TRCR ; TEST CONTROL REGISTER
 BEQ AS4
 MOV #142602, RS
 MOV @TRCR, R4
 MOV TRCR, REGIST
 ERROR 1 ; INCORRECT REGISTER MATCHUP
 TYPEF , MWLPFL

AS4: STAGAS=STAGAS+1

CMP #2040, @TRSR ; TEST STATUS REGISTER
 BEQ AS5
 MOV #2040, RS
 MOV @TRSR, R4
 MOV TRSR, REGIST
 ERROR 1 ; INCORRECT REGISTER MATCHUP
 TYPEL

AS5: STAGAS=STAGAS+1

BIS #PWRCLR, @TRCR ; DEVICE MASTER RESET
 64\$: BIT #PWRCLR, @TRCR ; INSTRUCTION CLEAR ?
 BNE 64\$; IF NO, WAIT FOR IT TO CLEAR
 RTS PC

; +-----
 ; THIS ROUTINE WRITES A END OF FILE MARK
 ; TAPE MUST BE POSITIONED PRIOR TO ENTERING THIS ROUTINE

RUNL: JSR PC, TESTAH
 MOV #50., 3\$
 2\$: TST 3\$
 BNE 1\$
 RTS PC

```

2076 014436 005337 014444 1$: DEC 3$
2077 014442 000401 BR 4$
2078 014444 000000 3$: 0
2079 014446 4$:
2080
2081 014446 TESTL:
2082 014446 004537 027546 JSR R5,TRYIT
2083 014452 001034 .WORD WEOF ;FUNCTION
2084 014454 000000 .WORD -0 ;WORD COUNT
2085 014456 000000 .WORD 0 ;BUS ADDRESS
2086 014460 022777 003234 164714 CMP #3234,@TRCR ;TEST COMMAND REGISTER
2087 014466 001412 BEQ A$6
2088 014470 012705 003234 MOV #3234,R5
2089 014474 017704 164702 MOV @TRCR,R4
2090 014500 013737 001402 001244 MOV TRCR,REGIST
2091 014506 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
2092 014510 104403 006342 TYPEF ,MWEFFL
2093
2094 014514 000007 A$6: $TAG$=$TAG$+1
2095
2096 014514 022777 002011 164664 CMP #2011,@TRSR ;TEST THE STATUS REGISTER
2097 014522 001411 BEQ A$7
2098 014524 012705 002011 MOV #2011,R5
2099 014530 017704 164652 MOV @TRSR,R4
2100 014534 013737 001406 001244 MOV TRSR,REGIST
2101 014542 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
2102 014544 104404 TYPEL
2103
2104 014546 000010 A$7: $TAG$=$TAG$+1
2105
2106
2107 014546 000207 RTS PC
2108
2109 ;+-----+
2110
2111 ;THIS ROUTINE ROUTINE TESTS THE DEROR CONDITION OF WRITING A END
2112 ;OF FILE MARK FROM LOAD POINT.
2113
2114 TESTM:
2115 014550 004737 013626 JSR PC,TESTI
2116 014554 004537 027546 JSR R5,TRYIT
2117 014560 001034 .WORD WEOF ;FUNCTION
2118 014562 000000 .WORD -0 ;WORD COUNT
2119 014564 000000 .WORD 0 ;BUS ADDRESS
2120 014566 022777 143234 164606 CMP #143234,@TRCR ;TEST COMMAND REGISTER
2121 014574 001412 BEQ A$10
2122 014576 012705 143234 MOV #143234,R5
2123 014602 017704 164574 MOV @TRCR,R4
2124 014606 013737 001402 001244 MOV TRCR,REGIST
2125 014614 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
2126 014616 104403 006367 TYPEF ,MWELPFL
2127
2128 014622 000011 A$10: $TAG$=$TAG$+1
2129
2130 014622 022777 002040 164556 CMP #2040,@TRSR ;TEST THE STATUS REGISTER
2131 014630 001411 BEQ A$11

```

CZTRAB0 MACY11 30(1046) 14-DEC-77 12:32 PAGE 48
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0047

```

2132 014632 012705 002040      MOV      #2040,R5
2133 014636 017704 164544      MOV      @TRSR,R4
2134 014642 013737 001406 001244  MOV      TRSR,REGIST
2135 014650 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
2136 014652 104404      TYPEL
2137
2138 014654 000012      AS11:  STAGAS=STAGAS+1
2139
2140 014654 004737 013234      JSR      PC,TESTG
2141 014660 000207      RTS      PC
2142
2143
2144
2145 ;-----*
2146
2147 ;THIS ROUTINE TESTS FOR TAPE TRANSPORT ON LINE AND EXITS WHEN
2148 ;IT IS.
2149
2150      A=.
2151 014662 000207      RTS      PC
2152
2153 014664 032777 002000 164514  TESTN:  BIT      #BIT10,@TRSR
2154 014672 001373      BNE      A      ;EXIT IF ON LINE
2155 014674 104402 006435      TYPE      ,MUOLN
2156
2157 014700 032777 002000 164500  A=.
2158 014706 001007      BIT      #BIT10,@TRSR
2159 014710 005237 010560      BNE      AS12
2160 014714 001003      INC      TEMP
2161 014716 012777 000207 164266  BNE      64$
2162 014724      MOV      #207,@TPDBR      ;RING THE BELL
2163
2164 014724 000765      BR      A
2165 014726 000013      AS12:  STAGAS=STAGAS+1
2166 014734 001403      BIT      #BIT15,@TRCR
2167 014736 104000      BEQ      AS13
2168 014740 104403 006520      ERROR      TYPEF      ,MOLEFL      ;ERROR-BIT WAS NOT CORRECT
2169
2170 014744 000014      AS13:  STAGAS=STAGAS+1
2171 014744 052777 004000 164430  BIS      #PWRCLR,@TRCR      ;DEVICE MASTER RESET
2172 014752 032777 004000 164422  64$:  BIT      #PWRCLR,@TRCR      ;INSTRUCTION CLEAR ?
2173 014760 001374      BNE      64$      ;IF NO, WAIT FOR IT TO CLEAR
2174 014762 000207      RTS      PC
2175
2176 ;-----*
2177 ;THIS ROUTINE IS USED TO TEST THE FUNCTION OF WRITING A ID
2178 ;BLOCK FROM LOAD POINT THIS IS THE MAIN ID BLOCK ROUTINE
2179
2180 014764
2181 014764 004737 013626      TESTO:  JSR      PC,TESTI
2182 014770 004537 027546      JSR      R5,TRYIT
2183 014774 000432      .WORD      WIDB      ;FUNCTION
2184 014776 000000      .WORD      -0      ;WORD COUNT
2185 015000 000000      .WORD      0      ;BUS ADDRESS
2186 015002 022777 002632 164372  CMP      #2632,@TRCR      ;TEST COMMAND REGISTER
2187 015010 001412      BEQ      AS14

```

```

2188 015012 012705 002632      MOV      #2632,R5
2189 015016 017704 164360      MOV      @TRCR,R4
2190 015022 013737 001402 001244  MOV      TRCR,REGIST
2191 015030 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
2192 015032 104403 006541      TYPEF      ,MWILPFL
2193
2194 015036 000015      AS14:   STAGAS=STAGAS+1
2195
2196 015036 022777 002021 164342  CMP      #2021,@TRSR      ;TEST THE STATUS REGISTER
2197 015044 001411      BEQ      AS15
2198 015046 012705 002021      MOV      #2021,R5
2199 015052 017704 164330      MOV      @TRSR,R4
2200 015056 013737 001406 001244  MOV      TRSR,REGIST
2201 015064 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
2202 015066 104404      TYPEL
2203
2204 015070 000016      AS15:   STAGAS=STAGAS+1
2205
2206
2207 015070 000207      RTS      PC
2208
2209
2210
2211 -----*
2212 ;THIS ROUTINE IS USED TO TAKE THE TAPE UNIT OFF LINE
2213 ;USING THE OFF LINE COMMAND
2214
2215 015072      TESTP:
2216 015072 004537 027546      JSR      R5,TRYIT      ;;;
2217 015076 001436      .WORD      OFFLINE      ;FUNCTION
2218 015100 000000      .WORD      -0      ;WORD COUNT
2219 015102 000000      .WORD      0      ;BUS ADDRESS
2220
2221 015104 032777 002000 164274  BIT      #BIT10,@TRSR
2222 015112 001403      BEQ      AS16
2223 015114 104000      ERROR      ;ERROR-BIT WAS NOT CORRECT
2224 015116 104403 006603      TYPEF      ,MUNOFLN
2225
2226 015122 000017      AS16:   STAGAS=STAGAS+1
2227 015122 032777 100000 164252  BIT      #BIT15,@TRCR
2228 015130 001003      BNE      AS17
2229 015132 104000      ERROR      ;ERROR-BIT WAS NOT CORRECT
2230 015134 104403 006637      TYPEF      ,MOFLEFL
2231
2232 015140 000020      AS17:   STAGAS=STAGAS+1
2233 015140 052777 004000 164234  BIS      #BIT11,@TRCR
2234 015146 013727 001240      MOV      HOLD,(PC)+
2235 015152 000000      64$:   .WORD      0      ;PICK UP TIME PARAMETER
2236 015154      66$:           ;USE THIS WORD AS A TIME COUNTER
2237 015154 005227 000000      INC      #0      ;IF NO,COUNT 1 OF 65535 TICKS
2238 015160 001375      BNE      66$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2239 015162 005337 015152      DEC      64$      ;HAS THE TOTAL TIME ELAPSED?
2240 015166 001372      BNE      66$      ;IF NO,GO WAIT A LITTLE LONGER
2241 015170
2242 015170 004737 014664      65$:   JSR      PC,TESTN
2243 015174 052777 004000 164200  BIS      #PWRCLR,@TRCR      ;DEVICE MASTER RESET

```

```

2244 015202 032777 004000 164172 67$: BIT #PWRCLR, @TRCR ; INSTRUCTION CLEAR ?
2245 015210 001374 BNE 67$ ; IF NO, WAIT FOR IT TO CLEAR
2246 015212 000207 RTS PC
2247
2248
2249
2250 ;-----*
2251 ; THIS ROUTINE TESTS THE OFF LINE FUNCTION OF POWER CLEAR WHILE
2252 ; REWINDING THE TAPE
2253
2254 015214 012737 001026 027536 TESTQ: MOV #GOEOT, USEA ; SET FUNCTION
2255 015222 012737 000000 027540 MOV #-0, USEB ; SET WORD COUNT
2256 015230 012737 000000 027542 MOV #0, USEC ; SET BUS ADDRESS
2257 015236 004737 030012 JSR PC, EOTTST
2258 015242 013727 001240 MOV HOLD, (PC)+ ; PICK UP TIME PARAMETER
2259 015246 000000 64$: .WORD 0 ; USE THIS WORD AS A TIME COUNTER
2260 015250 66$:
2261 015250 005227 000000 INC #0 ; IF NO, COUNT 1 OF 65535 TICKS
2262 015254 001375 BNE 66$ ; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2263 015256 005337 015246 DEC 64$ ; HAS THE TOTAL TIME ELAPSED?
2264 015262 001372 BNE 66$ ; IF NO, GO WAIT A LITTLE LONGER
2265 015264 65$:
2266 015264 013727 001240 MOV HOLD, (PC)+ ; PICK UP TIME PARAMETER
2267 015270 000000 67$: .WORD 0 ; USE THIS WORD AS A TIME COUNTER
2268 015272 69$:
2269 015272 005227 000000 INC #0 ; IF NO, COUNT 1 OF 65535 TICKS
2270 015276 001375 BNE 69$ ; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2271 015300 005337 015270 DEC 67$ ; HAS THE TOTAL TIME ELAPSED?
2272 015304 001372 BNE 69$ ; IF NO, GO WAIT A LITTLE LONGER
2273 015306 68$:
2274 015306 013727 001240 MOV HOLD, (PC)+ ; PICK UP TIME PARAMETER
2275 015312 000000 70$: .WORD 0 ; USE THIS WORD AS A TIME COUNTER
2276 015314 72$:
2277 015314 005227 000000 INC #0 ; IF NO, COUNT 1 OF 65535 TICKS
2278 015320 001375 BNE 72$ ; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2279 015322 005337 015312 DEC 70$ ; HAS THE TOTAL TIME ELAPSED?
2280 015326 001372 BNE 72$ ; IF NO, GO WAIT A LITTLE LONGER
2281 015330 71$:
2282 015330 052777 004000 164044 BIS #PWRCLR, @TRCR ; DEVICE MASTER RESET
2283 015336 032777 004000 164036 73$: BIT #PWRCLR, @TRCR ; INSTRUCTION CLEAR ?
2284 015344 001374 BNE 73$ ; IF NO, WAIT FOR IT TO CLEAR
2285 015346 004537 027546 JSR RS, TRYIT
2286 015352 001020 .WORD REWIND ; FUNCTION
2287 015354 000000 .WORD -0 ; WORD COUNT
2288 015356 000000 .WORD 0 ; BUS ADDRESS
2289 015360 052777 004000 164014 BIS #BIT11, @TRCR
2290
2291 015366 013727 001240 MOV HOLD, (PC)+ ; PICK UP TIME PARAMETER
2292 015372 000000 74$: .WORD 0 ; USE THIS WORD AS A TIME COUNTER
2293 015374 76$:
2294 015374 005227 000000 INC #0 ; IF NO, COUNT 1 OF 65535 TICKS
2295 015400 001375 BNE 76$ ; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2296 015402 005337 015372 DEC 74$ ; HAS THE TOTAL TIME ELAPSED?
2297 015406 001372 BNE 76$ ; IF NO, GO WAIT A LITTLE LONGER
2298 015410 75$:
2299 015410 022777 004020 163764 CMP #4020, @TRCR ; TEST COMMAND REGISTER

```

```

2300 015416 001412          BEQ      A$20
2301 015420 012705 000020    MOV      #20,R5
2302 015424 017704 163752    MOV      @TRCR,R4
2303 015430 013737 001402 001244  MOV      TRCR,REGIST
2304 015433 104001          ERROR    1          ;INCORRECT REGISTER MATCHUP
2305 015440 104403 006661    TYPEF    ,MRWOFL
2306
2307 015444 000021          A$20:  STAGAS=STAGAS+1
2308
2309 015444 022777 000001 163734  CMP      #1,@TRSR          ;TEST THE STATUS REGISTER
2310 015452 001411          BEQ      A$21
2311 015454 012705 000001    MOV      #1,R5
2312 015460 017704 163722    MOV      @TRSR,R4
2313 015464 013737 001406 001244  MOV      TRSR,REGIST
2314 015472 104001          ERROR    1          ;INCORRECT REGISTER MATCHUP
2315 015474 104404          TYPEL
2316
2317 015476 000022          A$21:  STAGAS=STAGAS+1
2318
2319 015476 005777 163714    TST      @TRBA          ;TEST BUFFER ADDRESS
2320 015502 001411          BEQ      A$22
2321 015504 012705 000000    MOV      #0,R5
2322 015510 017704 163702    MOV      @TRBA,R4
2323 015514 013737 001416 001244  MOV      TRBA,REGIST
2324 015522 104001          ERROR    1          ;INCORRECT REGISTER MATCHUP
2325 015524 104404          TYPEL
2326
2327 015526 000023          A$22:  STAGAS=STAGAS+1
2328
2329 015526 005777 163660    TST      @TRWC          ;TEST THE WORD COUNT
2330 015532 001411          BEQ      A$23
2331 015534 012705 000000    MOV      #0,R5
2332 015540 017704 163646    MOV      @TRWC,R4
2333 015544 013737 001412 001244  MOV      TRWC,REGIST
2334 015552 104001          ERROR    1          ;INCORRECT REGISTER MATCHUP
2335 015554 104404          TYPEL
2336
2337 015556 000024          A$23:  STAGAS=STAGAS+1
2338
2339          015556          A=.
2340 015556 032777 000002 163622  BIT      #BIT1,@TRSR
2341 015564 001374          BNE      A          ;WAIT FOR REWIND TO FINISH
2342 015566 032777 000040 163612  BIT      #BIT5,@TRSR
2343 015574 001770          BEQ      A          ;WAIT FOR LOADPOINT TO SET
2344 015576 004737 014664    JSR      PC,TESTN
2345 015602 052777 004000 163572  BIS      #PWRCLR,@TRCR          ;DEVICE MASTER RESET
2346 015610 032777 004000 163564 64$:  BIT      #PWRCLR,@TRCR          ;INSTRUCTION CLEAR ?
2347 015616 001374          BNE      64$          ;IF NO, WAIT FOR IT TO CLEAR
2348 015620 000207          RTS      PC
2349
2350
2351
2352
2353
2354
2355

```

-----*

; THIS ROUTINE CHECKS THE DATA JUST READ BY THE
; MAG TAPE UNIT AND REPORTS ANY DERORS THAT EXIST

2356
 2357
 2358
 2359 015622 012701 035640
 2360 015626 012702 031600
 2361 015632 013703 027540
 2362 015636
 2363 015636 021112
 2364 015640 001410
 2365 015642 011205
 2366 015644 011104
 2367 015646 010137 001244
 2368 015652 104001
 2369 015654 104402 006704
 2370 015660 010200
 2371
 2372
 2373 015662 000025
 2374 015662 022122
 2375 015664 005203
 2376 015666 001363
 2377
 2378 015670 012700 035640
 2379 015674 012701 177000
 2380 015700 000026
 2381 015700
 2382 015700 005020
 2383 015702 001376
 2384 015704 000207
 2385
 2386
 2387
 2388
 2389
 2390
 2391
 2392
 2393
 2394
 2395
 2396
 2397 015706
 2398 015706 012737 000062 015732
 2399 015714 005737 015732
 2400 015720 001001
 2401 015722 000207
 2402 015724 005337 015732
 2403 015730 000401
 2404 015732 000000
 2405 015734
 2406
 2407 015734
 2408 015734 004537 027546
 2409 015740 001410
 2410 015742 000000
 2411 015744 000000

;THIS ROUTINE ALSO CLEARS THE INPUT AREA AFTER TESTING
 ;THE DATA

TESTS: MOV #INPUT,R1 ;SET INDEX
 MOV #OUTPUT,R2 ;SET INDEX
 MOV USEB,R3 ;GET THE WORD COUNT
 A=.
 CMP R1,R2 ;TEST INPUT
 BEQ AS24
 MOV R2,R5
 MOV R1,R4
 MOV R1,REGIST
 ERROR 1 ;INCORRECT REGISTER MATCHUP
 TYPE MRDERR
 MOV R2,R0
 ;----- CONVRT CALL SHOULD GO HERE

AS24: STAGAS=STAGAS+1
 CMP (R1)+,(R2)+ ;ADVANCE INDEXES
 INC R3 ;INCREMENT WORD COUNT
 BNE A ;CONTINUE TESTING INPUT

MOV #INPUT,R0 ;SET INDEX
 MOV #-1000,R1 ;SET COUNTER

AS25: STAGAS=STAGAS+1
 A=.
 CLR (R0)+ ;CLEAR INPUT AREA
 BNE A ;CONTINUE CLEARING
 RTS PC

;+-----
 ;THIS ROUTINE CHECKS THE FUNCTION OF SPACING REVERSE OVER A
 ;END OF FILE

RUNU: MOV #50.,3\$
 2\$: TST 3\$
 BNE 1\$
 RTS PC
 1\$: DEC 3\$
 BR 4\$
 3\$: 0
 4\$:

TESTU: JSR R5,TRYIT
 .WORD SPACER ;FUNCTION
 .WORD -0 ;WORD COUNT
 .WORD 0 ;BUS ADDRESS

```

2412 015746 022777 003610 163426      CMP      #3610, @TRCR      ; TEST COMMAND REGISTER
2413 015754 001412                      BEQ      AS26
2414 015756 012705 003610              MOV      #3610, R5
2415 015762 017704 163414              MOV      @TRCR, R4
2416 015766 013737 001402 001244      MOV      TRCR, REGIST
2417 015774 104001                      ERROR    1      ; INCORRECT REGISTER MATCHUP
2418 015776 104403 006727              TYPEF    , MSREFFL
2419
2420 016002 000027                      AS26:    STAGAS=STAGAS+1
2421
2422 016002 022777 002011 163376      CMP      #2011, @TRSR      ; TEST THE STATUS REGISTER
2423 016010 001411                      BEQ      AS27
2424 016012 012705 002011              MOV      #2011, R5
2425 016016 017704 163364              MOV      @TRSR, R4
2426 016022 013737 001406 001244      MOV      TRSR, REGIST
2427 016030 104001                      ERROR    1      ; INCORRECT REGISTER MATCHUP
2428 016032 104404                      TYPEL
2429
2430 016034 000030                      AS27:    STAGAS=STAGAS+1
2431
2432 016034 005777 163356              TST      @TRBA      ; TEST BUFFER ADDRESS
2433 016040 001411                      BEQ      AS30
2434 016042 012705 000000              MOV      #0, R5
2435 016046 017704 163344              MOV      @TRBA, R4
2436 016052 013737 001416 001244      MOV      TRBA, REGIST
2437 016060 104001                      ERROR    1      ; INCORRECT REGISTER MATCHUP
2438 016062 104404                      TYPEL
2439
2440 016064 000031                      AS30:    STAGAS=STAGAS+1
2441
2442 016064 005777 163322              TST      @TRWC      ; TEST THE WORD COUNT
2443 016070 001411                      BEQ      AS31
2444 016072 012705 000000              MOV      #0, R5
2445 016076 017704 163310              MOV      @TRWC, R4
2446 016102 013737 001412 001244      MOV      TRWC, REGIST
2447 016110 104001                      ERROR    1      ; INCORRECT REGISTER MATCHUP
2448 016112 104404                      TYPEL
2449
2450 016114 000032                      AS31:    STAGAS=STAGAS+1
2451
2452 016114 000207                      RTS      PC
2453
2454 ;-----
2455 ; THIS ROUTINE CHECKS THE FUNCTION OF READING A END OF FILE
2456
2457 016116                                RUNV:
2458 016116 012737 000062 016142      MOV      #50., 3$
2459 016124 005737 016142      2$:    TST      3$
2460 016130 001001                      BNE     1$
2461 016132 000207                      RTS     PC
2462 016134 005337 016142      1$:    DEC     3$
2463 016140 000401                      BR      4$
2464 016142 000000      3$:    0
2465 016144      4$:
2466
2467 016144      TESTV:

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 54
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0053

