

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43

.REM *

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DFKTG-A-D
PRODUCT NAME: 11/34 MEMORY MANAGEMENT EXERCISER
DATE CREATED: DECEMBER 21, 1975
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: GLENN JOHNSON

COPYRIGHT (C) DIGITAL EQUIPMENT CORPORATION
1975

THE MATERIAL IN THIS DOCUMENT IS FOR INFORMATION
PURPOSES ONLY AND IS SUBJECT TO CHANGE WITHOUT NOTICE.
DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY
FOR THE USE OF SOFTWARE ON EQUIPMENT WHICH IS NOT
SUPPLIED BY IT.
DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY
FOR ANY ERRORS WHICH MAY APPEAR IN THE DOCUMENT.

44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95

1.0 ABSTRACT

THIS PROGRAM IS AN INTERACTIVE EXERCISER FOR A PDP-11/34
IT PERFORMS A TEST OF INSTRUCTIONS AND
CONCURRENT OPERATIONS OF I/O EQUIPMENT WHILE RELOCATING THRU MEMORY.
IT PROVIDES NUMEROUS MODES OF TESTING, FROM 4K EXECUTION WITH
THE MEMORY MANAG. TURNED OFF AND ONLY KERNEL MODE IN USE, TO 128K EXECUTION
WITH EACH USER PAGE MAPPED SEQUENTIALLY TO EVERY 4K BANK OF MEMORY,
THIS PROGRAM IS NOT TO BE CONSIDERED A TOTAL CHECK OF THE
SYSTEM. IF AN ERROR IS DETECTED IN AN I/O DEVICE,
IT WILL PROBABLY BE NECESSARY TO CORRECT THE MALFUNCTION WITH THE
RESPECTIVE DIAGNOSTIC FOR THAT DEVICE.

2.0 REQUIREMENTS

2.1 EQUIPMENT

PDP-11/34 STANDARD COMPUTER
TELETYPE OR EQUIVALENT

2.1.1 OPTIONAL HARDWARE THAT THE PROGRAM WILL EXERCISE

MEMORY UP TO 124 KW OF MEMORY-DOES NOT HAVE TO BE CONTIGUOUS,
BUT BLOCKS OF LESS THAN 4KW WILL NOT BE USED
RF11 DISK
RK11 DISK
TC11 DECTAPE-TRANSPORT ONE(1)
KW11-L LINE CLOCK
KL11 ASR33 OR ASR35 TELEPRINTER
LP11 LINE PRINTER

2.2 STORAGE

THIS PROGRAM USES MEMORY FROM 00000 TO 17760.

3.0 LOADING PROCEDURE

PROCEDURE FOR NORMAL ABSOLUTE TAPES SHOULD BE FOLLOWED.

4.0 STARTING PROCEDURE AND SWITCH SETTINGS

4.1 NORMAL STARTING PROCEDURE

SET DESIRED MEMORY MANAGEMENT OPTION SWITCHES (IN LOC. 174,MMOPT)(SEE 4.2)-ALL
ZERO FOR WORST CASE TESTING.
SET DESIRED SWITCH REGISTER BITS
(USE LOC. 176 FOR SOFTWARE SWITCH REG. IF NECESSARY)
START AT 200.

96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136

4.1 NORMAL STARTING PROCEDURE (CONTINUED)

THE PROGRAM WILL RING THE BELL (UNLESS THE TTY OUTPUT IS SELECTED) AT THE END OF EACH BANK. IF SWITCHES 0,1 AND 2 WERE ALL DOWN WHEN START WAS PRESSED (SELECTING THE USE OF 4K PHYSICAL ADDRESS SPACE AS 32K VIRTUAL ADDRESS SPACE-SEE 5.3.1) AN ASTERISK WILL BE TYPED AT THE END OF A FULL PASS THRU ALL MEMORY (UNLESS THE TTY OUTPUT IS SELECTED).

4.2 MEMORY MANAGEMENT SELECTION SWITCHES (BITS IN LOC. 174,MMOPT)..

THE SWITCHES SET BEFORE STARTUP DETERMINE THE WAY IN WHICH MEMORY IS MAPPED AND EXERCISED:

MMOPT BIT0=1 ---INHIBIT THE MEMORY MANAG. (SR0<0> WILL NOT BE SET AT ALL)
MMOPT BIT1=1 ---INHIBIT USE OF USER MODE.
(ALSO INHIBITS 4K AS 32K)
MMOPT BIT2=1 ---INHIBIT 4K AS 32 K (ALSO INHIBITED IF EITHER MMOPT BIT0 OR MMOPT BIT1 IS SET)-SEE SECTION 5.3.1 FOR EXPLANATION
MMOPT BIT5=1 ---INHIBIT VARIABLE CORE EXPANSION
=0 -CORE EXPAND UNLESS MMOPT BIT0, 1 AND 2 ARE ALL ZERO
(IN WHICH CASE 4K AS 32K IS RUN INSTEAD)

4.3 DEVICE SELECTION SWITCHES

THE DEVICE SELECTION SWITCHES ARE SET IN THE SWITCH REGISTER (USE LOC. 176 FOR SOFTWARE SR IF NECESSARY).
SEE ALSO 5.1.2. EACH SWITCH, IF SET, INHIBITS A SINGLE I/O DEVICE FROM BEING EXERCISED. IF A DEVICE DOES NOT EXIST, THE CORRESPONDING INHIBIT SWITCH DOES NOT HAVE TO BE SET.

SW0=1 ---INHIBIT TTY OUTPUT
SW3=1 ---INHIBIT RK11 DISK
SW4=1 ---INHIBIT LINE CLOCK
SW5=1 ---INHIBIT RF11 DISK
SW6=1 ---INHIBIT TC11 DECTAPE
SW7=1 ---INHIBIT LINE PRINTER (USE SA310 IF LP11 IS SELECTED)

137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187

4.4 RESTART PROCEDURE

USING RESTART ADDRESS 310 THE SWITCH REGISTER SETTINGS GIVEN PREVIOUSLY ARE USED (FOR BOTH MEMORY MANAGEMENT SELECTION AND DEVICE SELECTION). NO HALT OCCURS AFTER START IS PRESSED.

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

5.1.1 BASIC SWITCH SETTINGS=STARTUP

SEE SECTIONS 4.2 AND 4.3 FOR THE BASIC SWITCH SETTINGS USED AT STARTUP. THOSE SWITCHES ARE NOT RECHECKED AFTER THEY ARE INITIALLY STORED.

5.1.2 DYNAMIC SWITCH SETTINGS

@@NOTE: IF NO HARDWARE SWITCH REGISTER IS AVAILABLE, THE PROGRAM WILL AUTOMATICALLY USE THE CONTENTS OF LOC. 176 AS THE SOFTWARE SWITCH REGISTER.
THE USER SHOULD SET THIS LOCATION BEFORE STARTING THE PROGRAM.

THE FOLLOWING SWITCHES ARE RECHECKED PERIODICALLY DURING PROGRAM EXECUTION:

SW15=1 ---HALT ON ERROR
SW14=1 ---SCOPE LOOP
SW13=1 ---INHIBIT PRINT OUT
SW12=1 ---INHIBIT TRACE TRAPPING
SW11=1 ---INHIBIT SUB-PROGRAM ITERATION AND INHIBIT TESTS WHICH USE ALL COMBINATIONS OF NUMBERS
SW10=1 ---INHIBIT PROCESSOR TEST (ONCE SET, PROCESSOR TEST IS PERMANENTLY INHIBITED)

5.2 SUBROUTINE ABSTRACTS

5.2.1 SCOPE

THIS SUBROUTINE CALL IS PLACED BETWEEN EACH SUBTEST. IT RECORDS THE STARTING ADDRESS OF EACH SUBTEST AS IT IS BEING ENTERED. IF A SCOPE LOOP IS REQUESTED, IT WILL JUMP TO THE START OF THE SUBTEST THAT THE SCOPE LOOP IS REQUESTED FOR. IF A SCOPE LOOP IS NOT REQUESTED, THERE WILL BE 256 ITERATIONS ON THAT SUBTEST BEFORE THE NEXT SUBTEST IS ENTERED. SWITCH 11 ON A 1 INHIBITS ITERATION OF SUBTESTS.

188
189
190
191 5.2.2 HLT
192
193 THIS EMT CALLS THE SUBROUTINE PRINT, WHICH PRINTS OUT THE LOCATION
194 COUNTER AT THE TIME OF FAILURE, THE CONTENTS OF THE PROCESSOR STATUS
195 REGISTER, AND THE CONTENTS OF THE CURRENT BANK COUNTER. NOTE THAT
196 THE LOCATION COUNTER WILL BE THE VIRTUAL ADDRESS OF THE HLT PLUS
197 TWO.
198
199 5.2.3 TRAPCATCHER
200
201 THIS IS A SERIES OF INSTRUCTIONS STARTING AT LOCATION 0 DESIGNED
202 TO DETECT AND ISOLATE UNEXPECTED TRAPS AND INTERRUPTS TO THE
203 TRAP AND INTERRUPT VECTOR AREA OF MEMORY.
204
205 EACH VECTOR ENTRANCE ADDRESS IS LOADED WITH THE ADDRESS OF THE
206 NEXT LOCATION. THE NEXT LOCATION IS LOADED WITH A HALT (000000).
207 THUS AN ILLEGAL TRAP OR INTERRUPT WILL CAUSE A HALT AT THE TRAP
208 LOCATION PLUS TWO.
209
210 IF A HALT OCCURS IN THE TRAP OR INTERRUPT AREA EXAMINE KERNEL REGISTER
211 SIX. IT WILL CONTAIN THE CURRENT STACK ADDRESS. THE CONTENTS
212 OF THE CURRENT STACK ADDRESS IS THE VIRTUAL PC AT THE TIME THE TRAP
213 OR INTERRUPT OCCURRED.
214
215 5.2.4 EMTSRV (EMT HANDLER)
216
217 THIS ROUTINE DECODES THE EMT CALLS AND PASSES CONTROL TO THE
218 CORRECT SERVICE ROUTINE. THE ROUTINES HANDLED BY EMT CALLS ARE
219 PRINT (HLT CALL) AND EOBSRV (EOB CALL).
220
221 5.2.6 EOBSRV (END OF BANK SERVICE)
222
223 THE VARIOUS EXECUTION OPTIONS FOR THIS EXERCISER REQUIRE
224 SPECIAL HANDLING WHEN THE END OF THE PROCESSOR TESTS IS REACHED
225 IN A BANK. THIS SERVICE ROUTINE PERFORMS THE VARIOUS MAPPING
226 FUNCTIONS, DEPENDING UPON THE INITIAL SWITCH REGISTER SETTINGS.
227

228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257

5.2.7 BEGINX (CORE EXPANSION SPECIAL HANDLER)

WHEN CORE EXPANSION IS UTILIZED, A NUMBER OF SPECIAL ACTIONS MUST BE TAKEN AT THE BEGINNING OF EACH BANK. THE SCOPE ROUTINE VECTOR IS LOADED TO POINT TO THE NEW BANK, AND IF TC11 AND RF11 CODE AND BUFFER RELOCATION IS ALLOWED.

5.2.9 PFAIL (POWER FAIL)

IN THIS VERSION THE POWER FAIL ROUTINE IS NOT OPERABLE.

5.2.11 TYOUT (TTY OUTPUT)

THIS ROUTINE OUTPUTS A COUNT PATTERN IN THE INTERRUPT MODE TO THE TELEPRINTER.

5.2.12 RFSTART (RF11 DISK)

THIS ROUTINE PERFORMS A WRITE AND A WRITE CHECK OF THE DISK. THE DATA THAT IS WRITTEN ON THE DISK IS A PART OF THE TEST PROGRAM CODE THAT IS NEVER MODIFIED. THIS SEGMENT OF CORE IS WRITTEN IN CONTIGUOUS BLOCKS THRU THE DISK MEMORY. AFTER THE TOTAL DISK(S) HAS BEEN WRITTEN, A WRITE CHECK IS USED TO VERIFY THAT THE DATA HAS BEEN WRITTEN CORRECTLY ON THE DISK. NOTE THAT NO "DATI" ARE USED IN EXERCISING THE DISK (DATA IS NOT TRANSFERRED INTO MEMORY). THERE IS A LOCATION IN THE PROGRAM THAT IF MODIFIED WILL ALLOW EXERCISING UP TO EIGHT DISKS.

258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301

5.2.13 ENDZ (TC11 END ZONE HANDLER)

THIS ROUTINE IS PART OF THE TC11 SERVICE CODE. IT DRIVES THE DECTAPE INTO THE FORWARD OR REVERSE END ZONE, THEN REVERSES IT. IT ALSO DOES THE NECESSARY SETUP TO BEGIN READING OR WRITING THE TAPE.

5.2.14 REGEN (TC11 WRITE BUFFER REGENERATE ROUTINE)

THE TC11 CODE WRITES THE ENTIRE DECTAPE GOING FORWARD, THEN READS IT IN REVERSE. THE BUFFER IS REGENERATED BEFORE WRITING THE TAPE, AND IS CLEARED OUT ONCE THE ENTIRE TAPE HAS BEEN WRITTEN. THIS ROUTINE REGENERATES THE WRITE BUFFER.

5.2.15 RBN (TC11 READ BLOCK NUMBER SERVICE ROUTINE)

AT THE END OF EACH "BLOCK NUMBER FOUND" INTERRUPT, THIS ROUTINE IS ENTERED (UNLESS END ZONE IS BEING SEARCHED FOR). IT CHECKS FOR THE CORRECT SEQUENCE OF BLOCK NUMBERS, THEN SETS UP THE TC11 TO WRITE A BLOCK IF THE TAPE IS TRAVELLING FORWARD. IF IT IS GOING IN REVERSE, THE ROUTINE CHECKS TO SEE IF DATA IS STILL BEING CHECKED FROM A PREVIOUS READ. IF IT'S NOT, THE ROUTINE SETS UP TO READ A BLOCK. IF DATA IS STILL BEING CHECKED FROM BEFORE, IT SIMPLY DOES ANOTHER READ BLOCK NUMBER.

5.2.16 NXTBLK (TC11 READ BLOCK AND WRITE BLOCK SERVICE ROUTINE)

WHEN A READ BLOCK OR A WRITE BLOCK OPERATION IS COMPLETED, THIS ROUTINE IS ENTERED. IT CHECKS THE ERROR BIT, THEN SETS UP A CALL TO CHECK DATA IF DATA WAS JUST READ IN. THE ROUTINE ALSO SETS UP A READ BLOCK NUMBER OPERATION.

5.2.17 TCCK (TC11 CHECK DATA ROUTINE)

WHEN A READ BLOCK OPERATION HAS BEEN COMPLETED, THIS ROUTINE IS CALLED VIA A PRIORITY INTERRUPT REQUEST AT LEVEL 3. THE ENTIRE BUFFER IS CHECKED, AND THE CONTENTS OF THE BUFFER IS ALTERED AS THE CHECK PROGRESSES. THUS, IF A READ BLOCK OPERATION DOES NOT ACTUALLY READ IN ANY DATA, THE DATA CHECK ROUTINE WILL FIND BAD DATA INSTEAD OF SEEING GOOD DATA FROM AN EARLIER READ.

302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341

5.2.18 LCLK (LINE CLOCK)

THIS TEST OF THE LINE CLOCK IS IN THE INTERRUPT MODE. IF OPERATING CORRECTLY THE SYSTEM I/O WILL RUN AT FULL SPEED FOR 55 SECONDS, AND THEN ALL I/O AT LEVEL FOUR OR LESS (AND THE PROCESSOR TESTS) WILL STALL FOR 5 SECONDS, TIMES GIVEN ARE BASED ON 60 CYCLES AS THE LINE FREQUENCY.

5.2.19 LP1 (LINE PRINTER)

THIS ROUTINE OUTPUTS TO THE LINE PRINTER IN THE FLAG MODE WHILE FILLING THE BUFFER, AND IN THE INTERRUPT MODE WHILE THE BUFFER IS BEING PRINTED.

5.2.20 RKSTART (RK-11 DISK)

THIS ROUTINE PERFORMS A WRITE AND WRITE CHECK OF THE DISK. THE DATA THAT IS WRITTEN ON THE DISK IS PART OF THE TEST PROGRAM CODE THAT IS NEVER MODIFIED. THIS SEGMENT OF CORE IS WRITTEN IN CONTIGUOUS BLOCKS THRU THE DISK MEMORY. AFTER THE TOTAL DISK HAS BEEN WRITTEN, A WRITE CHECK IS USED TO VERIFY THAT DATA HAS BEEN WRITTEN CORRECTLY ON THE DISK.

5.2.22 CORE EXPANSION (DET1)

THIS ROUTINE IS CONTROLLED BY SWITCH 5. IF CALLED, THE PROCESSOR MAINLINE CODE WILL EXPAND TO THE MAXIMUM MEMORY THAT IS AVAILABLE (UP TO 28K). THE ROUTINE DETERMINES THE MAXIMUM MEMORY SIZE BY DOING A "DATO" TO A LOCATION IN EACH BANK. IF THE BANK DOES NOT EXIST, A TIMEOUT WILL OCCUR. AN IMAGE OF BANK 0 IS THEN TRANSFERRED TO EACH EXISTING BANK. THE CODE IN EACH BANK EXCEPT THE LAST IS MODIFIED TO CHANGE THE END OF BANK CALL TO A JUMP TO BEGINX (CORE EXPANSION SPECIAL HANDLER) IN THE NEXT BANK.

THE LISTING SHOWS ONLY THE CODE FOR BANK ZERO. WHEN AN ERROR OCCURS THAT IS NOT IN BANK ZERO, IGNORE THE BANK BITS OF THE PRINT OUT AND USE THE LISTING FOR BANK ZERO.

342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374

5.3 PROGRAM AND/OR OPERATOR ACTION

5.3.1 PROCESSOR TEST EXECUTION - 4K AS 32K

IF MMOPT BITS 0, 1, AND 2 ARE ALL ZERO (=0) AT STARTUP, THE PROCESSOR TEST WILL BE EXECUTED TREATING EACH 4K BANK AS 32K OF VIRTUAL ADDRESS SPACE. THE FOLLOWING DETAILS THIS MODE OF OPERATION.

USER PAGE 0 IS FIRST MAPPED RW, BANK 0, AND ALL OTHER USER PAGES ARE MAPPED NON-RESIDENT. THE PROCESSOR TESTS ARE EXECUTED IN USER THRU USER PAGE 0. WHEN DONE, USER PAGE 0 IS CHANGED TO NON-RESIDENT, AND USER PAGE 1 IS MAPPED RW, BANK 0. THE PC IS CHANGED TO ADDRESS THE START OF THE PROCESSOR TESTS THRU PAGE 1, AND ANOTHER PASS THRU THE PROCESSOR TESTS IS EXECUTED. AT THE END OF THIS PASS, USER PAGE 2 IS MAPPED RW, BANK 0, AND USER PAGE 1 IS MADE NON-RESIDENT. THE PC IS AGAIN CHANGED. THIS TIME TO ACCESS USER PAGE 2, AND THE PROCESSOR TESTS ARE EXECUTED THRU USER PAGE 2. THIS CYCLE IS REPEATED FOR THE REMAINING USER PAGES, MAPPING EACH IN TURN TO BANK 0 AND CHANGING THE PC TO EXECUTE THRU THE ONE CURRENTLY MAPPED. WHEN THE PASS USING USER PAGE 7 IS COMPLETED, A SEARCH IS MADE FOR THE NEXT 4K BANK OF MEMORY. WHEN A BANK IS FOUND, THE PROGRAM IS COPIED INTO THAT BANK FROM BANK 0. USER PAGE 0 IS MAPPED TO THE NEW BANK, AND THE PC IS CHANGED TO EXECUTE THRU USER PAGE 0. THE PREVIOUS CYCLE IS REPEATED, BUT THIS TIME EACH USER PAGE IS MAPPED IN TURN TO THE NEW BANK. ONCE EXECUTION THRU USER PAGE 7 IS COMPLETED, A SEARCH IS MADE FOR THE NEXT BANK. THE PREVIOUS BANK IS CLEARED (EXCEPT FOR THE LOADER), AND THE PROGRAM IS COPIED FROM BANK 0 INTO THE CURRENT BANK. THE CYCLE REPEATS UNTIL THE EXTERNAL BANK IS REACHED, AT WHICH POINT USER 0 IS MAPPED BACK TO BANK 0 AND THE PROCESS STARTS AGAIN.