```

2468 016144 004537 027546      JSR      R5,TRYIT      ;;;
2469 016150 000004      .WORD     READ          ;FUNCTION
2470 016152 000000      .WORD     -0           ;WORD COUNT
2471 016154 035640      .WORD     INPUT        ;BUS ADDRESS
2472 016156 022777 002204 163216  CMP      #2204,@TRCR    ;TEST COMMAND REGISTER
2473 016164 001412      BEQ      A$32
2474 016166 012705 002204      MOV      #2204,R5
2475 016172 017704 163204      MOV      @TRCR,R4
2476 016176 013737 001402 001244  MOV      TRCR,REGIST
2477 016204 104001      ERROR     1          ;INCORRECT REGISTER MATCHUP
2478 016206 104403 006771      TYPEF    ,MREFFL
2479
2480 016212 000033      A$32:    STAGAS=STAGAS+1
2481
2482 016212 022777 002011 163166  CMP      #2011,@TRSR    ;TEST THE STATUS REGISTER
2483 016220 001410      BEQ      A$33
2484 016222 012705 002011      MOV      #2011,R5
2485 016226 017704 163154      MOV      @TRSR,R4
2486 016232 013737 001406 001244  MOV      TRSR,REGIST
2487 016240 104001      ERROR     1          ;INCORRECT REGISTER MATCHUP
2488
2489 016242 000034      A$33:    STAGAS=STAGAS+1
2490
2491 016242 022777 035640 163146  CMP      #INPUT,@TRBA    ;TEST BUFFER ADDRESS
2492 016250 001410      BEQ      A$34
2493 016252 012705 035640      MOV      #INPUT,R5
2494 016256 017704 163134      MOV      @TRBA,R4
2495 016262 013737 001416 001244  MOV      TRBA,REGIST
2496 016270 104001      ERROR     1          ;INCORRECT REGISTER MATCHUP
2497
2498 016272 000035      A$34:    STAGAS=STAGAS+1
2499
2500 016272 005777 163114      TST      @TRWC          ;TEST THE WORD COUNT
2501 016276 001410      BEQ      A$35
2502 016300 012705 000000      MOV      #0,R5
2503 016304 017704 163102      MOV      @TRWC,R4
2504 016310 013737 001412 001244  MOV      TRWC,REGIST
2505 016316 104001      ERROR     1          ;INCORRECT REGISTER MATCHUP
2506 016320 000036      A$35:    STAGAS=STAGAS+1
2507
2508 016320 000207      RTS      PC
2509
2510 ;-----*
2511 ;THIS ROUTINE TESTS THE FUNCTION OF THE TIME OUT WHILE WRITING
2512
2513 016322      TESTW:
2514 016322 004737 021712      JSR      PC,TESTAH
2515 016326 042777 000001 163052  BIC      #BIT0,@TRSR
2516 016334 005077 163052      CLR      @TRWC          ;SET WORD COUNT TO MAX
2517 016340 012777 031600 163050  MOV      #OUTPUT,@TRBA    ;SET BUS ADDRESS AT START
2518 016346 012777 000402 163026  MOV      #WRITE,@TRCR    ;SET COMMAND REGISTER
2519 016354 052777 000001 163020  BIS      #BIT0,@TRCR    ;SET THE GO BIT
2520 016362 005037 010560      CLR      TEMP          ;CLEAR THE TIMER
2521 016366 005077 163020      A=:    CLR      @TRWC          ;RESET THE WORD COUNT
2522 016372 012777 031600 163016  MOV      #OUTPUT,@TRBA    ;RESET THE BUS ADDRESS

```

CZTRAB0 MACY11 30(1046) 14-DEC-77 12:32 PAGE 55
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0054

2524	016400	005237	010560		INC	TEMP		; INCREMENT TIME
2525	016404	001370			BNE	A		; DELAY MORE
2526		016406		A=.				
2527	016406	005077	163000		CLR	QTRWC		; RESET THE WORD COUNT
2528	016412	012777	031600	162776	MOV	#OUTPUT, QTRBA		; RESET THE BUS ADDRESS
2529	016420	005237	010560		INC	TEMP		; INCREMENT TIME
2530	016424	001370			BNE	A		; DELAY MORE
2531		016426		A=.				
2532	016426	005077	162760		CLR	QTRWC		; RESET THE WORD COUNT
2533	016432	012777	031600	162756	MOV	#OUTPUT, QTRBA		; RESET THE BUS ADDRESS
2534	016440	005237	010560		INC	TEMP		; INCREMENT TIME
2535	016444	001370			BNE	A		; DELAY MORE
2536	016446	022777	102602	162726	CMP	#102602, QTRCR		; TEST COMMAND REGISTER
2537	016454	001412			BEQ	AS36		
2538	016456	012705	102602		MOV	#102602, R5		
2539	016462	017704	162714		MOV	QTRCR, R4		
2540	016466	013737	001402	001244	MOV	TRCR, REGIST		
2541	016474	104001			ERROR	1		; INCORRECT REGISTER MATCHUP
2542	016476	104403	007015		TYPEF	, MWTOFL		
2543								
2544	016502	000037			AS36:	STAGAS=STAGAS+1		
2545								
2546	016502	017701	162700		MOV	QTRSR, R1		; GET THE STATUS
2547	016506	042701	170000		BIC	#170000, R1		; CLEAR BAD BITS
2548	016512	022701	002401		CMP	#2401, R1		; TEST THE STATUS REGISTER
2549	016516	001411			BEQ	AS37		
2550	016520	012705	002401		MOV	#2401, R5		
2551	016524	017704	162656		MOV	QTRSR, R4		
2552	016530	013737	001406	001244	MOV	TRSR, REGIST		
2553	016536	104001			ERROR	1		; INCORRECT REGISTER MATCHUP
2554	016540	104404			TYPEL			
2555								
2556	016542	000040			AS37:	STAGAS=STAGAS+1		
2557								
2558	016542	022777	031600	162646	CMP	#OUTPUT, QTRBA		; TEST BUFFER ADDRESS
2559	016550	001411			BEQ	AS40		
2560	016552	012705	031600		MOV	#OUTPUT, R5		
2561	016556	017704	162634		MOV	QTRBA, R4		
2562	016562	013737	001416	001244	MOV	TRBA, REGIST		
2563	016570	104001			ERROR	1		; INCORRECT REGISTER MATCHUP
2564	016572	104404			TYPEL			
2565								
2566	016574	000041			AS40:	STAGAS=STAGAS+1		
2567								
2568	016574	005777	162612		TST	QTRWC		; TEST THE WORD COUNT
2569	016600	001411			BEQ	AS41		
2570	016602	012705	000000		MOV	#0, R5		
2571	016606	017704	162600		MOV	QTRWC, R4		
2572	016612	013737	001412	001244	MOV	TRWC, REGIST		
2573	016620	104001			ERROR	1		; INCORRECT REGISTER MATCHUP
2574	016622	104404			TYPEL			
2575								
2576	016624	000042			AS41:	STAGAS=STAGAS+1		
2577								
2578	016624	052777	004000	162550	BIS	#PWRCLR, QTRCR		; DEVICE MASTER RESET
2579	016632	032777	004000	162542	BIT	#PWRCLR, QTRCR		; INSTRUCTION CLEAR "

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 56
 CZTRAB.MED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0055

2580 016640 001374 BNE 645 ;IF NO, WAIT FOR IT TO CLEAR
 2581 016642 004737 014764 JSR PC,TESTO
 2582 016646 000207 RTS PC

-----*
 ;THIS ROUTINE CHECKS THE FUNCTION OF THE TIME OUT WHILE READING

TESTX:

2583
 2584
 2585
 2586
 2587 016650
 2588 016650 004737 021712 JSR PC,TESTAH
 2589 016654 004737 026550 JSR PC,TESTCA
 2590 016660 004737 026650 JSR PC,TESTFO
 2591 016664 004737 026650 JSR PC,TESTFO
 2592 016670 042777 000001 162510 BIC #B1TO,TRSR
 2593 016676 005077 162510 CLR TRWC ;SET WORD COUNT TO MAX
 2594 016702 012777 035640 162506 MOV #INPUT,TRBA ;SET BUS ADDRESS AT START
 2595 016710 012777 001410 162464 MOV #SPACER,TRCR ;SET COMMAND REGISTER
 2596 016716 052777 000001 162456 BIS #B1TO,TRCR ;SET THE GO BIT
 2597 016724 004737 012524 JSR PC,TESTA
 2598
 2599 016730 022777 102050 162444 CMP #102050,TRCR ;TEST COMMAND REGISTER
 2600 016736 001412 BEQ AS42
 2601 016740 012705 102050 MOV #102050,R5
 2602 016744 017704 162432 MOV TRCR,R4
 2603 016750 013737 001402 001244 MOV TRCR,REGIST
 2604 016756 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
 2605 016760 104403 007037 TYPEF ,MRT0FL
 2606
 2607 016764 000043 AS42: STAGAS=STAGAS+1
 2608
 2609 016764 022777 002400 162414 CMP #2400,TRSR ;TEST THE STATUS REGISTER
 2610 016772 001411 BEQ AS43
 2611 016774 012705 002400 MOV #2400,R5
 2612 017000 017704 162402 MOV TRSR,R4
 2613 017004 013737 001406 001244 MOV TRSR,REGIST
 2614 017012 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
 2615 017014 104404 TYPEL
 2616
 2617 017016 000044 AS43: STAGAS=STAGAS+1
 2618
 2619 017016 022777 035640 162372 CMP #INPUT,TRBA ;TEST BUFFER ADDRESS
 2620 017024 001411 BEQ AS44
 2621 017026 012705 035640 MOV #INPUT,R5
 2622 017032 017704 162360 MOV TRBA,R4
 2623 017036 013737 001416 001244 MOV TRBA,REGIST
 2624 017044 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
 2625 017046 104404 TYPEL
 2626
 2627 017050 000045 AS44: STAGAS=STAGAS+1
 2628
 2629 017050 005777 162336 TST TRWC ;TEST THE WORD COUNT
 2630 017054 001411 BEQ AS45
 2631 017056 012705 000000 MOV #0,R5
 2632 017062 017704 162324 MOV TRWC,R4
 2633 017066 013737 001412 001244 MOV TRWC,REGIST
 2634 017074 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
 2635 017076 104404 TYPEL

```

2636
2637 017100 000046 AS45: STAGAS=STAGAS+1
2638
2639 017100 052777 004000 162274 64S: BIS #PWRCLR, @TRCR ; DEVICE MASTER RESET
2640 017106 032777 004000 162266 64S: BIT #PWRCLR, @TRCR ; INSTRUCTION CLEAR ?
2641 017114 001374 64S: BNE 64S ; IF NO, WAIT FOR IT TO CLEAR
2642 017116 000207 RTS PC
2643
2644
2645 ;-----*
2646 ; THIS ROUTINE FORCES A NXM AND BGL DEROR WHILE WRITING
2647
2648 017120 TESTY: JSR PC, TESTAH
2649 017120 004737 021712 JSR RS, TRYIT ;;;
2650 017124 004537 027546 .WORD WRITE ; FUNCTION
2651 017130 000402 .WORD -0 ; WORD COUNT
2652 017132 000000 .WORD -1 ; BUS ADDRESS
2653 017134 177777 MOV HOLD, (PC)+ ; PICK UP TIME PARAMETER
2654 017136 013727 001240 64S: .WORD 0 ; USE THIS WORD AS A TIME COUNTER
2655 017142 000000 66S: INC #0 ; IF NO, COUNT 1 OF 65535 TICKS
2656 017144 005227 000000 BNE 66S ; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2657 017150 001375 DEC 64S ; HAS THE TOTAL TIME ELAPSED?
2658 017152 005337 017142 BNE 66S ; IF NO, GO WAIT A LITTLE LONGER
2659
2660 017156 001372 65S: CMP #112602, @TRCR ; TEST COMMAND REGISTER
2661 017160 022777 112602 162214 BEQ AS46
2662 017166 001412 MOV #112602, R5
2663 017170 012705 112602 MOV @TRCR, R4
2664 017174 017704 162202 MOV TRCR, REGIST
2665 017174 017704 162202 MOV TRCR, REGIST
2666 017200 013737 001402 001244 ERROR 1 ; INCORRECT REGISTER MATCHUP
2667 017206 104001 TYPEF ,MWNXFL
2668 017210 104403 007060
2669
2670 017214 000047 AS46: STAGAS=STAGAS+1
2671
2672 017214 022777 006101 162164 CMP #6101, @TRSR ; TEST THE STATUS REGISTER
2673 017222 001411 BEQ AS47
2674 017224 012705 006101 MOV #6101, R5
2675 017230 017704 162152 MOV @TRSR, R4
2676 017234 013737 001406 001244 MOV TRSR, REGIST
2677 017242 104001 ERROR 1 ; INCORRECT REGISTER MATCHUP
2678 017244 104404 TYPEL
2679
2680 017246 000050 AS47: STAGAS=STAGAS+1
2681
2682 017246 022777 000001 162142 CMP #1, @TRBA ; TEST BUFFER ADDRESS
2683 017254 001411 BEQ AS50
2684 017256 012705 000001 MOV #1, R5
2685 017262 017704 162130 MOV @TRBA, R4
2686 017266 013737 001416 001244 MOV TRBA, REGIST
2687 017274 104001 ERROR 1 ; INCORRECT REGISTER MATCHUP
2688 017276 104404 TYPEL
2689
2690 017300 000051 AS50: STAGAS=STAGAS+1
2691

```

F05

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 58
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0057

```

2692 017300 022777 000001 162104      CMP      #1, @TRWC      ;TEST THE WORD COUNT
2693 017306 001411                      BEQ      A$51
2694 017310 012705 000001                      MOV      #1, R5
2695 017314 017704 162072                      MOV      @TRWC, R4
2696 017320 013737 001412 001244          MOV      TRWC, REGIST
2697 017326 104001                      ERROR    1      ;INCORRECT REGISTER MATCHUP
2698 017330 104404                      TYPEL
2700 017332 000052                      A$51:  STAGAS=STAGAS+1
2701
2702 017332 000207                      RTS      PC
2703
2704 ;-----*
2705 ;THIS ROUTINE FORCES A NXM DEROR USING BIT 12
2706
2707 017334      TESTZ:
2708 017334 004737 021712                      JSR      PC, TESTAH
2709 017340 012777 000402 162034          MOV      #WRITE, @TRCR      ;SET COMMAND
2710 017346 052777 010000 162026          BIS      #BIT12, @TRCR
2711 017354 042777 000001 162024          BIC      #BIT0, @TRSR
2712 017362 005077 162024                      CLR      @TRWC      ;CLEAR THE WORD COUNT
2713 017366 005077 162024                      CLR      @TRBA      ;CLEAR BUS ADDRESS
2714 017372 005277 162004                      INC      @TRCR      ;SET GO BIT
2715
2716 017376 013727 001240                      MOV      HOLD, (PC)+      ;PICK UP TIME PARAMETER
2717 017402 000000                      64$:  .WORD    0      ;USE THIS WORD AS A TIME COUNTER
2718 017404                      66$:
2719 017404 005227 000000                      INC      #0      ;IF NO,COUNT 1 OF 65535 TICKS
2720 017410 001375                      BNE      66$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2721 017412 005337 017402                      DEC      64$
2722 017416 001372                      BNE      66$      ;HAS THE TOTAL TIME ELAPSED?
2723 017420                      ;IF NO,GO WAIT A LITTLE LONGER
2724 017420 022777 112602 161754          65$:  CMP      #112602, @TRCR      ;TEST COMMAND REGISTER
2725 017426 001412                      BEQ      A$52
2726 017430 012705 112602                      MOV      #112602, R5
2727 017434 017704 161742                      MOV      @TRCR, R4
2728 017440 013737 001402 001244          MOV      TRCR, REGIST
2729 017446 104001                      ERROR    1      ;INCORRECT REGISTER MATCHUP
2730 017450 104403 007115                      TYPEF    ,MBNXFL
2731
2732 017454 000053                      A$52:  STAGAS=STAGAS+1
2733
2734 017454 022777 006101 161724          CMP      #6101, @TRSR      ;TEST THE STATUS REGISTER
2735 017462 001411                      BEQ      A$53
2736 017464 012705 006101                      MOV      #6101, R5
2737 017470 017704 161712                      MOV      @TRSR, R4
2738 017474 013737 001406 001244          MOV      TRSR, REGIST
2739 017502 104001                      ERROR    1      ;INCORRECT REGISTER MATCHUP
2740 017504 104404                      TYPEL
2741
2742 017506 000054                      A$53:  STAGAS=STAGAS+1
2743
2744 017506 022777 000002 161702          CMP      #2, @TRBA      ;TEST BUFFER ADDRESS
2745 017514 001411                      BEQ      A$54
2746 017516 012705 000002                      MOV      #2, R5
2747 017522 017704 161670                      MOV      @TRBA, R4

```

```

2748 017526 013737 001416 001244      MOV    TRBA,REGIST
2749 017534 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
2750 017536 104404      TYPEL
2751
2752 017540 000055      ASS4:   STAGAS=STAGAS+1
2753
2754 017540 022777 000001 161644      CMP     #1,@TRWC      ;TEST THE WORD COUNT
2755 017546 001411      BEQ     ASS5
2756 017550 012705 000001      MOV     #1,R5
2757 017554 017704 161632      MOV     @TRWC,R4
2758 017560 013737 001412 001244      MOV     TRWC,REGIST
2759 017566 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
2760 017570 104404      TYPEL
2761
2762 017572 000056      ASS5:   STAGAS=STAGAS+1
2763
2764 017572 05777 004000 161602      BIS     #PWRCLR,@TRCR ;DEVICE MASTER RESET
2765 017600 03777 004000 161574 64$:  BIT     #PWRCLR,@TRCR ;INSTRUCTION CLEAR ?
2766 017606 001374      BNE     64$   ;IF NO, WAIT FOR IT TO CLEAR
2767 017610 000207      RTS     PC
2768
2769 -----*
2770 ;THIS ROUTINE FORCES A NXM DEROR USING BIT 13
2771
2772 017612      TESTAA:
2773 017612 004737 021712      JSR     PC,TESTAH
2774 017616 012777 000402 161556      MOV     #WRITE,@TRCR ;SET COMMAND
2775 017624 052777 020000 161550      BIS     #BIT13,@TRCR
2776 017632 042777 000001 161546      BIC     #BIT0,@TRSR
2777 017640 005077 161546      CLR     @TRWC      ;CLEAR THE WORD COUNT
2778 017644 005077 161546      CLR     @TRBA      ;CLEAR BUS ADDRESS
2779 017650 005277 161526      INC     @TRCR      ;SET GO BIT
2780
2781 017654 013727 001240      MOV     HOLD,(PC)+ ;PICK UP TIME PARAMETER
2782 017660 000000 64$:  .WORD    0      ;USE THIS WORD AS A TIME COUNTER
2783 017662 66$:
2784 017662 005227 000000      INC     #0      ;IF NO,COUNT 1 OF 65535 TICKS
2785 017666 001375      BNE     66$   ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2786 017670 005337 017660      DEC     64$   ;HAS THE TOTAL TIME ELAPSED?
2787 017674 001372      BNE     66$   ;IF NO,GO WAIT A LITTLE LONGER
2788 017676 65$:
2789 017676 022777 122602 161476      CMP     #122602,@TRCR ;TEST COMMAND REGISTER
2790 017704 001412      BEQ     ASS6
2791 017706 012705 122602      MOV     #122602,R5
2792 017712 017704 161464      MOV     @TRCR,R4
2793 017716 013737 001402 001244      MOV     TRCR,REGIST
2794 017724 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
2795 017726 104403 007157      TYPEF    ,MNXBFL
2796
2797 017732 000057      ASS6:   STAGAS=STAGAS+1
2798
2799 017732 022777 006101 161446      CMP     #6101,@TRSR ;TEST THE STATUS REGISTER
2800 017740 001411      BEQ     ASS7
2801 017742 012705 006101      MOV     #6101,R5
2802 017746 017704 161434      MOV     @TRSR,R4
2803 017752 013737 001406 001244      MOV     TRSR,REGIST

```

```

2804 017760 104001          ERROR 1          ; INCORRECT REGISTER MATCHUP
2805 017762 104404          TYPEL
2806
2807 017764 000060          AS$7:  STAGAS=STAGAS+1
2808
2809 017764 022777 000002 161424      CMP      #2, @TRBA          ; TEST BUFFER ADDRESS
2810 017772 001411          BEQ      AS$6C
2811 017774 012705 000002          MOV      #2, R5
2812 020000 017704 161412          MOV      @TRBA, R4
2813 020004 013737 001416 001244      MOV      TRBA, REGIST
2814 020012 104001          ERROR 1          ; INCORRECT REGISTER MATCHUP
2815 020014 104404          TYPEL
2816
2817 020016 000061          AS$6:  STAGAS=STAGAS+1
2818
2819 020016 022777 000001 161366      CMP      #1, @TRWC          ; TEST THE WORD COUNT
2820 020024 001411          BEQ      AS$61
2821 020026 012705 000001          MOV      #1, R5
2822 020032 017704 161354          MOV      @TRWC, R4
2823 020036 013737 001412 001244      MOV      TRWC, REGIST
2824 020044 104001          ERROR 1          ; INCORRECT REGISTER MATCHUP
2825 020046 104404          TYPEL
2826
2827 020050 000062          AS$61: STAGAS=STAGAS+1
2828
2829 020050 052777 004000 161324      BIS      #PWRCLR, @TRCR ; DEVICE MASTER RESET
2830 020056 032777 004000 161316 64$: BIT      #PWRCLR, @TRCR ; INSTRUCTION CLEAR ?
2831 020064 001374          BNE      64$      ; IF NO, WAIT FOR IT TO CLEAR
2832 020066 000207          PTS      PC
2833
2834 ;-----*
2835 ; THIS ROUTINE FORCES A NXM DEROR USING BIT 12-13
2836
2837 020070          TESTAB:
2838 020070 004737 021712          JSR      PC, TESTAB
2839 020074 012777 000402 161300      MOV      #WRITE, @TRCR ; SET COMMAND
2840 020102 052777 030000 161272      BIS      #BIT12!BIT13, @TRCR ; SET THE EXTENDED ADDRESS BITS
2841 020110 042777 000001 161270      BIC      #BIT0, @TRSR
2842 020116 005077 161270          CLR      @TRWC          ; CLEAR THE WORD COUNT
2843 020122 012777 160000 161266      MOV      #160000, @TRBA ; SET UP AN I/O PAGE REFERENCE
2844 020130 005277 161246          INC      @TRCR          ; SET GO BIT
2845 020134 013727 001240          MOV      HOLD, (PC)+ ; PICK UP TIME PARAMETER
2846 020140 000000          64$: .WORD 0 ; USE THIS WORD AS A TIME COUNTER
2847 020142          66$:
2848 020142 005227 000000          INC      #0          ; IF NO, COUNT 1 OF 65535 TICKS
2849 020146 001375          BNE      66$          ; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2850 020150 005337 020140          DEC      64$          ; HAS THE TOTAL TIME ELAPSED?
2851 020154 001372          BNE      66$          ; IF NO, GO WAIT A LITTLE LONGER
2852 020156          65$:
2853 020156 022777 132602 161216      CMP      #132602, @TRCR ; TEST COMMAND REGISTER
2854 020164 001412          BEQ      AS$62
2855 020166 012705 132602          MOV      #132602, R5
2856 020172 017704 161204          MOV      @TRCR, R4
2857 020176 013737 001402 001244      MOV      TRCR, REGIST
2858 020204 104001          ERROR 1          ; INCORRECT REGISTER MATCHUP
2859 020206 104403 007227          TYPEF      , MNXMFL

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 61
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0060

```

2860 020212 000063 AS62: STAGS=STAGS+1
2861 020212 022777 006101 161166 CMP #6101,ATRSR ;TEST THE STATUS REGISTER
2862 020220 001411 BEQ AS63
2863 020222 012705 006101 MOV #6101,R5
2864 020226 017704 161154 MOV ATRSR,R4
2865 020232 013737 001406 001244 MOV TRSR,REGIST
2866 020240 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
2867 020242 104404 TYPEL
2868
2869 020244 000064 AS63: STAGS=STAGS+1
2870
2871 020244 032777 000002 161144 BIT #2,ATRBA ;TEST BUFFER ADDRESS
2872 020252 001011 BNE AS64
2873 020254 012705 000002 MOV #2,R5
2874 020260 017704 161132 MOV ATRBA,R4
2875 020264 013737 001416 001244 MOV TRBA,REGIST
2876 020272 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
2877 020274 104404 TYPEL
2878
2879 020276 000065 AS64: STAGS=STAGS+1
2880
2881 020276 022777 000001 161106 CMP #1,ATRWC ;TEST THE WORD COUNT
2882 020304 001411 BEQ AS65
2883 020306 012705 000001 MOV #1,R5
2884 020312 017704 161074 MOV ATRWC,R4
2885 020316 013737 001416 001244 MOV TRWC,REGIST
2886 020324 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
2887 020326 104404 TYPEL
2888
2889 020330 000066 AS65: STAGS=STAGS+1
2890
2891 020330 052777 004000 161044 BIS #PWRCLR,ATRCR ;DEVICE MASTER RESET
2892 020336 032777 004000 161036 64$: BIT #PWRCLR,ATRCR ;INSTRUCTION CLEAR ?
2893 020344 001374 BNE 64$ ;IF NO, WAIT FOR IT TO CLEAR
2894 020346 000207 RTS PC
2895
2896 ;-----*
2897 ;THIS ROUTINE FORCS A READ COUNT DEROR WHILE READING A LONG RECORD
2898
2899 020350 TESTAC:
2900 020350 004737 021712 JSR PC,TESTAH
2901 020354 004737 026610 JSR PC,TESTCK
2902 020360 004737 026630 JSR PC,TESTEK
2903 020364 004537 027546 JSR R5,TRYIT
2904 020370 000004 .WORD READ ;FUNCTION
2905 020372 177701 .WORD -77 ;WORD COUNT
2906 020374 035640 .WORD INPUT ;BUS ADDRESS
2907 020376 013727 001240 MOV HOLD,(PC)+ ;PICK UP TIME PARAMETER
2908 020402 000000 64$: .WORD 0 ;USE THIS WORD AS A TIME COUNTER
2909 020404 66$:
2910 020404 005227 000000 INC #0 ;IF NO,COUNT 1 OF 65535 TICKS
2911 020410 001375 BNE 66$ ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2912 020412 005337 020402 DEC 64$ ;HAS THE TOTAL TIME ELAPSED?
2913 020416 001372 BNE 66$ ;IF NO,GO WAIT A LITTLE LONGER
2914 020420 65$:
2915

```

2916	020420	022777	102204	160754		CMP	#102204,@TRCR	;TEST COMMAND REGISTER
2917	020426	001412				BEQ	A\$66	
2918	020430	012705	102204			MOV	#102204,R5	
2919	020434	017704	160742			MOV	@TRCR,R4	
2920	020440	013737	001402	001244		MOV	TRCR,REGIST	
2921	020446	104001				ERROR	1	;INCORRECT REGISTER MATCHUP
2922	020450	104403	007301			TYPEF	,MRCFL	
2923								
2924	020454	000067			A\$66:	STAGS=STAGS+1		
2925								
2926	020454	022777	003001	160724		CMP	#3001,@TRSR	;TEST THE STATUS REGISTER
2927	020462	001411				BEQ	A\$67	
2928	020464	012705	003001			MOV	#3001,R5	
2929	020470	017704	160712			MOV	@TRSR,R4	
2930	020474	013737	001406	001244		MOV	TRSR,REGIST	
2931	020502	104001				ERROR	1	;INCORRECT REGISTER MATCHUP
2932	020504	104404				TYPEL		
2933								
2934	020506	000070			A\$67:	STAGS=STAGS+1		
2935								
2936	020506	022777	036036	160702		CMP	#INPUT+176,@TRBA	;TEST BUFFER ADDRESS
2937	020514	001411				BEQ	A\$70	
2938	020516	012705	036036			MOV	#INPUT+176,R5	
2939	020522	017704	160670			MOV	@TRBA,R4	
2940	020526	013737	001416	001244		MOV	TRBA,REGIST	
2941	020534	104001				ERROR	1	;INCORRECT REGISTER MATCHUP
2942	020536	104404				TYPEL		
2943								
2944	020540	000071			A\$70:	STAGS=STAGS+1		
2945								
2946	020540	005777	160646			TST	@TRWC	;TEST THE WORD COUNT
2947	020544	001411				BEQ	A\$71	
2948	020546	012705	000000			MOV	#0,R5	
2949	020552	017704	160634			MOV	@TRWC,R4	
2950	020556	013737	001412	001244		MOV	TRWC,REGIST	
2951	020564	104001				ERROR	1	;INCORRECT REGISTER MATCHUP
2952	020566	104404				TYPEL		
2953								
2954	020570	000072			A\$71:	STAGS=STAGS+1		
2955								
2956	020570	052777	004000	160604		BIS	#PWRCLR,@TRCR	;DEVICE MASTER RESET
2957	020576	032777	004000	160576	64\$:	BIT	#PWRCLR,@TRCR	;INSTRUCTION CLEAR ?
2958	020604	001374				BNE	64\$	;IF NO, WAIT FOR IT TO CLEAR
2959	020606	000207				RTS	PC	
2960								
2961								
2962								
2963								
2964	020610				TESTAD:			
2965	020610	004737	021712			JSR	PC,TESTAH	
2966	020614	004737	026610			JSR	PC,TESTCK	
2967	020620	004737	026630			JSR	PC,TESTEK	
2968	020624	004537	027546			JSR	R5,TRYIT	
2969	020630	000004				.WORD	READ	;FUNCTION
2970	020632	177677				.WORD	-101	;WORD COUNT
2971	020634	035640				.WORD	INPUT	;BUS ADDRESS

```

2972 020636 013727 001240      MOV    HOLD,(PC)+      ;PICK UP TIME PARAMETER
2973 020642 000000      64$: .WORD    0      ;USE THIS WORD AS A TIME COUNTER
2974 020644      66$:
2975 020644 005227 000000      INC     #0      ;IF NO,COUNT 1 OF 65535 TICKS
2976 020650 001375      BNE     66$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
2977 020652 005337 020642      DEC     64$      ;HAS THE TOTAL TIME ELAPSED?
2978 020656 001372      BNE     66$      ;IF NO,GO WAIT A LITTLE LONGER
2979 020660
2980
2981 020660 022777 102204 160514      CMP     #102204,@TRCR      ;TEST COMMAND REGISTER
2982 020666 001411      BEQ     A$72
2983 020670 012705 102204      MOV     #102204,R5
2984 020674 017704 160502      MOV     @TRCR,R4
2985 020700 013737 001402 001244      MOV     TRCR,REGIST
2986 020706 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
2987 020710 104404      TYPEL
2988
2989 020712 000073      A$72: $TAGAS=$TAGAS+1
2990
2991 020712 022777 003001 160466      CMP     #3001,@TRSR      ;TEST THE STATUS REGISTER
2992 020720 001411      BEQ     A$73
2993 020722 012705 003001      MOV     #3001,R5
2994 020726 017704 160454      MOV     @TRSR,R4
2995 020732 013737 001406 001244      MOV     TRSR,REGIST
2996 020740 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
2997 020742 104404      TYPEL
2998
2999 020744 000074      A$73: $TAGAS=$TAGAS+1
3000
3001 020744 022777 036040 160444      CMP     #INPUT+200,@TRBA      ;TEST BUFFER ADDRESS
3002 020752 001411      BEQ     A$74
3003 020754 012705 036040      MOV     #INPUT+200,R5
3004 020760 017704 160432      MOV     @TRBA,R4
3005 020764 013737 001416 001244      MOV     TRBA,REGIST
3006 020772 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3007 020774 104404      TYPEL
3008
3009 020776 000075      A$74: $TAGAS=$TAGAS+1
3010
3011 020776 022777 177777 160406      CMP     #-1,@TRWC      ;TEST THE WORD COUNT
3012 021004 001411      BEQ     A$75
3013 021006 012705 177777      MOV     #-1,R5
3014 021012 017704 160374      MOV     @TRWC,R4
3015 021016 013737 001412 001244      MOV     TRWC,REGIST
3016 021024 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3017 021026 104404      TYPEL
3018
3019 021030 000076      A$75: $TAGAS=$TAGAS+1
3020
3021 021030 052777 004000 160344      BIS     #PWRCLR,@TRCR      ;DEVICE MASTER RESET
3022 021036 032777 004000 160336 64$: BIT     #PWRCLR,@TRCR      ;INSTRUCTION CLEAR ?
3023 021044 001374      BNE     64$      ;IF NO, WAIT FOR IT TO CLEAR
3024 021046 000207      RTS     PC
3025
3026 -----*
3027 ;THIS ROUTINE CHECKS THE FUNCTION OF SPACING REVERSE WITH A

```

```

3028                                     ;SHORT WORD COUNT
3029
3030 TESTAE:
3031     021050 004737 021712      JSR      PC,TESTAH
3032     021054 004737 026610      JSR      PC,TESTCK
3033     021060 004537 027546      JSR      R5,TRYIT
3034     021064 001410              .WORD    SPACER          ;FUNCTION
3035     021066 177701              .WORD    -77             ;WORD COUNT
3036     021070 000000              .WORD    0               ;BUS ADDRESS
3037
3038     021072 022777 003610 160302  CMP      #3610,ATRCR      ;TEST COMMAND REGISTER
3039     021100 001411              BEQ      A$76
3040     021102 012705 003610      MOV      #3610,R5
3041     021106 017704 160270      MOV      ATRCR,R4
3042     021112 013737 001402 001244  MOV      ATRCR,REGIST
3043     021120 104001              ERROR    1               ;INCORRECT REGISTER MATCHUP
3044     021122 104404              TYPEL
3045
3046     021124 000077      A$76:   STAGAS=STAGAS+1
3047
3048     021124 022777 002001 160254  CMP      #2001,ATRSR      ;TEST THE STATUS REGISTER
3049     021132 001411              BEQ      A$77
3050     021134 012705 002001      MOV      #2001,R5
3051     021140 017704 160242      MOV      ATRSR,R4
3052     021144 013737 001406 001244  MOV      ATRSR,REGIST
3053     021152 104001              ERROR    1               ;INCORRECT REGISTER MATCHUP
3054     021154 104404              TYPEL
3055
3056     021156 000100      A$77:   STAGAS=STAGAS+1
3057
3058     021156 005777 160234      TST      ATRBA          ;TEST BUFFER ADDRESS
3059     021162 001411              BEQ      A$100
3060     021164 012705 000000      MOV      #0,R5
3061     021170 017704 160222      MOV      ATRBA,R4
3062     021174 013737 001416 001244  MOV      ATRBA,REGIST
3063     021202 104001              ERROR    1               ;INCORRECT REGISTER MATCHUP
3064     021204 104404              TYPEL
3065
3066     021206 000101      A$100:  STAGAS=STAGAS+1
3067
3068     021206 022777 177701 160176  CMP      #177701,ATRWC      ;TEST THE WORD COUNT
3069     021214 001411              BEQ      A$101
3070     021216 012705 177701      MOV      #177701,R5
3071     021222 017704 160164      MOV      ATRWC,R4
3072     021226 013737 001412 001244  MOV      ATRWC,REGIST
3073     021234 104001              ERROR    1               ;INCORRECT REGISTER MATCHUP
3074     021236 104404              TYPEL
3075
3076     021240 000102      A$101:  STAGAS=STAGAS+1
3077
3078     021240 000207      RTS      PC
3079
3080 -----*
3081 ;THIS ROUTINE CHECKS THE FUNCTION OF SPACING WITH A LONG WORD COUNT
3082
3083 TESTAF:

```

M05

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 65
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0064

```

3084 021242 004737 021712      JSR      PC,TESTAH
3085 021246 004737 026610      JSR      PC,TESTCK
3086 021252 004537 027546      JSR      RS,TRYIT
3087 021256 001410              .WORD     SPACER
3088 021260 177677              .WORD     -101
3089 021262 000000              .WORD     0
3090
3091 021264 022777 003610 160110  CMP      #3610,ATCR
3092 021272 001411              BEQ      AS102
3093 021274 012705 003610      MOV      #3610,R5
3094 021300 017704 160076      MOV      ATCR,R4
3095 021304 013737 001402 001244 MOV      TRCR,REGIST
3096 021312 104001              ERROR     1
3097 021314 104404              TYPEL
3098
3099 021316 000103              AS102:  STAGAS=STAGAS+1
3100
3101 021316 022777 002001 160062  CMP      #2001,ATRSR
3102 021324 001410              BEQ      AS103
3103 021326 012705 002001      MOV      #2001,R5
3104 021332 017704 160050      MOV      ATRSR,R4
3105 021336 013737 001406 001244 MOV      TRSR,REGIST
3106 021344 104001              ERROR     1
3107
3108 021346 000104              AS103:  STAGAS=STAGAS+1
3109
3110 021346 005777 160044      TST      ATBBA
3111 021352 001410              BEQ      AS104
3112 021354 012705 000000      MOV      #0,R5
3113 021360 017704 160032      MOV      ATBBA,R4
3114 021364 013737 001416 001244 MOV      TRBA,REGIST
3115 021372 104001              ERROR     1
3116
3117 021374 000105              AS104:  STAGAS=STAGAS+1
3118
3119 021374 022777 177677 160010  CMP      #177677,ATWRC
3120 021402 001410              BEQ      AS105
3121 021404 012705 177677      MOV      #177677,R5
3122 021410 017704 157776      MOV      ATWRC,R4
3123 021414 013737 001412 001244 MOV      TRWC,REGIST
3124 021422 104001              ERROR     1
3125
3126 021424 000106              AS105:  STAGAS=STAGAS+1
3127
3128 021424 000207              RTS      PC
3129
3130
3131
3132
3133
3134 021426
3135 021426 052777 004000 157746  TESTAG:  BIS      #PWRCLR,ATCR
3136 021434 032777 004000 157740 64$:     BIT      #PWRCLR,ATCR
3137 021442 001374              BNE      64$
3138 021444 012777 001026 157730      MOV      #GOEOT,ATCR
3139 021452 042777 000001 157726      BIC      #BIT0,ATRSR

```

; ; ;
 ; FUNCTION
 ; WORD COUNT
 ; BUS ADDRESS
 ; TEST COMMAND REGISTER
 ; INCORRECT REGISTER MATCHUP
 ; TEST THE STATUS REGISTER
 ; INCORRECT REGISTER MATCHUP
 ; TEST BUFFER ADDRESS
 ; INCORRECT REGISTER MATCHUP
 ; TEST THE WORD COUNT
 ; INCORRECT REGISTER MATCHUP
 ; DEVICE MASTER RESET
 ; INSTRUCTION CLEAR ?
 ; IF NO, WAIT FOR IT TO CLEAR
 ; SET COMMAND REGISTER
 ; CLEAR INHIBIT

-----*
 ; THIS ROUTINE TESTS THE END OF TAPE SENCING FUNCTION
 ; BY GOING FAST FORWARD TO END OF TAPE

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 66
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0065

3140	021460	012777	177777	157730		MOV	#-1, @TRBA	; SET BUFFER ADDRESS
3141	021466	012777	177777	157716		MOV	#-1, @TRWC	; SET WORD COUNT
3142	021474	005277	157702			INC	@TRCR	; SET THE GO BIT
3143								
3144		021500			A=.			
3145	021500	032777	000200	157700		BIT	#BIT7, @TRSR	
3146	021506	001774				BEQ	A	; BRANCH IF CLEAR
3147								
3148	021510	013727	001240			MOV	HOLD, (PC)+	; PICK UP TIME PARAMETER
3149	021514	000000			65\$:	.WORD	0	; USE THIS WORD AS A TIME COUNTER
3150	021516				67\$:			
3151	021516	005227	000000			INC	#0	; IF NO, COUNT 1 OF 65535 TICKS
3152	021522	001375				BNE	67\$	; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
3153	021524	005337	021514			DEC	65\$	; HAS THE TOTAL TIME ELAPSED?
3154	021530	001372				BNE	67\$	; IF NO, GO WAIT A LITTLE LONGER
3155	021532				66\$:			
3156	021532	004737	012524			JSR	PC, TESTA	
3157	021536	022777	003226	157636		CMP	#3226, @TRCR	; TEST COMMAND REGISTER
3158	021544	001412				BEQ	AS106	
3159	021546	012705	003226			MOV	#3226, R5	
3160	021552	017704	157624			MOV	@TRCR, R4	
3161	021556	013737	001402	001244		MOV	TRCR, REGIST	
3162	021564	104001				ERROR	1	; INCORRECT REGISTER MATCHUP
3163	021566	104403	007317			TYPEF	, METFL	
3164								
3165	021572	000107			AS106:	\$TAGAS=\$TAGAS+1		
3166								
3167	021572	022777	002201	157606		CMP	#2201, @TRSR	; TEST THE STATUS REGISTER
3168	021600	001411				BEQ	AS107	
3169	021602	012705	002201			MOV	#2201, R5	
3170	021606	017704	157574			MOV	@TRSR, R4	
3171	021612	013737	001406	001244		MOV	TRSR, REGIST	
3172	021620	104001				ERROR	1	; INCORRECT REGISTER MATCHUP
3173	021622	104404				TYPEL		
3174								
3175	021624	000110			AS107:	\$TAGAS=\$TAGAS+1		
3176								
3177	021624	022777	177777	157564		CMP	#-1, @TRBA	; TEST BUFFER ADDRESS
3178	021632	001411				BEQ	AS110	
3179	021634	012705	177777			MOV	#-1, R5	
3180	021640	017704	157552			MOV	@TRBA, R4	
3181	021644	013737	001416	001244		MOV	TRBA, REGIST	
3182	021652	104001				ERROR	1	; INCORRECT REGISTER MATCHUP
3183	021654	104404				TYPEL		
3184								
3185	021656	000111			AS110:	\$TAGAS=\$TAGAS+1		
3186								
3187	021656	022777	177777	157526		CMP	#-1, @TRWC	; TEST THE WORD COUNT
3188	021664	001411				BEQ	AS111	
3189	021666	012705	177777			MOV	#-1, R5	
3190	021672	017704	157514			MOV	@TRWC, R4	
3191	021676	013737	001412	001244		MOV	TRWC, REGIST	
3192	021704	104001				ERROR	1	; INCORRECT REGISTER MATCHUP
3193	021706	104404				TYPEL		
3194								
3195	021710	000112			AS111:	\$TAGAS=\$TAGAS+1		

```

3196
3197 021710 000207          RTS    PC
3198
3199
3200      ;-----*
3201      ; THIS ROUTINE INSURES THE TAPE IS NOT AT LOAD POINT AND THERE
3202      ; IS GOOD DATA ON THE TAPE
3203
3204 021712 052777 004000 157462 TESTAH:
3205 021720 032777 004000 157454 64$: BIS    #PWRCLR, @TRCR    ; DEVICE MASTER RESET
3206 021726 001374          BNE    64$    ; INSTRUCTION CLEAR ?
3207 021730 032777 000040 157450 64$: BIT    #PWRCLR, @TRCR    ; IF NO, WAIT FOR IT TO CLEAR
3208 021736 001402          BNE    64$
3209 021740 004737 014764 64$: BIT    #BITS, @TRSR
3210 021744 000113 64$: BEQ    A$112, @TRSR
3211 021744 004737 026650 64$: JSR    PC, TESTO
3212 021750 000207          JSR    PC, TESTFO
3213          RTS    PC
3214
3215      ;-----*
3216      ; THIS ROUTINE DELAYS ABOUT 10 SECONDS
3217
3218 021752 012737 000024 021776 TESTAJ:
3219 021760 005737 021776 2$: MOV    #20., 3$
3220 021764 001001          TST    3$
3221 021766 000207          BNE    1$
3222 021770 005337 021776 1$: RTS    PC
3223 021774 000401          DEC    3$
3224 021776 000000          BR     4$
3225 022000          3$: O
3226 022000 013727 001240 4$: MOV    HOLD, (PC)+    ; PICK UP TIME PARAMETER
3227 022004 000000          .WORD 0    ; USE THIS WORD AS A TIME COUNTER
3228 022006
3229 022006 005227 000000 64$: INC    #0    ; IF NO, COUNT 1 OF 65535 TICKS
3230 022012 001375          BNE    66$    ; HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
3231 022014 005337 022004 64$: DEC    64$    ; HAS THE TOTAL TIME ELAPSED?
3232 022020 001372          BNE    66$    ; IF NO, GO WAIT A LITTLE LONGER
3233 022022
3234 022022 000207          65$: RTS    PC
3235
3236      ;-----*
3237
3238      ; THIS ROUTINE TELLS THE OPERATOR TO RESET THE
3239      ; TAPE TRANSPORT AND TESTS THE REGISTERS AFTER HE DOES
3240
3241 022024
3242 022024 004737 013626 TESTAK:
3243 022030 104402 007336          JSR    PC, TESTI
3244          TYPE    ,MASTU
3245
3246 022034 022034 005237 010560 A=: INC    TEMP
3247 022040 001003 64$: BNE    64$
3248 022042 012777 000207 157142 64$: MOV    #207, @TPDBR    ; RING THE BELL
3249 022050
3250 022050 032777 002000 157330 64$: BIT    #BIT10, @TRSR
3251 022056 001366          BNE    A    ; BRANCH IF CLEAR

```

```

3252
3253
3254 022060 013727 001240      MOV    HOLD,(PC)+      ;PICK UP TIME PARAMETER
3255 022064 000000      65$: .WORD    0      ;USE THIS WORD AS A TIME COUNTER
3256 022066
3257 022066 005227 000000      67$: INC     #0      ;IF NO,COUNT 1 OF 65535 TICKS
3258 022072 001375      BNE     67$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
3259 022074 005337 022064      DEC     65$      ;HAS THE TOTAL TIME ELAPSED?
3260 022100 001372      BNE     67$      ;IF NO,GO WAIT A LITTLE LONGER
3261 022102
3262 022102 032777 100000 157272 66$: BIT     #100000,ATCR      ;TEST COMMAND REGISTER
3263 022110 001012      BNE     AS113
3264 022112 012705 100026      MOV     #100026,R5
3265 022116 017704 157260      MOV     ATCR,R4
3266 022122 013737 001402 001244 MOV     TRCR,REGIST
3267 022130 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3268 022132 104403 007361      TYPEF   ,MMRSFL
3269
3270 022136 000114      AS113: $TAGAS=$TAGAS+1
3271
3272 022136 017701 157244      MOV     ATRSR,R1      ;GET STATUS
3273 022142 042701 000040      BIC     #BITS,R1      ;CLEAR BIT 5
3274
3275 022146 022701 000001      CMP     #1,R1      ;TEST THE STATUS
3276 022152 001411      BEQ     AS114
3277 022154 012705 000001      MOV     #1,R5
3278 022160 017704 157222      MOV     ATRSR,R4
3279 022164 013737 001406 001244 MOV     TRSR,REGIST
3280 022172 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3281 022174 104404      TYPEL
3282
3283 022176 000115      AS114: $TAGAS=$TAGAS+1
3284
3285 022176 005777 157214      TST     ATBBA      ;TEST BUFFER ADDRESS
3286 022202 001411      BEQ     AS115
3287 022204 012705 000000      MOV     #0,R5
3288 022210 017704 157202      MOV     ATBBA,R4
3289 022214 013737 001416 001244 MOV     TRBA,REGIST
3290 022222 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3291 022224 104404      TYPEL
3292
3293 022226 000116      AS115: $TAGAS=$TAGAS+1
3294
3295 022226 005777 157160      TST     ATWBC      ;TEST THE WORD COUNT
3296 022232 001411      BEQ     AS116
3297 022234 012705 000000      MOV     #0,R5
3298 022240 017704 157146      MOV     ATWBC,R4
3299 022244 013737 001412 001244 MOV     TRWC,REGIST
3300 022252 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3301 022254 104404      TYPEL
3302
3303 022256 000117      AS116: $TAGAS=$TAGAS+1
3304
3305 022256 000207      RTS     PC
3306
3307 ;-----*

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 69
 CZTRAB.MED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0068

```

3308 ;THIS ROUTINE WRITES RECORDS PAST EOT
3309
3310 022260 032777 000200 157120 TESTAL:
3311 022260 001033 BIT #BIT7, @TRSR
3312 022266 012737 BNE AS117
3313 022270 012737 001026 027536 MOV #GOEOT, USEA ;SET FUNCTION
3314 022276 012737 000001 027540 MOV #-1, USEB ;SET WORD COUNT
3315 022304 012737 177777 027542 MOV #-1, USEC ;SET BUS ADDRESS
3316 022312 004737 030012 JSR PC, EOTTST
3317
3318 022316 032777 000200 157062 A=:
3319 022324 001774 BIT #BIT7, @TRSR
3320 022326 032777 000040 157046 BEQ A ;BRANCH IS CLEAR
3321 022334 001370 BIT #BIT5, @TRCR
3322 022336 032777 002000 157036 BNE A ;BRANCH IF SET
3323 022344 001764 BIT #BIT10, @TRCR
3324 022346 032777 000200 157026 BEQ A ;BRANCH IF CLEAR
3325 022354 001760 BIT #BIT7, @TRCR
3326 022356 032777 000016 027536 BEQ A ;BRANCH IF CLEAR
3327 022356 012737 177000 027540 AS117: $TAGS=$TAGS+1
3328 022364 012737 031600 027542 MOV #ERASE, USEA ;SET FUNCTION
3329 022372 012737 030012 JSR PC, EOTTST ;SET WORD COUNT
3330 022400 004737 004000 156770 MOV #-1000, USEB ;SET BUS ADDRESS
3331 022412 032777 004000 156762 BIS #PWCLR, @TRCR ;DEVICE MASTER RESET
3332 022420 001374 BIT #PWCLR, @TRCR ;INSTRUCTION CLEAR ?
3333 022422 012737 000012 022446 64$: BNE 64$ ;IF NO, WAIT FOR IT TO CLEAR
3334 022430 005737 022446 2$: MOV #10., 3$
3335 022434 001001 TST 3$
3336 022436 000207 BNE 1$
3337 022440 005337 022446 1$: RTS PC
3338 022444 000401 DEC 3$
3339 022446 000000 BR 4$
3340 022450 012737 000402 027536 MOV #WRITE, USEA ;SET FUNCTION
3341 022456 012737 177700 027540 MOV #-100, USEB ;SET WORD COUNT
3342 022464 012737 031600 027542 MOV #OUTPUT, USEC ;SET BUS ADDRESS
3343 022472 004737 030012 JSR PC, EOTTST
3344
3345 022476 022777 002602 156676 CMP #2602, @TRCR ;TEST COMMAND REGISTER
3346 022504 001412 BEQ AS120
3347 022506 012705 002602 MOV #2602, R5
3348 022512 017704 156664 MOV @TRCR, R4
3349 022516 013737 001402 001244 MOV TRCR, REGIST
3350 022524 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
3351 022526 104403 007401 TYPEF ,MWPETFL
3352
3353 022532 000121 AS120: $TAGS=$TAGS+1
3354
3355 022532 021777 002201 156646 CMP #2201, @TRSR ;TEST THE STATUS REGISTER
3356 022540 001411 BEQ AS121
3357 022542 012705 002201 MOV #2201, R5
3358 022546 017704 156634 MOV @TRSR, R4