375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405

5.3.2 PROCESSOR TEST EXECUTION - CORE EXPANSION

IF MMOPT BIT, 1, OR 2 ARE ONE AND BIT 5 IS ZERO AT STARTUP, THE PROCESSOR TESTS WILL BE CORE EXPANDED THRU ALL AVAILABLE MEMORY UP TP 28K. THR ROUTINE DET1 DOES THIS CORE EXPANSION, COPYING BANK 0 INTO EACH OF THE OTHER BANKS. THE EMT CALL AT THE END OF EACH BANK (EOB) WHICH CALLS THE END OF BANK SERVICE ROUTINE IS CHANGED TO A JUMP TO BEGINX IN THE NEXT BANK. THE EOB CALL IN THE LAST BANK IS LEFT ALONE. IF MMOPT BITS 0 AND 1 WERE BOTH ZERO AT STARTUP, USER PAGES 0 THRU 6 ARE MAPPED SO THAT THE PHYSICAL AND VIRTUAL ADDRESSES CORRESPOND, AND THE PROCESSOR TESTS ARE THEN RUN IN USER. IF BIT0 WAS ZERO BUT BIT1 WAS SET, KERNEL PAGES 0-6 ARE MAPPED SO THAT THE PHYSICAL AND VIRTUAL ADDRESSES ARE THE SAME, AND THE PROCESSOR TESTS ARE THEN RUN IN KERNEL MODE. IF BIT0 WAS SET, ORDINARY CORE EXPANSION IS RUN WITH NO SPECIAL MAPPING REQUIRED (MEMORY MANAG. IS TURNED OFF).

5.3.3 PROCESSOR TEST EXECUTION - BANK 0 ONLY

IF MMOPT BITS 0, 1 OR 2 ARE ONE AND BITS IS ONE AT STARTUP, ONLY BANK 0 IS UTILIZED, IN THIS CASE, IF BIT0 AND BIT1 WERE ZERO THE PROCESSOR TESTS ARE EXECUTED IN USER, WITH USER PAGE 0 MAPPED TO BANK 0. IF BIT0 WAS ZERO AND BIT1 WAS ONE, THE PROCESSOR TESTS ARE EXECUTED IN KERNEL, WITH KERNEL PAGE 0 MAPPED TO BANK 0. IF BIT0 WAS ONE, THE MEMORY MANAG. IS TURNED OFF AND THE PROCESSOR TESTS ARE EXECUTED IN KERNEL MODE OR USER MODE (DEPENDING ON BIT1) IN BANK 0 ONLY.

406
407
408
409
410
411
412
413 6.0 ERRORS
414
415 6.1 ERROR PRINTOUT
416
417 PRINTOUTS ARE IN AN EXTENDED VERSION OF THE STANDARD FORMAT,
418 USING THREE WORDS. THE FIRST WORD IS THE OCTAL VALUE OF THE
419 VIRTUAL PC+2 OF THE DETECTED ERROR, THE SECOND WORD IS THE
420 CONTENTS OF THE PROCESSOR STATUS REGISTER WHEN THE ERROR WAS DETECTED.
421 THE THIRD IS THE TOP 12 BITS OF THE 18-BIT ADDRESS OF
422 THE BANK BEING CURRENTLY USED FOR EXECUTION OF THE PROCESSOR TEST.
423 THE FOURTH IS RETURN WHICH IS THE RETURN ADDRESS IN THE
424 CURRENT BANK OF MEMORY.
425 TO GET THE STARTING ADDRESS OF THE CURRENT BANK SIMPLY APPEND
426 TWO ZEROS TO THE END OF THE OCTAL VALUE PRINTED OUT (I.E.
427 007400 INDICATES THE BANK BEGINNING AT PHYSICAL ADDRESS
428 740000).
429
430 6.2 ERROR RECOVERY
431
432 IN GENERAL, TEST FAILURES WILL PRINTOUT AN ERROR MESSAGE AND
433 CONTINUE. IF THE "HALT ON ERROR" SWITCH IS SET, HITTING CONTINUE
434 WILL RECOVER. IF THE PROGRAM HANGS UP IN A LOOP, THE ERROR IS
435 LIKELY TO BE A SIGNAL WHICH WAS NEVER RECEIVED. IF A HALT
436 OCCURS IN THE TRAP AND VECTOR AREA THE PROGRAM MUST BE RESTARTED.
437 IF THE PROGRAM HALTS IN THE MAIN FLOW, CONSULT THE LISTING IF NO
438 MESSAGE IS TYPED OUT. FOR TTY READER AND MSR, TAPE MUST BE
439 REPOSITIONED TO LEADER BEFORE RESTARTING THE TEST.
440

441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472

6.3 FINDING WHICH PROCESSOR TEST WAS BEING EXECUTED WHEN AN ERROR OCCURRED

SOME ERRORS ARE DEPENDENT ON THE PROCESSOR TEST BEING RUN (SUCH AS LATENCY ERRORS WHICH ONLY SHOW UP IN WORST-CASE PROCESSOR TIMING). THE SCOPE ROUTINE CONTAINS A LOCATION CALLED "RETURN" WHICH STORES THE STARTING ADDRESS OF THE PROCESSOR TEST CURRENTLY BEING EXECUTED. NOTE THAT THE SCOPE ROUTINE IS EXECUTED IN USER MODE IF SW1 IS DOWN AT STARTUP, AND IS THEREFORE RELOCATED WITH THE PROCESSOR TESTS. THUS, TO DETERMINE WHICH PROCESSOR TEST WAS BEING EXECUTED WHEN A FAILURE OCCURRED, FIRST CHECK THE CONTENTS OF CURBNK IN BANK 0. THIS LOCATION CONTAINS THE ADDRESS OF THE CURRENT PHYSICAL BANK, SHIFTED RIGHT 6 PLACES. BY APPENDING 2 ZEROES TO IT, YOU HAVE THE 18-BIT ADDRESS OF THE CURRENT BANK OF MEMORY. ADD TO THIS THE ADDRESS OF RETURN IN BANK 0 AND YOU HAVE THE ADDRESS OF RETURN IN THE CURRENT BANK OF MEMORY. THE CONTENTS OF RETURN IN THE CURRENT BANK OF MEMORY IS THE VIRTUAL ADDRESS OF THE START OF THE CURRENT PROCESSOR TEST.

7.0 RESTRICTIONS

PROGRAM MUST BE LOADED INTO LOWER 4K OF MEMORY.

THE INHIBIT SWITCHES MUST ONLY BE SET FOR ALL DEVICES THAT ARE PART OF THE SYSTEM BUT WHICH YOU DO NOT WISH TO RUN,

IF THE LINE PRINTER IS USED, STARTING ADDRESS 310 MUST BE USED.

473
474
475
476 8.0 MISCELLANEOUS
477
478 8.1 EXECUTION TIME
479
480 EXECUTION TIME VARIES WITH THE AMOUNT OF MEMORY, THE TYPES OF
481 MEMORY, THE DEVICES RUN, AND THE OPTIONAL MODES OF EXECUTION USED.
482
483 A PASS RUN WITH CORE EXPANSION AND 4K AS 32K RELOCATION BOTH
484 INHIBITED TAKES LESS THAN 10 SECONDS (RUNNING NO I/O).
485
486 A PASS RUN WITH 4K AS 32K, IN CORE MEMORY WITH NO I/O, TAKES
487 ABOUT 5MINUTES PER 4K BANK. (AN ASTERISK IS PRINTED AT THE
488 END OF A FULL PASS, AND THE BELL IS RUNG AT THE END OF EACH
489 4K BANK).
490
491
492 8.2 STACK POINTERS
493
494 THE KERNEL STACK POINTER IS INITIALIZED TO 17760.
495
496 THE USER STACK POINTER IS INITIALIZED TO 400. IT IS RELOCATED
497 THRU ALL USER PAGES AND TO EVERY 4K BANK IF THE 4K AS 32K
498 MODE OF EXECUTION IS RUN.
499

500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532

9.0 PROGRAM DESCRIPTION

THIS MEMORY MANAGEMENT EXERCISER IS DESIGNED TO RUN BACKGROUND PROCESSOR TESTS AND FOREGROUND CONCURRENT I/O WITH MEMORY MANAGEMENT UTILIZED IN ANY OF SEVERAL DIFFERENT MODES. THE VARIOUS MODES AVAILABLE FOR UTILIZING MEMORY MANAGEMENT ARE INCLUDED TO AID IN FAULT ISOLATION BY PROVIDING A SERIES OF STEPS FROM SIMPLE TO COMPLEX. MEMORY MANAGEMENT CAN BE LEFT TURNED OFF AND THE PROCESSOR TESTS CAN STILL BE RUN IN 4K ONLY OR CORE EXPANDED UP TO 28K. WITH MEMORY MANAGEMENT ON, THE PROGRAM CAN BE RUN USING ONLY 4K, WITH EVERYTHING MAPPED IN KERNEL SPACE OR WITH USER AND KERNEL BOTH USED. AT THE NEXT LEVEL OF COMPLEXITY, CORE EXPANSION CAN BE RUN WITH MEMORY MANAGEMENT ON, USING KERNEL ONLY OR USING BOTH MODES AS DESIRED. FINALLY, ALL AVAILABLE MEMORY (IN 4K PIECES) CAN BE UTILIZED BY RUNNING 4K AS 32K. THERE IS NO MONITOR IN THE CONVENTIONAL SENSE. EACH DEVICE THAT IS TO BE EXERCISED HAS ITS OWN STAND ALONE ROUTINE THAT OPERATES IN THE INTERRUPT MODE. THESE ROUTINES NEED NO SUPERVISION OR MONITORING AFTER THEY ARE INITIATED. THERE IS A PRIMER AREA THAT CHECKS THE SWITCH REGISTER TO SEE WHAT DEVICES ARE TO BE INITIATED. IT SETS THE INTERRUPT ENABLE BIT IN THE DEVICE STATUS REGISTER, INITIALIZES THE DATA PATTERN, AND INITIATES AN OPERATION TO RAISE DATA FLAGS ON DEVICES THAT CAN NOT INITIATE THEM THEMSELVES. THE PRIMER CODE THEN ENTERS THE MEMORY MANAG. SETUP CODE. THE RP11 AND TC11 PRIMER CODE IS IN WITH THE MEMORY MANAG. SETUP CODE SINCE THEY REQUIRE CERTAIN PARTS OF THE MEMORY MANAG. CODE TO BE RUN FIRST. AFTER MEMORY MANAGEMENT IS TURNED ON, EXECUTION OF THE BACKGROUND PROCESSOR TESTS BEGINS, AND THE I/O DEVICES ARE SERVICED WHEN THEY INTERRUPT.

```

533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586

```

*
)THIS PROGRAM IS A MODIFICATION OF THE 11/40 DIAGNOSTIC, DBKTB.
)THIS TEST HAS BEEN MODIFIED TO PROVIDE SOFTWARE SWITCH CAPABILITY
)AND TO ACCOUNT FOR ANY 11/34 - 11/40 DIFFERENCES.
)THIS PROGRAM IS INTENDED FOR USE ON ONLY 11/34 PROCESSORS.

)COPYRIGHT 1975, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS. 01754
)PDP11/34 SYSTEM EXERCISER, WITH MEMORY MANAG. --- TTY,PC11,KW11-L
)LP11,RF11,TC11
)TEST SIMULTANEOUS RUNNING OF I/O, WITH PROCESSOR INSTRUCTION TEST AND
)WITH TRACE BIT ENABLED TO BE CONSIDERED MAINLINE CODE

)I/O RUNS IN KERNEL MODE
)CPU TESTS RUN IN USER MODE UNLESS INHIBITED BY SR SETTINGS
)MEMORY MANAG. IS UTILIZED

)R6 IS THE STACK POINTER
)R6) IS THE PC+2 OF LOCATION WHERE THE TRAP ORIGINATED
)FOR NORMAL OPERATION RUN WITH ALL SWITCHES DOWN
)SA = 200
)RESTART = 310 (OPTION SETTINGS PREVIOUSLY MADE ARE USED)

)AT STARTUP, MMOPT(LOC. 174) SETTINGS ARE:
)MMOPT BIT 0=1 --- RUN WITHOUT MEMORY MANAG.
)MMOPT BIT 1=1 --- RUN ALL IN KERNEL MODE (INHIBITS RUNNING 4K AS 32K)
)MMOPT BIT 2=1 --- INHIBIT RUNNING 28K USER MEMORY MANAG. FROM EVERY 4K
)BANK (ALLOW NORMAL CORE EXPANSION)
)MMOPT BIT 5=1 ---INHIBIT VARIABLE CORE EXPANSION

)SR (USE LOC. 176 IF NECESSARY) BIT SETTINGS ARE:
)SR 15=1 ---HALT ON ERROR
)SR 14=1 ---SCOPE LOOP
)SR 13=1 ---INHIBIT PRINT OUT
)SR 12=1 ---INHIBIT TRACE TRAPPING
)SR 11=1 ---INHIBIT SUB-PROGRAM ITERATION AND INHIBIT TESTS WHICH
)USE ALL COMBINATIONS OF NUMBERS
)SR 10=1 ---INHIBIT PROCESSOR TEST

)SPECIAL DELETE SWITCHES-SET RESPECTIVE SWITCH TO A 1 TO INHIBIT
)INITIATION OF DEVICE
)SW 0=1 INHIBIT TTY OUTPUT
)SW 3=1 INHIBIT RK11 DISK
)SW 4=1 INHIBIT LINE CLOCK
)SW 5=1 INHIBIT RF11 DISK
)SW 6=1 INHIBIT TC11 DECTAPE
)SW 7=1 INHIBIT LINE PRINTER

)DEFINITIONS
)NDR=240)SYSTEM NULL OPERATION
)SCOPE=TRAP)TRAP USED SCOPE LOOP AND ITERATION
)TCSR=TTCSR
)TDBR=TTDBR
)PSR=17776

```

582      000240
583      104400
584      000410
585      000412
586      177776

```

```

587      104006      HLT=104006      )ERROR PRINTOUT CALL
588      104010      EOB=104010      )END OF BANK CALL
589      000000      R0=X0
590      000001      R1=X1
591      000002      R2=X2
592      000003      R3=X3
593      000004      R4=X4
594      000005      R5=X5
595      000006      SP=X6
596      000006      R6=SP
597      000007      PC=X7
598
599
600      )SBTTL MAIN
601      )TRAP CATCHER
602      000000      000002      )+2
603      000002      000000      )HALT )TRAP ENTRANCE
604      000004      000006      )+2 )TRAPPED TO PREVIOUS LOCATION
605      000006      000008      )HALT )TRAP ENTRANCE
606      000010      000012      )+2 )TRAPPED TO PREVIOUS LOCATION
607      000012      000014      )HALT )TRAP ENTRANCE
608      000014      000016      )+2 )TRAPPED TO PREVIOUS LOCATION
609      000016      000018      )HALT )TRAP ENTRANCE
610      000020      000022      )+2 )TRAPPED TO PREVIOUS LOCATION
611      000022      000024      )HALT )TRAP ENTRANCE
612      000024      000026      )+2 )TRAPPED TO PREVIOUS LOCATION
613      000026      000028      )HALT )TRAP ENTRANCE
614      000030      000032      )+2 )TRAPPED TO PREVIOUS LOCATION
615      000032      000034      )HALT )TRAP ENTRANCE
616      000034      000036      )+2 )TRAPPED TO PREVIOUS LOCATION
617      000036      000038      )HALT )TRAP ENTRANCE
618      000040      000042      )+2 )TRAPPED TO PREVIOUS LOCATION
619      000042      000044      )HALT )TRAP ENTRANCE
620      000044      000046      )+2 )TRAPPED TO PREVIOUS LOCATION
621      000046      000048      )HALT )TRAP ENTRANCE
622      000050      000052      )+2 )TRAPPED TO PREVIOUS LOCATION
623      000052      000054      )HALT )TRAP ENTRANCE
624      000054      000056      )+2 )TRAPPED TO PREVIOUS LOCATION
625      000056      000058      )HALT )TRAP ENTRANCE
626      000060      000062      )+2 )TRAPPED TO PREVIOUS LOCATION
627      000062      000064      )HALT )TRAP ENTRANCE
628      000064      000066      )+2 )TRAPPED TO PREVIOUS LOCATION
629      000066      000068      )HALT )TRAP ENTRANCE
630      000070      000072      )+2 )TRAPPED TO PREVIOUS LOCATION
631      000072      000074      )HALT )TRAP ENTRANCE
632      000074      000076      )+2 )TRAPPED TO PREVIOUS LOCATION
633      000076      000078      )HALT )TRAP ENTRANCE
634      000100      000102      )+2 )TRAPPED TO PREVIOUS LOCATION
635      000102      000104      )HALT )TRAP ENTRANCE
636      000104      000106      )+2 )TRAPPED TO PREVIOUS LOCATION
637      000106      000108      )HALT )TRAP ENTRANCE
638      000110      000112      )+2 )TRAPPED TO PREVIOUS LOCATION
639      000112      000114      )HALT )TRAP ENTRANCE
640      000114      000116      )+2 )TRAP ENTRANCE

```

641	000116	000000	HALT	TRAPPED TO PREVIOUS LOCATION
642	000120	000122	.+2	TRAP ENTRANCE
643	000122	000000	HALT	TRAPPED TO PREVIOUS LOCATION
644	000124	000126	.+2	TRAP ENTRANCE
645	000126	000000	HALT	TRAPPED TO PREVIOUS LOCATION
646	000130	000132	.+2	TRAP ENTRANCE
647	000132	000000	HALT	TRAPPED TO PREVIOUS LOCATION
648	000134	000136	.+2	TRAP ENTRANCE
649	000136	000000	HALT	TRAPPED TO PREVIOUS LOCATION
650	000140	000142	.+2	TRAP ENTRANCE
651	000142	000000	HALT	TRAPPED TO PREVIOUS LOCATION
652	000144	000146	.+2	TRAP ENTRANCE
653	000146	000000	HALT	TRAPPED TO PREVIOUS LOCATION
654	000150	000152	.+2	TRAP ENTRANCE
655	000152	000000	HALT	TRAPPED TO PREVIOUS LOCATION
656	000154	000156	.+2	TRAP ENTRANCE
657	000156	000000	HALT	TRAPPED TO PREVIOUS LOCATION
658	000160	000162	.+2	TRAP ENTRANCE
659	000162	000000	HALT	TRAPPED TO PREVIOUS LOCATION
660	000164	000166	.+2	TRAP ENTRANCE
661	000166	000000	HALT	TRAPPED TO PREVIOUS LOCATION
662	000170	000172	.+2	TRAP ENTRANCE
663	000172	000000	HALT	TRAPPED TO PREVIOUS LOCATION
664	000174	000176	.+2	TRAP ENTRANCE
665	000176	000000	HALT	TRAPPED TO PREVIOUS LOCATION
666	000200	000202	.+2	TRAP ENTRANCE
667	000202	000000	HALT	TRAPPED TO PREVIOUS LOCATION
668	000204	000206	.+2	TRAP ENTRANCE
669	000206	000000	HALT	TRAPPED TO PREVIOUS LOCATION
670	000210	000212	.+2	TRAP ENTRANCE
671	000212	000000	HALT	TRAPPED TO PREVIOUS LOCATION
672	000214	000216	.+2	TRAP ENTRANCE
673	000216	000000	HALT	TRAPPED TO PREVIOUS LOCATION
674	000220	000222	.+2	TRAP ENTRANCE
675	000222	000000	HALT	TRAPPED TO PREVIOUS LOCATION
676	000224	000226	.+2	TRAP ENTRANCE
677	000226	000000	HALT	TRAPPED TO PREVIOUS LOCATION
678	000230	000232	.+2	TRAP ENTRANCE
679	000232	000000	HALT	TRAPPED TO PREVIOUS LOCATION
680	000234	000236	.+2	TRAP ENTRANCE
681	000236	000000	HALT	TRAPPED TO PREVIOUS LOCATION
682	000240	000242	.+2	TRAP ENTRANCE
683	000242	000000	HALT	TRAPPED TO PREVIOUS LOCATION
684	000244	000246	.+2	TRAP ENTRANCE
685	000246	000000	HALT	TRAPPED TO PREVIOUS LOCATION
686	000250	000252	.+2	TRAP ENTRANCE
687	000252	000000	HALT	TRAPPED TO PREVIOUS LOCATION
688	000254	000256	.+2	TRAP ENTRANCE
689	000256	000000	HALT	TRAPPED TO PREVIOUS LOCATION
690	000260	000262	.+2	TRAP ENTRANCE
691	000262	000000	HALT	TRAPPED TO PREVIOUS LOCATION
692	000264	000266	.+2	TRAP ENTRANCE
693	000266	000000	HALT	TRAPPED TO PREVIOUS LOCATION
694	000270	000272	.+2	TRAP ENTRANCE