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 70
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0069

```

3364 022552 013737 001406 001244      MOV    TRSR,REGIST
3365 022560 104001      ERROR    1          ;INCORRECT REGISTER MATCHUP
3366 022562 104404      TYPEL
3367
3368 022564 000122      AS121:  STAGAS=STAGAS+1
3369
3370
3371 022564 005777 156622      TST     @TRWC          ;TEST THE WORD COUNT
3372 022570 001411      BEQ     AS122
3373 022572 012705 000000      MOV     @0,R5
3374 022576 017704 156610      MOV     @TRWC,R4
3375 022602 013737 001412 001244      MOV     TRWC,REGIST
3376 022610 104001      ERROR    1          ;INCORRECT REGISTER MATCHUP
3377 022612 104404      TYPEL
3378
3379 022614 000123      AS122:  STAGAS=STAGAS+1
3380
3381 022614 000207      RTS     PC
3382
3383 ;-----*
3384 ;;THIS ROUTINE TESTS THE FUNCTION OF SPACE REVERSE PAST EOT
3385
3386 022616
3387 022616 004737 022260      TESTAM: JSR     PC,TESTAL
3388
3389 022622 012737 001410 027536      MOV     @SPACER,USEA      ;SET FUNCTION
3390 022630 012737 000001 027540      MOV     @--1,USEB      ;SET WORD COUNT
3391 022636 012737 177777 027542      MOV     @-1,USEC      ;SET BUS ADDRESS
3392 022644 004737 030012      JSR     PC,EOTTST
3393
3394 022650 022777 003610 156524      CMP     @3610,@TRCR      ;TEST COMMAND REGISTER
3395 022656 001412      BEQ     AS123
3396 022660 012705 003610      MOV     @3610,R5
3397 022664 017704 156512      MOV     @TRCR,R4
3398 022670 013737 001402 001244      MOV     TRCR,REGIST
3399 022676 104001      ERROR    1          ;INCORRECT REGISTER MATCHUP
3400 022700 104403 007434      TYPEF    ,MSRETFI
3401
3402 022704 000124      AS123:  STAGAS=STAGAS+1
3403
3404 022704 022777 002201 156474      CMP     @2201,@TRSR      ;TEST THE STATUS REGISTER
3405 022712 001411      BEQ     AS124
3406 022714 012705 002201      MOV     @2201,R5
3407 022720 017704 156462      MOV     @TRSR,R4
3408 022724 013737 001406 001244      MOV     TRSR,REGIST
3409 022732 104001      ERROR    1          ;INCORRECT REGISTER MATCHUP
3410 022734 104404      TYPEL
3411
3412 022736 000125      AS124:  STAGAS=STAGAS+1
3413
3414 022736 022777 177777 156452      CMP     @-1,@TRBA      ;TEST BUFFER ADDRESS
3415 022744 001411      BEQ     AS125
3416 022746 012705 177777      MOV     @-1,R5
3417 022752 017704 156440      MOV     @TRBA,R4
3418 022756 013737 001416 001244      MOV     TRBA,REGIST
3419 022764 104001      ERROR    1          ;INCORRECT REGISTER MATCHUP

```

```

3420 022766 104404          TYPEL
3421
3422 022770 000126          AS125: STAGAS=STAGAS+1
3423
3424 022770 022777 000001 156414      CMP      #1, @TRWC          ;TEST THE WORD COUNT
3425 022776 001411          BEQ      AS126
3426 023000 012705 000001          MOV      #1, R5
3427 023004 017704 156402          MOV      @TRWC, R4
3428 023010 013737 001412 001244      MOV      TRWC, REGIST
3429 023016 104001          ERROR      1          ;INCORRECT REGISTER MATCHUP
3430 023020 104404          TYPEL
3431
3432 023022 000127          AS126: STAGAS=STAGAS+1
3433
3434 023022 000207          RTS      PC
3435
3436
3437
3438
3439
3440
3441 023024 012737 000004 027536      TESTAN: MOV      #READ, USEA          ;SET FUNCTION
3442 023032 012737 177700 027540      MOV      #-100, USEB          ;SET WORD COUNT
3443 023040 012737 035640 027542      MOV      #INPUT, USEC          ;SET BUS ADDRESS
3444 023046 004737 030012          JSR      PC, EOT1ST
3445
3446 023052 022777 002204 156322      CMP      #2204, @TRCR          ;TEST COMMAND REGISTER
3447 023060 001412          BEQ      AS127
3448 023062 012705 002204          MOV      #2204, R5
3449 023066 017704 156310          MOV      @TRCR, R4
3450 023072 013737 001402 001244      MOV      TRCR, REGIST
3451 023100 104001          ERROR      1          ;INCORRECT REGISTER MATCHUP
3452 023102 104403 007476          TYPEF      ,MRPETFL
3453
3454 023106 000130          AS127: STAGAS=STAGAS+1
3455
3456 023106 022777 002201 156272      CMP      #2201, @TRSR          ;TEST THE STATUS REGISTER
3457 023114 001411          BEQ      AS130
3458 023116 012705 002201          MOV      #2201, R5
3459 023122 017704 156260          MOV      @TRSR, R4
3460 023126 013737 001406 001244      MOV      TRSR, REGIST
3461 023134 104001          ERROR      1          ;INCORRECT REGISTER MATCHUP
3462 023136 104404          TYPEL
3463
3464 023140 000131          AS130: STAGAS=STAGAS+1
3465
3466
3467 023140 005777 156246          TST      @TRWC          ;TEST THE WORD COUNT
3468 023144 001411          BEQ      AS131
3469 023146 012705 000000          MOV      #0, R5
3470 023152 017704 156234          MOV      @TRWC, R4
3471 023156 013737 001412 001244      MOV      TRWC, REGIST
3472 023164 104001          ERROR      1          ;INCORRECT REGISTER MATCHUP
3473 023166 104404          TYPEL
3474
3475 023170 000132          AS131: STAGAS=STAGAS+1

```

```

3476
3477 023170 000207          RTS      PC
3478
3479          ;-----*
3480          ;THIS ROUTINE WRITES A EOF PAST END OF TAPE
3481
3482 023172          TESTAO:
3483 023172 004737 022260      JSR      PC,TESTAL
3484
3485 023176 012737 001034 027536      MOV      #WEOF,USEA          ;SET FUNCTION
3486 023204 012737 000001 027540      MOV      #--1,USEB          ;SET WORD COUNT
3487 023212 012737 177777 027542      MOV      #-1,USEC          ;SET BUS ADDRESS
3488 023220 004737 030012          JSR      PC,EOFTST
3489
3490 023224 022777 003234 156150      CMP      #3234,@TRCR          ;TEST COMMAND REGISTER
3491 023232 001412          BEQ      AS132
3492 023234 012705 003234          MOV      #3234,R5
3493 023240 017704 156136          MOV      @TRCR,R4
3494 023244 013737 001402 001244      MOV      TRCR,REGIST
3495 023252 104001          ERROR      1          ;INCORRECT REGISTER MATCHUP
3496 023254 104403 007527          TYPEF      ,MWEFETF
3497
3498 023260 000133          AS132: STAGAS=STAGAS+1
3499
3500 023260 022777 002211 156120      CMP      #2211,@TRSR          ;TEST THE STATUS REGISTER
3501 023266 001411          BEQ      AS133
3502 023270 012705 002211          MOV      #2211,R5
3503 023274 017704 156106          MOV      @TRSR,R4
3504 023300 013737 001406 001244      MOV      TRSR,REGIST
3505 023306 104001          ERROR      1          ;INCORRECT REGISTER MATCHUP
3506 023310 104404          TYPEL
3507
3508 023312 000134          AS133: STAGAS=STAGAS+1
3509
3510 023312 022777 177777 156076      CMP      #-1,@TRBA          ;TEST BUFFER ADDRESS
3511 023320 001411          BEQ      AS134
3512 023322 012705 177777          MOV      #-1,R5
3513 023326 017704 156064          MOV      @TRBA,R4
3514 023332 013737 001416 001244      MOV      TRBA,REGIST
3515 023340 104001          ERROR      1          ;INCORRECT REGISTER MATCHUP
3516 023342 104404          TYPEL
3517
3518 023344 000135          AS134: STAGAS=STAGAS+1
3519
3520 023344 022777 000001 156040      CMP      #1,@TRWC          ;TEST THE WORD COUNT
3521 023352 001411          BEQ      AS135
3522 023354 012705 000001          MOV      #1,R5
3523 023360 017704 156026          MOV      @TRWC,R4
3524 023364 013737 001412 001244      MOV      TRWC,REGIST
3525 023372 104001          ERROR      1          ;INCORRECT REGISTER MATCHUP
3526 023374 104404          TYPEL
3527
3528 023376 000136          AS135: STAGAS=STAGAS+1
3529
3530 023376 000207          RTS      PC
3531

```

```

3532
3533
3534
3535
3536 023400
3537 023400 004737 023172
3538
3539 023404 012737 001410 027536
3540 023412 012737 000001 027540
3541 023420 012737 177777 027542
3542 023426 004737 030012
3543
3544 023432 022777 003610 155742
3545 023440 001412
3546 023442 012705 003610
3547 023446 017704 155730
3548 023452 013737 001402 001244
3549 023460 104001
3550 023462 104403 007600
3551
3552 023466 000137
3553
3554 023466 022777 002211 155712
3555 023474 001411
3556 023476 012705 002211
3557 023502 017704 155700
3558 023506 013737 001406 001244
3559 023514 104001
3560 023516 104404
3561
3562 023520 000140
3563
3564 023520 022777 177777 155670
3565 023526 001411
3566 023530 012705 177777
3567 023534 017704 155656
3568 023540 013737 001416 001244
3569 023546 104001
3570 023550 104404
3571
3572 023552 000141
3573
3574 023552 022777 000001 155632
3575 023560 001411
3576 023562 012705 000001
3577 023566 017704 155620
3578 023572 013737 001412 001244
3579 023600 104001
3580 023602 104404
3581
3582 023604 000142
3583
3584 023604 000207
3585
3586
3587

```

-----*

```

; THIS ROUTINE TESTS THE FUNCTION OF SPACING REVERSE OVER
; END OF FILE PAST EOT

TESTAP:
      JSR      PC,TESTAO
      MOV      #SPACER,USEA      ;SET FUNCTION
      MOV      #-1,USEB          ;SET WORD COUNT
      MOV      #-1,USEC          ;SET BUS ADDRESS
      JSR      PC,EOTTST

      CMP      #3610,ATRCR      ;TEST COMMAND REGISTER
      BEQ      AS136
      MOV      #3610,R5
      MOV      ATRCR,R4
      MOV      TRCR,REGIST
      ERROR    1                ;INCORRECT REGISTER MATCHUP
      TYPEF    ,MREFETF

AS136:  STAGAS=STAGAS+1

      CMP      #2211,ATRSR      ;TEST THE STATUS REGISTER
      BEQ      AS137
      MOV      #2211,R5
      MOV      ATRSR,R4
      MOV      TRSR,REGIST
      ERROR    1                ;INCORRECT REGISTER MATCHUP
      TYPEL

AS137:  STAGAS=STAGAS+1

      CMP      #-1,ATRBA        ;TEST BUFFER ADDRESS
      BEQ      AS140
      MOV      #-1,R5
      MOV      ATRBA,R4
      MOV      TRBA,REGIST
      ERROR    1                ;INCORRECT REGISTER MATCHUP
      TYPEL

AS140:  STAGAS=STAGAS+1

      CMP      #1,ATRCW        ;TEST THE WORD COUNT
      BEQ      AS141
      MOV      #1,R5
      MOV      ATRWC,R4
      MOV      TRWC,REGIST
      ERROR    1                ;INCORRECT REGISTER MATCHUP
      TYPEL

AS141:  STAGAS=STAGAS+1

      RTS      PC

;-----*
; THIS ROUTINE TESTS THE FUNCTION OF READING A EOF PAST EOT

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 74
 CZTRAB.MED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0073

```

3588
3589 023606          TESTAQ: JSR      PC,TESTAP
3590 023606 004737 023400
3591
3592 023612 012737 000004 027536      MOV      #READ,USEA      ;SET FUNCTION
3593 023620 012737 000001 027540      MOV      #-1,USEB      ;SET WORD COUNT
3594 023626 012737 177777 027542      MOV      #-1,USEC      ;SET BUS ADDRESS
3595 023634 004737 030012
3596
3597 023640 022777 002204 155534      CMP      #2204,@TRCR      ;TEST COMMAND REGISTER
3598 023646 001412
3599 023650 012705 002204      BEQ      AS142
3600 023654 017704 155522      MOV      #2204,R5
3601 023660 013737 001402 001244      MOV      @TRCR,R4
3602 023666 104001      MOV      TRCR,REGIST
3603 023670 104403 007600      ERROR    1      ;INCORRECT REGISTER MATCHUP
3604      TYPEF    ,MREFETF
3605 023674 000143      AS142: STAGAS=STAGAS+1
3606
3607 023674 022777 002211 155504      CMP      #2211,@TRSR      ;TEST THE STATUS REGISTER
3608 023702 001411      BEQ      AS143
3609 023704 012705 002211      MOV      #2211,R5
3610 023710 017704 155472      MOV      @TRSR,R4
3611 023714 013737 001406 001244      MOV      TRSR,REGIST
3612 023722 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3613 023724 104404      TYPEL
3614
3615 023726 000144      AS143: STAGAS=STAGAS+1
3616
3617 023726 022777 177777 155462      CMP      #-1,@TRBA      ;TEST BUFFER ADDRESS
3618 023734 001411      BEQ      AS144
3619 023736 012705 177777      MOV      #-1,R5
3620 023742 017704 155450      MOV      @TRBA,R4
3621 023746 013737 001416 001244      MOV      TRBA,REGIST
3622 023754 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3623 023756 104404      TYPEL
3624
3625 023760 000145      AS144: STAGAS=STAGAS+1
3626
3627 023760 022777 000001 155424      CMP      #1,@TRWC      ;TEST THE WORD COUNT
3628 023766 001411      BEQ      AS145
3629 023770 012705 000001      MOV      #1,R5
3630 023774 017704 155412      MOV      @TRWC,R4
3631 024000 013737 001412 001244      MOV      TRWC,REGIST
3632 024006 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3633 024010 104404      TYPEL
3634
3635 024012 000146      AS145: STAGAS=STAGAS+1
3636
3637 024012 000207      RTS      PC
3638
3639 -----*
3640 ;THIS ROUTINE TRIES TO WRITE ON TAPE WITHOUT A WRITE
3641 ;RING
3642
3643 024014      RUNAS:

```

CZTRAB0 MACY11 30(1046) 14-DEC-77 12:32 PAGE 75
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0074

```

3644 024014 012737 000012 024040      MOV      #10.,3$
3645 024022 005737 024040      2$:      TST      3$
3646 024026 001001              BNE      1$
3647 024030 000207              RTS      PC
3648 024032 005337 024040      1$:      DEC      3$
3649 024036 000401              BR       4$
3650 024040 000000      3$:      0
3651 024042      4$:
3652
3653 024042      TESTAS:
3654 024042 052777 004000 155332      BIS      #PWRCLR,@TRCR ;DEVICE MASTER RESET
3655 024050 032777 004000 155324      64$:      BIT      #PWRCLR,@TRCR ;INSTRUCTION CLEAR ?
3656 024056 001374              BNE      64$ ;IF NO, WAIT FOR IT TO CLEAR
3657 024060 004537 027546      JSR      RS,TRYIT ;...
3658 024064 000402              .WORD   WRITE ;FUNCTION
3659 024066 177700              .WORD   -100 ;WORD COUNT
3660 024070 031600              .WORD   OUTPUT ;BUS ADDRESS
3661
3662 024072 013727 001240      MOV      HOLD,(PC)+ ;PICK UP TIME PARAMETER
3663 024076 000000      65$:      .WORD   0 ;USE THIS WORD AS A TIME COUNTER
3664 024100      67$:
3665 024100 005227 000000      INC      #0 ;IF NO,COUNT 1 OF 65535 TICKS
3666 024104 001375              BNE      67$ ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
3667 024106 005337 024076      DEC      65$ ;HAS THE TOTAL TIME ELAPSED?
3668 024112 001372              BNE      67$ ;IF NO,GO WAIT A LITTLE LONGER
3669 024114
3670
3671 024114 022777 142602 155260      CMP      #142602,@TRCR ;TEST COMMAND REGISTER
3672 024122 001412      BEQ      AS146
3673 024124 012705 142602      MOV      #142602,R5
3674 024130 017704 155246      MOV      @TRCR,R4
3675 024134 013737 001402 001244      MOV      TRCR,REGIST
3676 024142 104001      ERROR   1 ;INCORRECT REGISTER MATCHUP
3677 024144 104403 007650      TYPEF   ,MWENFL
3678
3679 024150 000147      AS146:  STAGAS=STAGAS+1
3680
3681 024150 017701 155232      MOV      @TRSR,R1 ;GET STATUS REGISTER
3682 024154 042701 000040      BIC      #BITS,R1 ;CLEAR UNWANTED BITS
3683
3684 024160 022701 002004      CMP      #2004,R1 ;TEST THE STATUS REGISTER
3685 024164 001411      BEQ      AS147
3686 024166 012705 002004      MOV      #2004,R5
3687 024172 017704 155210      MOV      @TRSR,R4
3688 024176 013737 001406 001244      MOV      TRSR,REGIST
3689 024204 104001      ERROR   1 ;INCORRECT REGISTER MATCHUP
3690 024206 104404      TYPEL
3691
3692 024210 000150      AS147:  STAGAS=STAGAS+1
3693
3694
3695 024210 022777 031600 155200      CMP      #OUTPUT,@TRBA ;TEST BUFFER ADDRESS
3696 024216 001411      BEQ      AS150
3697 024220 012705 031600      MOV      #OUTPUT,R5
3698 024224 017704 155166      MOV      @TRBA,R4
3699 024230 013737 001416 001244      MOV      TRBA,REGIST

```

```

3700 024236 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
3701 024240 104404          TYPEL
3702
3703 024242 000151          AS150: STAGAS=STAGAS+1
3704
3705 024242 022777 177700 155142      CMP      #177700,@TRWC      ;TEST THE WORD COUNT
3706 024250 001411          BEQ      AS151
3707 024252 012705 177700      MOV      #177700,R5
3708 024256 017704 155130      MOV      @TRWC,R4
3709 024262 013737 001412 001244      MOV      TRWC,REGIST
3710 024270 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
3711 024272 104404          TYPEL
3712
3713 024274 000152          AS151: STAGAS=STAGAS+1
3714
3715 024274 052777 004000 155100      BIS      #PWRCLR,@TRCR      ;DEVICE MASTER RESET
3716 024302 032777 004000 155072      BIT      #PWRCLR,@TRCR      ;INSTRUCTION CLEAR ?
3717 024310 001374          BNE      64$
3718 024312 000207          RTS      PC      ;IF NO, WAIT FOR IT TO CLEAR
3719
3720          ;-----*
3721          ;THIS ROUTINE TRIES TO SPACE REVERSE AT LOAD POINT
3722
3723          TESTAT:
3724 024314 004737 013626      JSR      PC,TESTI
3725 024320 004537 027546      JSR      RS,TRYIT
3726 024324 001410          .WORD    SPACER      ;:FUNCTION
3727 024326 177777          .WORD    -1          ;:WORD COUNT
3728 024330 000000          .WORD    0          ;:BUS ADDRESS
3729
3730
3731 024332 022777 143610 155042      CMP      #143610,@TRCR      ;TEST COMMAND REGISTER
3732 024340 001412          BEQ      AS152
3733 024342 012705 143610      MOV      #143610,R5
3734 024346 017704 155030      MOV      @TRCR,R4
3735 024352 013737 001402 001244      MOV      TRCR,REGIST
3736 024360 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
3737 024362 104403 007670          TYPEF      ,MSRLPFL
3738
3739 024366 000153          AS152: STAGAS=STAGAS+1
3740
3741 024366 022777 002040 155012      CMP      #2040,@TRSR      ;TEST THE STATUS REGISTER
3742 024374 001411          BEQ      AS153
3743 024376 012705 002040      MOV      #2040,R5
3744 024402 017704 155000      MOV      @TRSR,R4
3745 024406 013737 001406 001244      MOV      TRSR,REGIST
3746 024414 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
3747 024416 104404          TYPEL
3748
3749 024420 000154          AS153: STAGAS=STAGAS+1
3750
3751 024420 005777 154772          TST      @TRBA          ;TEST BUFFER ADDRESS
3752 024424 001411          BEQ      AS154
3753 024426 012705 000000      MOV      #0,R5
3754 024432 017704 154760      MOV      @TRBA,R4
3755 024436 013737 001416 001244      MOV      TRBA,REGIST

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 77
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0076

```

3756 024444 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
3757 024446 104404          TYPEL
3758
3759 024450 000155          AS154: STAGAS=STAGAS+1
3760
3761 024450 022777 177777 154734      CMP      #-1, @TRWC          ;TEST THE WORD COUNT
3762 024456 001411          BEQ      AS155
3763 024460 012705 177777          MOV      #-1, R5
3764 024464 017704 154722          MOV      @TRWC, R4
3765 024470 013737 001412 001244      MOV      TRWC, REGIST
3766 024476 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
3767 024500 104404          TYPEL
3768
3769 024502 000156          AS155: STAGAS=STAGAS+1
3770
3771 024502 052777 004000 154672      BIS      #PWRCLR, @TRCR      ;DEVICE MASTER RESET
3772 024510 032777 004000 154664      BIT      #PWRCLR, @TRCR      ;INSTRUCTION CLEAR ?
3773 024516 001374          BNE      64$          ;IF NO, WAIT FOR IT TO CLEAR
3774 024520 000207          RTS      PC
3775
3776
3777
3778          ;-----*
3779          ;THIS ROUTINE TESTS THE BUS ADDRESS BITS BY USING A COUNT PATTERN
3780
3781 024522 005001          TESTAU: CLR      R1          ;CLEAR COUNT
3782 024524          A=.
3783 024524 010177 154666          MOV      R1, @TRBA          ;SET BUS ADDRESS
3784 024530 020177 154662          CMP      R1, @TRBA          ;TEST BUS ADDRESS
3785 024534 001407          BEQ      AS156
3786 024536 010105          MOV      R1, R5
3787 024540 017704 154652          MOV      @TRBA, R4
3788 024544 013737 001416 001244      MOV      TRBA, REGIST
3789 024552 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
3790
3791 024554 000157          AS156: STAGAS=STAGAS+1
3792 024554 005201          INC      R1          ;INCREMENT COUNT
3793 024556 001362          BNE      A          ;CONTINUE
3794 024560 000207          RTS      PC
3795
3796
3797
3798          ;-----*
3799          ;THIS ROUTINE SPACES REVERSE OVER ID BLOCK
3800
3801 024562          TESTAV:
3802 024562 004737 014764          JSR      PC, TESTO
3803
3804 024566 004537 027546          JSR      R5, TRYIT
3805 024572 001410          .WORD      SPACER          ;FUNCTION
3806 024574 177777          .WORD      -1          ;WORD COUNT
3807 024576 177777          .WORD      -1          ;BUS ADDRESS
3808
3809 024600 022777 003610 154574      CMP      #3610, @TRCR      ;TEST COMMAND REGISTER
3810 024606 001412          BEQ      AS157
3811 024610 012705 003610          MOV      #3610, R5