695	000272	000000	HALT	TRAPPED TO PREVIOUS LOCATION
696	000274	000276	.+2	TRAP ENTRANCE
697	000276	000000	HALT	TRAPPED TO PREVIOUS LOCATION
698	000300	000302	.+2	TRAP ENTRANCE
699	000302	000000	HALT	TRAPPED TO PREVIOUS LOCATION
700	000304	000306	.+2	TRAP ENTRANCE
701	000306	000000	HALT	TRAPPED TO PREVIOUS LOCATION
702	000310	000312	.+2	TRAP ENTRANCE
703	000312	000000	HALT	TRAPPED TO PREVIOUS LOCATION
704	000314	000316	.+2	TRAP ENTRANCE
705	000316	000000	HALT	TRAPPED TO PREVIOUS LOCATION
706	000320	000322	.+2	TRAP ENTRANCE
707	000322	000000	HALT	TRAPPED TO PREVIOUS LOCATION
708	000324	000326	.+2	TRAP ENTRANCE
709	000326	000000	HALT	TRAPPED TO PREVIOUS LOCATION
710	000330	000332	.+2	TRAP ENTRANCE
711	000332	000000	HALT	TRAPPED TO PREVIOUS LOCATION
712	000334	000336	.+2	TRAP ENTRANCE
713	000336	000000	HALT	TRAPPED TO PREVIOUS LOCATION
714	000340	000342	.+2	TRAP ENTRANCE
715	000342	000000	HALT	TRAPPED TO PREVIOUS LOCATION
716	000344	000346	.+2	TRAP ENTRANCE
717	000346	000000	HALT	TRAPPED TO PREVIOUS LOCATION
718	000350	000352	.+2	TRAP ENTRANCE
719	000352	000000	HALT	TRAPPED TO PREVIOUS LOCATION
720	000354	000356	.+2	TRAP ENTRANCE
721	000356	000000	HALT	TRAPPED TO PREVIOUS LOCATION
722	000360	000362	.+2	TRAP ENTRANCE
723	000362	000000	HALT	TRAPPED TO PREVIOUS LOCATION
724	000364	000366	.+2	TRAP ENTRANCE
725	000366	000000	HALT	TRAPPED TO PREVIOUS LOCATION
726	000370	000372	.+2	TRAP ENTRANCE
727	000372	000000	HALT	TRAPPED TO PREVIOUS LOCATION
728	000374	000376	.+2	TRAP ENTRANCE
729	000376	000000	HALT	TRAPPED TO PREVIOUS LOCATION
730				
731			LOAD VECTOR AREA	
732		000024	.#24	
733	000024	016634	PFail	POWER FAIL TRAP
734	000026	000340	340	
735		000030	.#30	
736	000030	015050	EMTSRV	EMT CALLS
737	000032	000340	340	HIGHEST PRIORITY
738		000034	.#34	
739	000034	014642	SCOPEC	USER TRAP
740	000036	000000	0	
741		000046	.#46	
742	000046	015662	LOGICAL	
743		000052	.#52	
744	000052	040000	40000	
745		000174	.#174	
746	000174	000000	MMOPT: 0	MEMORY MANAG OPTION SEL.
747	000176	000000	SWREG: 0	SOFTWARE SWITCH REG
748		000176	SREG2=SWREG	


```

749
750
751
752          000200          JLOAD STARTING AREA
753 000200 000137 000640          .#200
754          000300          JMP      ##START
755 000300 000137 000640          .#300
756          000310          JMP      ##START
757 000310 000137 000616          .#310
758          JMP      ##RSTRT
759
760          000400          JDATA AREA
761 000400 000000          USUFF:  0          ;BUFFER FOR USER SP
762          000406          .#.#+4          ;FOR STACK OVERRUN
763 000406 177560          TRCSR: 177560       ;TTY READER STATUS REGISTER
764 000410 177564          TTCSR: 177564       ;TTY PUNCH STATUS REGISTER
765 000412 177566          TTDBR: 177566
766 000414 000064          TTPVC: 64
767 000416 000066          TTPST: 66
768 000420 000000          TTSAV: 0
769 000422 000100          KWLVC: 100
770 000424 000102          KWLST: 102
771 000426 177546          LKCSR: 177546
772 000430 177514          LPCSR: 177514
773 000432 177516          LPDBR: 177516
774 000434 000200          LPVC: 200
775 000436 000202          LPST: 202
776 000440 177470          RFDAE: 177470       ;DISK ADDRESS AND ERROR
777 000442 177466          RFDA: 177466        ;DISK ADDRESS REGISTER
778 000444 177462          RFNC: 177462        ;WORD COUNT REGISTER
779 000446 177464          RFCA: 177464        ;CURRENT ADDRESS REGISTER
780 000450 177460          RFCSR: 177460       ;STATUS REGISTER
781 000452 177461          RFCSRH: 177461     ;HIGH BYTE ADDRESS OR CSR
782 000454 000204          RFVC: 204
783 000456 000206          RFST: 206
784 000460 177413          RKDAH: 177413       ;HIGH BYTE DISK ADR
785 000462 177412          RKDAE: 177412       ;DISK ADDRESS REGISTER
786 000464 177406          RKWC: 177406        ;WORD COUNT REGISTER
787 000466 177410          RKBAR: 177410       ;CURRENT ADDRESS REGISTER
788 000470 177404          RKCSR: 177404       ;STATUS REGISTER
789 000472 177405          RKCSRH: 177405     ;HIGH BYTE OF CSR
790 000474 000220          RKVC: 220
791 000476 000222          RKST: 222
792 000500 177572          SR0: 177572         ;MEMORY MANAG. REGISTERS
793 000502 177600          UPDR0: 177600
794 000504 177602          UPDR1: 177602
795 000506 177616          UPDR7: 177616
796 000510 177640          UPAR0: 177640
797 000512 177642          UPAR1: 177642
798 000514 177656          UPAR7: 177656
799 000516 172300          KPDR0: 172300
800 000520 172302          KPDR1: 172302
801 000522 172304          KPDR2: 172304
802 000524 172316          KPDR7: 172316

```

```

803 000526 172340          KPAR0: 172340
804 000530 172342          KPAR1: 172342
805 000532 172344          KPAR2: 172344
806 000534 172356          KPAR7: 172356
807
808 000536 177600          IPDRTAB:177600
809 000540 177640          177640
810 000542 172300          172300
811 000544 172340          IPREND:172340
812 000546 177570          SR: 177570          ;SWITCH REGISTER POINTER
813 000550 177571          SRH: 177571        ;HIGH BYTE SWITCH REG POINTER
814 000552 177342          TCCM: 177342       ;CONTROL AND FUNCTION
815 000554 177340          TCST: 177340       ;GENERAL STATUS
816 000556 177350          TCDT: 177350       ;DATA
817 000560 177344          TCWC: 177344       ;WORD COUNT
818 000562 177346          TCBA: 177346       ;BUS ADDRESS
819 000564 000214          TCIV: 214          ;DECTAPE INTERRUPT VECTOR
820 000566 000216          TCSTA: 216
821 000570 000000          CURBNK: 0          ;$AP TO POINT TO CURRENT BANK
822 000572 000000          OLDBNK: 0
823 000574 000000          CURPAR: 0          ;ADDRESS OF CURRENT ISAR
824 000576 000000          CURPDR: 0
825 000600 000000          BNKSTR: 0          ; PC TO POINT TO BEGIN THRU CURRENT SEGMENT
826 000602 000000          TRPB: 0
827
828          ;THE NEXT TWO WORDS ARE THE MEMORY MAP. THE FIRST WORD REPRESENTS
829          ;0-64K WITH ONE BIT REPRESENTING A 4K CONTIGUOUS BLOCK, IF THE
830          ;BIT=1 THAT 4K BLOCK IS PRESENT. THE LSB REPRESENTS 0-4K, THE NEXT
831          ;SIGNIFICANT BIT REPRESENTS 4-8K ANS SO ON.
832 000604 177777          MEM0: 177777       ;0-64K
833 000606 077777          MEM1: 077777       ;64-124K
834 000610 000001          COREPT: 1
835 000612 000604          MEMUT: MEM0
836 000614 000000          TBank: 0
837
838          ;RESTART ADD USING INITIAL SR SETTINGS
839 000622 012706 017760 000024 RSTRT: MOV      #KSTACK,R6
840 000630 117737 177714 000177      MOV      #PFALL,#24
841 000636 000452          BR       #SRH,#$REG2+1
842          BR       START1
843
844          ;START UP FOR MINI MONITOR
845 000640 012706 017760 000024 START: MOV      #KSTACK,R6          ;SET UP STACK
846 000644 012737 000137 000200      MOV      #137,#200          ;RESTORE 200 IF START AT 300
847 000652 012737 000640 000202      MOV      #START,#202
848 000660 013746 000004          MOV      #4,-(SP)          ;SAVE ERROR VECTOR
849 000664 013746 000006          MOV      #6,-(SP)
850 000670 012747 000704 177106      MOV      #15,4          ;SET UP TIME OUT VECTOR
851 000676 005777 17644          TST      #SR          ;TRY TO REFERENCE HARDWARE SWR
852 000702 000404          BR       2$          ;BRANCH IF NO TIMEOUT TRAP OCCURS
853 000704 012747 000176 177634 1$1 MOV      #SWREG,SR          ;POINT TO SOFTWARE SWR
854 000712 022626          CNP      (SP)+,(SP)+          ;RESTORE STACK
855 000714 016767 177626 177626 2$1 MOV      SR,SRH
856 000722 005267 177622          INC      SRH          ;POINT TO HIGH BYTE OF SR

```

```

857 000726 012637 000006 MOV (SP)+,#06 ;RESTORE ERROR VECTOR
858 000732 012637 000004 MOV (SP)+,#04
859 000736 005737 000042 TST ##4 ;CHECK FOR MONITOR OPERATION
860 000742 001405 BEQ STARTX
861 000744 005037 000174 CLR ##MMOPT ;RUN ALL SW DOWN IF MONITOR
862 000750 005037 000176 CLR ##SREG2
863 000754 000403 BR START1
864 000756 STARTX1
865 000756 017737 177564 000176 MOV #SR,##SREG2
866 000764 004767 014016 START1: JSR X7,NRALL
867 000770 012777 077406 177520 MOV #77406,#KPDOR
868 000776 012777 007600 177530 MOV #7600,#KPAR7 ;MAP PAGE 7 TO EXT BANK
869 001004 012777 077406 177512 MOV #77406,#KPDOR7
870 001012 005067 177576 CLR TBANK
871 001016 012767 177777 177560 MOV #177777,MEM0 ;SET UP CORE MAPS
872 001024 012767 077777 177554 MOV #777777,MEM1
873 001032 012767 000001 177550 MOV #1,COREPT ;SET UP 4K POINTER
874 001040 012767 000604 177544 MOV #MEM0,MEMUT
875 001046 012767 077406 177446 MOV #77406,#KPDOR2 ;REING CHECKED FOR
876 001054 012737 001124 000004 MOV #TMEMEX,#04 ;SET UP FOR TIME OUTS
877 001062 005037 000006 CLR ##6
878 001066 052777 000001 177404 BIS #1,#SR0
879 001074 016777 177514 177430 MAP1: MOV TBANK,#KPAR2 ;MAP KERNEL PAGE 2 TO BANK
880 001102 005737 041000 TST ##41000 ;1ST K PRESENT
881 001106 005737 045000 TST ##45000 ;2ND K PRESENT
882 001112 005737 051000 TST ##51000 ;3RD K PRESENT
883 001116 005737 055000 TST ##55000 ;4TH K PRESENT
884 001122 000404 BR MOVEPT ;OK, FULL 4K BLOCK PRESENT
885 001124 046777 177460 177460 TMEMEX: BIC COREPT,#MEMUT ;NO, BLOCK NOT PRESENT
886 001132 022626 CMP (SP)+,(SP)+ ;ADJUST STACK POINTER
887 001134 062767 000200 177452 MOVEPT: ADD #200,TBANK ;UPDATE BANK POINTER
888 001142 006367 177442 ASL COREPT
889 001146 103006 BCC MAP2 ;THIS 1ST MEM WORD DONE
890 001150 012767 000001 177432 MOV #1,COREPT
891 001156 012767 000606 177426 MOV #MEM1,MEMUT
892 001164 022767 007600 177422 MAP2: CMP #7600,TBANK ;EXTERNAL BANK YET
893 001172 001340 BNE MAP1 ;NO,NOT YET?
894 001174 012767 000001 177406 MOV #1,COREPT ;RE-INIT
895 001202 012767 000604 177402 MOV #MEM0,MEMUT
896 001210 042777 000001 177262 BIC #1,#SR0
897 001216 012767 000001 015510 MOV #1,ICOUNT
898 001224 004767 015352 JSR X7,CRLF
899 001230 012737 014642 000034 MOV #SCOPEC,#034
900 001236 005037 000036 CLR ##36 ;INITIALIZE SCOPE CALL TO KERNEL STATUS
901 001242 012737 015050 000030 MOV #EMTSRV,#030
902 001250 012737 000340 000032 MOV #340,#032
903 001256 012737 005452 014740 MOV #BEGIN,#RETURN
904 001264 005037 014736 CLR ##SCOPEF
905 001270 012737 000340 177776 MOV #340,#PSR ;LOCK OUT INTERRUPTS
906 001276 005037 016352 CLR ##PRTON ;PRINT ROUTINE BUSY FLAG
907 001302 000005 RESET
908 001304 012737 002314 000004 MOV #NODEV,##4 ;RETURN FOR NO DEVICE
909 001312 005037 000006 CLR ##6
910 001316 005067 001444 CLR DATA2 ;BASE DATA FOR TTY TELEPRINTER

```

```

911 001322 005737 000042 TST ##42 ;ACT 11?
912 001326 001405 BEQ ST3A ;YES
913 001330 052737 000001 000176 BIS #1,##SREG2 ;INHIBIT TTY OUT
914 001336 033727 000176 000001 ST3A: BIT ##SREG2,#1 ;INHIBIT TTY OUTPUT?
915 001344 001006 BNE ST3 ;YES, GO CHECK NEXT.
916 001346 012777 003000 177040 MOV #TYOUTR,#ATT PVC ;NO, SETUP INTERRUPT VECTOR
917 001354 052777 000100 177026 BIS #100,#TTCSR ;START TTY OUTPUT
918 001362 012700 000010 ST3: MOV #10,R0
919 001366 032737 000010 000176 BIT #10,##SREG2 ;INHIBIT RK DISK
920 001374 001026 BNE ST4 ;YES, SKIP OVER
921 001376 005777 177066 TST #RKCSR ;PRESENT
922 001402 012777 003376 177064 MOV #IRK,#RKVC ;SETUP VECTOR RETURNS
923 001410 012777 000240 177060 MOV #240,#RKST
924 001416 012767 043503 002014 MOV #43503,#RKFUNCT
925 001424 005077 177032 CLR #RKDAE ;INIT
926 001430 016777 002144 177030 MOV #LLIMIT,#RKBAR ;CORE BASE
927 001436 016777 002140 177020 MOV #WORDCT,#RKWC ;TRANSFER LENGTH
928 001444 116777 001770 177016 MOV #RKFUNCT,#RKCSR
929 001452 006300 ASL R0
930 001454 033727 000176 000020 ST4: BIT ##SREG2,#20 ;INHIBIT LINE CLOCK?
931 001462 001015 BNE ST5 ;YES, GO CK NEXT
932 001464 005777 176736 TST #LKCSR ;PRESENT
933 001470 012777 003056 176724 MOV #LK3,#KWLVC
934 001476 012777 000300 176720 MOV #300,#KWLST
935 001504 005067 001442 CLR TIME ;NO, INITIALIZE COUNT
936 001510 052777 000100 176710 BIS #100,#LKCSR ;START LINE CLOCK
937 001516 006300 ASL R0
938 001520 033727 000176 000040 ST5: BIT ##SREG2,#40 ;TEST FOR INHIBITING RF11 DISK
939 001526 001026 BNE ST6 ;SKIP IF SET
940 001530 005777 176714 TST #RFCSR ;PRESENT?
941 001534 012777 003472 176712 MOV #IRF,#RFVC ;SET UP TRAP RETURN
942 001542 012777 000240 176706 MOV #240,#RFST
943 001550 012767 043503 002020 MOV #43503,#RFUNCT ;WRITE CHECK/WRITE
944 001556 105277 176670 INCB #RFCSR ;INITIALIZE DISK-DAR,DAE
945 001562 016777 002014 176654 MOV #WORDCT,#RFWC ;LENGTH OF TRANSFER
946 001570 016777 002004 176650 MOV #LLIMIT,#RFRC ;CORE ADDRESS OF START OF TRANSFER
947 001576 116777 001774 176644 MOV #RFUNCT,#RFCSR ;START RF11 READ OR WRITE
948 001604 006300 ASL R0
949 001606 033727 000176 000100 ST6: BIT ##SREG2,#100 ;CHECK FOR INHIBITING TC11 DECTAPE
950 001614 001015 BNE ST7 ;SKIP IF SET
951 001616 005777 176732 TST #TCST ;PRESENT?
952 001622 012777 003612 176730 MOV #FENDZ,#TCIV ;GO TO END ZONE ON INTERRUPT
953 001630 012777 000300 176730 MOV #300,#TCSTA
954 001636 012777 004503 176706 MOV #R+IE+RB+DO,#TCM ;START REVERSE READ BLOCK NUMBER
955 001644 006300 ASL R0
956 001646 005737 000042 ST7: TST ##42 ;ACT 11?
957 001652 001402 BEQ 15 ;YES
958 001654 050037 000176 BIS #R0,##SREG2
959 001660 033727 000176 000200 18: BIT ##SREG2,#200 ;INHIBIT LINE PRINTER?
960 001666 001032 BNE ST8 ;YES, GO CK NEXT
961 001670 005777 176534 TST #LPCSR ;PRESENT?
962 001674 012737 001754 000004 MOV #S78,##4 ;DON'T CHANGE 200 IF NO SUCH DEVICE
963 001702 012767 000137 001246 MOV #137,SOLPAT ;RESET FOR START OF LINE PATTERN
964 001710 012767 000117 001332 MOV #79,CLINCT ;LINE COUNT