```

```

3812 024614 017704 154562      MOV      @TRCR,R4
3813 024620 013737 001402 001244  MOV      TRCR,REGIST
3814 024626 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
3815 024630 104403 007727      TYPEF      ,MSRIFL
3816
3817 024634 000160      AS157:  STAGAS=STAGAS+1
3818
3819 024634 022777 002061 154544      CMP      #2061,@TRSR      ;TEST THE STATUS REGISTER
3820 024642 001411      BEQ      AS160
3821 024644 012705 002061      MOV      #2061,R5
3822 024650 017704 154532      MOV      @TRSR,R4
3823 024654 013737 001406 001244      MOV      TRSR,REGIST
3824 024662 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
3825 024664 104404      TYPEL
3826
3827 024666 000161      AS160:  STAGAS=STAGAS+1
3828
3829 024666 022777 177777 154522      CMP      #-1,@TRBA      ;TEST BUFFER ADDRESS
3830 024674 001411      BEQ      AS161
3831 024676 012705 177777      MOV      #-1,R5
3832 024702 017704 154510      MOV      @TRBA,R4
3833 024706 013737 001416 001244      MOV      TRBA,REGIST
3834 024714 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
3835 024716 104404      TYPEL
3836
3837 024720 000162      AS161:  STAGAS=STAGAS+1
3838
3839 024720 022777 177777 154464      CMP      #-1,@TRWC      ;TEST THE WORD COUNT
3840 024726 001411      BEQ      AS162
3841 024730 012705 177777      MOV      #-1,R5
3842 024734 017704 154452      MOV      @TRWC,R4
3843 024740 013737 001412 001244      MOV      TRWC,REGIST
3844 024746 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
3845 024750 104404      TYPEL
3846
3847 024752 000163      AS162:  STAGAS=STAGAS+1
3848
3849 024752 052777 004000 154422      BIS      #PWRCLR,@TRCR      ;DEVICE MASTER RESET
3850 024760 032777 004000 154414 64$:  BIT      #PWRCLR,@TRCR      ;INSTRUCTION CLEAR ?
3851 024766 001374      BNE      64$      ;IF NO, WAIT FOR IT TO CLEAR
3852 024770 000207      RTS      PC
3853
3854
3855 ;-----*
3856 ;THIS ROUTINE READS A ID BLOCK
3857
3858 024772      TESTAW:
3859 024772 004737 013626      JSR      PC,TESTI
3860 024776 005037 035640      CLR      INPUT      ;CLEAR INPUT AREA
3861
3862 025002 004537 027546      JSR      R5,TRYIT
3863 025006 000004      .WORD   READ      ;FUNCTION
3864 025010 177777      .WORD   -1      ;WORD COUNT
3865 025012 035640      .WORD   INPUT     ;BUS ADDRESS
3866
3867 025014 022777 002204 154360      CMP      #2204,@TRCR      ;TEST COMMAND REGISTER

```

```

3868 025022 001412      BEQ      A$163
3869 025024 012705 002204  MOV     #2204,R5
3870 025030 017704 154346  MOV     @TRCR,R4
3871 025034 013737 001402 001244  MOV     TRCR,REGIST
3872 025042 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3873 025044 104403 007764  TYPEF   ,MRIBFL
3874
3875 025050 000164      A$163:  STAGAS=STAGAS+1
3876
3877 025050 022777 002021 154330  CMP     #2021,@TRSR      ;TEST THE STATUS REGISTER
3878 025056 001411      BEQ      A$164
3879 025060 012705 002021  MOV     #2021,R5
3880 025064 017704 154316  MOV     @TRSR,R4
3881 025070 013737 001406 001244  MOV     TRSR,REGIST
3882 025076 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3883 025100 104404      TYPEL
3884
3885 025102 000165      A$164:  STAGAS=STAGAS+1
3886
3887 025102 022777 035640 154306  CMP     #INPUT,@TRBA    ;TEST BUFFER ADDRESS
3888 025110 001411      BEQ      A$165
3889 025112 012705 035640  MOV     #INPUT,R5
3890 025116 017704 154274  MOV     @TRBA,R4
3891 025122 013737 001416 001244  MOV     TRBA,REGIST
3892 025130 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3893 025132 104404      TYPEL
3894
3895 025134 000166      A$165:  STAGAS=STAGAS+1
3896
3897 025134 022777 177777 154250  CMP     #-1,@TRWC      ;TEST THE WORD COUNT
3898 025142 001411      BEQ      A$166
3899 025144 012705 177777  MOV     #-1,R5
3900 025150 017704 154236  MOV     @TRWC,R4
3901 025154 013737 001412 001244  MOV     TRWC,REGIST
3902 025162 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3903 025164 104404      TYPEL
3904
3905 025166 000167      A$166:  STAGAS=STAGAS+1
3906
3907 025166 005737 035640      TST     INPUT      ;TEST INPUT AREA
3908 025172 001411      BEQ      A$167
3909 025174 012705 000000  MOV     #0,R5
3910 025200 013704 035640  MOV     INPUT,R4
3911 025204 012737 035640 001244  MOV     #INPUT,REGIST
3912 025212 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3913 025214 104404      TYPEL
3914 025216 000170      A$167:  STAGAS=STAGAS+1
3915 025216 052777 004000 154156  BIS     #PWRCLR,@TRCR  ;DEVICE MASTER RESET
3916 025224 032777 004000 154150 64$:  BIT     #PWRCLR,@TRCR ;INSTRUCTION CLEAR ?
3917 025232 001374      BNE     64$      ;IF NO, WAIT FOR IT TO CLEAR
3918
3919 025234 000207      RTS     PC
3920
3921 -----*
3922 ;THIS ROUTINE WRITES EVEN PARITY
3923

```

```

3924 025236          TESTAX:
3925 025236 004737 021712      JSR      PC,TESTAH
3926
3927 025242 004537 027546      JSR      R5,TRYIT
3928 025246 000402              .WORD   WRITE
3929 025250 177700              .WORD   -100
3930 025252 031574              .WORD   BADCUT
3931 025254 013727 001240      MOV      HOLD,(PC)+
3932 025260 000000      64$:      .WORD   0
3933 025262              66$:
3934 025262 005227 000000      INC      #0
3935 025266 001375              BNE      66$
3936 025270 005337 025260      DEC      64$
3937 025274 001372              BNE      66$
3938 025276              65$:
3939
3940 025276 022777 102602 154076  CMP      #102602,@TRCR ;TEST COMMAND REGISTER
3941 025304 001412      BEQ      AS170
3942 025306 012705 102602      MOV      #102602,R5
3943 025312 017704 154064      MOV      @TRCR,R4
3944 025316 013737 001402 001244  MOV      TRCR,REGIST
3945 025324 104001      ERROR      1 ;INCORRECT REGISTER MATCHUP
3946 025326 104403 010005      TYPEF      ,MPARFL
3947
3948 025332 000171      AS170:  STAGAS=STAGAS+1
3949
3950 025332 022777 162001 154046  CMP      #162001,@TRSR ;TEST THE STATUS REGISTER
3951 025340 001411      BEQ      AS171
3952 025342 012705 162001      MOV      #162001,R5
3953 025346 017704 154034      MOV      @TRSR,R4
3954 025352 013737 001406 001244  MOV      TRSR,REGIST
3955 025360 104001      ERROR      1 ;INCORRECT REGISTER MATCHUP
3956 025362 104404      TYPEL
3957
3958 025364 000172      AS171:  STAGAS=STAGAS+1
3959
3960 025364 013701 027540      MOV      USEB,R1 ;GET WORD COUNT
3961 025370 005101      COM      R1 ;FIX IT
3962 025372 005201      INC      R1 ;INCREMENT IT
3963 025374 006301      ASL      R1 ;DOUBLE IT
3964 025376 062701 031574      ADD      #BADOUT,R1 ;FINISH IT
3965
3966 025402 020177 154010      CMP      R1,@TRBA ;TEST BUFFER ADDRESS
3967 025406 001410      BEQ      AS172
3968 025410 010105      MOV      R1,R5
3969 025412 017704 154000      MOV      @TRBA,R4
3970 025416 013737 001416 001244  MOV      TRBA,REGIST
3971 025424 104001      ERROR      1 ;INCORRECT REGISTER MATCHUP
3972 025426 104404      TYPEL
3973
3974 025430 000173      AS172:  STAGAS=STAGAS+1
3975
3976 025430 005777 153756      TST      @TRWC ;TEST THE WORD COUNT
3977 025434 001411      BEQ      AS173
3978 025436 012705 000000      MOV      #0,R5
3979 025442 017704 153744      MOV      @TRWC,R4

```

```

3980 025446 013737 001412 001244      MOV    TRWC,REGIST
3981 025454 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
3982 025456 104404      TYPEL
3983
3984 025460 000174      AS173:  STAGAS=STAGAS+1
3985
3986 025460 052777 004000 153714      BIS    #PWRCLR,ATCR  ;DEVICE MASTER RESET
3987 025466 032777 004000 153706 64$:  BIT    #PWRCLR,ATCR  ;INSTRUCTION CLEAR ?
3988 025474 001374      BNE     64$      ;IF NO, WAIT FOR IT TO CLEAR
3989 025476 000207      RTS     PC
3990
3991
3992 ;-----*
3993 ;THIS ROUTINE SPACES REVERSE OVER EVEN PARITY
3994
3995 025500 004737 025236      TESTAY: JSR    PC,TESTAX
3996
3997 025504 004537 027546      JSR    R5,TRYIT
3998 025510 001410      .WORD   SPACER      ;FUNCTION
3999 025512 177700      .WORD   -100      ;WORD COUNT
4000 025514 035640      .WORD   INPUT      ;BUS ADDRESS
4001 025516 013727 001240      MOV    HOLD,(PC)+  ;PICK UP TIME PARAMETER
4002 025522 000000 64$:  .WORD   0      ;USE THIS WORD AS A TIME COUNTER
4003 025524
4004 025524 005227 000000      INC     #0      ;IF NO,COUNT 1 OF 65535 TICKS
4005 025530 001375      BNE     66$      ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
4006 025532 005337 025522      DEC     64$      ;HAS THE TOTAL TIME ELAPSED?
4007 025536 001372      BNE     66$      ;IF NO,GO WAIT A LITTLE LONGER
4008 025540
4009
4010
4011 025540 022777 103610 153634      CMP     #103610,ATCR  ;TEST COMMAND REGISTER
4012 025546 001412      BEQ     AS174
4013 025550 012705 103610      MOV     #103610,R5
4014 025554 017704 153622      MOV     ATCR,R4
4015 025560 013737 001402 001244      MOV     TRCR,REGIST
4016 025566 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
4017 025570 104403 010024      TYPEF   ,MEPSRFL
4018
4019 025574 000175      AS174:  STAGAS=STAGAS+1
4020
4021 025574 022777 122001 153604      CMP     #122001,ATRSR ;TEST THE STATUS REGISTER
4022 025602 001411      BEQ     AS175
4023 025604 012705 122001      MOV     #122001,R5
4024 025610 017704 153572      MOV     ATRSR,R4
4025 025614 013737 001406 001244      MOV     TRSR,REGIST
4026 025622 104001      ERROR    1      ;INCORRECT REGISTER MATCHUP
4027 025624 104404      TYPEL
4028
4029 025626 000176      AS175:  STAGAS=STAGAS+1
4030
4031 025626 022777 035640 153562      CMP     #INPUT,ATRBA ;TEST BUFFER ADDRESS
4032 025634 001411      BEQ     AS176
4033 025636 012705 035640      MOV     #INPUT,R5
4034 025642 017704 153550      MOV     ATRBA,R4
4035 025646 013737 001416 001244      MOV     TRBA,REGIST

```

```

4036 025654 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
4037 025656 104404          TYPEL
4038
4039 025660 000177          AS176: STAGS=STAGS+1
4040
4041 025660 022777 177700 153524      CMP      #177700, @TRWC      ;TEST THE WORD COUNT
4042 025666 001411          BEQ      AS177
4043 025670 012705 177700          MOV      #177700, R5
4044 025674 017704 153512          MOV      @TRWC, R4
4045 025700 013737 001412 001244      MOV      TRWC, REGIST
4046 025706 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
4047 025710 104404          TYPEL
4048
4049 025712 000200          AS177: STAGS=STAGS+1
4050
4051 025712 052777 004000 153462      BIS      #PWRCLR, @TRCR      ;DEVICE MASTER RESET
4052 025720 032777 004000 153454      64$: BIT      #PWRCLR, @TRCR      ;INSTRUCTION CLEAR ?
4053 025726 001374          BNE      64$
4054 025730 000207          RTS      PC          ;IF NO, WAIT FOR IT TO CLEAR
4055
4056 ;-----*
4057 ;THIS ROUTINE READS EVEN PARITY
4058
4059 025732          TESTAZ:
4060 025732 004737 025500          JSR      PC, TESTAY
4061
4062 025736 004537 027546          JSR      R5, TRYIT
4063 025742 000004          .WORD      READ          ;FUNCTION
4064 025744 177700          .WORD      -100         ;WORD COUNT
4065 025746 035640          .WORD      INPUT        ;BUS ADDRESS
4066 025750 013727 001240          MOV      HOLD, (PC)+ ;PICK UP TIME PARAMETER
4067 025754 000000      64$: .WORD      0          ;USE THIS WORD AS A TIME COUNTER
4068 025756          66$:
4069 025756 005227 000000          INC      #0          ;IF NO, COUNT 1 OF 65535 TICKS
4070 025762 001375          BNE      66$          ;HAS THE TIME EXPIRED? IF NO, LOOP AGAIN
4071 025764 005337 025754          DEC      64$
4072 025770 001372          BNE      66$          ;HAS THE TOTAL TIME ELAPSED?
4073 025772          65$:
4074
4075
4076 025772 022777 102204 153402      CMP      #102204, @TRCR      ;TEST COMMAND REGISTER
4077 026000 001412          BEQ      AS200
4078 026002 012705 102204          MOV      #102204, R5
4079 026006 017704 153370          MOV      @TRCR, R4
4080 026012 013737 001402 001244      MOV      TRCR, REGIST
4081 026020 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
4082 026022 104403 010061          TYPEF      ,MREPFL
4083
4084 026026 000201          AS200: STAGS=STAGS+1
4085
4086 026026 022777 163001 153352      CMP      #163001, @TRSR      ;TEST THE STATUS REGISTER
4087 026034 001411          BEQ      AS201
4088 026036 012705 163001          MOV      #163001, R5
4089 026042 017704 153340          MOV      @TRSR, R4
4090 026046 013737 001406 001244      MOV      TRSR, REGIST
4091 026054 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP

```

```

4092 026056 104404          TYPEL
4093
4094 026060 000202          AS201: STAGAS=STAGAS+1
4095
4096 026060 022777 035642 153330      CMP      #INPUT+2, @TRBA ;TEST BUFFER ADDRESS
4097 026066 001410          BEQ      AS202
4098 026070 010105          MOV      R1, R5
4099 026072 017704 153320          MOV      @TRBA, R4
4100 026076 013737 001416 001244      MOV      TRBA, REGIST
4101 026104 104001          ERROR     1 ;INCORRECT REGISTER MATCHUP
4102 026106 104404          TYPEL
4103
4104 026110 000203          AS202: STAGAS=STAGAS+1
4105
4106 026110 022777 177701 153274      CMP      #177701, @TRWC ;TEST THE WORD COUNT
4107 026116 001411          BEQ      AS203
4108 026120 012705 177701          MOV      #177701, R5
4109 026124 017704 153262          MOV      @TRWC, R4
4110 026130 013737 001412 001244      MOV      TRWC, REGIST
4111 026136 104001          ERROR     1 ;INCORRECT REGISTER MATCHUP
4112 026140 104404          TYPEL
4113
4114 026142 000204          AS203: STAGAS=STAGAS+1
4115
4116 026142 000207          RTS      PC
4117
4118 ;-----*
4119 ;THIS ROUTINE TESTS THE BITS IN THE WORD COUNT USING A COUNT PATTERN
4120 ;-----*
4121 026144 005001          TESTBA: CLR      R1 ;CLEAR COUNT
4122 026146 005001          A=.
4123 026146 010177 153240          MOV      R1, @TRWC ;SET WORD COUNT
4124 026152 020177 153234          CMP      R1, @TRWC ;TEST WORD COUNT
4125 026156 001411          BEQ      AS204
4126 026160 010105          MOV      R1, R5
4127 026162 017704 153224          MOV      @TRWC, R4
4128 026166 013737 001412 001244      MOV      TRWC, REGIST
4129 026174 104001          ERROR     1 ;INCORRECT REGISTER MATCHUP
4130 026176 104403 010105          TYPEF ,MWCFL
4131
4132 026202 000205          AS204: STAGAS=STAGAS+1
4133 026202 005201          INC      R1 ;INCREMENT COUNT
4134 026204 001360          BNE      A ;CONTINUE
4135 026206 000207          RTS      PC
4136
4137 ;-----*
4138 ;THIS ROUTINE TESTS THE STATUS AND COMMAND REGISTERS
4139 ;AFTER VALID WRITE FUNCTIONS
4140
4141
4142
4143
4144
4145 026210 022777 002602 153164      TESTCX: CMP      #2602, @TRCR ;TEST COMMAND REGISTER
4146 026216 001422          BEQ      AS205
4147 026220 032777 000200 153160      BIT      #200, @TRSR ;EOT??

```

```

4148 026226 001404      BEQ      1$      ; NO
4149 026230 022626      CMP      (SP)+,(SP)+
4150 026232 022626      CMP      (SP)+,(SP)+
4151 026234 022626      CMP      (SP)+,(SP)+
4152 026236 000207      RTS      PC
4153 026240      1$:
4154 026240 012705 002602      MOV      #2602,R5
4155 026244 017704 153132      MOV      @TRCR,R4
4156 026250 013737 001402 001244      MOV      TRCR,REGIST
4157 026256 104001      ERROR      1      ; INCORRECT REGISTER MATCHUP
4158 026260 104403 005667      TYPEF      ,MWRTFL
4159
4160 026264 000206      AS205: STAGAS=STAGAS+1
4161
4162 026264 022777 002001 153114      CMP      #2001,@TRSR      ; TEST THE STATUS REGISTER
4163 026272 001422      BEQ      AS206
4164 026274 032777 000200 153104      BIT      #200,@TRSR      ; AT EOT??
4165 026302 001404      BEQ      1$      ; NO
4166 026304 022626      CMP      (SP)+,(SP)+
4167 026306 022626      CMP      (SP)+,(SP)+
4168 026310 022626      CMP      (SP)+,(SP)+
4169 026312 000207      RTS      PC
4170 026314      1$:
4171 026314 012705 002001      MOV      #2001,R5
4172 026320 017704 153062      MOV      @TRSR,R4
4173 026324 013737 001406 001244      MOV      TRSR,REGIST
4174 026332 104001      ERROR      1      ; INCORRECT REGISTER MATCHUP
4175 026334 104403 005667      TYPEF      ,MWRTFL
4176
4177 026340 000207      AS206: STAGAS=STAGAS+1
4178
4179 026340 005777 153046      TST      @TRWC      ; TEST WORD COUNT
4180 026344 001412      BEQ      AS207
4181 026346 012705 000000      MOV      #0,R5
4182 026352 017704 153034      MOV      @TRWC,R4
4183 026356 013737 001412 001244      MOV      TRWC,REGIST
4184 026364 104001      ERROR      1      ; INCORRECT REGISTER MATCHUP
4185 026366 104403 005667      TYPEF      ,MWRTFL
4186
4187 026372 000210      AS207: STAGAS=STAGAS+1
4188 026372 000207      RTS      PC
4189
4190      ; -----*
4191      ; THIS ROUTINE TESTS THE COMMAND AND STATUS AFTER VALID
4192      ; READ FUNCTIONS
4193
4194 026374 022777 002204 153000 TESTDX: CMP      #2204,@TRCR      ; TEST COMMAND REGISTER
4195 026402 001421      BEQ      AS210
4196 026404 032777 000200 152774      BIT      #200,@TRSR      ; AT EOT??
4197 026412 001403      BEQ      1$      ; NO
4198 026414 022626      CMP      (SP)+,(SP)+
4199 026416 005726      TST      (SP)+
4200 026420 000207      RTS      PC
4201 026422      1$:
4202 026422 012705 002204      MOV      #2204,R5
4203 026426 017704 152750      MOV      @TRCR,R4

```

```

4204 026432 013737 001402 001244      MOV      TRCR,REGIST
4205 026440 104001                      ERROR      1          ;INCORRECT REGISTER MATCHUP
4206 026442 104403 005700      TYPEF      ,MRDFL
4207
4208 026446 000211          AS210:  STAGAS=STAGAS+1
4209
4210 026446 022777 002001 152732      CMP      #2001,ATRSR      ;TEST THE STATUS REGISTER
4211 026454 001421                      BEQ      AS21
4212 026456 032777 000200 152722      BIT      #200,ATRSR      ;AT EOT??
4213 026464 001403                      BEQ      1$          ;NO
4214 026466 022626                      CMP      (SP)+,(SP)+
4215 026470 005726                      TST      (SP)+
4216 026472 000207                      RTS      PC
4217 026474
4218 026474 012705 002001          1$:  MOV      #2001,R5
4219 026500 017704 152702          MOV      ATRSR,R4
4220 026504 013737 001406 001244      MOV      TRSR,REGIST
4221 026512 104001                      ERROR      1          ;INCORRECT REGISTER MATCHUP
4222 026514 104403 005700      TYPEF      ,MRDFL
4223
4224 026520 000212          AS211:  STAGAS=STAGAS+1
4225
4226 026520 005777 152666          TST      ATRWC          ;TEST WORD COUNT
4227 026524 001412                      BEQ      AS212
4228 026526 012705 000000          MOV      #0,R5
4229 026532 017704 152654          MOV      ATRWC,R4
4230 026536 013737 001412 001244      MCV      TRWC,REGIST
4231 026544 104001                      ERROR      1          ;INCORRECT REGISTER MATCHUP
4232 026546 104403 005700      TYPEF      ,MRDFL
4233
4234 026552 000213          AS212:  STAGAS=STAGAS+1
4235
4236 026552 004737 015622          JSR      PC,TESTS
4237 026556 000207          RTS      PC
4238
4239
4240          TESTCA:
4241 026560 004737 031212          JSR      PC,ENAINIT
4242 026564 004537 027546          JSR      R5,TRYIT
4243 026570 000402          .WORD      WRITE          ;FUNCTION
4244 026572 177777          .WORD      -1          ;WORD COUNT
4245 026574 031600          .WORD      OUTPUT          ;BUS ADDRESS
4246 026576 004737 031252          JSR      PC,CHKINT
4247 026602 004737 026210          JSR      PC,TESTCX
4248 026606 000207          RTS      PC
4249
4250          TESTCK:
4251 026610 004537 027546          JSR      R5,TRYIT
4252 026614 000402          .WORD      WRITE          ;FUNCTION
4253 026616 177700          .WORD      -100          ;WORD COUNT
4254 026620 031600          .WORD      OUTPUT          ;BUS ADDRESS
4255 026622 004737 026210          JSR      PC,TESTCX
4256 026626 000207          RTS      PC
4257
4258          TESTEK:
4259 026630 004537 027546          JSR      R5,TRYIT
4260 026634 001410          .WORD      SPACER          ;FUNCTION

```

H07

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 86
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0085

```

4260 026636 177700      :WORD      -100      :WORD COUNT
4261 026640 000000      :WORD      0      :BUS ADDRESS
4262 026642 004737 026670 JSR      PC,TESTEX
4263 026646 000207      RTS      PC
4264
4265
4266
4267 026650
4268 026650 004537 027546 TESTFO: JSR      R5,TRYIT      ;;;
4269 026654 000016      :WORD      ERASE      :FUNCTION
4270 026656 177400      :WORD      -400      :WORD COUNT
4271 026660 031600      :WORD      OUTPUT     :BUS ADDRESS
4272 026662 004737 027016 JSR      PC,TESTFX
4273 026666 000207      RTS      PC
4274
4275
4276
4277 ;-----*
4278 ;THIS ROUTINE TESTS THE COMMAND AND STATUS AFTER VALID
4279 ;SPACE REVERSE FUNCTIONS
4280 026670 022777 003610 152504 TESTEX: CMP      #3610,@TRCR      ;TEST COMMAND REGISTER
4281 026676 001412      BEQ      A$213
4282 026700 012705 003610      MOV      #3610,R5
4283 026704 017704 152472      MOV      @TRCR,R4
4284 026710 013737 001402 001244 MOV      TRCR,REGIST
4285 026716 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
4286 026720 104403 005710      TYPEF      ,MSRVFL
4287
4288 026724 000214      A$213: STAGAS=STAGAS+1
4289
4290 026724 022777 002001 152454      CMP      #2001,@TRSR      ;TEST THE STATUS REGISTER
4291 026732 001412      BEQ      A$214
4292 026734 012705 002001      MOV      #2001,R5
4293 026740 017704 152442      MOV      @TRSR,R4
4294 026744 013737 001406 001244 MOV      TRSR,REGIST
4295 026752 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
4296 026754 104403 005710      TYPEF      ,MSRVFL
4297
4298 026760 000215      A$214: STAGAS=STAGAS+1
4299
4300 026760 023777 027540 152424      CMP      USEB,@TRWC      ;TEST WORD COUNT
4301 026766 001412      BEQ      A$215
4302 026770 013705 027540      MOV      USEB,R5
4303 026774 017704 152412      MOV      @TRWC,R4
4304 027000 013737 001412 001244 MOV      TRWC,REGIST
4305 027006 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
4306 027010 104403 005710      TYPEF      ,MSRVFL
4307
4308 027014 000216      A$215: STAGAS=STAGAS+1
4309
4310 027014 000207      RTS      PC
4311
4312
4313
4314 ;-----*
4315 ;THIS ROUTINE TESTS THE COMMAND AND STATUS REGISTERS AFTER

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 87
 CZTRAB.HED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0086

```

4316 ;AN ERASE FUNCTION
4317
4318 027016 022777 002216 152356 TESTFX: CMP #2216, @TRCR ;TEST COMMAND REGISTER
4319 027024 001412 BEQ AS216
4320 027026 012705 002216 MOV #2216, R5
4321 027032 017704 152344 MOV @TRCR, R4
4322 027036 013737 001402 001244 MOV TRCR, REGIST
4323 027044 104001 ERROR 1 ;INCOPRECT REGISTER MATCHUP
4324 027046 104403 005731 TYPEF ,MERFL
4325
4326 027052 000217 AS216: STAGAS=STAGAS+1
4327
4328 027052 022777 002001 152326 CMP #2001, @TRSR ;TEST THE STATUS REGISTER
4329 027060 001412 BEQ AS217
4330 027062 012705 002001 MOV #2001, R5
4331 027066 017704 152314 MOV @TRSR, R4
4332 027072 013737 001406 001244 MOV TRSR, REGIST
4333 027100 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
4334 027102 104403 005731 TYPEF ,MERFL
4335
4336 027106 000220 AS217: STAGAS=STAGAS+1
4337
4338 027106 013701 027540 MOV USEB, R1 ;GET WORD COUNT
4339 027112 005101 COM R1 ;FIX IT
4340 027114 005201 INC R1
4341 027116 006301 ASL R1 ;DOUBLE IT
4342 027120 062701 031600 ADD #OUTPUT, R1 ;COMPUTE BUS ADDRESS
4343
4344 027124 020177 152266 CMP R1, @TRBA ;TEST BUFFER ADDRESS
4345 027130 001411 BEQ AS220
4346 027132 010105 MOV R1, R5
4347 027134 017704 152256 MOV @TRBA, R4
4348 027140 013737 001416 001244 MOV TRBA, REGIST
4349 027146 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
4350 027150 104403 005731 TYPEF ,MERFL
4351
4352 027154 000221 AS220: STAGAS=STAGAS+1
4353
4354 027154 005777 152232 TST @TRWC ;TEST THE WORD COUNT
4355 027160 001412 BEQ AS221
4356 027162 012705 000000 MOV #0, R5
4357 027166 017704 152220 MOV @TRWC, R4
4358 027172 013737 001412 001244 MOV TRWC, REGIST
4359 027200 104001 ERROR 1 ;INCORRECT REGISTER MATCHUP
4360 027202 104403 005731 TYPEF ,MERFL
4361
4362 027206 070222 AS221: STAGAS=STAGAS+1
4363
4364 027206 000207 RTS PC
4365
4366 ;-----*
4367 ;THIS ROUTINE WRITES, SPACES REVERSE, AND READS
4368
4369 027210 012704 000017 TESTXX: MOV #15, R4 ;LOAD THE TOTAL SUBTEST PASS COUNT
4370 027214 012703 027356 MOV #SIZTAB, R3 ;POINT TO TABLE OF BLOCK SIZES
4371 027220 012737 000001 001264 MOV #1, WRTFLG ;SET FLAG FOR RETURN

```

CZTRAB0 MACY11 30(1046) 14-DEC-77 12:32 PAGE 38
 CZTRAB.MED 14-DEC-77 12:19 SUB-TEST ROUTINES

SEQ 0087

```

4372 027226 012337 027306      1$: MOV      (R3)+,4$
4373 027232 012702 000003      MOV      #3,R2          ;KEEP TRACK OF THE NUMBER OF FUNCTIONS TO DO
4374 027236 012705 027422      MOV      #BUFTAB,R5      ;POINT TO TABLE OF BUFFERS
4375 027242 012701 027414      MOV      #FUNCTB,R1      ;POINT TO TABLE OF FUNCTIONS
4376 027246 012700 027430      MOV      #SUBTST,R0      ;POINT TO TABLE OF FOLLOWUP TESTS
4377 027252 004737 031212      JSR      PC,ENAINI
4378 027256 012137 027304      2$: MOV      (R1)+,3$      ;LOAD A FUNCTION IN SUBROUTINE INTERFACE
4379 027262 012537 027310      MOV      (R5)+,5$      ;LOAD THE NAME OF THE BUFFER
4380 027266 010146      MOV      R1,-(SP)      ;SAVE REGISTERS.
4381 027270 010246      MOV      R2,-(SP)
4382 027272 010346      MOV      R3,-(SP)
4383 027274 010446      MOV      R4,-(SP)
4384 027276 010546      MOV      R5,-(SP)
4385 027300 004537 027546      JSR      R5,TRYIT      ;GO EXECUTE THE FUNCTION
4386 027304 000000      3$: .WORD      0
4387 027306 000000      4$: .WORD      0
4388 027310 000000      5$: .WORD      0
4389 027312 004730      JSR      PC,2(R0)+      ;DO THE APPROPRIATE FOLLOWUP TEST
4390 027314 012605      MOV      (SP)+,R5      ;RESTORE THE REGISTERS
4391 027316 012604      MOV      (SP)+,R4
4392 027320 012603      MOV      (SP)+,R3
4393 027322 012602      MOV      (SP)+,R2
4394 027324 012601      MOV      (SP)+,R1
4395 027326 005302      DEC      R2          ;REDUCE THE COUNT. ALL FUNCTIONS DONE?
4396 027330 001352      BNE      2$          ;IF NOT, GO DO THE NEXT ONE
4397 027332 005304      DEC      R4          ;REDUCE THE OVERALL COUNT. ALL BLOCK SIZES DONE?
4398 027334 001334      BNE      1$          ;IF NOT, GO DO THE NEXT ONE
4399 027336 004737 014446      JSR      PC,TESTL
4400 027342 004737 015734      JSR      PC,TESTU
4401 027346 004737 016144      JSR      PC,TESTV
4402 027352 000137 027210      JMP      TESTXX
4403 027356 177777 177776 177775 SIZTAB: .WORD      -1,-2,-3,-4,-5,-6,-7,-10,-20,-40,-100,-200,-400,-1000,-2000
(1) 027414 000402 001410 000004 FUNCTB: .WORD      WRITE,SPACER,READ
(1) 027422 031600 000000 035640 BUFTAB: .WORD      OUTPUT,0,INPUT
(1) 027430 026210 026670 026374 SUBTST: .WORD      TESTCX,TESTEX,TESTDX
4404 ;-----*
4405 ;THIS ROUTINE READS DATA WRITTEN BY TESTXX.
4406
4407 027436 012704 000017      TESTXY: MOV      #15,R4          ;COUNT THE NUMBER OF ITERATIONS
4408 027442 012703 027356      MOV      #SIZTAB,R3      ;POINT TO THE TABLE OF BLOCK SIZES
4409 027446 012737 000002 001264 MOV      #2,WRTFLG      ;SET RETURN FLAG
4410 027454 004737 031212      JSR      PC,ENAINI
4411 027460 012337 027476      1$: MOV      (R3)+,2$      ;LOAD SUBROUTINE INTERFACE
4412 027464 010346      MOV      R3,-(SP)      ;SAVE REGISTERS
4413 027466 010446      MOV      R4,-(SP)
4414 027470 004537 027546      JSR      R5,TRYIT      ;GO DO A READ FUNCTION
4415 027474 000004      .WORD      READ      ;EXECUTE A READ
4416 027476 000000      2$: .WORD      0      ;BLOCK SIZE
4417 027500 035640      INPUT      ;BUFFER TO USE
4418 027502 004737 026374      JSR      PC,TESTDX      ;EXECUTE THE FOLLOWUP ROUTINE
4419 027506 012604      MOV      (SP)+,R4      ;RESTORE REGISTERS.
4420 027510 012603      MOV      (SP)+,R3
4421 027512 005304      DEC      R4          ;REDUE COUNT. ALL BLOCK SIZES DONE?
4422 027514 001361      BNE      1$          ;IF NO, GO DO THE NEXT ONE
4423 027516 004737 016144      JSR      PC,TESTV

```

```

4424 027522 000137 027436          JMP      TESTXY
4425
4426
4427
4428 ;-----*
4429
4430
4431 ;THIS ROUTINE PERFORMS TAPE FUNCTION IT IS CALLED
4432 ;BY THE TAPFUN MACRO
4433
4434 027526 000000          USEA0: 0
4435 027530 000000          USEB0: 0
4436 027532 000000          USEC0: 0
4437 027534 000000          USED0: 0
4438
4439 027536 000000          USEA: 0          ;FUNCTION
4440 027540 000000          USEB: 0          ;WORD COUNT
4441 027542 000000          USEC: 0          ;INITIAL BUFFER ADDRESS
4442 027544 000000          USED: 0          ;TEMP STORAGE LOCATION
4443
4444 027546 012537 027536    TRYIT: MOV      (R5)+,USEA          ;GET THE FUNCTION
4445 027552 012537 027540    MOV      (R5)+,USEB          ;GET THE WORD COUNT
4446 027556 012537 027542    MOV      (R5)+,USEC          ;GET THE BUS ADDRESS
4447 027562 000223          AS222: STAGAS=STAGAS+1
4448 027562 013737 027536 027526    MOV      USEA,USEA0          ;SAVE LAST FUNCTION
4449 027570 013737 027540 027530    MOV      USEB,USEB0          ;SAVE LAST WORD COUNT
4450 027576 013737 027542 027532    MOV      USEC,USEC0          ;SAVE LAST BUS ADDRESS
4451 027604 001737 014664          JSR      PC,TESTN
4452 027610 027610          A=
4453 027610 000224          AS223: STAGAS=STAGAS+1
4454 027610 017737 151566 027544    MOV      @TRCR,USED          ;STORE COMMAND REGISTER
4455 027616 042737 175577 027544    BIC      #175577,USED          ;CLEAR UNWANTED BITS
4456 027624 022737 002200 027544    CMP      #002200,USED          ;TEST THE COMMAND REGISTER
4457 027632 001413          BEQ      AS224
4458 027634 012705 002200          MOV      #02200,R5
4459 027640 013704 027544          MOV      USED,R4
4460 027644 013737 001402 001244    MOV      TRCR,REGIST
4461 027652 104001          ERROR 1          ;INCORRECT REGISTER MATCHUP
4462 027654 104402 010155          TYPE ,MUNRDY
4463
4464 027660 000753          BR      A          ;GET RID OF DERORS
4465 027662 000225          AS224: STAGAS=STAGAS+1
4466
4467 027662 042777 000001 151516    BIC      #BIT0,@TRSR
4468 027670 052737 000001 027536    BIS      #BIT0,USEA          ;SET GO BIT IN COMMAND
4469
4470 027676 013777 027540 151506    MOV      USEB,@TRWC          ;SET WORD COUNT
4471 027704 013777 027542 151504    MOV      USEC,@TRBA          ;SET INITIAL BUS ADDRESS
4472 027712 000240          NOP
4473 ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
4474 ;EQUIPMENT STATUS OR PROGRAM OPERATION
4475 027714 013777 027536 151460    MOV      USEA,@TRCR          ;SET CONTROL REGISTER
4476
4477 027722 004737 012524          JSR      PC,TESTA
4478 027726 000240          NOP
4479 ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
;EQUIPMENT STATUS OR PROGRAM OPERATION

```

```

4480 027730 032777 000200 151450      BIT      #BIT7, @TRSR
4481 027736 001424      BEQ      A$225
4482 027740 005737 001264      TST      WRTFLG      ;TEST FOR RETURN FLAG
4483 027744 001417      BEQ      1$      ;NO FLAG
4484 027746 032737 000001 001264      BIT      #1, WRTFLG      ;WRITE RETURN?
4485 027754 001003      BNE      2$      ;YES
4486 027756 022626      CMP      (SP)+, (SP)+      ;OK READ
4487 027760 005726      TST      (SP)+
4488 027762 000403      BR       3$
4489 027764 022626      2$:      CMP      (SP)+, (SP)+
4490 027766 022626      CMP      (SP)+, (SP)+
4491 027770 022626      CMP      (SP)+, (SP)+
4492 027772 005037 001264      3$:      CLR      WRTFLG
4493 027776 004737 013626      JSR      PC, TESTI
4494 030002 000207      RTS      PC
4495 030004      1$:
4496 030004 004737 013626      JSR      PC, TESTI
4497 030010 000226      A$225:    $TAGAS=$TAGAS+1
4498 030010 000205      RTS      RS      ;RETURN TO CALLING PROCEDURE
4499
4500
4501
4502
4503      ;-----*
4504      ;THIS ROUTINE PERFORMS TAPE FUNCTION IT IS CALLED
4505      ;BY THE EOTFUN MACRO THIS IS USED FOR EOT FUNCTIONS ONLY
4506 030012 005737 031326      EOTTST: TST      INTFLG      ;CHECK INTERRUPT FLAG
4507 030016 001403      BEQ      A$226
4508 030020 052737 000100 027536      BIS      #BIT6, USEA      ;ENABLE INTERRUPT
4509 030026 000227      A$226:    $TAGAS=$TAGAS+1
4510 030026 013737 027536 027526      MOV      USEA, USEA0      ;SAVE LAST FUNCTION
4511 030034 013737 027540 027530      MOV      USEB, USEB0      ;SAVE LAST WORD COUNT
4512 030042 013737 027542 027532      MOV      USEC, USEC0      ;SAVE LAST BUS ADDRESS
4513 030050 004737 014664      JSR      PC, TESTN
4514 030054 032777 100000 151320      BIT      #BIT15, @TRCR
4515 030062 001412      BEQ      A$227
4516 030064      A=:
4517 030064 104000      ERROR      ;ERROR-BIT WAS NOT CORRECT
4518 030066 104403 010123      TYPEF      , MUNERR
4519
4520 030072 052777 004000 151302      BIS      #PWRCLR, @TRCR      ;DEVICE MASTER RESET
4521 030100 032777 004000 151274      64$:    BIT      #PWRCLR, @TRCR      ;INSTRUCTION CLEAR ?
4522 030106 001374      BNE      64$      ;IF NO, WAIT FOR IT TO CLEAR
4523 030110 000230      A$227:    $TAGAS=$TAGAS+1
4524 030110 017737 151266 027544      MOV      @TRCR, USED      ;STORE COMMAND REGISTER
4525 030116 042737 175577 027544      BIC      #175577, USED      ;CLEAR UNWANTED BITS
4526 030124 022737 002200 027544      CMP      #002200, USED      ;TEST THE COMMAND REGISTER
4527 030132 001413      BEQ      A$230
4528 030134 012705 002200      MOV      #02200, R5
4529 030140 013704 027544      MOV      USED, R4
4530 030144 013737 001402 001244      MOV      TRCR, REGIST
4531 030152 104001      ERROR      1      ;INCORRECT REGISTER MATCHUP
4532 030154 104403 010155      TYPEF      , MUNRDY
4533
4534 030160 000741      BR       A
4535 030162 000231      A$230:    $TAGAS=$TAGAS+1      ;GET RID OF DERORS

```

```

4536
4537 030162 042777 000001 151216 BIC #BIT0,ATRSR ;SET GO BIT IN COMMAND
4538 030170 052737 000001 027536 BIS #BIT0,USEA
4539
4540 030176 013777 027540 151206 MOV USEB,ATRCW ;SET WORD COUNT
4541 030204 013777 027542 151204 MOV USEC,ATRBA ;SET INITIAL BUS ADDRESS
4542 030212 000240 NOP ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
4543 ;EQUIPMENT STATUS OR PROGRAM OPERATION
4544 030214 013777 027536 151160 MOV USEA,ATRCR ;SET CONTROL REGISTER
4545
4546 030222 004737 012524 JSR PC,TESTA
4547 030226 000240 NOP ;THIS LOCATION MAY BE CHANGED TO A HALT TO INSPECT
4548 ;EQUIPMENT STATUS OR PROGRAM OPERATION
4549
4550 030230 000207 RTS PC
4551
4552 ;-----*
4553 ;THIS ROUTINE PUTS THE CORRECT PARITY ON ALL WORDS IN
4554 ;THE OUTPUT BUFFER
4555
4556 030232 012700 031600 FIXBUF: MOV #OUTPUT,RO ;SET INDEX
4557 A=.
4558 030236 011001 MOV (RO),R1 ;GET THE WORD
4559 030240 005002 CLR R2 ;CLEAR THE COUNTER
4560
4561 B=
4562 030242 105701 TSTB R1 ;TEST R1 LOWER
4563 030244 100001 BPL AS231
4564 030246 005202 INC R2 ;INCREMENT COUNTER
4565 030250 000232 AS231: STAGAS=STAGAS+1
4566 030252 105701 TSTB R1 ;TEST R1 LOWER
4567 030254 001402 BEQ AS232
4568 030256 006301 ASL R1 ;SHIFT R1
4569 030260 000771 BR B ;GO BACK TO B
4570 030262 000233 AS232: STAGAS=STAGAS+1
4571 030264 032702 000001 BIT #BIT0,R2 ;ODD OR EVEN
4572 030266 052710 000400 BNE AS233
4573 030272 000234 AS233: STAGAS=STAGAS+1
4574 030274 022700 035640 TST (RO)+ ;ALL DONE
4575 030300 001356 CMP #INPUT,RO ;NO
4576 030302 000207 BNE A
4577
4578 ;-----*
4579 ;SBTTL DEROR HANDLING ROUTINE
4580
4581 ;-----*
4582
4583
4584
4585 030304 012737 177777 001266 WRONG: MOV #-1,LOC ;SET UP FOR ERROR ;ERROR-BIT WAS N
4586 030312 012737 177777 001252 MOV #-1,ERR ;SET DEROR FLAG
4587 030320 012600 MOV (SP)+,RO ;GET PC
4588 030322 005740 TST -(RO) ;FIX IT
4589 030324 010037 001254 MOV RO,XPC ;STORE IT
4590 030330 011637 001256 MOV (SP),XSR ;STORE SR
4591 030334 005746 TST -(SP) ;FIX THE STACK

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 92
 CZTRAB.HED 14-DEC-77 12:19 DEROR HANDLING ROUTINE

SEQ 0091

```

4592 030336 005737 031210          TST      VECSTR      ;TEST VECTOR FLAG
4593 030342 001402                      BEQ      AS234      ;
4594 030344 000137 031152          JMP      VECTR      ;RETURN TO VECTOR
4595 030350 000235          AS234: STAGAS=STAGAS+1
4596 030350 000002          RTI
4597
4598
4599 030352                      WRONGX: JSR      R5,SAVE
4600 030352 004537 031414          BIT      #BIT4,JSWR      ;TEST SWITCH BIT 04
4601 030356 032777 000020 150616          BEQ      AS235      ;
4602 030364 001406                      CMP      ERRCNT,MAXERR ;TEST DEROR COUNT
4603 030366 023727 001232 000062          BHI      WRONGY      ;BRANCH IF HIGHER
4604 030374 101024                      INC      ERRCNT      ;INCREMENT COUNT
4605 030376 005237 001232          AS235: STAGAS=STAGAS+1
4606 030402 000236                      TYPE     WHED1
4607 030402 104402 030500          CMP      #-1,LOC      ;TEST LOCATION FOR ERROR
4608 030406 022737 177777 001266          BEQ      AS236      ;ERROR-B
4609 030414 001402                      TYPE     WHED2
4610 030416 104402 030537          AS236: STAGAS=STAGAS+1
4611 030422 000237                      TYPEF     ,MCRLF
4612 030422 104403 005320          CNVRT      ;TYPE A CARRIAGE RETURN
4613 030426 104413          WRDATA      ;OCTAL TO ASCII CONVERT ROUTINE
4614 030430 030454          CMP      #-1,LOC      ;POINTER TO DATA TO PRINT
4615 030432 022737 177777 001266          BEQ      WRONGY      ;YES-EXIT
4616 030440 001402          CNVRT      ;OCTAL TO ASCII CONVERT ROUTINE
4617 030442 104413          WRDAT2      ;DATA TO CONVERT AND PRINT
4618 030444 030466          INC      ERRCNT      ;INCREMENT 1 OVER LIMIT
4619 030446 005237 001232          RTS      PC
4620 030452 000207          WRDATA: .WORD     3      ;NUMBER OF DATA TO PRINT
4621 030454 000003          .BYTE     6,2      ;NUMBER OF CHARACTERS AND SPACES TO PRINT
4622 030456 006      002          TSTPTR      ;LOCATIONS OF DATA
4623 030460 001250          XPC
4624 030462 001254          XSR
4625 030464 001256          WRDAT2: .WORD     3      ;NUMBER OF WORDS TO PRINT
4626 030466 000003          .BYTE     6,2      ;NUMBER OF CHARACTERS, SPACES
4627 030470 006      002          COR      ;LOCATION OF DATA
4628 030472 001260          ACT
4629 030474 001262          LOC
4630 030476 001266          WHED1: .ASCIZ    <377>/SUBTEST  PC      STATUS /
4631 030500 051777 041125 042524
4632 030506 052123 020040 020040
4633 030514 041520 020040 020040
4634 030522 020040 020040 051440
4635 030530 040524 052524 020123
4636 030536 000
4637 030537 040      020040 020040 WHED2: .ASCIZ    /      CORRECT  ACTUAL  LOCATION /
4638 030544 047503 051122 041505
4639 030552 020124 020040 040440
4640 030560 052103 040525 020114
4641 030566 020040 020040 047514
4642 030574 040503 044524 047117
4643 030602 000040
4644
4645
4646
4647

```

.EVEN

-----*
 .SBTTL CORE SIZE ROUTINE

```

4648
4649 ;THIS ROUTINE TESTS THE CORE SIZE AND INSURES IT IS OK
4650
4651 030604 000414 BR CORE
4652 030606 030606 X=.
4653 030606 162700 000002 SUB #2,RO
4654 030612 010037 030716 MOV RO, LASTAD ;SET LAST ADDRESS
4655 030616 022626 POP2SP
4656 030620 023727 030716 040000 CMP LASTAD, #CORSIZ ;TEST CORE SIZE
4657 030626 103001 BHS AS237
4658 030630 104000 ERROR ;ERROR-BIT WAS NOT CORRECT
4659 030632 000240 AS237: STAGAS=STAGAS+1
4660 030632 000137 003502 JMP RESTRT
4661 030636 012737 030606 000004 CORE: MOV #X,4 ;SET VECTOR
4662 030644 012737 000000 000006 MOV #0,6 ;SET STATUS
4663 030652 005000 CLR RO ;SET RO TO ZERO
4664 A=.
4665 030654 105720 TSTB (RO)+ ;TEST ADDRESS
4666 030656 000776 BR A ;CONTINUE
4667
4668 030660 012737 177777 030720 X=.
4669 030666 000002 MOV #-1, ADDERR ;SET DEROR FLAG
4670 030670 005037 030720 ADDTST: RTI
4671 030674 012737 030660 000004 CLR ADDERR ;CLEAR DEROR FLAG
4672 030702 012737 000340 000006 MOV #X,4 ;SET VECTOR
4673 030710 005710 MOV #340,6 ;SET STATUS
4674 030712 000240 TST (RO) ;TEST ADDRESS
4675 030714 000207 NOP ;DELAY
4676 RTS PC
4677
4678
4679 LASTAD: 0 ;LAST ADDRESS IN CORE
4680 030716 000000 ADDERR: 0 ;ADDRESS DEROR FLAG
4681 030720 000000 VECADD:
4682 030722 000000 VECSTA:
4683 030722 000000 PRIOR: 0 ;PRIORITY LEVEL OF INTERRUPT
4684 030724 104402 030732 TTYTAB: TYPE MTAB
4685 030730 000207 RTS PC
4686 030732 000011 MTAB: .ASCIZ<11>
4687
4688 .EVEN
4689
4690 ;-----*
4691 ;SBTTL TTY DECIMAL OUTPUT ROUTINE
4692
4693 ;THIS ROUTINE CONVERTS A OCTAL VALUE IN THE EXCHANGE REGISTER TO
4694 ;A DECIMAL VALUE AND PRINTS IT AT THE TTY.
4695
4696
4697
4698 030734 000207 X=.
4699 030736 030736 RTS PC
4700 CTR=.
4701 030736 000000 0
4702 A=.
4703 030740 162700 000012 SUB #12,RO ;SUBTRACT 12