```

```

965 001716 012767 000137 001234 MOV #137,CURPAT
966 001724 012777 000014 176500 MOV #14,PLPOBR ;/LINE FEED TO POSITION BUFFER
967 001732 012777 003200 176474 MOV #LPINTR,PLPVC ;/INTERRUPT ENABLE
968 001740 012777 000200 176470 MOV #200,PLPST ;/PROCESSOR LEVEL 4
969 001746 012777 000100 176454 MOV #100,PLPCSR ;/INTERRUPT ENABLE
970 001754 005037 000602 ST0: CLR #*TRPB ;/NO *T* BIT FIRST PASS
971 001760 005037 000006 CLR #*6 ;/CHANGE ADDRESS ERROR VECTOR TO CAUSE
972 001764 012737 000006 000004 MOV #6,*#4 ;/HALT ON A TRAP TO 4
973 001772 004737 016676 JSR X7,*#USER ;/FOR USER I/O PROGRAM INSERTION
974 001776 004767 000332 JSR X7,DET1 ;/CHECK FOR CORE EXPANSION
975 002002 032737 000001 000174 BIT #1,*#MMOPT ;/INHIBIT MEMORY MANAG.?
976 002010 001106 BNE MODE ;/YES - GO SETUP USER
977 002012 004767 012770 JSR X7,NRALL ;/NO - MAKE ALL SEGMENTS INITIALLY NON-RESIDENT
978 002016 012777 077406 176500 MOV #77406,*KPDRT
979 002024 012777 007600 176502 MOV #7600,*KPAR7
980 002032 032737 000006 000174 BIT #6,*#MMOPT ;/INHIBIT USER/KERNEL OR 4K AS 32K?
981 002040 001415 BEQ SEG1 ;/NO - BRANCH
982 002042 012701 000007 MOV #7,R1 ;/YES - MAP KERNEL ASR'S 0-6 TO PA
983 002046 016702 176454 MOV KPAR0,R2
984 002052 005003 CLR R3
985 002054 010312 SETEX1 MOV R3,*R2
986 002056 012762 077406 177740 MOV #77406,-40(R2)
987 002064 005722 TST (R2)+
988 002066 062703 000200 ADD #200,R3
989 002072 077110 SOB R1,SETEX
990 002074 012777 077406 176414 SEG1: MOV #77406,*KPDRT ;/MAP KERNEL 0 TO BANK 0, RW
991 002102 032737 000004 000174 BIT #4,*#MMOPT ;/INHIBIT RUNNING 4K AS 32K?
992 002110 001416 BEQ USEALL ;/NO, SETUP FOR RUNNING 4K AS 32K
993 002112 012701 000010 MOV #10,R1 ;/YES, MAP ALL USER ASR'S TO PA
994 002116 016702 176366 MOV UPAR0,R2
995 002122 005003 CLR R3
996 002124 010312 SETUSE1 MOV R3,(R2)
997 002126 062703 000200 ADD #200,R3
998 002132 012762 077406 177740 MOV #77406,-40(R2)
999 002140 005722 TST (R2)+
1000 002142 077110 SOB R1,SETUSE
1001 002144 000425 BR SETSEG
1002 002146 012777 077406 176326 USEALL: MOV #77406,*UPDR0 ;/MAP USER ASR0 TO BANK 0, RW
1003 002154 012737 000000 000570 MOV #0,*CURBNK ;/CURRENT SAR CONTENTS
1004 002162 012767 000001 176420 MOV #1,COREPT ;/INIT MAP POINTERS
1005 002170 012767 000604 176414 MOV #MEM0,MEMUT
1006 002176 016767 176306 176370 MOV UPAR0,CURPAR ;/CURRENT SEGMENT REGISTER ADDRESSES
1007 002204 016767 176272 176364 MOV UPDR0,CURPDR
1008 002212 012767 005452 176360 MOV #8BEGIN,BNKSTR ;/CURRENT STARTING PC
1009 002220 052777 000001 176252 SETSEG: BIS #1,*SR0 ;/SET MEMORY MANAG. BIT
1010 002226 042737 000340 177776 MODE: BIC #340,*#PSR ;/PRIORITY LEVEL 0
1011 002234 032737 000002 000174 BIT #2,*#MMOPT ;/INHIBIT USER/KERNEL?
1012 002242 001016 BNE MAIN+2 ;/YES - SKIP OVER
1013 002244 052737 140000 000036 BIS #140000,*#36 ;/SET USER BIT IN SCOPE STATUS
1014 002252 012746 000400 MOV #UBUFF,-(R6)
1015 002256 052737 030000 177776 BIS #30000,*#PSR
1016 002264 006606 MTP1 SP ;/SET UP USER STACK
1017 002266 012737 140000 177776 MOV #140000,*#PSR ;/CHANGE TO USER
1018 002274 000001 BR ,+4

```

```

1019 002276 000001 HAIN: WAIT
1020 002300 033727 000176 002000 BIT #*SREG2,*#2000 ;/INHIBIT PROCESSOR TEST
1021 002306 001373 BNE MAIN
1022 002310 000167 003136 JMP BEGIN
1023
1024 ;/NON-EXISTING DEVICE SERVICE
1025 002314 050037 000176 NODEV: BIS R0,*#SREG2 ;/SET INHIBIT BIT
1026 002320 162716 000006 SUR #6,(SP) ;/ALTER PC RETURN
1027 002324 042766 000017 000002 BIC #17,2(SP) ;/CLEAR Z BIT ON STACK
1028 002332 000002 RTI
1029
1030 ;/PDP-11 MEMORY DETERMINATION AND SETUP/
1031 ;/USE WITH VARIABLE CORE QUANTITY SYSTEMS/
1032 002334 012767 104010 012236 DET1: MOV #E0B,DONE ;/RESTORE INITIAL CODE
1033 002342 032767 000007 175624 BIT #7,*#MMOPT ;/INHIBIT RUNNING 4K AS 32K USER?
1034
1035 002350 001001 BNE ,+4 ;/YES - ALLOW CORE EXPANSION
1036 002352 000207 RTS X7 ;/NO - INHIBIT SEGMENTATION?
1037 002354 032737 000040 000174 BIT #40,*#MMOPT ;/CHECK VARIABLE CORE SWITCH
1038 002362 001401 BEQ DET4 ;/USE VARIABLE CORE ROUTINE
1039 002364 000207 RTS X7 ;/4K ONLY (SWITCH SET)
1040 002366 012737 002452 000004 DET4: MOV #DET2,*#4 ;/TRAP VECTOR SETUP
1041 002374 012737 000340 000006 MOV #340,*#6 ;/TRAP STATUS SETUP
1042 002402 000241 CLC
1043 002404 005537 037770 EIGHT: ADC #*37770 ;/CHECK FOR 8K
1044 002410 000240 NOP
1045 002412 005537 057770 ADC #*57770 ;/CHECK FOR 12K
1046 002416 000240 NOP
1047 002420 005537 077770 ADC #*077770 ;/CHECK FOR 16K
1048 002424 000240 NOP
1049 002426 005537 117770 ADC #*117770 ;/CHECK FOR 20K
1050 002432 000240 NOP
1051 002434 005537 137770 ADC #*137770 ;/CHECK FOR 24K
1052 002440 000240 NOP
1053 002442 005537 157770 ADC #*157770 ;/CHECK FOR 28K
1054 002446 000240 NOP
1055 002450 000437 BR STRT28
1056 002452 012602 DET2: MOV (6)+,X2 ;/RETRIEVE TRAP PC
1057 002454 005726 TST (6)+ ;/DISCARD TRAP STATUS WORD
1058 002456 062702 000074 ADD #STRT4-EIGHT=4,R2
1059 002462 000112 JMP #R2
1060
1061 002464 005000 MOVE: CLR X0 ;/SET UP MAIN CORE POINTER
1062 002466 010102 MOV X1,X2
1063 002470 062702 015006 ADD #0+2,X2 ;/SET UP MAX CORE MOVE
1064 002474 012021 MOV (0)+,(1)+ ;/MOVE WORD
1065 002476 002021 CMP X2,X1 ;/MOVE COMPLETE?
1066 002500 001375 BNE ,+4 ;/MOVE ANOTHER WORD
1067 002502 000207 RTS X7 ;/MOVE COMPLETE
1068 002504 000521 STRT4: BR DET3
1069 002506 000240 NOP
1070 002510 000240 NOP
1071 002512 004767 000110 JSR X7,XFER8 ;/START 8K TRANSFER
1072 002516 000506 BR MOD4 ;/START 4K MODIFY

```

```

1073 002520 004767 000072 JSR X7,XFER12 ;START 12K TRANSFER
1074 002524 000475 BR MOD0 ;START 8K MODIFY
1075 002526 004767 000054 JSR X7,XFER16 ;START 16K TRANSFER
1076 002532 000464 BR MOD12 ;START 12K MODIFY
1077 002534 004767 000036 JSR X7,XFER20 ;START 20K TRANSFER
1078 002540 000453 BR MOD16 ;START 16K MODIFY
1079 002542 004767 000020 JSR X7,XFER24 ;START 24K TRANSFER
1080 002546 000442 BR MOD20 ;START 20K MODIFY
1081 002550 004767 000002 STRT20: JSR X7,XFER28 ;START 28K TRANSFER
1082 002554 000431 BR MOD24 ;START 24K MODIFY
1083 002556 012701 140000 XFER20: MOV #140000,X1 ;SET UP MOVE START LOCATION
1084 002562 004767 177676 JSR X7,MOVE ;GO TO MOVE SUBROUTINE
1085 002566 012701 120000 XFER24: MOV #120000,X1
1086 002572 004767 177666 JSR X7,MOVE
1087 002576 012701 100000 XFER20: MOV #100000,X1
1088 002602 004767 177656 JSR X7,MOVE
1089 002606 012701 060000 XFER16: MOV #60000,X1
1090 002612 004767 177646 JSR X7,MOVE
1091 002616 012701 040000 XFER12: MOV #40000,X1
1092 002622 004767 177636 JSR X7,MOVE
1093 002626 012701 020000 XFER08: MOV #20000,X1
1094 002632 004767 177626 JSR X7,MOVE
1095 002636 000207 RTS X7 ;RETURN FROM TRANSFERS
1096 002640 012767 000137 131732 MOD24: MOV #137,DONE+120000
1097 002646 012767 145420 131726 MOV #0EGINX+140000,DONE+120000
1098 002654 012767 000137 111716 MOD20: MOV #137,DONE+100000
1099 002662 012767 125420 111712 MOV #0EGINX+120000,DONE+100000
1100 002670 012767 000137 071702 MOD16: MOV #137,DONE+60000
1101 002676 012767 105420 071676 MOV #0EGINX+100000,DONE+60000
1102 002704 012767 000137 051666 MOD12: MOV #137,DONE+40000
1103 002712 012767 065420 051662 MOV #0EGINX+60000,DONE+40000
1104 002720 012767 000137 031652 MOD8: MOV #137,DONE+20000
1105 002726 012767 045420 031646 MOV #0EGINX+40000,DONE+20000
1106 002734 012767 000137 011636 MOD4: MOV #137,DONE
1107 002742 012767 025420 011632 MOV #0EGINX+20000,DONE+2
1108 002750 000037 000006 DET3: CLR #6
1109 002754 012737 000006 MOV #6,#4
1110 002762 000207 RTS X7
1111
1112 ;TTY TRANSMITTER PRINT VALUES 0 TO 377/
1113 002764 005027 000000 TYOUT: CLR #0 ;INITIAL DATA
1114 002766 DATA2#,-2
1115 002770 016777 177772 175414 TYOUT1: MOV DATA2,#TT08R ;OUTPUT TO DEVICE
1116 002776 000002 RTI ;RETURN TO MAINLINE**
1117 003000 017767 175404 175412 TYOUTR: MOV #TTCSR,TTSAV
1118 003006 105767 175406 TSTB TTSAV ;TEST FOR DONE
1119 003012 100401 BMI +4 ;BRANCH IF FLAG FOUND
1120 003014 104006 HLT ;FALSE INTERRUPT RETURN
1121 003016 005267 177744 INC DATA2 ;INCREMENT DATA
1122 003022 022767 000400 177736 CMP #400,DATA2 ;TEST DATA FOR UPPER LIMIT
1123 003030 001755 BEQ TYOUT ;AT UPPER LIMIT START OVER
1124 003032 000756 BR TYOUT1 ;FINISH REST OF DATA
1125
1126 ;TEST OF LINE CLOCK, INTERRUPT FOR 55 SECONDS THEN STALL FOR 5 SECONDS.

```

```

1127 003034 005037 003152 LK1: CLR #0TIME ;CLEAR LINE CLOCK TIMER
1128 003040 052777 000100 175360 BIC #100,#LKCSR
1129 003046 052737 000100 177776 BIC #100,#PSR
1130 003054 000002 LK2: RTI
1131 003056 105777 175344 LK3: TSTB #LKCSR
1132 003062 100401 BMI +4
1133 003064 104006 HLT ;FALSE INTERRUPT
1134 003066 042777 000200 175332 BIC #200,#LKCSR
1135 003074 005237 003152 LK4: INC #0TIME ;HERE ON INTERRUPTS
1136 003100 022737 006344 003152 CMP #3300,#0TIME ;55 SEC YET?
1137 003106 103362 LK2 BMS ;BR IF NOT
1138 003110 042777 000100 175310 BIC #100,#LKCSR
1139 003116 042737 000100 177776 BIC #100,#PSR
1140 003124 022737 007020 003152 CMP #3600,#0TIME ;ONE MINUTE YET
1141 003132 001740 BEQ LK1 ;YES RESET TIMER
1142 003134 105777 175266 TSTB #LKCSR ;NO, SKIP TILL MINUTE UP
1143 003140 100375 BMI +4
1144 003142 042777 000200 175256 BIC #200,#LKCSR ;CLEAR FLAG
1145 003150 000751 BR LK4
1146 003152 000000 TIME: 0
1147
1148 ;LINE PRINTER SHOULD RAISE PROCESSOR PRIORITY TO LEVEL OF LINE PRINTER/
1149 ;INTERRUPT VECTOR IS 200/
1150 003154 012727 000000 000000 LP1: MOV #0,#0 ;START OF LINE TO CURRENT
1151 003160 003160 CURPAT#,-2 ;CHARACTER BEING PRINTED
1152 003156 003156 SOLPAT#,-4 ;START OF LINE CHARACTER
1153 003162 016777 177772 175242 LP2: MOV CURPAT,#LP0BR ;CURRENT PATTERN TO LINE PRINTER
1154 003170 105777 175234 TSTB #LPCSR
1155 003174 100420 BMI LP6
1156 003176 000002 RTI ;RETURN TO MAIN LINE
1157 003200 105777 175224 LPINTR: TSTB #LPCSR ;TEST FOR FLAG
1158 003204 100414 BMI LP6
1159 003206 005737 000042 TST #042 ;MONITOR LOAD
1160 003212 001410 BEQ LP7 ;NO, ERROR
1161 003214 032777 100000 175206 BIT #100000,#LPCSR ;YES, IS ERROR SET
1162 003222 001404 BEQ LP7 ;NO, ERROR
1163 003224 042777 000100 175176 BIC #100,#LPCSR ;DIS ABLE INTERRUPT
1164 003232 000002 HLT
1165 003234 104006 ;FALSE RETURN FROM MAIN LINE
1166 003236 026727 000006 000117 LP6: CMP CLINCT,#79 ;TEST FOR END OF LINE
1167 003244 001415 BEQ LP4 ;GO GENERATE CR/LF
1168 003246 005227 000000 INC #0 ;INCREMENT LINE POSITION COUNT
1169 003250 CLINCT#,-2 ;POSITION OF LINE
1170 003252 026727 177702 000137 CMP CURPAT,#137 ;TEST FOR MAXIMUM PATTERN
1171 003260 001403 BEQ LP3 ;YES - GO TO LP3 AND RESET
1172 003262 005267 177672 INC CURPAT ;NO - INCREMENT TO NEXT PATTERN
1173 003266 000735 BR LP2 ;GO SEND IT TO LINE PRINTER
1174 003270 012767 000040 177662 LP3: MOV #40,CURPAT ;RESET PATTERN AND SEND TO PRINTER
1175 003276 000731 BR LP2 ;SENT TO LINE PRINTER
1176 003300 005067 177744 LP6: CLR CLINCT ;RESET LINE COUNT
1177 003304 012777 000012 175120 LP6: MOV #12,#LP0BR ;LINE FEED
1178 003312 105777 175112 TSTB #LPCSR
1179 003316 100375 BMI +4
1180 003320 026727 177632 000137 CMP SOLPAT,#137 ;START OF LINE PATTERN

```

```

1181 003326 001403          BEQ     LPS
1182 003330 005267          INC     SOLPAT          ;INCREMENT START OF LINE
1183 003334 000707          BR      LP1
1184 003336 012767          MOV     #40,SOLPAT      ;RESET START OF LINE
1185 003344 000703          BR      LP1             ;PRINT
1186
1187
1188          ;RKF11 DISK TEST
1188 003346 005077 175110          JRK11  CLR          ;INTERRUPT LEVEL 5, 2000 WORD TRANSFERS
1189 003352 013777 003600          RK11   #0          ;CORE BASE
1190 003360 013777 003602 175076          MOV     #0,LLIMIT,RK11 ;TRANSFER LENGTH
1191 003366 113777 003440 175074          MOV     #0,WORDOCT,RK11 ;WRITE OR WRITE CK TO DSK
1192 003374 000002          RTI    #0,RKFCR        ;RETURN TO MAINLINE
1193 003376 032777 100200 175064          BIT     #100200,RKFCR   ;INTERRUPT RETURN
1194 003404 003002          BGT    .+6
1195 003406 104006          HLT
1196 003410 000756          BR      RKSTART        ;DISK AT UPPER LIMIT?
1197 003412 032777 000037 175042          BIT     #37,RKDAE
1198 003420 001354          BNE    RK1             ;DISK AT UPPER LIMIT?
1199 003422 122777 000031 175030          CMPB   #31,RKDAH
1200 003430 001350          RFI
1201 003432 000337 003440          SNE    #0,RKFUNCT
1202 003436 000743          SWAB   RKSTART        ;CHANGE COMMAND
1203 003440 000000          BR      RKSTART        ;RESTART NEW TRANSFER OF DISK
1204
1205          RKFUNCT: 0
1206
1206 003442 105277 175004          ;RKF11 DISK
1207 003446 013777 003600          RFSTART: INCB   #0,RFCSRH   ;INITIALIZE DISK - DAR-DAE
1208 003454 013777 003602 174772          RF1:   MOV     #0,LLIMIT,RF1 ;CORE BASE
1209 003462 113777 003576 174760          MOV     #0,WORDOCT,RF1 ;LENGTH OF TRANSFER
1210 003470 000002          MOVB   #0,RFUNCT,RF1   ;WRITE OR WRITE CHECK TO DISK
1211 003472 105777 174752          RTI    #0,RFCSR        ;RETURN TO MAINLINE CODE
1212 003476 100402          TSTB   #0,RFCSR        ;INTERRUPT VECTOR POINTS HERE
1213 003500 104006          BMI    .+6
1214 003502 000757          HLT
1215 003504 005777 174740          BR      RFSTART        ;RKF11 READY NOT UP
1216 003510 100012          TST    #0,RFCSR        ;ERROR SET?
1217 003512 032777 020000 174730          BPL    ERROK           ;BRANCH IF NOT
1218 003520 001404          BIT     #20000,RFCSR   ;YES=WRITE CHECK ERROR?
1219 003522 104006          BEQ    ERRSET          ;NO-BRANCH
1220 003524 000337 003576          HLT    #0,RFUNCT
1221 003530 000744          SWAB   RFSTART        ;CHANGE COMMAND TO DO WRITE
1222 003532 104006          BR      RFSTART        ;RKF11 ERROR SET=NOT WRITE CHECK
1223 003534 000742          HLT
1224 003536 005777 174702          ERRSET: TST    #0,RFWC
1225 003542 100002          BPL    .+6
1226 003544 104006          HLT
1227 003546 000735          BR      RFSTART        ;RKF-11 WORD COUNT NOT ZERO
1228 003550 122777 000003 174662          CMPB   #3,RFDAE
1229 003556 001333          BNE    RF1             ;DISK AT UPPER LIMIT? 7=2, 17=4, 37=8
1230 003560 027727 174656 174000          CMP    #0,RF0AR,#174000 ;NO
1231 003566 101727          BLOS   RF1             ;AS FAR ON DISK AS WE CAN GO
1232 003570 000337 003576          SWAB   RF1             ;NO
1233 003574 000722          BR      RFSTART        ;CHANGE COMMAND
1234 003576 000000          BR      RFSTART        ;RESTART NEW TRANSFER OF DISK
1235
1236
1237          RKFUNCT: 0
1238
1239          ;RKFUNCT: 0
1240
1241          ;RKFUNCT: 0
1242
1243          ;RKFUNCT: 0
1244
1245          ;RKFUNCT: 0
1246
1247          ;RKFUNCT: 0
1248
1249          ;RKFUNCT: 0
1250
1251          ;RKFUNCT: 0
1252
1253          ;RKFUNCT: 0
1254
1255          ;RKFUNCT: 0
1256
1257          ;RKFUNCT: 0
1258
1259          ;RKFUNCT: 0
1260
1261          ;RKFUNCT: 0
1262
1263          ;RKFUNCT: 0
1264
1265          ;RKFUNCT: 0
1266
1267          ;RKFUNCT: 0
1268
1269          ;RKFUNCT: 0
1270
1271          ;RKFUNCT: 0
1272
1273          ;RKFUNCT: 0
1274
1275          ;RKFUNCT: 0
1276
1277          ;RKFUNCT: 0
1278
1279          ;RKFUNCT: 0
1280
1281          ;RKFUNCT: 0
1282
1283          ;RKFUNCT: 0
1284
1285          ;RKFUNCT: 0
1286
1287          ;RKFUNCT: 0
1288          ;RKFUNCT: 0

```

```

1235 003600 005452          LLIMIT: BEGIN          ;FIRST CORE ADDRESS OF TRANSFER
1236 003602 176000          WORDCT: =2000        ;LENGTH OF TRANSFER
1237
1238          ;DECTAPE DIAGNOSTIC ROUTINE. THE TAPE IS FIRST DRIVEN TO THE FORWARD
1239          ;END ZONE. THE DESIRED DATA IS THEN GENERATED IN THE DECTAPE BUFFER AREA
1240          ;AND DATA IS WRITTEN ONTO ALL BLOCKS FROM THE BLOCK NUMBER IN TCFRST
1241          ;THRU THE BLOCK NUMBER IN TCLAST. BLOCK NUMBERS ARE ALSO CHECKED FOR
1242          ;BEING IN ORDER. AFTER THE BLOCK NUMBER IN TCLAST IS WRITTEN, TAPE IS
1243          ;DRIVEN INTO THE REVERSE END ZONE.
1244          ;THE TAPE IS THEN STARTED IN REVERSE, AND WHEN THE CLOSEST BLOCK THAT
1245          ;WAS WRITTEN (TCLAST) IS FOUND, IT IS READ INTO THE DECTAPE BUFFER AREA.
1246          ;THE PROGRAM INTERRUPT REQUEST FACILITY IS THEN USED TO BOOK A REQUEST
1247          ;FOR CHECKING THE DATA AT LEVEL 3, AND NO FURTHER DATA IS READ IN
1248          ;UNTIL THAT DATA HAS BEEN CHECKED. AFTER IT IS CHECKED, THE DATA IS
1249          ;SCRAMBLED TO GUARANTEE THAT NEW DATA IS REALLY READ IN NEXT TIME. WHILE
1250          ;THIS IS GOING ON, BLOCK NUMBERS ARE CHECKED FOR BEING IN ORDER AS THE
1251          ;TAPE TRAVELS TOWARD THE FORWARD END ZONE,ONCE THE DATA IS FULLY CHECKED
1252          ;THE NEXT BLOCK THAT COMES UP IS READ IN AND THE PROCESS REPEATED. ONCE
1253          ;THE BLOCK WHOSE NUMBER IS IN TCFRST HAS BEEN READ, THE TAPE IS DRIVEN
1254          ;INTO THE FORWARD END ZONE AND THE WHOLE SEQUENCE IS REPEATED.
1255
1256          ;FUNCTION VALUES IN CSR
1257          ;DT11 DEC TAPE
1258          RD=4          ;READ DATA
1259          WD=14         ;WRITE DATA
1260          RB=2
1261          IE=500       ;INTERRUPT ENABLE*UNIT 1
1262          DO=1         ;DO = THE FUNCTION
1263          R=4000      ;REVERSE
1264
1265 003604 000000          TCFIRST: 0          ;FIRST BLOCK TO BE SEARCHED FOR
1266 003606 001101          TCLAST: 577        ;LAST BLOCK TO BE SEARCHED FOR
1267 003610 000000          TCXPE: 0          ;THE BLOCK THAT IS EXPECTED
1268
1269          ;GO TO FORWARD END ZONE
1270 003612 012777 003612 174744          FENDZ: MOV     #FENDZ,#TC1V ;END ZONE VECTOR SETUP
1271 003620 005777 174730          TST    #TCST         ;TEST FOR END ZONE
1272 003624 100403          BMI    FEND1         ;AT END ZONE?
1273 003626 105277 174720          INCB   #TC1V         ;SET DO = NO DELAY
1274 003632 000002          RTI    #TC1V         ;NO = WAIT SOME MORE
1275 003634 012777 003664 174722          FEND1: MOV     #TCF1,#TC1V ;YES = NEW VECTOR
1276 003642 042777 104000 174702          BIC    #104000,#TC1V ;SEARCH BLOCK FOWARD
1277 003650 016767 177730 177732          MOV     TCFIRST,TCXPE ;COUNT WHEN THIS BLOCK IS FOUND
1278 003656 105277 174670          TCF1A: INCB   #TC1V         ;SET DO
1279 003662 000002          RTI
1280 003664 032777 100200 174660          TCF1:  BIT     #100200,#TC1V ;RETURN ON NEXT BLOCK
1281 003672 100001          BPL    .+4           ;ANY ERROR ON READ?
1282 003674 104006          HLT
1283 003676 001001          BNE    .+4           ;TC ERROR SET = FORWARD READ BLOCK
1284 003700 104006          HLT                 ;DONE FLAG UP?
1285 003702 027767 174650 177700          CMP    #0,TCXPE     ;FALSE INTERRUPT
1286 003710 002762          BLT    TCF1A         ;IS THIS OUR BLOCK FOR SYNC
1287 003712 001401          BEQ    TCF2         ;NO=READ SOME MORE BLOCKS
1288 003714 104006          HLT                 ;YES

```

```

1289
1290 003716 012777 003732 174640 TCF2: MOV #TCF3,#TCIV ;VECTOR FOR SEQUENTIAL READS
1291 003724 105277 174622 INCB #TCCH ;ISET DO
1292 003730 000002 RTI ;RETURN AND TEST SEQUENTIAL BLOCKS
1293
1294 ;FIND SEQUENTIAL BLOCK AT FOWARD DIRECTION
1295 003732 032777 100200 174612 TCF3: BIT #100200,#TCCH ;TEST ERROR AND READY
1296 003740 100001 BPL .+4
1297 003742 104006 HLT ;FOWARD READ ERROR TC=11
1298 003744 001001 BNE .+4
1299 003746 104006 HLT ;FALSE INTERRUPT ON TC=11
1300 003750 027767 174602 177630 CMP #TCDD,TCLAST ;HAVE WE TESTED ALL BLOCKS
1301 003756 001414 BEQ RENDZ ;YES DRIVE UNIT IN END ZONE TO START OVER
1302 003760 005267 177624 TCEXPZ ;NO-INCREMENT EXPECTED COUNT
1303 003764 027767 174566 177616 CMP #TCDD,TCEXPZ ;IS CURRENT BLOCK CORRECT
1304 003772 001401 BEQ .+4
1305 003774 104006 HLT ;FAILED IN FOWARD READ TO FIND NEXT BLOCK
1306 003776 000427 BR TCWBK ;THIS ROUTINE WRITES A BLOCK
1307 004000 105277 174546 TCF4: INCB #TCCH ;ISET DO
1308 004004 000002 RTI
1309 004006 000701 XFENDZ: BR FENOZ ;INDIRECT LINK
1310
1311 ;MOVE TAPE TO REVERSE END ZONE
1312 004010 012777 004010 174546 RENDZ: MOV #RENDZ,#TCIV ;END ZONE VECTOR SETUP
1313 004016 016767 177564 177564 MOV TCLAST,TCEXPZ ;SET UP FOR REVERSE SEARCH
1314 004024 005777 174524 TST #TCST ;IN END ZONE
1315 004030 100403 BMI REND1 ;YES - START TO TURN UNIT AROUND
1316 004032 105277 174514 INCB #TCCH ;ISET DO
1317 004036 000002 RTI ;NO - WAIT TILL WE ARE
1318 004040 012777 004503 174504 REND1: MOV #R+IE+RB+DO,#TCCH ;FUNCTION = READ BLOCK, REVERSE AND GO
1319 004046 012777 004136 174510 MOV #TCR1,#TCIV ;SET UP NEW INTERRUPT VECTOR
1320 004054 000002 RTI
1321 ;WRITE FORWARD ALL BLOCKS EXCEPT 0
1322
1323 004056 012777 004110 174500 TCWBK: MOV #TCWB1,#TCIV ;INTERRUPT VECTOR FOR WRITE
1324 004064 012777 177400 174466 MOV #-400,#TCWC ;ONE BLOCK
1325 004072 012777 004420 174462 MOV #TCWBUF,#TCBA ;THE WRITE BUFFER ADDRESS
1326 004100 112777 000515 174444 MOVVB #IE+WD+DO,#TCCH ;WRITE THE BLOCK
1327 004106 000002 RTI ;RETURN WHEN BLOCK IS WRITTEN
1328 004110 005777 174436 TCWB1: TST #TCCH ;ANY ERRORS
1329 004114 100001 BPL .+4
1330 004116 104006 HLT
1331 004120 012777 003732 174436 MOV #TCF3,#TCIV ;SEARCH BLOCK VECTOR
1332 004126 112777 000502 174416 MOVVB #IE+RB,#TCCH ;READ BLOCK
1333 004134 000721 TCF4 ;FIND THE NEXT BLOCK
1334
1335 004136 032777 100200 174406 TCR1: BIT #100200,#TCCH ;TEST FOR ERROR AND READY
1336 004144 100001 BPL .+4
1337 004146 104006 HLT ;DECTAPE ERROR ON READ BLOCK REVERSE
1338 004150 001001 BNE .+4
1339 004152 104006 HLT ;FALSE INTERRUPT FROM DECTAPE
1340 004154 027767 174376 177426 CMP #TCDD,TCEXPZ ;IS IT OUR FIRST BLOCK
1341 004162 001406 BEQ TCR2 ;YES - GO TEST THE REST
1342 004164 002002 BGE TCR1A ;NO - HAVE WE PASSED THE BLOCK
    
```

```

1343 004166 104006 HLT ;WE PASS OUR BLOCK
1344 004170 000707 BR RENDZ ;GO TO END ZONE AND TRY AGAIN
1345 004172 105277 174354 TCR1A: INCB #TCCH ;ISET DO
1346 004176 000002 RTI ;WE FOUND OUR FIRST BLOCK
1347 004200 012777 004214 174356 TCR2: MOV #TCR3,#TCIV ;SET UP INTERRUPT TO TEST ALL BLOCKS
1348 004206 105277 174340 INCB #TCCH ;ISET DO
1349 004212 000002 RTI ;WAIT FOR NEXT BLOCK TO INTERRUPT
1350
1351 ;FIND SEQUENTIAL BLOCK IN REVERSE DIRECTION
1352 004214 032777 100200 174330 TCR3: BIT #100200,#TCCH ;TEST FOR READ AND ERROR
1353 004222 100001 BPL .+4
1354 004224 104006 HLT ;ERROR READING SEQUENTIAL BLOCK IN REVERSE
1355 004226 001001 BNE .+4
1356 004230 104006 HLT ;FALSE DECTAPE INTERRUPT
1357 004232 026777 177346 174316 CMP TCFIRST,#TCDD ;DO WE DO ALL THE BLOCKS
1358 004240 001662 XFENDZ ;YES - GO TO END ZONE TO RESTART
1359 004242 005367 177342 DEC TCEXPZ ;NO - DECREMENT BLOCK NUMBER
1360 004246 027767 174304 177334 CMP #TCDD,TCEXPZ ;TEST SEQUENTIAL BLOCK IN REVERSE
1361 004254 001401 BEQ .+4
1362 004256 104006 HLT ;TEST SEQUENTIAL READ BLOCK IN REVERSE FAILED
1363 004260 000403 BR TCRBK ;THIS ROUTINE READ A BLOCK
1364 004262 105277 174264 TCR4: INCB #TCCH ;ISET DO
1365 004266 000002 RTI ;LETS TRY A NEW BLOCK
1366
1367 ;READ REVERSE ALL BLOCK EXCEPT BLOCK 1101
1368 004270 012777 004326 174266 TCRBK: MOV #TCRB1,#TCIV ;SET UP INTERRUPT VECTOR
1369 004276 012777 177400 174254 MOV #-400,#TCWC ;READ ONE BLOCK
1370 004304 012777 004420 174250 MOV #TCRBUF,#TCBA ;WHERE BUFFER IS
1371 004312 112777 000505 174232 MOVVB #IE+RD+DO,#TCCH ;READ THE BLOCK
1372 004320 004767 000030 JSR X7,TC1 ;CHECK DATA BUFFER
1373 004324 000002 RTI ;EXIT - RETURN WHEN BLOCK IS READ
1374 004326 005777 174220 TCRB1: TST #TCCH ;AND ERRORS
1375 004332 100001 BPL .+4
1376 004334 104006 HLT ;DECTAPE ERROR
1377 004336 012777 004214 174220 MOV #TCR3,#TCIV ;NEW VECTOR FOR BLOCK SEARCH
1378 004344 112777 000502 174200 MOVVB #IE+RB,#TCCH ;READ BLOCK FUNCTION
1379 004352 000743 BR TCR4 ;RETURN TO BLOCK SEARCH
1380
1381 ;THIS ROUTINE CHECKS THE READ DATA BUFFER TC11
1382 ;BY DOING A CHECK SUM ON THE DATA
1383 004354 010146 TC1: MOV X1,-(6) ;SAVE THESE ON THE STACK
1384 004356 010246 MOV X2,-(6)
1385 004360 010346 MOV X3,-(6)
1386 004362 005003 CLR X3 ;SUM OF DATA
1387 004364 012701 004420 MOV #TCRBUF,X1 ;ADDRESS OF READ BUFFER
1388 004370 012702 005420 MOV #TCRBUF+1000,X2 ;END OF READ BUFFER
1389 004374 062103 TC2: ADD (1),X3 ;EVEN ADD
1390 004376 062103 ADD (1)+,X3 ;ODD ADD +2'S COMPLIMENT
1391 004400 001401 BEQ .+4
1392 004402 104006 HLT ;DATA ERROR TC=11
1393 004404 020102 CMP X1,X2 ;AT END OF BUFFER?
1394 004406 001372 BNE TC2 ;NO - SUM THE REST
1395 004410 012603 MOV (6)+,X3 ;RESTORE THE REGISTERS
1396 004412 012602 MOV (6)+,X2
    
```

1397	004414	012601	MOV	(6)+,X1	
1398	004416	000207	RTS	X7	JEXIT
1399					
1400					}THIS WRITE BUFFER LOOK THE SAME FORWARD OR REVERSE
1401	004420				TCNBUF:
1402	004420				TCRBUF:
1403		000001		N=N+1	
1404	004420	000001		N	IDECTAPE WRITE BUFFER
1405	004422	177777		=N	
1406		000002		N=N+1	
1407	004424	000002		N	IDECTAPE WRITE BUFFER
1408	004426	177776		=N	
1409		000003		N=N+1	
1410	004430	000003		N	IDECTAPE WRITE BUFFER
1411	004432	177775		=N	
1412		000004		N=N+1	
1413	004434	000004		N	IDECTAPE WRITE BUFFER
1414	004436	177774		=N	
1415		000005		N=N+1	
1416	004440	000005		N	IDECTAPE WRITE BUFFER
1417	004442	177773		=N	
1418		000006		N=N+1	
1419	004444	000006		N	IDECTAPE WRITE BUFFER
1420	004446	177772		=N	
1421		000007		N=N+1	
1422	004450	000007		N	IDECTAPE WRITE BUFFER
1423	004452	177771		=N	
1424		000010		N=N+1	
1425	004454	000010		N	IDECTAPE WRITE BUFFER
1426	004456	177770		=N	
1427		000011		N=N+1	
1428	004460	000011		N	IDECTAPE WRITE BUFFER
1429	004462	177767		=N	
1430		000012		N=N+1	
1431	004464	000012		N	IDECTAPE WRITE BUFFER
1432	004466	177766		=N	
1433		000013		N=N+1	
1434	004470	000013		N	IDECTAPE WRITE BUFFER
1435	004472	177765		=N	
1436		000014		N=N+1	
1437	004474	000014		N	IDECTAPE WRITE BUFFER
1438	004476	177764		=N	
1439		000015		N=N+1	
1440	004500	000015		N	IDECTAPE WRITE BUFFER
1441	004502	177763		=N	
1442		000016		N=N+1	
1443	004504	000016		N	IDECTAPE WRITE BUFFER
1444	004506	177762		=N	
1445		000017		N=N+1	
1446	004510	000017		N	IDECTAPE WRITE BUFFER
1447	004512	177761		=N	
1448		000020		N=N+1	
1449	004514	000020		N	IDECTAPE WRITE BUFFER
1450	004516	177760		=N	

1451		000021		N=N+1	
1452	004520	000021		N	IDECTAPE WRITE BUFFER
1453	004522	177757		=N	
1454		000022		N=N+1	
1455	004524	000022		N	IDECTAPE WRITE BUFFER
1456	004526	177756		=N	
1457		000023		N=N+1	
1458	004530	000023		N	IDECTAPE WRITE BUFFER
1459	004532	177755		=N	
1460		000024		N=N+1	
1461	004534	000024		N	IDECTAPE WRITE BUFFER
1462	004536	177754		=N	
1463		000025		N=N+1	
1464	004540	000025		N	IDECTAPE WRITE BUFFER
1465	004542	177753		=N	
1466		000026		N=N+1	
1467	004544	000026		N	IDECTAPE WRITE BUFFER
1468	004546	177752		=N	
1469		000027		N=N+1	
1470	004550	000027		N	IDECTAPE WRITE BUFFER
1471	004552	177751		=N	
1472		000030		N=N+1	
1473	004554	000030		N	IDECTAPE WRITE BUFFER
1474	004556	177750		=N	
1475		000031		N=N+1	
1476	004560	000031		N	IDECTAPE WRITE BUFFER
1477	004562	177747		=N	
1478		000032		N=N+1	
1479	004564	000032		N	IDECTAPE WRITE BUFFER
1480	004566	177746		=N	
1481		000033		N=N+1	
1482	004570	000033		N	IDECTAPE WRITE BUFFER
1483	004572	177745		=N	
1484		000034		N=N+1	
1485	004574	000034		N	IDECTAPE WRITE BUFFER
1486	004576	177744		=N	
1487		000035		N=N+1	
1488	004600	000035		N	IDECTAPE WRITE BUFFER
1489	004602	177743		=N	
1490		000036		N=N+1	
1491	004604	000036		N	IDECTAPE WRITE BUFFER
1492	004606	177742		=N	
1493		000037		N=N+1	
1494	004610	000037		N	IDECTAPE WRITE BUFFER
1495	004612	177741		=N	
1496		000040		N=N+1	
1497	004614	000040		N	IDECTAPE WRITE BUFFER
1498	004616	177740		=N	
1499		000041		N=N+1	
1500	004620	000041		N	IDECTAPE WRITE BUFFER
1501	004622	177737		=N	
1502		000042		N=N+1	
1503	004624	000042		N	IDECTAPE WRITE BUFFER
1504	004626	177736		=N	