```

```

4704 030744 005237 030736      INC      CTR          ; INCREMENT COUNTER
4705 030750 000407              BR      AS240
4706 030752 032777 000001 150222 TYPTEN: BIT      #BIT0,ASWR    ; TEST INHIBIT PRINT BIT
4707 030760 001365              BNE      X              ; EXIT IF SET
4708 030762 005046              CLR      -(SP)          ; STORE STOP CODE
4709              030764      B=
4710 030764 005037 030736      CLR      CTR          ; SET THE COUNTER TO 000
4711 030770 000241      AS240: STAGAS=STAGAS+1
4712 030770 020027 000012      CMP      RO,#12          ; TEST EXCHANGE
4713 030774 103361              BHS      A              ; BRANCH IF MORE THAN 12
4714 030776 062700 000260      ADD      #260,RO         ; CONVERT OT TTY CODE
4715 031002 010046              MOV      RO,-(SP)        ; STACK THE CODE
4716 031004 013700 030736      MOV      CTR,RO         ; SET RO
4717 031010 001365              BNE      B              ; BRANCH IF RO NOT 000
4718              ; THIS ROUTINE PRINTS TTY CODES PREVIOUSLY PLACED ON THE STACK.
4719              ; TYPING A 00 CODE WILL TERMINATE THIS ROUTINE.
4720
4721
4722
4723 031012 031012      A=
4724              TYPSTK:
4725 031012 105777 150172      SAS=
4726 031016 100375              TSTB     #TPCSR
4727 031020 012677 150166      MOV      (SP)+,#TPDBR    ; PRINT
4728 031024 001372              BNE      A              ; NO-CONTINUE
4729 031026 031026      SAS=
4730 031026 105777 150156      TSTB     #TPCSR
4731 031032 100375              BPL      SAS
4732 031034 000207              RTS      PC
4733
4734
4735
4736
4737
4738
4739
4740
4741 031036 005001
4742 031040 012702 000002      A=
4743 031044 031044      CLR      R1          ; PRESET R1
4744 031044 010221      MOV      #2,R2        ; PRESET R2
4745 031046 022222      MOV      R2,(R1)+      ; SET ADDRESS
4746 031050 012721 020367      CMP      (R2)+,(R2)+    ; INCREMENT R2
4747 031054 022701 001000      MOV      #20367,(R1)+   ; SET STATUS
4748 031060 003371      CMP      #1000,R1        ; ALL ADDRESSES SET
4749 031062 012737 000340 000016      BGT      B        ; NO
4750 031070 012737 031114 000014      MOV      #340,16   ; SET TRAP STATUS
4751 031076 012737 000340 000022      MOV      #VECTOR,14 ; SET TRAP VECTOR
4752 031104 012737 030304 000020      MOV      #340,22   ; SET UP FOR
4753 031112 003207      MOV      #WRONG,20          ; DEROR VECTOR
4754 031114 005737 000000      RTS      PC
4755 031120 001746      VECTOR: TST      0          ; TEST ADDRESS 000
4756 031122 021627 001000      BEQ      A          ; BRANCH IF ZERO
4757 031126 003005      CMP      (SP),#1000        ; TEST THE ADDRESS
4758 031130 011637 031210      BGT      AS241
4759 031134 022626      MOV      (SP),VECSTR      ; STORE VECTOR ADDRESS
4760              POP2SP

```

A=.

TYPSTK:

```

SAS=
TSTB     #TPCSR
BPL      SAS
MOV      (SP)+,#TPDBR    ; PRINT
BNE      A              ; NO-CONTINUE
SAS=
TSTB     #TPCSR
BPL      SAS
RTS      PC

```

-----*

SBTTL INTERRUPT VECTOR ROUTINE

THIS ROUTINE SETS UP CORE ADDRESSES FROM 0 TO 1000 AND WATCHES
FOR ILLEGAL INTERRUPTS

A=.

```

CLR      R1          ; PRESET R1
MOV      #2,R2        ; PRESET R2
B=
MOV      R2,(R1)+      ; SET ADDRESS
CMP      (R2)+,(R2)+    ; INCREMENT R2
MOV      #20367,(R1)+  ; SET STATUS
CMP      #1000,R1        ; ALL ADDRESSES SET
BGT      B            ; NO
MOV      #340,16       ; SET TRAP STATUS
MOV      #VECTOR,14    ; SET TRAP VECTOR
MOV      #340,22       ; SET UP FOR
MOV      #WRONG,20      ; DEROR VECTOR
RTS      PC
VECTOR:  TST      0      ; TEST ADDRESS 000
        BEQ      A      ; BRANCH IF ZERO
        CMP      (SP),#1000 ; TEST THE ADDRESS
        BGT      AS241
        MOV      (SP),VECSTR ; STORE VECTOR ADDRESS
        POP2SP

```

```

4760      031136 000137 030304      A=.
4761      031142 000242      AS241: JMP      WRONG      ;GET PC AND SR
4762      031142 012737 177777 031210      MOV      #-1,VECSTR ;SET VECSTR TO -1
4763      031150 000772      BR      A
4764      031152 004737 030352      VECTR: JSR      PC,WRONGX
4765      031152 022737 177777 031210      CMP      #-1,VECSTR ;TEST FOR BREAK POINT
4766      031164 001410      BEQ      AS242
4767      031166 013700 031210      MOV      VECSTR,R0 ;GET ADDRESS
4768      031172 162700 000006      SUB      #6,R0 ;FIX IT
4769      031176 000000      A=.
4770      031200 005037 031210      HALT
4771      031204 000002      CLR      VECSTR ;CLEAR STORAGE
4772      031206 000243      RTI
4773      031206 000773      AS242: STAGAS=STAGAS+1
4774      031210 000000      BR      A
4775      031210 000000      VECSTR: 0
4776
4777      ;-----*
4778      ;SBTTL INTERRUPT HANDLING ROUTINE
4779
4780      ;THIS ROUTINE HANDLES INTERRUPTS FROM THE EQUIPMENT BEING
4781      ;TESTED.
4782
4783      ENAINT: MOV      #-1,INTFLG ;SET INTERRUPT TEST FLAG
4784      MOV      #340,PS ;SET PROCESSOR TO 7
4785      MOV      #INTIN,INTVCT ;SET UP VECTOR
4786      MOV      #340,INTARRIS ;SET UP STATUS
4787      BIS      #100,INTCR ;ENABLE INTERRUPT
4788      RTS      PC
4789
4790      CHKINT: CLR      PS ;CLEAR PROCESSOR PRIORITY
4791      NOP
4792      NOP ;DELAY
4793      TST      INTFLG ;TEST INTERRUPT FLAG
4794      BEQ      AS243
4795      ERROR
4796
4797      AS243: STAGAS=STAGAS+1
4798
4799      DISINT: BIC      #100,164000 ;DISABLE INTERRUPT
4800      CLR      INTFLG ;CLEAR INTERRUPT FLAG
4801      CLR      0 ;CLEAR VECTOR FLAG
4802      JSR      R5,SAVE
4803      JSR      PC,VECTOR
4804      JSR      R5,SETALL
4805      RTS      PC
4806
4807      INTFLG: 0 ;INTERRUPT FLAG = 0 WHEN INTERRUPT OCCURS
4808
4809      INTIN: BIC      #100,164000 ;DISABLE INTERRUPT
4810      CLR      INTFLG ;CLEAR THE FLAG
4811      NOP
4812
4813
4814
4815

```

```

4816 031344 000002          RTI
4817
4818
4819
4820
4821 ;-----*
4822 ;SBTTL  PASS NUMBER ROUTINE
4823
4824 ;THIS ROUTINE KEEPS TRACK OF THE CURRENT PASS COUNT AND
4825 ;DISPLAYS IT WHEN REQUESTED BY THE OPERATOR OR OTHER ROUTINES.
4826
4827          PASSUB=.
4828 031346 000000          0
4829          031350          PASTOL=.
4830 031350 000000          0
4831
4832 A=.
4833 PAST:  MOV      PASTOL,R0          ;SET EXCHANGE
4834        ADD      PASSUB,R0          ;ADD SUB TOTAL
4835        JSR      R5,SAVE
4836        JSR      PC,TYPTEN
4837        JSR      R5,SETALL
4838        B=.
4839        RTS      PC
4840
4841 ;-----*
4842 ;SBTTL  AUTO RESTART ROUTINE
4843
4844 AUTRES: NOP
4845 ;PRINT TOTAL PASSES COMPLETED
4846        JSR      PC,PRETST
4847        MOV      XTST,--(SP)        ;STACK THE TEST ADDRESS
4848        RTS      PC
4849
4850 ;-----*
4851 ;SBTTL  SAVE AND SET REGISTERS ROUTINE
4852
4853
4854 SAVE:  MOV      R0,--(SP)
4855        MOV      R1,--(SP)
4856        MOV      R2,--(SP)
4857        MOV      R3,--(SP)
4858        MOV      R4,--(SP)
4859        RTS      R5
4860
4861
4862
4863 SET:   TST      (SP)+
4864        MOV      (SP)+,R4
4865        MOV      (SP)+,R3
4866        MOV      (SP)+,R2
4867        MOV      (SP)+,R1
4868        TST      (SP)+
4869        RTS      R5          ;DONT RESTORE R0
4870
4871

```

```

4872
4873
4874
4875
4876
4877 031446 005726      SETALL: TST      (SP)+
4878 031450 012604      MOV      (SP)+,R4
4879 031452 012603      MOV      (SP)+,R3
4880 031454 012602      MOV      (SP)+,R2
4881 031456 012601      MOV      (SP)+,R1
4882 031460 000205      RTS      R5
4883
4884
4885      ;-----*
4886      ;SBTTL  PRESET ROUTINE
4887
4888      ;THIS ROUTINE IS USED TO PRESET ALL VARIABLES, REGISTERS ECT. PRIOR TO
4889      ;PERFORMING A TEST.
4890
4891
4892 031462 012600      PRESET: MOV      (SP)+,R0      ;GET THE RETURN
4893 031464 012706 001200      MOV      #STACK,SP      ;SET STACK
4894 031470 010046      MOV      R0,-(SP)      ;RESTORE RETURN
4895 031472 005000      CLR      R0      ;CLEAR THE REGISTERS
4896 031474 005001      CLR      R1
4897 031476 005002      CLR      R2
4898 031500 005003      CLR      R3
4899 031502 005004      CLR      R4
4900 031504 005005      CLR      R5
4901 031506 005077 147472      CLR      @TKCSR      ;CLEAR STATUS WORDS
4902 031512 005077 147472      CLR      @TPCSR
4903 031516 005077 147464      CLR      @TKDBR      ;CLEAR BUFFERS
4904 031522 005077 147464      CLR      @TPDBR
4905 031526 005037 177776      CLR      PS      ;CLEAR PROCESSOR STATUS
4906 031532 005037 031346      CLR      PASSUB      ;CLEAR PASS SUBTO TOTAL
4907 031536 005037 031350      CLR      PASTOL      ;CLEAR PASS TOTAL
4908 031542 005077 147434      CLR      @SWR      ;CLEAR SWITCH REGISTER
4909 031546 005037 031210      CLR      VECSTR
4910 031552 005037 031326      CLR      INTFLG      ;CLEAR INTERRUPT ENABLE
4911 031556 005037 000000      CLR      0
4912 031562 005037 001252      CLR      ERR      ;CLEAR THE DEROR FLAG
4913 031566 005037 001232      CLR      ERRCNT      ;CLEAR THE DEROR COUNT
4914 031572 000207      RTS      PC
4915
4916      ;-----*
4917      ;SBTTL  INPUT - OUTPUT BUFFERS
4918
4919 031574 000000 000000      BADOUT: 0,0
4920 031600      OUTPUT:      ;OUTPUT BUFFER STARTS HERE
4921
(1) 031600 000652 000525 000652 652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1) 031640 000652 000525 000652 652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1) 031700 000652 000525 000652 652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1) 031740 000652 000525 000652 652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1) ;100

```

(1)	032000	000001	000002	000403	001,002,403,004,405,406,007,010,411,412,013,414,015,016,417,020
(1)	032040	000421	000422	000023	421,422,023,424,025,026,427,430,031,032,433,034,435,436,037,040
(1)	032100	000441	000442	000043	441,442,043,444,045,046,447,450,051,052,453,054,455,456,057,460
(1)	032140	000061	000062	000463	061,062,463,064,465,466,067,070,471,472,073,474,075,076,477,100
(1)					;200
(1)					
(1)	032200	000501	000502	000103	501,502,103,504,105,106,507,510,111,112,513,114,515,516,117,520
(1)	032240	000121	000122	000523	121,122,523,124,525,526,127,130,531,532,133,534,135,136,537,540
(1)	032300	000141	000142	000543	141,142,543,144,545,546,147,150,551,552,153,554,155,156,557,160
(1)	032340	000561	000562	000163	561,562,163,564,165,166,567,570,171,172,573,174,575,576,177,200
(1)					;300
(1)					
(1)	032400	000601	000602	000203	601,602,203,604,205,206,607,610,211,212,613,214,615,616,217,620
(1)	032440	000221	000222	000623	221,222,623,224,625,626,227,230,631,632,233,634,235,236,637,640
(1)	032500	000241	000242	000643	241,242,643,244,645,646,247,250,651,652,253,654,255,256,657,260
(1)	032540	000661	000662	000263	661,662,263,664,265,266,667,670,271,272,673,274,675,676,277,700
(1)					;400
(1)					
(1)	032600	000301	000302	000703	301,302,703,304,705,706,307,310,711,712,313,714,315,316,717,320
(1)	032640	000721	000722	000323	721,722,323,724,325,326,727,730,331,332,733,334,735,736,337,340
(1)	032700	000741	000742	000343	741,742,343,744,345,346,747,750,351,352,753,354,755,756,357,760
(1)	032740	000361	000362	000763	361,362,763,364,765,766,367,370,771,772,373,774,375,376,777,400
(1)					;500
(1)					
(1)	033000	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)	033040	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)	033100	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)	033140	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)					;600
(1)					
(1)	033200	000777	000400	000777	777,400,777,400,777,400,777,400,777,400,777,400,777,400,777,400
(1)	033240	000001	000001	000001	001,001,001,001,001,001,001,001,001,001,001,001,001,001,001
(1)	033300	000010	000010	000010	010,010,010,010,010,010,010,010,010,010,010,010,010,010,010
(1)	033340	000100	000100	000100	100,100,100,100,100,100,100,100,100,100,100,100,100,100,100
(1)					;700
(1)					
(1)	033400	000777	000400	000777	777,400,777,400,777,400,777,400,777,400,777,400,777,400,777,400
(1)	033440	000002	000002	000002	002,002,002,002,002,002,002,002,002,002,002,002,002,002,002
(1)	033500	000020	000020	000020	020,020,020,020,020,020,020,020,020,020,020,020,020,020,020
(1)	033540	000200	000200	000200	200,200,200,200,200,200,200,200,200,200,200,200,200,200,200
(1)					;1000
(1)	033600	000777	000400	000777	777,400,777,400,777,400,777,400,777,400,777,400,777,400,777,400
(1)	033640	000004	000004	000004	004,004,004,004,004,004,004,004,004,004,004,004,004,004,004
(1)	033700	000040	000040	000040	040,040,040,040,040,040,040,040,040,040,040,040,040,040,040
(1)	033740	000400	000400	000400	400,400,400,400,400,400,400,400,400,400,400,400,400,400,400
(1)					;1100
(1)					
(1)	034000	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)	034040	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)	034100	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)	034140	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)					;1200
(1)					
(1)	034200	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)	034240	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525
(1)	034300	000652	000525	000652	652,525,652,525,652,525,652,525,652,525,652,525,652,525,652,525

5
6
7
8
9
0
1
2
3
4
5
6
7
8
9

```

:THIS UTILITY CALCULATES PRIORITY LEVEL,SETS UP CSR'S,SETS UP VECTORS.
TRLEV: ASL      TRPRT      ;BUILD PRIORITY IN THIS LOCATION
        ASL      TRPRT      ;USING ARITHMETIC SHIFTS, ROTATE
        ASL      TRPRT      ;THE PRIORITY LEVEL PAST
        ASL      TRPRT      ;THE BIT POSITIONS CORRE-
        ASL      TRPRT      ;SPONDING TO THE CONDITION CODES
        MOV      TRPRT,LESS1 ;MOVE THIS TO LESS1
        DEC      LESS1 ;CREATE THE NEXT LOWEST PRIORITY
        BIC      #37,LESS1 ;INSURE THAT THE INZVC BITS ARE CLEAR
        MOV      TRVCT,RO   ;PLACE THE BASE VECTOR ADDRESS IN RO
        TST      (RO)+      ;CALCULATE THE RECEIVER INTERRUPT STATUS ADDR.
        MOV      RO,TRRIS   ;STORE IT HERE

;THIS SEGMENT SETS UP POINTERS FOR THE GIVEN TR79. TRBASE IS THE BASE ADDRESS
;OF THE DEVICE

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 100
 CZTRAB.HED 14-DEC-77 12:19 INPUT - OUTPUT BUFFERS

SEQ 0099

```

4940 041716 013700 001242      MOV      TRBASE,RO      ;COPY THE BASE ADDRESS INTO A SCRATCH REGISTER
4941 041722 010037 001402      MOV      RO,TRCA ;XXX0
4942 041726 005200              INC      RO
4943 041730 010037 001404      MOV      RO,HTRCR      ;XXX1
4944 041734 005200              INC      RO
4945 041736 010037 001406      MOV      RO,TRSR ;XXX2
4946 041742 005200              INC      RO
4947 041744 010037 001410      MOV      RO,HTRSR      ;XXX3
4948 041750 005200              INC      RO
4949 041752 010037 001412      MOV      RO,TRWC ;XXX4
4950 041756 005720              TST      (RO)+
4951 041760 010037 001416      MOV      RO,TRBA ;XXX6
4952 041764 000207              RTS      PC
4953 041766 000200              TRPRT:  LEVEL4
4954 041770 000140              LESS1:  LEVEL3 ;LEVEL TO ALLOW INTERRUPTS
4955
4956              ;THIS ROUTINE CALCULATES ODD PARITY FOR AN 8 BIT CHAR
4957 041772 013700 001304      ODD8:   MOV      TEMP1,RO      ;SAVE TEMP1
4958 041776 005001              CLR      R1      ;USE R1 TO CREATE PARITY BIT
4959 042000 012727 000010      MOV      #8.,(PC)+    ;COUNT THE NUMBER OF BITS TO CALCULATE
4960 042004 000000              4$:      0      ;USE THIS WORD AS A LOOP COUNTER
4961 042006 006000              1$:      ROR      RO      ;GET A BIT INTO THE CARRY BIT
4962 042010 005501              ADC      R1      ;ADD THE CARRY BIT TO THE BIT COUNT
4963 042012 005337 042004      DEC      4$      ;REDUCE THE COUNT. ARE 8 BITS CHECKED?
4964 042016 001373              BNE      1$      ;IF NOT, GO CHECK THE NEXT BIT
4965 042020 006001              ROR      R1      ;IF SO, GET THE SCALE OF THE SUM INTO THE C BIT
4966 042022 103404              BCS      2$      ;IF IT IS ODD, GO SET UP ODD PARITY
4967 042024 052737 000400 001304  BIS      #BIT8,TEMP1    ;SET ODD PARITY
4968 042032 000403              BR       3$
4969 042034 042737 000400 001304  2$:      BIC      #BIT8,TEMP1    ;CLR EVEN PARITY
4970              ;TEMP1 NOW HAS ODD PARITY CHARACTER
4971 042042 000207              3$:      RTS      PC
4972
4973              ;THIS ROUTINE CALCULATES EVEN PARITY FOR AN 8 BIT CHARACTER
4974 042044 013700 001304      EVEN8:  MOV      TEMP1,RO      ;SAVE TEMP1
4975 042050 005001              CLR      R1
4976 042052 012727 000010      MOV      #8.,(PC)+
4977 042056 000000              4$:      0
4978 042060 006000              1$:      ROR      RO
4979 042062 005501              ADC      R1
4980 042064 005337 042056      DEC      4$
4981 042070 001373              BNE      1$
4982 042072 006001              ROR      R1
4983 042074 103004              BCC      2$
4984 042076 052737 000400 001304  BIS      #BIT8,TEMP1    ;SET EVEN PARITY
4985 042104 000403              BR       3$
4986 042106 042737 000400 001304  2$:      BIC      #BIT8,TEMP1    ;CLR ODD PARITY
4987              ;TEMP1 NOW HAS EVEN PARITY CHARACTER
4988 042114 000207              3$:      RTS      PC
4989 042116 011637 001304      TRPREG: MOV      (SP),TEMP1    ;SAVE PC
4990 042122 104007              ERROR   7      ;DEVICE NON-EXISTENT
4991 042124 000002              RTI

```

```

4992
4993 042126 000000
4994 042130 000000
4995 042132 000000
4996
4997 042134 042260
4998 042136 042713
4999 042140 043122
5000
5001 042142 042302
5002 042144 042713
5003 042146 043122
5004
5005 042150 042326
5006 042152 042713
5007 042154 043122
5008
5009 042156 042344
5010 042160 000000
5011 042162 000000
5012
5013 042164 042405
5014 042166 000000
5015 042170 000000
5016
5017 042172 042440
5018 042174 042750
5019 042176 043140
5020
5021 042200 042455
5022 042202 043005
5023 042204 043156
5024
5025 042206 042504
5026 042210 043033
5027 042212 043170
5028
5029 042214 042527
5030 042216 000000
5031 042220 000000
5032
5033 042222 042546
5034 042224 043044
5035 042226 043176
5036
5037 042230 042344
5038 042232 043067
5039 042234 043210
5040
5041 042236 042567
5042 042240 000000
5043 042242 000000
5044
5045 042244 042626
5046 042246 000000
5047 042250 000000

```

		; ERROR TABLE			
.ERRTAB:		0	;ERROR 0	BIT ERROR (GENERAL)	
		0			
		0			
		EM1	;ERROR 1	REGISTER ERROR	
		DH1			
		DT1			
		EM2	;ERROR 2	CONTROLLER ERROR	
		DH1			
		DT1			
		EM3	;ERROR 3	UNIT ERROR	
		DH1			
		DT1			
		MRECF	;ERROR 4 REC FAILED TO INTERRUPT		
		0			
		0			
		MTRANF	;ERROR 5 TRANS FAILED TO INTERRUPT		
		0			
		0			
		MDATAE	;ERROR 6 DATA ERROR		
		DH2			
		DT2			
		MBUS	;ERROR 7	DEVICE NON-EXISTENT	
		DH3			
		DT3			
		MUNITR	;ERROR 10	MISC. UNIT ERROR	
		DH4			
		DT4			
		MWUNIT	;ERROR 11	INVALID UNIT	
		0			
		0			
		SIZER0	;ERROR 12	AUTOSIZE ERROR	
		DH5			
		DT5			
		MRECF	;ERROR 13	AUTOSIZE ERROR	
		DH6			
		DT6			
		SIZER1	;ERROR 14	AUTOSIZE ERROR	
		0			
		0			
		SIZER2	;ERROR 15	AUTOSIZE ERROR	
		0			
		0			

```

5048
5049 042252 042663          SIZER3 ;ERROR 16      AUTOSIZE ERROR
5050 042254 043067          DH6
5051 042256 043210          DT6
5052
5053
5054
5055
5056 042260 020377 042522 044507 EM1: ;ERROR MESSAGES
(1) 042302 020377 047503 052116 EM2: .ASCIZ <377>/ REGISTER ERROR /
(1) 042326 020377 047125 052111 EM3: .ASCIZ <377>/ CONTROLLER ERROR /
(1) 042344 041777 047117 051124 MRECF: .ASCIZ <377>/ UNIT ERROR /
(1) 042405 377 047125 052111 MTRANF: .ASCIZ <377>/CONTROLLER FAILED TO INTERRUPT /
(1) 042440 042377 052101 020101 MDATAE: .ASCIZ <377>/UNIT FAILED TO INTERRUPT /
(1) 042455 377 042504 044526 MBUS: .ASCIZ <377>/DATA ERROR /
(1) 042504 046777 051511 027103 MBUS: .ASCIZ <377>/DEVICE NON-EXISTENT /
(1) 042527 377 047111 040526 MUNITR: .ASCIZ <377>/MISC. UNIT ERROR /
(1) 042546 040777 052125 051517 MWUNIT: .ASCIZ <377>/INVALID UNIT /
(1) 042567 377 051503 020122 SIZER0: .ASCIZ <377>/AUTOSIZE ERROR /
(1) 042626 052377 040522 020520 SIZER1: .ASCIZ<377>/CSR DID NOT AUTOSIZE IN RANGE/
(1) 042663 377 047503 052116 SIZER2: .ASCIZ<377>/TRAP! - CSR ADDRESS FAILURE/
(1) SIZER3: .ASCIZ<377>/CONTROLLER INTERRUPTED/
(1) ;DATA HEADERS FOR ERROR MESSAGES
(1) (1) 042713 377 042522 044507 DH1: .ASCIZ <377>/REGISTER EXPECTED ACTUAL/
(1) (1) 042750 052777 044516 020124 DH2: .ASCIZ <377>/UNIT # EXPECTED ACTUAL/
(1) (1) 043005 377 051105 047522 DH3: .ASCIZ <377>/ERROR PC ADDRESS /
(1) (1) 043033 377 047125 052111 DH4: .ASCIZ <377>/UNIT # /
(1) (1) 043044 042777 050130 041505 DH5: .ASCIZ <377>/EXPECTED ACTUAL/
(1) (1) 043067 377 050103 020125 DH6: .ASCIZ <377>/CPU PRIORITY LEVEL WAS: /
(1)
(1) 043122 .EVEN
(1)
(1) 043122 000003 DT1: 3 ;DATA TABLES FOR ERROR MESSAGES
(1) 043124 006 004 .BYTE 6,4
(1) 043126 001244 REGIST ;DEVICE REGISTER
(1) 043130 006 004 .BYTE 6,4
(1) 043132 001330 SAVR5 ;EXPECTED DATA
(1) 043134 006 002 .BYTE 6,2
(1) 043136 001326 SAVR4 ;ACTUAL DATA
(1)
(1) 043140 000003 DT2: 3
(1) 043142 003 007 .BYTE 3,7
(1) 043144 001236 SAVLIN ;UNIT #
(1) 043146 003 007 .BYTE 3,7
(1) 043150 001330 SAVR5 ;EXPECTED DATA
(1) 043152 003 002 .BYTE 3,2
(1) 043154 001326 SAVR4 ;ACTUAL DATA
(1)
(1) 043156 000002 DT3: 2
(1) 043160 006 004 .BYTE 6,4
(1) 043162 001304 TEMPI ;PC
(1) 043164 006 002 .BYTE 6,2
(1) 043166 001244 REGIST ;DEVICE REGISTER
(1)
(1) 043170 000001 DT4: 1
(1) 043172 003 002 .BYTE 3,2
(1) 043174 001236 SAVLIN ;UNIT #

```

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 103
CZTRAB.HED 14-DEC-77 12:19 INPUT - OUTPUT BUFFERS

SEQ 0102

```
(1)
(1) 043176 000002          DT5: 2
(1) 043200 006          .BYTE 6,5
(1) 043202 001336        SAVED ;EXPECTED DATA
(1) 043204 006          .BYTE 6,2
(1) 043206 001340        SAVE1 ;ACTUAL DATA
(1)
(1) 043210 000001          DT6: 1
(1) 043212 001          .BYTE 1,2
(1) 043214 001340        SAVE1 ;ACTUAL DATA
(1) 043216
      CORMAX:
      .NLIST SEQ,LOC,BIN
      ;.ERROR ;YOU HAVE EXCEEDED 8K CORE LIMITS.
      .END
```

02TRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 105

2TRAB.HED 14-DEC-77 12:19

CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0103

[illegible]

AS146	024150	3672	3679#
AS147	024210	3685	3692#
AS15	015070	2197	2204#
AS150	024242	3696	3703#
AS151	024274	3706	3713#
AS152	024366	3732	3739#
AS153	024420	3742	3749#
AS154	024450	3752	3759#
AS155	024502	3762	3769#
AS156	024554	3785	3791#
AS157	024634	3810	3817#
AS16	015122	2222	2226#
AS160	024666	3820	3827#
AS161	024720	3830	3837#
AS162	024752	3840	3847#
AS163	025050	3868	3875#
AS164	025102	3878	3885#
AS165	025134	3888	3895#
AS166	025166	3898	3905#
AS167	025216	3908	3914#
AS17	015140	2228	2232#
AS170	025332	3941	3948#
AS171	025364	3951	3958#
AS172	025430	3967	3974#
AS173	025460	3977	3984#
AS174	025574	4012	4019#
AS175	025626	4022	4029#
AS176	025660	4032	4039#
AS177	025712	4042	4049#
AS2	014202	2002	2010#
AS20	015444	2300	2307#
AS200	026026	4077	4084#
AS201	026060	4087	4094#
AS202	026110	4097	4104#
AS203	026142	4107	4114#
AS204	026202	4125	4132#
AS205	026264	4146	4160#
AS206	026340	4163	4177#
AS207	026372	4180	4187#
AS21	015476	2310	2317#
AS210	026446	4195	4208#
AS211	026520	4211	4224#
AS212	026552	4227	4234#
AS213	026724	4281	4288#
AS214	026760	4291	4298#
AS215	027014	4301	4308#
AS216	027052	4319	4326#
AS217	027106	4329	4336#
AS22	015526	2320	2327#
AS220	027154	4345	4352#
AS221	027206	4355	4362#
AS222	027562	4447#	
AS223	027610	4453#	
AS224	027662	4457	4465#
AS225	030010	4481	4497#
AS226	030026	4507	4509#

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 107
CZTRAB.WED 14-DEC-77 12:19 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0105

AS227	030110	4515	4523#
AS23	015556	2330	2337#
AS230	030162	4527	4535#
AS231	030250	4563	4565#
AS232	030260	4567	4570#
AS233	030272	4572	4574#
AS234	030350	4593	4595#
AS235	030402	4602	4606#
AS236	030422	4609	4611#
AS237	030632	4657	4659#
AS24	015662	2364	2373#
AS240	030770	4705	4711#
AS241	031142	4757	4762#
AS242	031206	4768	4775#
AS243	031272	4797	4800#
AS25	015700	2380#	
AS26	016002	2413	2420#
AS27	016034	2423	2430#
AS3	014234	2013	2021#
AS30	016064	2433	2440#
AS31	016114	2443	2450#
AS32	016212	2473	2480#
AS33	016242	2483	2489#
AS34	016272	2492	2498#
AS35	016320	2501	2506#
AS36	016502	2537	2544#
AS37	016542	2549	2556#
AS4	014342	2042	2049#
AS40	016574	2559	2566#
AS41	016624	2569	2576#
AS42	016764	2600	2607#
AS43	017016	2610	2617#
AS44	017050	2620	2627#
AS45	017100	2630	2637#
AS46	017214	2663	2670#
AS47	017246	2673	2680#
AS5	014374	2052	2059#
AS50	017300	2683	2690#
AS51	017332	2693	2700#
AS52	017454	2725	2732#
AS53	017505	2735	2742#
AS54	017540	2745	2752#
AS55	017572	2755	2762#
AS56	017732	2790	2797#
AS57	017764	2800	2807#
AS6	014514	2087	2094#
AS60	020016	2810	2817#
AS61	020050	2820	2827#
AS62	020212	2854	2860#
AS63	020244	2862	2869#
AS64	020276	2872	2879#
AS65	020330	2882	2889#
AS66	020454	2917	2924#
AS67	020506	2927	2934#
AS7	014546	2097	2104#
AS70	020540	2937	2944#

[illegible]

PARAM =	104407	534#	630	642	653	1266	1611										
PARAM1	004162	978#	996														
PARERR	004236	981	983	985	995#	1002	1004	1006									
PASCNT	001230	458#	585*	835*	836	866											
PASRAT=	000012	1295#															
PASSUB=	031346	4827#	4833	4906*													
PAST	031352	4832#															
PASTOL=	031350	4829#	4832	4907*													
PFTAB	005306	1222	1226#														
POPRO =	012600	333#	1188														
POP1SP=	005726	331#															
POP2SP=	022626	335#	603	683	732	740	741	777	892	4655	4759						
PRESET	031462	4892#															
PRETST	011044	1303#	1360	4845													
PRHLT	010323	691	728	734	774	1229#											
PRIOR	030722	4684#															
PS =	177776	324#	583*	598*	715*	753*	810*	1778*	1801*	4786*	4793*	4905*					
PUSHRO=	010046	332#	1185														
PUSHIS=	005746	330#															
PUSH2S=	024646	334#															
PWRCLR=	004000	387#	1313	1314	1685	1686	1708	1709	1712	1713	1723	1724	1915	1916			
		1925	1926	1937	1938	1965	1966	2022	2023	2061	2062	2171	2172	2243			
		2244	2282	2283	2345	2346	2578	2579	2639	2640	2764	2765	2829	2830			
		2851	2892	2956	2957	3021	3022	3135	3136	3204	3205	3333	3334	3654			
		3655	3715	3716	3771	3772	3849	3850	3915	3916	3986	3987	4051	4052			
		4520	4521														
QV.FLG	001347	508#	587*	846*	882												
READ =	000004	378#	2469	2904	2969	3441	3592	3862	4063	4403	4415						
REGIST	001244	468#	1321*	1340*	1351*	1726*	1863*	1875*	1900*	1910*	1949*	1959*	2005*	2016*			
		2045*	2055*	2090*	2100*	2124*	2134*	2190*	2200*	2303*	2313*	2323*	2333*	2367*			
		2416*	2426*	2436*	2446*	2476*	2486*	2495*	2504*	2540*	2552*	2562*	2572*	2603*			
		2613*	2623*	2633*	2666*	2676*	2686*	2696*	2728*	2738*	2748*	2758*	2793*	2803*			
		2813*	2823*	2857*	2865*	2875*	2885*	2920*	2930*	2940*	2950*	2985*	2995*	3005*			
		3015*	3042*	3052*	3062*	3072*	3095*	3105*	3114*	3123*	3161*	3171*	3181*	3191*			
		3266*	3279*	3289*	3299*	3354*	3364*	3375*	3398*	3408*	3418*	3428*	3450*	3460*			

CZTRABD MACY11 30(1046) 14-DEC-77 12:32 PAGE 113

CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0111

SAVE1	001340	498#	681*	693	695	716*	738*	739*	742	744	749*	765	770*	771
		778	780	5056										
SAVLIN	001236	461#	5056											
SAVPC	001334	496#	1028*	1200										
SAVR0	001316	489#	1037*	1042										
SAVR1	001320	490#	1036*	1043										
SAVR2	001322	491#	1035*	1044										
SAVR3	001324	492#	1034*	1045										
SAVR4	001326	493#	1033*	1046	5056									
SAVR5	001330	494#	1032*	1047	5056									
SAVSP	001332	495#												
SAV05 =	104410	536#	1138											
SCOPE =	104400	520#	1395	1443	1490	1542	1587	1632						
SCOPI =	104401	522#												
SET	031430	4863#												
SETALL	031446	4808	4836	4877#										
SIZE	010401	665	1229#											
SIZER0	042546	5033	5056#											
SIZER1	042567	5041	5056#											
SIZER2	042626	5045	5056#											
SIZER3	042663	5049	5056#											
SIZTAB	027356	4370	4403#	4408										
SPACER=	001+10	380#	2409	2595	3034	3087	3389	3539	3726	3804	3998	4259	4403	
SPACNT	004573	1097#												
STACK =	001200	325#	582	668	701	752	811	1195	1217	1380	1428	1475	1526	1567
		1607	4893											
STAT	001246	469#												
SUBTST	027430	4376	4403#											
SV05	004330	1032#												
SWR	001202	440#	599	601*	611	661	814	875	900	1127	1132	1183	1190	1192
		1260	1358	1375	1390	1438	1485	1537	1563	1578	1584	1627	4601	4706
		4908#												
SWREG	000176	431#	601											
SW00 =	000001	228#	611											
SW01 =	000002	227#	1260											
SW02 =	000004	226#												
SW03 =	000010	222#												
SW04 =	000020	221#												
SW05 =	000040	219#												
SW06 =	000100	218#												
SW07 =	000200	217#												
SW08 =	000400	215#	1190											
SW09 =	001000	214#	900											
SW10 =	002000	213#	1192											
SW11 =	004000	212#												
SW12 =	010000	211#	1127											
SW13 =	020000	210#	1132											
SW14 =	040000	209#												
SW15 =	100000	208#												
TABLE1	011460	1379	1401#											
TABLE2	011624	1427	1449#											
TABLE3	011764	1474	1496#											
TABLE4	012136	1525	1548#											
TABLE5	012306	1566	1591#											
TABLE6	012520	1606	1614	1617	1645#									
TBA =	164006	398#												

TC	001302	483#	1379*	1382	1383	1384*	1427*	1430	1431	1432*	1474*	1477	1478	1479*
TCR	= 164000	1525*	1528	1529	1530*	1566*	1569	1570	1571*	1606*	1617*	1618	1619	1620*
TEMP	010560	395#												
TEMP1	001304	1241#	1679*	1699*	1841*	2159*	2520*	2524*	2529*	2534*	3246*			
		484#	748*	753	769*	787*	793*	795*	1122*	1231	4957	4967*	4969*	4974
		4984*	4986*	4989*	5056									
TEMP2	001306	485#	789*	1233										
TEMP3	001310	486#												
TEMP4	001312	487#												
TEMP5	001314	488#												
TESTA	012524	1655#	2597	3156	4476	4546								
TESTAA	017612	2772#												
TESTAB	020070	1458	2837#											
TESTAC	020350	1506	2899#											
TESTAD	020610	1507	2964#											
TESTAE	021050	1508	3030#											
TESTAF	021242	1509	3083#											
TESTAG	021426	3134#												
TESTAH	021712	2071	2514	2588	2649	2708	2773	2838	2900	2965	3031	3084	3203#	3925
TESTAJ	021752	1404	3217#											
TESTAK	022024	1406	3241#											
TESTAL	022260	1408	3310#	3387	3483									
TESTAM	022616	1409	3386#											
TESTAN	023024	1410	3440#											
TESTAO	023172	1411	3482#	3537										
TESTAP	023400	1412	3536#	3590										
TESTAQ	023606	1413	3589#											
TESTAS	024042	3653#												
TESTAT	024314	1457	3723#											
TESTAU	024522	1459	3781#											
TESTAV	024562	1500	3800#											
TESTAW	024772	1501	1592	3857#										
TESTAX	025236	1510	3924#	3995										
TESTAY	025500	1511	3994#	4060										
TESTAZ	025732	1512	4059#											
TESTB	012564	1401	1672#											
TESTBA	026144	1460	4121#											
TESTC	012646	1403	1450	1496	1548	1694#								
TESTCA	026560	2589	4240#											
TESTCK	026610	1889	2901	2966	3032	3085	4250#							
TESTCX	026210	4145#	4247	4255	4403									
TESTD	012760	1451	1722#											
TESTDX	026374	4194#	4403	4418										
TESTE	013146	672	1452	1778#										
TESTEK	026630	2902	2967	4257#										

[illegible]

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 116
CZTRAB.HED 14-DEC-77 12:19 CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

[illegible]

4771	4827	4829	4831
810#			
543	1054#		
541	1053#		
832#	1586	1605	
417	1127#		
1146	4993#		
533	958#		
531	937#		
941#	961		
939*	942#		
535	969#		
415	584	1208#	1216
539	1042#		
537	1028#		
521	874#		

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 119
 CZTRAB.HED 14-DEC-77 12:19 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0117

.SCOP1	003662	523	900#	
.START	001510	433	581#	592
.TRPSR	004576	419	1107#	
.TRPTA	001352	519#	1112	
.TYPE	003706	525	910#	
.TYPEF	003752	527	926#	
.TYPEL	003764	529	928#	

ADDTST	1#														
ASKOPR	1#														
AUTRES	1#														
BELL	1#	1679	1699	2159	3246										
BREAK	1#	1386	1434	1481	1532	1573	1622	4472	4477	4542	4547				
CHKINT	1#	4246													
CRLF	1#	4612													
DEFOR	1#	1319	1338	1349	1861	1873	1898	1908	1947	1957	2003	2014	2043	2053	2088
	2098	2122	2132	2188	2198	2301	2311	2321	2331	2365	2414	2424	2434	2444	2474
	2484	2493	2502	2538	2550	2560	2570	2601	2611	2621	2631	2664	2674	2684	2694
	2726	2736	2746	2756	2791	2801	2811	2821	2855	2863	2873	2883	2918	2928	2938
	2948	2983	2993	3003	3013	3040	3050	3060	3070	3093	3103	3112	3121	3159	3169
	3179	3189	3264	3277	3287	3297	3352	3362	3373	3396	3406	3416	3426	3448	3458
	3469	3492	3502	3512	3522	3546	3556	3566	3576	3599	3609	3619	3629	3673	3686
	3697	3707	3733	3743	3753	3763	3786	3811	3821	3831	3841	3869	3879	3889	3899
	3909	3942	3952	3968	3978	4013	4023	4033	4043	4078	4088	4098	4108	4126	4153
	4170	4181	4201	4217	4228	4282	4292	4302	4320	4330	4346	4356	4458	4528	
DISINT	1#														
DO	1#	4806	4834												
ENAINI	1#	1977	4240	4377	4410										
EOTFUN	1#	2253	3313	3329	3345	3389	3440	3485	3539	3592					
ERROR	336#	690	696	727	733	745	767	782	1322	1341	1352	1730	1737	1745	1753
	1761	1768	1782	1797	1818	1848	1864	1876	1901	1911	1950	1960	2006	2017	2046
	2056	2091	2101	2125	2135	2167	2191	2201	2223	2229	2304	2314	2324	2334	2368
	2417	2427	2437	2447	2477	2487	2496	2505	2541	2553	2563	2573	2604	2614	2624
	2634	2667	2677	2687	2697	2729	2739	2749	2759	2794	2804	2814	2824	2858	2866
	2876	2886	2921	2931	2941	2951	2986	2996	3006	3016	3043	3053	3063	3073	3096
	3106	3115	3124	3162	3172	3182	3192	3267	3280	3290	3300	3355	3365	3376	3399
	3409	3419	3429	3451	3461	3472	3495	3505	3515	3525	3549	3559	3569	3579	3602
	3612	3622	3632	3676	3689	3700	3710	3736	3746	3756	3766	3789	3814	3824	3834
	3844	3872	3882	3892	3902	3912	3945	3955	3971	3981	4016	4026	4036	4046	4081
	4091	4101	4111	4129	4157	4174	4184	4205	4221	4231	4285	4295	4305	4323	4333
	4349	4359	4461	4517	4531	4658	4798	4990							
GETADD	1#														

TYPTAB	1#														
TYPTEN	1#	4834													
VECTOR	1#	4806													
WRONG	1#														
WRONGX	1#														
\$BUFFE	1#	1235													
\$CONFI	1#	4925													
\$CYCLE	1#	1245													
\$DELAY	1#	1305	1327	1822	1830	1990	2234	2258	2266	2274	2291	2654	2716	2781	2845
	2907	2972	3148	3226	3254	3662	3931	4001	4066						
\$DPOIN	1#														
\$EOP	1#	826													
\$ERTAB	1#	4992													
\$FINI	1#	5056													
\$GETPA	1#	627	639	651	1264	1609									
\$HEAD	1#	186													
\$HERES	1#	1953	1963	2010	2021	2049	2059	2094	2104	2128	2138	2164	2170	2194	2204
	2226	2232	2307	2317	2327	2337	2373	2380	2420	2430	2440	2450	2480	2489	2498
	2506	2544	2556	2566	2576	2607	2617	2627	2637	2670	2680	2690	2700	2732	2742
	2752	2762	2797	2807	2817	2827	2860	2869	2879	2889	2924	2934	2944	2954	2989
	2999	3009	3019	3046	3056	3066	3076	3099	3108	3117	3126	3165	3175	3185	3195
	3210	3270	3283	3293	3303	3328	3358	3368	3379	3402	3412	3422	3432	3454	3464
	3475	3498	3508	3518	3528	3552	3562	3572	3582	3605	3615	3625	3635	3679	3692
	3703	3713	3739	3749	3759	3769	3791	3817	3827	3837	3847	3875	3885	3895	3905
	3914	3948	3958	3974	3984	4019	4029	4039	4049	4084	4094	4104	4114	4132	4160
	4177	4187	4208	4224	4234	4288	4298	4308	4326	4336	4352	4362	4447	4453	4465
	4497	4509	4523	4535	4565	4570	4574	4595	4606	4611	4659	4711	4762	4775	4800
\$INTER	1#														
\$INTSE	1#														
\$JUNK	1#	567													
\$LINEU	1#														
\$LVLT	1#														
\$MRESE	1#	1313	1685	1708	1712	1722	1915	1924	1937	1965	2022	2061	2171	2243	2282
	2345	2578	2639	2764	2829	2891	2956	3021	3134	3203	3333	3653	3715	3771	3849
	3915	3986	4051	4520											
\$MRR	1#														
\$MRRD	1#														
\$MRRW	1#														
\$MRRWR	1#														
\$MRWD	1#														
\$MSG	1#	1229													
\$NOSLA	1#	4989													
\$PARB	1#	4956													
\$PFAIL	1#	1204													
\$PRINT	1#														
\$QUEST	1#	594													
\$RESTO	1#														
\$SCOPE	1#	870													
\$SKIP\$	1#	1946	1956	2002	2013	2042	2052	2087	2097	2121	2131	2158	2166	2187	2197
	2222	2228	2300	2310	2320	2330	2364	2413	2423	2433	2443	2473	2483	2492	2501
	2537	2549	2559	2569	2600	2610	2620	2630	2663	2673	2683	2693	2725	2735	2745
	2755	2790	2800	2810	2820	2854	2862	2872	2882	2917	2927	2937	2947	2982	2992
	3002	3012	3039	3049	3059	3069	3092	3102	3111	3120	3158	3168	3178	3188	3208
	3263	3276	3286	3296	3312	3351	3361	3372	3395	3405	3415	3425	3447	3457	3468
	3491	3501	3511	3521	3545	3555	3565	3575	3598	3608	3618	3628	3672	3685	3696
	3706	3732	3742	3752	3762	3785	3810	3820	3830	3840	3868	3878	3888	3898	3908

CZTRABO MACY11 30(1046) 14-DEC-77 12:32 PAGE 124

CZTRAB.HED 14-DEC-77 12:19

CROSS REFERENCE TABLE -- MACRO NAMES

SEQ 0121

	3941	3951	3967	3977	4012	4022	4032	4042	4077	4087	4097	4107	4125	4146	4163
	4180	4195	4211	4227	4281	4291	4301	4319	4329	4345	4355	4457	4481	4507	4515
	4527	4563	4567	4572	4593	4602	4609	4657	4705	4757	4768	4797			
STRPDE	1#	520	522	524	526	528	530	532	534	536	538	540	542		
STR79	1#	1292													
STSTN	1#	1369	1420	1467	1518	1556	1599								
STUNIT	1#														
SUNIBU	1#														
SVARIA	1#	435													
SXZ	1#	1366	1368	1417	1419	1464	1466	1515	1517	1553	1555	1596	1598		
SSSSSB	1#														

. ABS. 043216 000

ERRORS DETECTED: 0

CZTRAB.BIN,CZTRAB.LST/CRF/SOL/NL:TOC=CZTRAB.P11,CZTRAB.HED

RUN-TIME: 13 18 2 SECONDS

RUN-TIME RATIO: 165/34=4.7

CORE USED: 37K (73 PAGES)

F10