1505		000043	N=N+1	
1506	004630	000043	N	IDECTAPE WRITE BUFFER
1507	004632	177735	=N	
1508		000044	N=N+1	
1509	004634	000044	N	IDECTAPE WRITE BUFFER
1510	004636	177734	=N	
1511		000045	N=N+1	
1512	004640	000045	N	IDECTAPE WRITE BUFFER
1513	004642	177733	=N	
1514		000046	N=N+1	
1515	004644	000046	N	IDECTAPE WRITE BUFFER
1516	004646	177732	=N	
1517		000047	N=N+1	
1518	004650	000047	N	IDECTAPE WRITE BUFFER
1519	004652	177731	=N	
1520		000050	N=N+1	
1521	004654	000050	N	IDECTAPE WRITE BUFFER
1522	004656	177730	=N	
1523		000051	N=N+1	
1524	004660	000051	N	IDECTAPE WRITE BUFFER
1525	004662	177727	=N	
1526		000052	N=N+1	
1527	004664	000052	N	IDECTAPE WRITE BUFFER
1528	004666	177726	=N	
1529		000053	N=N+1	
1530	004670	000053	N	IDECTAPE WRITE BUFFER
1531	004672	177725	=N	
1532		000054	N=N+1	
1533	004674	000054	N	IDECTAPE WRITE BUFFER
1534	004676	177724	=N	
1535		000055	N=N+1	
1536	004700	000055	N	IDECTAPE WRITE BUFFER
1537	004702	177723	=N	
1538		000056	N=N+1	
1539	004704	000056	N	IDECTAPE WRITE BUFFER
1540	004706	177722	=N	
1541		000057	N=N+1	
1542	004710	000057	N	IDECTAPE WRITE BUFFER
1543	004712	177721	=N	
1544		000060	N=N+1	
1545	004714	000060	N	IDECTAPE WRITE BUFFER
1546	004716	177720	=N	
1547		000061	N=N+1	
1548	004720	000061	N	IDECTAPE WRITE BUFFER
1549	004722	177717	=N	
1550		000062	N=N+1	
1551	004724	000062	N	IDECTAPE WRITE BUFFER
1552	004726	177716	=N	
1553		000063	N=N+1	
1554	004730	000063	N	IDECTAPE WRITE BUFFER
1555	004732	177715	=N	
1556		000064	N=N+1	
1557	004734	000064	N	IDECTAPE WRITE BUFFER
1558	004736	177714	=N	

1559		000065	N=N+1	
1560	004740	000065	N	IDECTAPE WRITE BUFFER
1561	004742	177713	=N	
1562		000066	N=N+1	
1563	004744	000066	N	IDECTAPE WRITE BUFFER
1564	004746	177712	=N	
1565		000067	N=N+1	
1566	004750	000067	N	IDECTAPE WRITE BUFFER
1567	004752	177711	=N	
1568		000070	N=N+1	
1569	004754	000070	N	IDECTAPE WRITE BUFFER
1570	004756	177710	=N	
1571		000071	N=N+1	
1572	004760	000071	N	IDECTAPE WRITE BUFFER
1573	004762	177707	=N	
1574		000072	N=N+1	
1575	004764	000072	N	IDECTAPE WRITE BUFFER
1576	004766	177706	=N	
1577		000073	N=N+1	
1578	004770	000073	N	IDECTAPE WRITE BUFFER
1579	004772	177705	=N	
1580		000074	N=N+1	
1581	004774	000074	N	IDECTAPE WRITE BUFFER
1582	004776	177704	=N	
1583		000075	N=N+1	
1584	005000	000075	N	IDECTAPE WRITE BUFFER
1585	005002	177703	=N	
1586		000076	N=N+1	
1587	005004	000076	N	IDECTAPE WRITE BUFFER
1588	005006	177702	=N	
1589		000077	N=N+1	
1590	005010	000077	N	IDECTAPE WRITE BUFFER
1591	005012	177701	=N	
1592		000100	N=N+1	
1593	005014	000100	N	IDECTAPE WRITE BUFFER
1594	005016	177700	=N	
1595		000101	N=N+1	
1596		000100	N=N-1	
1597	005020	177700	=N	
1598	005022	000100	N	IDECTAPE WRITE BUFFER
1599		000077	N=N-1	
1600	005024	177701	=N	
1601	005026	000077	N	IDECTAPE WRITE BUFFER
1602		000076	N=N+1	
1603	005030	177702	=N	
1604	005032	000076	N	IDECTAPE WRITE BUFFER
1605		000075	N=N-1	
1606	005034	177703	=N	
1607	005036	000075	N	IDECTAPE WRITE BUFFER
1608		000074	N=N-1	
1609	005040	177704	=N	
1610	005042	000074	N	IDECTAPE WRITE BUFFER
1611		000073	N=N+1	
1612	005044	177705	=N	

1613	005046	000073	N	IDEC TAPE WRITE BUFFER
1614		000072	NNN=1	
1615	005050	177706	=N	
1616	005052	000072	N	IDEC TAPE WRITE BUFFER
1617		000071	NNN=1	
1618	005054	177707	=N	
1619	005056	000071	N	IDEC TAPE WRITE BUFFER
1620		000070	NNN=1	
1621	005060	177710	=N	
1622	005062	000070	N	IDEC TAPE WRITE BUFFER
1623		000067	NNN=1	
1624	005064	177711	=N	
1625	005066	000067	N	IDEC TAPE WRITE BUFFER
1626		000066	NNN=1	
1627	005070	177712	=N	
1628	005072	000066	N	IDEC TAPE WRITE BUFFER
1629		000065	NNN=1	
1630	005074	177713	=N	
1631	005076	000065	N	IDEC TAPE WRITE BUFFER
1632		000064	NNN=1	
1633	005100	177714	=N	
1634	005102	000064	N	IDEC TAPE WRITE BUFFER
1635		000063	NNN=1	
1636	005104	177715	=N	
1637	005106	000063	N	IDEC TAPE WRITE BUFFER
1638		000062	NNN=1	
1639	005110	177716	=N	
1640	005112	000062	N	IDEC TAPE WRITE BUFFER
1641		000061	NNN=1	
1642	005114	177717	=N	
1643	005116	000061	N	IDEC TAPE WRITE BUFFER
1644		000060	NNN=1	
1645	005120	177720	=N	
1646	005122	000060	N	IDEC TAPE WRITE BUFFER
1647		000057	NNN=1	
1648	005124	177721	=N	
1649	005126	000057	N	IDEC TAPE WRITE BUFFER
1650		000056	NNN=1	
1651	005130	177722	=N	
1652	005132	000056	N	IDEC TAPE WRITE BUFFER
1653		000055	NNN=1	
1654	005134	177723	=N	
1655	005136	000055	N	IDEC TAPE WRITE BUFFER
1656		000054	NNN=1	
1657	005140	177724	=N	
1658	005142	000054	N	IDEC TAPE WRITE BUFFER
1659		000053	NNN=1	
1660	005144	177725	=N	
1661	005146	000053	N	IDEC TAPE WRITE BUFFER
1662		000052	NNN=1	
1663	005150	177726	=N	
1664	005152	000052	N	IDEC TAPE WRITE BUFFER
1665		000051	NNN=1	
1666	005154	177727	=N	

1667	005156	000051	N	IDEC TAPE WRITE BUFFER
1668		000050	NNN=1	
1669	005160	177730	=N	
1670	005162	000050	N	IDEC TAPE WRITE BUFFER
1671		000047	NNN=1	
1672	005164	177731	=N	
1673	005166	000047	N	IDEC TAPE WRITE BUFFER
1674		000046	NNN=1	
1675	005170	177732	=N	
1676	005172	000046	N	IDEC TAPE WRITE BUFFER
1677		000045	NNN=1	
1678	005174	177733	=N	
1679	005176	000045	N	IDEC TAPE WRITE BUFFER
1680		000044	NNN=1	
1681	005200	177734	=N	
1682	005202	000044	N	IDEC TAPE WRITE BUFFER
1683		000043	NNN=1	
1684	005204	177735	=N	
1685	005206	000043	N	IDEC TAPE WRITE BUFFER
1686		000042	NNN=1	
1687	005210	177736	=N	
1688	005212	000042	N	IDEC TAPE WRITE BUFFER
1689		000041	NNN=1	
1690	005214	177737	=N	
1691	005216	000041	N	IDEC TAPE WRITE BUFFER
1692		000040	NNN=1	
1693	005220	177740	=N	
1694	005222	000040	N	IDEC TAPE WRITE BUFFER
1695		000037	NNN=1	
1696	005224	177741	=N	
1697	005226	000037	N	IDEC TAPE WRITE BUFFER
1698		000036	NNN=1	
1699	005230	177742	=N	
1700	005232	000036	N	IDEC TAPE WRITE BUFFER
1701		000035	NNN=1	
1702	005234	177743	=N	
1703	005236	000035	N	IDEC TAPE WRITE BUFFER
1704		000034	NNN=1	
1705	005240	177744	=N	
1706	005242	000034	N	IDEC TAPE WRITE BUFFER
1707		000033	NNN=1	
1708	005244	177745	=N	
1709	005246	000033	N	IDEC TAPE WRITE BUFFER
1710		000032	NNN=1	
1711	005250	177746	=N	
1712	005252	000032	N	IDEC TAPE WRITE BUFFER
1713		000031	NNN=1	
1714	005254	177747	=N	
1715	005256	000031	N	IDEC TAPE WRITE BUFFER
1716		000030	NNN=1	
1717	005260	177750	=N	
1718	005262	000030	N	IDEC TAPE WRITE BUFFER
1719		000027	NNN=1	
1720	005264	177751	=N	

```

1721 005266 000027 N /DEC TAPE WRITE BUFFER
1722 000026 N=N-1
1723 005270 177752 =N
1724 005272 000026 N /DEC TAPE WRITE BUFFER
1725 000025 N=N-1
1726 005274 177753 =N
1727 005276 000025 N /DEC TAPE WRITE BUFFER
1728 000024 N=N-1
1729 005300 177754 =N
1730 005302 000024 N /DEC TAPE WRITE BUFFER
1731 000023 N=N-1
1732 005304 177755 =N
1733 005306 000023 N /DEC TAPE WRITE BUFFER
1734 000022 N=N-1
1735 005310 177756 =N
1736 005312 000022 N /DEC TAPE WRITE BUFFER
1737 000021 N=N-1
1738 005314 177757 =N
1739 005316 000021 N /DEC TAPE WRITE BUFFER
1740 000020 N=N-1
1741 005320 177760 =N
1742 005322 000020 N /DEC TAPE WRITE BUFFER
1743 000017 N=N-1
1744 005324 177761 =N
1745 005326 000017 N /DEC TAPE WRITE BUFFER
1746 000016 N=N-1
1747 005330 177762 =N
1748 005332 000016 N /DEC TAPE WRITE BUFFER
1749 000015 N=N-1
1750 005334 177763 =N
1751 005336 000015 N /DEC TAPE WRITE BUFFER
1752 000014 N=N-1
1753 005340 177764 =N
1754 005342 000014 N /DEC TAPE WRITE BUFFER
1755 000013 N=N-1
1756 005344 177765 =N
1757 005346 000013 N /DEC TAPE WRITE BUFFER
1758 000012 N=N-1
1759 005350 177766 =N
1760 005352 000012 N /DEC TAPE WRITE BUFFER
1761 000011 N=N-1
1762 005354 177767 =N
1763 005356 000011 N /DEC TAPE WRITE BUFFER
1764 000010 N=N-1
1765 005360 177770 =N
1766 005362 000010 N /DEC TAPE WRITE BUFFER
1767 000007 N=N-1
1768 005364 177771 =N
1769 005366 000007 N /DEC TAPE WRITE BUFFER
1770 000006 N=N-1
1771 005370 177772 =N
1772 005372 000006 N /DEC TAPE WRITE BUFFER
1773 000005 N=N-1
1774 005374 177773 =N
    
```

```

1775 005376 000005 N /DEC TAPE WRITE BUFFER
1776 000004 N=N-1
1777 005400 177774 =N
1778 005402 000004 N /DEC TAPE WRITE BUFFER
1779 000003 N=N-1
1780 005404 177775 =N
1781 005406 000003 N /DEC TAPE WRITE BUFFER
1782 000002 N=N-1
1783 005410 177776 =N
1784 005412 000002 N /DEC TAPE WRITE BUFFER
1785 000001 N=N-1
1786 005414 177777 =N
1787 005416 000001 N /DEC TAPE WRITE BUFFER
1788 000000 N
1789 005420 010701 BEGINX: MOV PC,R1 /SET UP R1 TO SELECT CURBNK
1790 005422 042701 017777 BIC #17777,R1
1791 005426 042737 160000 000034 BIC #160000,##34 /SET SCOPE RET TO CURRENT BANK
1792 005430 050137 000034 BIS R1,##34
1793 005440 000301 SWAB R1
1794 005442 006201 ASR R1
1795 005444 006201 ASR R1
1796 005446 010137 000570 MOV R1,##CURBNK
1797 000000 N
1798 000000 N
1799 000000 N
1800 000000 N
1801 005452 010767 007262 /SBTTL BACKGROUND CPU TESTS
1802 005456 062767 000016 007254 /BINARY INSTRUCTIONS
1803 005464 012767 000400 007242 /INDEX, AND INDIRECT TEST OF PDP-11
1804 000000 N
1805 005472 012700 177770 BEGIN: MOV PC,RETURN /FOR SCOPING - SETUP ADDRESS OF BEGIN: IN
1806 005476 026027 014752 125252 ADD #16,RETURN /THIS BANK THRU CURRENT ASR
1807 005504 001401 MOV #400,ICOUNT /ITERATION COUNT
1808 005506 104006 /TEST COMPARE INSTRUCTION INDEXED
1809 005510 104400 /MINUS 10 TO REG 0
1810 000000 N / (A INDEX BY MINUS 10) TO #125252
1811 005512 012700 000010 MOV #10,X0 /COMPARE WITH INDEX FAILED
1812 005516 022760 052525 014752 CMP #052525,A(0)
1813 005524 001401 BEQ ,+4
1814 005526 104006 HLT
1815 005530 104400 SCOPE
1816 000000 N
1817 005532 012700 177770 MOV #10,X0
1818 005536 026060 014752 014752 CMP A(0),A(0)
1819 005544 001401 BEQ ,+4
1820 005546 104006 HLT
1821 005550 104400 SCOPE
1822 000000 N
1823 005552 012700 000010 MOV #10,X0
1824 005556 026060 014752 014752 CMP A(0),A(0)
1825 005564 001401 BEQ ,+4
1826 005566 104006 HLT
1827 005570 104400 SCOPE
1828 000000 N
    
```

1829	005572	012700	177774		MOV	#=4,X0
1830	005576	012701	000010		MOV	#=10,X1
1831	005602	020061	014752	014752	CMR	A(0),A(1)
1832	005610	001401			BEQ	.+4
1833	005612	104006			HLT	
1834	005614	104400			SCOPE	
1835						
1836	005616	012700	177774		MOV	#=4,X0
1837	005622	012701	000010		MOV	#10,X1
1838	005626	020160	014752	014752	CMR	A(1),A(0)
1839	005634	001401			BEQ	.+4
1840	005636	104006			HLT	
1841	005640	104400			SCOPE	
1842						
1843						
1844						
1845	005642	012700	177770		MOV	#=10,X0
1846	005646	010067	014752	007120	MOV	A(0),TEMP
1847	005654	026727	007114	125252	CMR	TEMP,#125252
1848	005662	001401			BEQ	.+4
1849	005664	104006			HLT	
1850	005666	104400			SCOPE	
1851						
1852	005670	012700	177770		MOV	#=10,X0
1853	005674	012760	125252	014774	MOV	#125252,TEMP(0)
1854	005702	023727	014764	125252	CMR	#C,#125252
1855	005710	001401			BEQ	.+4
1856	005712	104006			HLT	
1857	005714	104400			SCOPE	
1858						
1859						
1860	005716	012767	177777	007050	MOV	#=1,TEMP
1861	005724	012700	177770		MOV	#=10,X0
1862	005730	046067	014752	007036	BIC	A(0),TEMP
1863	005736	026727	007032	052525	CMR	TEMP,#052525
1864	005744	001401			BEQ	.+4
1865	005746	104006			HLT	
1866	005750	104400			SCOPE	
1867						
1868	005752	012700	177770		MOV	#=10,X0
1869	005756	012767	177777	007000	MOV	#=1,TEMP-10
1870	005764	042767	052525	006772	BIC	#052525,TEMP-10
1871	005772	026727	006766	125252	CMR	TEMP-10,#125252
1872	006000	001401			BEQ	.+4
1873	006002	104006			HLT	
1874	006004	104400			SCOPE	
1875						
1876	006006	012737	125252	014774	MOV	#125252,##TEMP
1877	006014	012700	177770		MOV	#=10,X0
1878	006020	166760	006716	015004	SUB	B,TEMP+10(0)
1879	006026	001401			BEQ	.+4
1880	006030	104006			HLT	
1881	006032	104400			SCOPE	
1882						

1883	006034	012737	052525	014774	MOV	#052525,##TEMP
1884	006042	012700	000010		MOV	#10,X0
1885	006046	166760	006710	014764	SUB	A+10,C(0)
1886	006054	001401			BEQ	.+4
1887	006056	104006			HLT	
1888	006060	104400			SCOPE	
1889						
1890						
1891						
1892	006062	012737	177777	014774	MOV	#=1,##TEMP
1893	006070	012700	000010		MOV	#=10,X0
1894	006074	005060	014764		CLR	C(0)
1895	006100	005737	014774		TST	#TEMP
1896	006104	001401			BEQ	.+4
1897	006106	104006			HLT	
1898	006110	104400			SCOPE	
1899						
1900	006112	012737	177777	014774	MOV	#=1,##TEMP
1901	006120	012700	000010		MOV	#10,X0
1902	006124	005160	014764		COM	C(0)
1903	006130	005737	014774		TST	#TEMP
1904	006134	001401			BEQ	.+4
1905	006136	104006			HLT	
1906	006140	104400			SCOPE	
1907						
1908	006142	012737	177777	014774	MOV	#=1,##TEMP
1909	006150	012700	177770		MOV	#=10,X0
1910	006154	005260	015004		INC	D(0)
1911	006160	005737	014774		TST	#TEMP
1912	006164	001401			BEQ	.+4
1913	006166	104006			HLT	
1914	006170	104400			SCOPE	
1915						
1916	006172	012737	000001	014774	MOV	#1,##TEMP
1917	006200	012700	177770		MOV	#=10,X0
1918	006204	005360	015004		DEC	D(0)
1919	006210	005737	014774		TST	#TEMP
1920	006214	001401			BEQ	.+4
1921	006216	104006			HLT	
1922	006220	104400			SCOPE	
1923						
1924	006222	012737	000001	014774	MOV	#1,##TEMP
1925	006230	012700	000010		MOV	#10,X0
1926	006234	005360	014764		DEC	C(0)
1927	006240	005737	014774		TST	#TEMP
1928	006244	001401			BEQ	.+4
1929	006246	104006			HLT	
1930	006250	104400			SCOPE	
1931						
1932	006252	012737	000001	014774	MOV	#1,##TEMP
1933	006260	012700	177770		MOV	#=10,X0
1934	006264	005660	015004		NEG	D(0)
1935	006270	022737	177777	014774	CMR	#=1,##TEMP
1936	006276	001401			BEQ	.+4

```

1937 006300 104006          HLT
1938 006302 104400          SCOPE
1939
1940 006304 012737 000001 014774      MOV    #1,TEMP
1941 006312 012700 000010          MOV    #+10,X0
1942 006316 005460 014764          NEG    C(0)
1943 006322 022737 177777 014774      CMP    #-1,TEMP
1944 006330 001401          BEQ    ,+4
1945 006332 104006          HLT
1946 006334 104400          SCOPE
1947
1948 006336 012737 177777 014774      MOV    #-1,TEMP
1949 006344 012700 177770          MOV    #-10,X0
1950 006350 000261          SEC
1951 006352 005560 015004          ADC    D(0)
1952 006356 005737 014774          TST   TEMP
1953 006362 001401          BEQ    ,+4
1954 006364 104006          HLT
1955 006366 104400          SCOPE
1956
1957 006370 012737 177777 014774      MOV    #-1,TEMP
1958 006376 012700 000010          MOV    #+10,X0
1959 006402 000261          SEC
1960 006404 005560 014764          ADC    C(0)
1961 006410 005737 014774          TST   TEMP
1962 006414 001401          BEQ    ,+4
1963 006416 104006          HLT
1964 006420 104400          SCOPE
1965
1966 006422 012737 000001 014774      MOV    #1,TEMP
1967 006430 012700 177770          MOV    #-10,X0
1968 006434 000261          SEC
1969 006436 005560 015004          SBC    D(0)
1970 006442 005737 014774          TST   TEMP
1971 006446 001401          BEQ    ,+4
1972 006450 104006          HLT
1973 006452 104400          SCOPE
1974
1975 006454 012737 000001 014774      MOV    #1,TEMP
1976 006462 012700 000010          MOV    #+10,X0
1977 006466 000261          SEC
1978 006470 005560 014764          SBC    C(0)
1979 006474 005737 014774          TST   TEMP
1980 006500 001401          BEQ    ,+4
1981 006502 104006          HLT
1982 006504 104400          SCOPE
1983
1984                                JTEST JMP INDIRECT
1985 006506 010700          MOV    X7,X0
1986 006510 062700 000010          ADD    #10,X0
1987 006514 000110          JMP    #X0
1988 006516 104006          HLT
1989 006520 000240          NOP
1990 006522 104400          SCOPE
    
```

```

1991
1992 006524 010700          MOV    X7,X0
1993 006526 062700 000010          ADD    #10,X0
1994 006532 000110          JMP    #X0
1995 006534 104006          HLT
1996 006536 000240          NOP
1997 006540 104400          SCOPE
1998
1999                                JTEST INDIRECT ADDRESSING
2000                                JTEST COMPARE INSTRUCTION
2001 006542 023727 014742 125252      CMP    #0B,#125252
2002 006550 001401          BEQ    ,+4
2003 006552 104006          HLT
2004 006554 104400          SCOPE
2005
2006 006556 022737 125252 014742      CMP    #125252,#0B
2007 006564 001401          BEQ    ,+4
2008 006566 104006          HLT
2009 006570 104400          SCOPE
2010
2011 006572 023737 014742 014742      CMP    #0B,#0B
2012 006600 001401          BEQ    ,+4
2013 006602 104006          HLT
2014 006604 104400          SCOPE
2015
2016                                JTEST MOVE INSTRUCTIONS
2017 006606 013700 014742          MOV    #0B,X0
2018 006612 022700 125252          CMP    #125252,X0
2019 006616 001401          BEQ    ,+4
2020 006620 104006          HLT
2021 006622 104400          SCOPE
2022
    
```

2023	006624	012737	125252	014774	MOV	#125252, #TEMP
2024	006632	023737	014742	014774	CMP	#B, #TEMP
2025	006640	001401			BEQ	, +4
2026	006642	104006			HLT	
2027	006644	104000			SCOPE	
2028						
2029	006646	013737	014742	014764	MOV	#B, #C
2030	006654	023737	014742	014764	CMP	#B, #C
2031	006662	001401			BEQ	, +4

2032	006664	104006			HLT	
------	--------	--------	--	--	-----	--

2033	006666	104400			SCOPE
2034					
2035					TEST BIC INSTRUCTION INDIRECT
2036	006670	012700	177777		MOV #-1,X0

2037	006674	043700	014742		BIC #0,X0
2038	006700	020027	052525		CMP X0,#052525
2039	006704	001401			BEQ ,+4
2040	006706	104006			HLT
2041	006710	104400			SCOPE
2042					
2043	006712	012737	177777	014774	MOV #-1,##TEMP
2044	006720	042737	125252	014774	BIC #125252,##TEMP
2045	006726	022737	052525	014774	CMP #052525,##TEMP
2046	006734	001401			BEQ ,+4
2047	006736	104006			HLT
2048	006740	104400			SCOPE

```

2049
2050 006742 012737 177777 014764      MOV    #-1,##C
2051 006750 043737 014742 014764      BIC    ##B,##C
2052 006756 023727 014764 052525      CMP    ##C,#52525
2053 006764 001401                BEQ    .+4
2054 006766 104006                HLT
2055 006770 104400                SCOPE
2056
2057
2058 006772 012700 125252      ;TEST SUBTRACT INSTRUCTION
2059 006776 163700 014742      MOV    #125252,X0
2060 007002 020027 000000      SUB    ##B,X0
2061 007006 001401                CMP    X0,#0
2062 007010 104006                BEQ    .+4
2063 007012 104400                HLT
2064                SCOPE
2065 007014 012737 125252 014774      MOV    #125252,##TEMP
2066 007022 166737 005714 014774      SUB    B,##TEMP
2067 007030 001401                BEQ    .+4
2068 007032 104006                HLT
2069 007034 104400                SCOPE
2070
2071 007036 012767 125252 005730      MOV    #125252,TEMP
2072 007044 163767 014742 005722      SUB    ##B,TEMP
2073 007052 005767 005716      TST   TEMP
2074 007056 001401                BEQ    .+4
2075 007060 104006                HLT
2076 007062 104400                SCOPE
2077
2078                ;TEST ADD INDIRECT
2079 007064 005000                CLR    X0
2080 007066 063700 014742      ADD    ##B,X0
2081 007072 022700 125252      CMP    #125252,X0
2082 007076 001401                BEQ    .+4
2083 007100 104006                HLT
2084 007102 104400                SCOPE
2085
2086 007104 005037 014774                CLR    ##TEMP
2087 007110 062737 125252 014774      ADD    #125252,##TEMP
2088 007116 022737 125252 014774      CMP    #125252,##TEMP
2089 007124 001401                BEQ    .+4
2090 007126 104006                HLT
2091 007130 104400                SCOPE
2092
2093 007132 012737 125252 014774      MOV    #125252,##TEMP
2094 007140 067737 005614 014774      ADD    #A+0,##TEMP
2095 007146 023727 014774 177777      CMP    ##TEMP,#-1
2096 007154 001401                BEQ    .+4
2097 007156 104006                HLT
2098 007160 104400                SCOPE
2099
2100                ;TEST UNARYS INDIRECT
2101 007162 012737 177777 014774      MOV    #-1,##TEMP
2102 007170 005037 014774                CLR    ##TEMP

```

```

2103 007174 005737 014774                TST   ##TEMP
2104 007200 001401                BEQ    .+4
2105 007202 104006                HLT
2106 007204 104400                SCOPE
2107
2108 007206 012737 125252 014774      MOV    #125252,##TEMP
2109 007214 005137 014774                COM   ##TEMP
2110 007220 022737 052525 014774      CMP    #052525,##TEMP
2111 007226 001401                BEQ    .+4
2112 007230 104006                HLT
2113 007232 104400                SCOPE
2114
2115 007234 005037 014774                CLR    ##TEMP
2116 007240 005237 014774                INC   ##TEMP
2117 007244 022737 000001 014774      CMP    #1,##TEMP
2118 007252 001401                BEQ    .+4
2119 007254 104006                HLT
2120 007256 104400                SCOPE
2121
2122 007260 005037 014774                CLR    ##TEMP
2123 007264 005377 005506                DEC   ##TEMP+2
2124 007270 023727 014774 177777      CMP    ##TEMP,#-1
2125 007276 001401                BEQ    .+4
2126 007300 104006                HLT
2127 007302 104400                SCOPE
2128
2129 007304 012737 000001 014774      MOV    #1,##TEMP
2130 007312 005437 014774                NEG   ##TEMP
2131 007316 022737 177777 014774      CMP    #-1,##TEMP
2132 007324 001401                BEQ    .+4
2133 007326 104006                HLT
2134 007330 104400                SCOPE
2135
2136                ;TEST INDIRECT ADDRESSING WITH INDEXING
2137                ;TEST COMPARE INSTRUCTION
2138 007332 027727 005406 125252      CMP    #0+2,#125252
2139 007340 001401                BEQ    .+4
2140 007342 104006                HLT
2141 007344 104400                SCOPE
2142
2143 007346 022777 125252 005370      CMP    #125252,##B+2
2144 007354 001401                BEQ    .+4
2145 007356 104006                HLT
2146 007360 104400                SCOPE
2147
2148 007362 027777 005356 005354      CMP    ##B+2,##B+2
2149 007370 001401                BEQ    .+4
2150 007372 104006                HLT
2151 007374 104400                SCOPE
2152
2153                ;TEST MOVE INSTRUCTIONS
2154 007376 017700 005342      MOV    ##B+2,X0
2155 007402 022700 125252      CMP    #125252,X0
2156 007406 001401                BEQ    .+4

```

```

2157 007410 104006 HLT
2158 007412 104400 SCOPE
2159
2160 007414 012777 125252 005354 MOV #125252,TEMP+2
2161 007422 023737 014742 014774 CMP #0,TEMP
2162 007430 001401 BEQ ,+4
2163 007432 104006 HLT
2164 007434 104400 SCOPE
2165
2166 007436 017777 005302 005322 MOV #0+2,PC+2
2167 007444 023737 014742 014764 CMP #0,PC
2168 007452 001401 BEQ ,+4
2169 007454 104006 HLT
2170 007456 104400 SCOPE
2171
2172 JTEST BIC INSTRUCTION INDIRECT WITH INDEXING
2173 007460 012700 177777 MOV #-1,X0
2174 007464 047700 005254 BIC #0+2,X0
2175 007470 020027 052525 CMP X0,#52525
2176 007474 001401 BEQ ,+4
2177 007476 104006 HLT
2178 007500 104400 SCOPE
2179
2180 007502 012737 177777 014774 MOV #-1,TEMP
2181 007510 042777 125252 005260 BIC #125252,TEMP+2
2182 007516 022737 052525 014774 CMP #52525,TEMP
2183 007524 001401 BEQ ,+4
2184 007526 104006 HLT
2185 007530 104400 SCOPE
2186
2187 007532 012737 177777 014764 MOV #-1,PC
2188 007540 047777 005200 005220 BIC #0+2,PC+2
2189 007546 026737 005210 014764 CMP A+10,PC
2190 007554 001401 BEQ ,+4
2191 007556 104006 HLT
2192 007560 104400 SCOPE
2193
2194 007562 012700 125252 MOV #125252,X0
2195 007566 167700 005152 SUB #0+2,X0
2196 007572 020027 000000 CMP X0,#0
2197 007576 001401 BEQ ,+4
2198 007600 104006 HLT
2199 007602 104400 SCOPE
2200
2201 007604 012737 125252 014774 MOV #125252,TEMP
2202 007612 166777 005124 005156 SUB B,TEMP+2
2203 007620 001401 BEQ ,+4
2204 007622 104006 HLT
2205 007624 104400 SCOPE
2206
2207 007626 012737 125252 014774 MOV #125252,TEMP
2208 007634 167777 005104 005134 SUB #0+2,TEMP+2
2209 007642 005737 014774 TST TEMP
2210 007646 001401 BEQ ,+4

```

```

2211 007650 104006 HLT
2212 007652 104400 SCOPE
2213
2214 JTEST ADD INDIRECT WITH INDEXING
2215 007654 005000 CLR X0
2216 007656 067700 005062 ADD #0+2,X0
2217 007662 022700 125252 CMP #125252,X0
2218 007666 001401 BEQ ,+4
2219 007670 104006 HLT
2220 007672 104400 SCOPE
2221
2222 007674 005037 014774 CLR TEMP
2223 007700 062777 125252 005070 ADD #125252,TEMP+2
2224 007706 022737 125252 014774 CMP #125252,TEMP
2225 007714 001401 BEQ ,+4
2226 007716 104006 HLT
2227 007720 104400 SCOPE
2228
2229 007722 012737 125252 014774 MOV #125252,TEMP
2230 007730 067777 005024 005040 ADD #A+6,TEMP+2
2231 007736 023727 014774 177777 CMP TEMP,#-1
2232 007744 001401 BEQ ,+4
2233 007746 104006 HLT
2234 007750 104400 SCOPE
2235
2236 JTEST UNARYS INDIRECT WITH INDEXING
2237 007752 012737 177777 014774 MOV #-1,TEMP
2238 007760 005077 005012 CLR TEMP+2
2239 007764 003737 014774 TST TEMP
2240 007770 001401 BEQ ,+4
2241 007772 104006 HLT
2242 007774 104400 SCOPE
2243
2244 007776 012737 125252 014774 MOV #125252,TEMP
2245 010004 005177 004766 COM TEMP+2
2246 010010 022737 052525 014774 CMP #052525,TEMP
2247 010016 001401 BEQ ,+4
2248 010020 104006 HLT
2249 010022 104400 SCOPE
2250
2251 010024 005037 014774 CLR TEMP
2252 010030 005277 004742 INC TEMP+2
2253 010034 022737 000001 014774 CMP #1,TEMP
2254 010042 001401 BEQ ,+4
2255 010044 104006 HLT
2256 010046 104400 SCOPE
2257
2258 010050 005037 014774 CLR TEMP
2259 010054 005377 004716 DEC TEMP+2
2260 010060 023727 014776 177777 CMP TEMP,#-1
2261 010066 001401 BEQ ,+4
2262 010070 104006 HLT
2263 010072 104400 SCOPE
2264

```


2265	010074	012737	000001	014774	MOV	#1, #TEMP
2266	010102	005477	004670		NEG	#TEMP+2
2267	010106	022737	177777	014774	CMP	#-1, #TEMP
2268	010114	001401			BEQ	,+4
2269	010116	104006			HLT	
2270	010120	104400			SCOPE	
2271						
2272	010122	012737	177777	014774	MOV	#-1, #TEMP
2273	010130	000261			SEC	
2274	010132	005577	004640		ADC	#TEMP+2
2275	010136	005737	014774		TST	#TEMP
2276	010142	001401			BEQ	,+4
2277	010144	104006			HLT	
2278	010146	104400			SCOPE	
2279						
2280	010150	012737	000001	014774	MOV	#1, #TEMP
2281	010156	000261			SEC	
2282	010160	005677	004612		SBC	#TEMP+2
2283	010164	005737	014774		TST	#TEMP
2284	010170	001401			BEQ	,+4
2285	010172	104006			HLT	
2286	010174	104400			SCOPE	
2287						
2288						
2289	010176	012700	177772		MOV	#-6, X0
2290	010202	027027	014752	125252	CMP	#A(0), #125252
2291	010210	001401			BEQ	,+4
2292	010212	104006			HLT	
2293	010214	104400			SCOPE	
2294						
2295	010216	012700	177772		MOV	#-6, X0
2296	010222	022770	125252	014752	CMP	#125252, #A(0)
2297	010230	001401			BEQ	,+4
2298	010232	104006			HLT	
2299	010234	104400			SCOPE	
2300						
2301	010236	012700	177772		MOV	#-6, X0
2302	010242	012701	000002		MOV	#+2, X1
2303	010246	027071	014752	014752	CMP	#A(0), #A(1)
2304	010254	001401			BEQ	,+4
2305	010256	104006			HLT	
2306	010260	104400			SCOPE	
2307						
2308						
2309	010262	012700	000006		MOV	#+6, X0
2310	010266	012767	177777	004500	MOV	#-1, TEMP
2311	010274	047067	014752	004472	BIC	#A(0), TEMP
2312	010302	022767	125252	004464	CMP	#125252, TEMP
2313	010310	001401			BEQ	,+4
2314	010312	104006			HLT	
2315	010314	104400			SCOPE	
2316						
2317	010316	012700	177772		MOV	#-6, X0
2318	010322	012737	177777	014764	MOV	#-1, #C

ITEST OF COMBINED INDEXING AND INDIRECT

ITEST BIC INSTRUCTION

2319	010330	042770	125252	014774	BIC	#125252, #TEMP(0)
2320	010336	023727	014764	052525	CMP	#C, #052525
2321	010344	001401			BEQ	,+4
2322	010346	104006			HLT	
2323	010350	104400			SCOPE	
2324						
2325	010352	012737	177777	014764	MOV	#-1, #C
2326	010360	012700	177772		MOV	#-6, X0
2327	010364	012701	177772		MOV	#-6, X1
2328	010370	047071	014752	014774	BIC	#A(0), #TEMP(1)
2329	010376	022737	052525	014764	CMP	#052525, #C
2330	010404	001401			BEQ	,+4
2331	010406	104006			HLT	
2332	010410	104400			SCOPE	
2333						
2334						
2335						
2336						
2337	010412	012700	177770		MOV	#-10, X0
2338	010416	126027	014752	000252	CMPB	A(0), #000252
2339	010424	001401			BEQ	,+4
2340	010426	104006			HLT	
2341	010430	104400			SCOPE	
2342						
2343	010432	012700	177770		MOV	#-10, X0
2344	010436	122760	000252	014752	CMPB	#000252, A(0)
2345	010444	001401			BEQ	,+4
2346	010446	104006			HLT	
2347	010450	104400			SCOPE	
2348						
2349	010452	012700	000010		MOV	#10, X0
2350	010456	126027	014752	000125	CMPB	A(0), #000125
2351	010464	001401			BEQ	,+4
2352	010466	104006			HLT	
2353	010470	104400			SCOPE	
2354						
2355	010472	012700	000010		MOV	#10, X0
2356	010476	122760	000125	014752	CMPB	#000125, A(0)
2357	010504	001401			BEQ	,+4
2358	010506	104006			HLT	
2359	010510	104400			SCOPE	
2360						
2361	010512	012700	177770		MOV	#-10, X0
2362	010516	126060	014752	014752	CMPB	A(0), A(0)
2363	010524	001401			BEQ	,+4
2364	010526	104006			HLT	
2365	010530	104400			SCOPE	
2366						
2367	010532	012700	000010		MOV	#+10, X0
2368	010536	126060	014752	014752	CMPB	A(0), A(0)
2369	010544	001401			BEQ	,+4
2370	010546	104006			HLT	
2371	010550	104400			SCOPE	
2372						

IBINARY INSTRUCTIONS
 IINDEX, AND INDIRECT TEST OF POP=11
 ITEST COMPARE INSTRUCTION INDEXED

I MINUS 10 TO REG 0
 I(A INDEX BY MINUS 10) TO #125252

I COMPARE WITH INDEX FAILED

I FOR INDEX
 I A INDEXED

I INDEX

```

2373 010552 012700 177770      MOV    #-10,X0
2374 010556 012701 000004      MOV    #+4,X1
2375 010562 126061 014752 014752  CMPB  A(0),A(1)
2376 010570 001401      BEQ    .+4
2377 010572 104006      HLT
2378 010574 104400      SCOPE
2379
2380 010576 126160 014752 014752  CMPB  A(1),A(0)
2381 010604 001401      BEQ    .+4
2382 010606 104006      HLT
2383 010610 104400      SCOPE
2384
2385 010612 012700 177774      MOV    #-4,X0
2386 010616 012701 000010      MOV    #+10,X1
2387 010622 126061 014752 014752  CMPB  A(0),A(1)
2388 010630 001401      BEQ    .+4
2389 010632 104006      HLT
2390 010634 104400      SCOPE
2391
2392 010636 012700 177774      MOV    #-4,X0
2393 010642 012701 000010      MOV    #10,X1
2394 010646 126160 014752 014752  CMPB  A(1),A(0)
2395 010654 001401      BEQ    .+4
2396 010656 104006      HLT
2397 010660 104400      SCOPE
2398
2399
2400 010662 012700 177770      JTEST MOVE INSTRUCTION FOR INDEX
2401 010666 116067 014752 004100      MOV    #-10,X0
2402 010674 126727 004074 000252      MOVB  A(0),TEMP
2403 010702 001401      CMPB  TEMP,#000252
2404 010704 104006      BEQ    .+4
2405 010706 104400      HLT
2406 010710 012700 000010      SCOPE
2407 010714 116067 014752 004052      MOV    #+10,X0
2408 010722 126727 004046 000125      MOVB  A(0),TEMP
2409 010730 001401      CMPB  TEMP,#000125
2410 010732 104006      BEQ    .+4
2411 010734 104400      HLT
2412 010736 104400      SCOPE
2413
2414 010738 012700 177770      MOV    #-10,X0
2415 010742 112760 125252 014774      MOVB  #125252,TEMP(0)
2416 010750 123727 014764 125252      CMPB  #C,#125252
2417 010756 001401      BEQ    .+4
2418 010760 104006      HLT
2419 010762 104400      SCOPE
2420
2421 010764 012700 000010      MOV    #+10,X0
2422 010770 112760 052525 014774      MOVB  #052525,TEMP(0)
2423 010776 123727 015004 052525      CMPB  #TEMP+10,#052525
2424 011004 001401      BEQ    .+4
2425 011006 104006      HLT
2426 011010 104400      SCOPE

```

```

2427
2428
2429 011012 012767 177777 003754      JTEST BIC INSTRUCTION FOR INDEXING
2430 011020 012700 177770      MOV    #-1,TEMP
2431 011024 146067 014752 003742      MOV    #-10,X0
2432 011032 126727 003736 177525      BICB  A(0),TEMP
2433 011040 001401      CMPB  TEMP,#177525
2434 011042 104006      BEQ    .+4
2435 011044 104400      HLT
2436 011046 104400      SCOPE
2437 011046 012767 177777 003720      MOV    #-1,TEMP
2438 011054 012700 000010      MOV    #10,X0
2439 011060 146067 014752 003706      BICB  A(0),TEMP
2440 011066 126727 003702 007652      CMPB  TEMP,#007652
2441 011074 001401      BEQ    .+4
2442 011076 104006      HLT
2443 011100 104400      SCOPE
2444
2445 011102 012737 177777 015004      MOV    #-1,#TEMP+10
2446 011110 012700 000010      MOV    #10,X0
2447 011114 142760 125252 014774      BICB  #125252,TEMP(0)
2448 011122 123727 015004 002525      CMPB  #TEMP+10,#2525
2449 011130 001401      BEQ    .+4
2450 011132 104006      HLT
2451 011134 104400      SCOPE
2452
2453 011136 012700 177770      MOV    #-10,X0
2454 011142 012767 177777 003614      MOV    #-1,TEMP-10
2455 011150 142767 052525 003606      BICB  #052525,TEMP-10
2456 011156 126727 003602 125252      CMPB  TEMP-10,#125252
2457 011164 001401      BEQ    .+4
2458 011166 104006      HLT
2459 011170 104400      SCOPE
2460
2461
2462 011172 012737 177777 014774      JTEST UNARYS INDEXED
2463 011200 012700 177770      MOV    #-1,#TEMP
2464 011204 105060 015004      MOV    #-10,X0
2465 011210 105737 014774      CLRB  D(0)
2466 011214 001401      TSTB #TEMP
2467 011216 104006      BEQ    .+4
2468 011220 104400      HLT
2469 011222 104400      SCOPE
2470 011222 012737 177777 014774      MOV    #-1,#TEMP
2471 011230 012700 177770      MOV    #-10,X0
2472 011234 105060 015004      CLRB  D(0)
2473 011240 023727 014774 177400      CMP  #TEMP,#177400
2474 011246 001401      BEQ    .+4
2475 011250 104006      HLT
2476 011252 104400      SCOPE
2477
2478 011254 012737 177777 014774      MOV    #-1,#TEMP
2479 011262 012700 177771      MOV    #-7,X0
2480 011266 105060 015004      CLRB  D(0)

```

2481	011272	023727	014774	000377	CHP	##TEMP,#000377
2482	011300	001401			BEQ	.,+4
2483	011302	104006			HLT	
2484	011300	104400			SCOPE	
2485						
2486	011306	012737	177777	014774	MOV	#-1,##TEMP
2487	011314	012700	000010		MOV	##-10,%0
2488	011320	105600	014764		CLRB	C(0)
2489	011324	105737	014774		TSTB	##TEMP
2490	011330	001401			BEQ	.,+4
2491	011332	104006			HLT	
2492	011334	104400			SCOPE	
2493						
2494	011336	012737	177777	014774	MOV	#-1,##TEMP
2495	011344	012700	177770		MOV	##-10,%0
2496	011350	105100	015004		COMB	D(0)
2497	011354	105737	014774		TSTB	##TEMP
2498	011360	001401			BEQ	.,+4
2499	011362	104006			HLT	
2500	011364	104400			SCOPE	
2501						
2502	011366	012737	177777	014774	MOV	#-1,##TEMP
2503	011374	012700	000010		MOV	##-10,%0
2504	011400	105200	014764		INCB	C(0)
2505	011404	105737	014774		TSTB	##TEMP
2506	011410	001401			BEQ	.,+4
2507	011412	104006			HLT	
2508	011414	104400			SCOPE	
2509						
2510	011416	012737	000001	014774	MOV	#1,##TEMP
2511	011424	012700	177770		MOV	##-10,%0
2512	011430	105300	015004		DECB	D(0)
2513	011434	105737	014774		TSTB	##TEMP
2514	011440	001401			BEQ	.,+4
2515	011442	104006			HLT	
2516	011444	104400			SCOPE	
2517						
2518	011446	012737	000001	014774	MOV	#1,##TEMP
2519	011454	012700	000010		MOV	##-10,%0
2520	011460	105400	014764		NEGB	C(0)
2521	011464	023727	014774	000377	CHP	##TEMP,#377
2522	011472	001401			BEQ	.,+4
2523	011474	104006			HLT	
2524	011476	104400			SCOPE	
2525						
2526	011500	012737	177777	014774	MOV	#-1,##TEMP
2527	011506	012700	177770		MOV	##-10,%0
2528	011512	000261			SEC	
2529	011514	105500	015004		ADCB	D(0)
2530	011520	023727	014774	177400	CHP	##TEMP,#177400
2531	011526	001401			BEQ	.,+4
2532	011530	104006			HLT	
2533	011532	104400			SCOPE	
2534						

2535	011534	012737	000001	014774	MOV	#1,##TEMP
2536	011542	012700	000010		MOV	##-10,%0
2537	011546	000261			SEC	
2538	011550	105600	014764		SBCB	C(0)
2539	011554	005737	014774		TST	##TEMP
2540	011560	001401			BEQ	.,+4
2541	011562	104006			HLT	
2542	011564	104400			SCOPE	
2543						
2544						
2545						
2546	011566	123727	014742	000252	JTEST COMPARE INSTRUCTION	##0,##000252
2547	011574	001401			BEQ	.,+4
2548	011576	104006			HLT	
2549	011600	104400			SCOPE	
2550						
2551	011602	122737	125252	014742	CMPB	#125252,##0
2552	011610	001401			BEQ	.,+4
2553	011612	104006			HLT	
2554	011614	104400			SCOPE	
2555						
2556						
2557	011616	113700	014742		JTEST MOVE INSTRUCTIONS	##0,%0
2558	011622	122700	000252		CMPB	##000252,%0
2559	011626	001401			BEQ	.,+4
2560	011630	104006			HLT	
2561	011632	104400			SCOPE	
2562						
2563	011634	112737	125252	014774	MOVB	#125252,##TEMP
2564	011642	126737	003074	014774	CMPB	##,##TEMP
2565	011650	001401			BEQ	.,+4
2566	011652	104006			HLT	
2567	011654	104400			SCOPE	
2568						
2569						
2570	011656	012737	177777	014774	JTEST UNARYS INDIRECT	##-1,##TEMP
2571	011664	105037	014774		CLRB	##TEMP
2572	011670	023727	014774	177400	CMP	##TEMP,#177400
2573	011676	001401			BEQ	.,+4
2574	011700	104006			HLT	
2575	011702	104400			SCOPE	
2576						
2577	011704	012737	125252	014774	MOV	#125252,##TEMP
2578	011712	105137	014775		COMB	##TEMP+1
2579	011716	022737	052652	014774	CMP	##052652,##TEMP
2580	011724	001401			BEQ	.,+4
2581	011726	104006			HLT	
2582	011730	104400			SCOPE	
2583						
2584	011732	005037	014774		CLR	##TEMP
2585	011736	105237	014775		INCB	##TEMP+1
2586	011742	022737	000400	014774	CMP	##400,##TEMP
2587	011750	001401			BEQ	.,+4
2588	011752	104006			HLT	

```

2589 011754 104400 SCOPE
2590
2591 011756 005037 014774 CLR #TEMP
2592 011762 105377 003010 DECB #TEMP+2
2593 011766 023727 014774 000377 CMP #TEMP,#377
2594 011774 001401 BEQ ,+4
2595 011776 104006 HLT
2596 012000 104400 SCOPE
2597
2598 012002 005037 014774 CLR #TEMP
2599 012006 112737 000001 014775 MOVB #1,#TEMP+1
2600 012014 105437 014775 NEGB #TEMP+1
2601 012020 022737 177400 014774 CMP #177400,#TEMP
2602 012026 001401 BEQ ,+4
2603 012030 104006 HLT
2604 012032 104400 SCOPE
2605
2606 JTEST INDIRECT ADDRESSING WITH INDEXING
2607 JTEST COMPARE INSTRUCTION
2608
2609 012034 122777 125252 002702 CMPB #125252,#B+2
2610 012042 001401 BEQ ,+4
2611 012044 104006 HLT
2612 012046 104400 SCOPE
2613
2614 012050 127777 002670 002666 CMPB #B+2,#B+2
2615 012056 001401 BEQ ,+4
2616 012060 104006 HLT
2617 012062 104400 SCOPE
2618
2619 JTEST MOVE INSTRUCTIONS
2620 012064 117700 002654 MOVB #B+2,X0
2621 012070 127700 125252 CMPB #125252,X0
2622 012074 001401 BEQ ,+4
2623 012076 104006 HLT
2624 012100 104400 SCOPE
2625
2626 012102 112777 125252 002666 MOVB #125252,#TEMP+2
2627 012110 126737 002626 014774 CMPB #,#TEMP
2628 012116 001401 BEQ ,+4
2629 012120 104006 HLT
2630 012122 104400 SCOPE
2631
2632 012124 117777 002614 002634 MOVB #B+2,#C+2
2633 012132 126737 002600 014764 CMPB #,#C
2634 012140 001401 BEQ ,+4
2635 012142 104006 HLT
2636 012144 104400 SCOPE
2637
2638 JTEST BIC INSTRUCTION INDIRECT WITH INDEXING
2639 012146 012700 177777 MOV #-1,X0
2640 012152 147700 002566 BICB #B+2,X0
2641 012156 120027 052525 CMPB X0,#52525
2642 012162 001401 BEQ ,+4
    
```

```

2643 012164 104006 HLT
2644 012166 104400 SCOPE
2645
2646 012170 012737 177777 014774 MOV #-1,#TEMP
2647 012176 142777 125252 002572 BICB #125252,#TEMP+2
2648 012204 122737 052525 014774 CMPB #52525,#TEMP
2649 012212 001401 BEQ ,+4
2650 012214 104006 HLT
2651 012216 104400 SCOPE
2652
2653 012220 012737 177777 014764 MOV #-1,#C
2654 012226 147777 002512 002532 BICB #B+2,#C+2
2655 012234 126737 002522 014764 CMPB #A+10,#C
2656 012242 001401 BEQ ,+4
2657 012244 104006 HLT
2658 012246 104400 SCOPE
2659
2660 JTEST UNARYS INDIRECT WITH INDEXING
2661 012250 012737 177777 014774 MOV #-1,#TEMP
2662 012256 105077 002514 CLRB #TEMP+2
2663 012262 105737 014774 TSTB #TEMP
2664 012266 001401 BEQ ,+4
2665 012270 104006 HLT
2666 012272 104400 SCOPE
2667
2668 012274 005037 014774 CLR #TEMP
2669 012300 105277 002472 INCB #TEMP+2
2670 012304 122737 000001 014774 CMPB #1,#TEMP
2671 012312 001401 BEQ ,+4
2672 012314 104006 HLT
2673 012316 104400 SCOPE
2674
2675 012320 005037 014774 CLR #TEMP
2676 012324 105377 002446 DECB #TEMP+2
2677 012330 123727 014774 177777 CMPB #TEMP,#-1
2678 012336 001401 BEQ ,+4
2679 012340 104006 HLT
2680 012342 104400 SCOPE
2681
2682 012344 012737 000001 014774 MOV #1,#TEMP
2683 012352 105477 002420 NEGB #TEMP+2
2684 012356 122737 177777 014774 CMPB #-1,#TEMP
2685 012364 001401 BEQ ,+4
2686 012366 104006 HLT
2687 012370 104400 SCOPE
2688
2689 012372 012737 177777 014774 MOV #-1,#TEMP
2690 012400 000261 SEC
2691 012402 105377 002370 ADCB #TEMP+2
2692 012406 022737 177400 014774 CMP #177400,#TEMP
2693 012414 001401 BEQ ,+4
2694 012416 104006 HLT
2695 012420 105737 014774 TSTB #TEMP
2696 012424 001401 BEQ ,+4
    
```

2697	012426	104006			HLT	
2698	012430	104400			SCOPE	
2699						
2700	012432	012737	000001	014774	MOV	#1,#TEMP
2701	012440	000261			SEC	
2702	012442	105377	002330		DECR	#TEMP+2
2703	012446	005737	014774		TST	#TEMP
2704	012452	001401			BEQ	,+4
2705	012454	104006			HLT	
2706	012456	104400			SCOPE	
2707						
2708						
2709	012460	012700	177772			ITEST OF COMBINED INDEXING AND INDIRECT
2710	012464	127027	014752	125252	MOV	#-6,X0
2711	012472	001401			CMPS	#A(0),#125252
2712	012474	104006			BEQ	,+4
2713	012476	104400			HLT	
2714					SCOPE	
2715	012500	012700	177772		MOV	#-6,X0
2716	012504	012701	000002		MOV	#+2,X1
2717	012510	127071	014752	014752	CMPS	#A(0),#A(1)
2718	012516	001401			BEQ	,+4
2719	012520	104006			HLT	
2720	012522	104400			SCOPE	
2721						
2722						ITEST BIC INSTRUCTION
2723	012524	012700	000006		MOV	#+6,X0
2724	012530	012767	177777	002236	MOV	#-1,TEMP
2725	012536	147067	014752	002230	BICB	#A(0),TEMP
2726	012544	122767	125252	002222	CMPS	#125252,TEMP
2727	012552	001401			BEQ	,+4
2728	012554	104006			HLT	
2729	012556	104400			SCOPE	
2730						
2731	012560	012700	177772		MOV	#-6,X0
2732	012564	012737	177777	014764	MOV	#-1,#C
2733	012572	142770	125252	014774	BICB	#125252,#TEMP(0)
2734	012600	123727	014764	000125	CMPS	#C,#000125
2735	012606	001401			BEQ	,+4
2736	012610	104006			HLT	
2737	012612	104400			SCOPE	
2738						
2739	012614	012700	014744		MOV	#B+2,X0 ;ADDRESS OF ADDRESS OF B
2740	012620	025067	002116		CMPS	#(0),B
2741	012624	001401			BEQ	,+4
2742	012626	104006			HLT	
2743	012630	104400			SCOPE	
2744						
2745	012632	012700	014746		MOV	#B+4,X0
2746	012636	025067	002100		CMPS	#-(0),B
2747	012642	001401			BEQ	,+4
2748	012644	104006			HLT	
2749	012646	104400			SCOPE	
2750						

2751	012650	012700	014746		MOV	#B+4,X0
2752	012654	125067	002062		CMPS	#-(0),B
2753	012660	001401			BEQ	,+4
2754	012662	104006			HLT	
2755	012664	104400			SCOPE	
2756						
2757	012666	012700	014770		MOV	#C+4,X0
2758	012672	012737	177777	014764	MOV	#-1,#C
2759	012700	105050			CLRB	#-(0)
2760	012702	023727	014764	177400	CMPS	#C,#177400
2761	012710	001401			BEQ	,+4
2762	012712	104006			HLT	
2763	012714	104400			SCOPE	
2764						
2765	012716	012737	177777	014764	MOV	#-1,#C
2766	012724	012700	177772		MOV	#-6,X0
2767	012730	012701	177772		MOV	#-6,X1
2768	012734	147071	014752	014774	BICB	#A(0),#TEMP(1)
2769	012742	022737	177525	014764	CMPS	#177525,#C
2770	012750	001401			BEQ	,+4
2771	012752	104006			HLT	
2772	012754	104400			SCOPE	
2773						
2774						
2775	012756	012700	052525		MOV	#52525,X0 ;THIS IS CHECKED LATER IN PROGRAM
2776						
2777						
2778	012762	004767	000002			ITEST JSR INSTRUCTION
2779	012766	000405			JSR	X7,TJSR2 ;PLACE PC ON STACK
2780	012770	121627	012766		TJSR1: BR	TJSR3 ;RETURN HERE ON RTS X19
2781	012774	001401			TJSR2: CMPS	#X0,#TJSR1 ;CHECK FOR CORRECT PC ON STACK
2782	012776	104006			BEQ	,+4
2783	013000	000207			HLT	
2784	013002	104400			RTS	X7 ;INCORRECT PC ON STACK
2785					TJSR3: SCOPE	;RETURN TO IMST AFTER JSR
2786	013004	000257				
2787	013006	004717			CCC	X7,#X7 ;INSTRUCTION UNDER TEST
2788	013010	121627	013010		JSR	#X0,#TJSR3+6 ;TEST THE STACK
2789	013014	001401			CMPS	,+4
2790	013016	104006			BEQ	,+4
2791	013020	005726			HLT	
2792	013022	104400			TST	{6}+ ;PC OF JSR DID NOT GO TO STACK
2793					SCOPE	;REPOSITION THE STACK
2794						
2795	013024	000257				ITEST NESTED SUBROUTINES
2796	013026	004767	001602		CCC	
2797	013032	104001			JSR	X7,SUBR6 ;CLEAR CONDITION CODES
2798	013034	104006			BMI	,+4
2799	013036	001401			HLT	
2800	013040	104006			BEQ	,+4 ;JSR OR RTS FAILED
2801	013042	104001			HLT	
2802	013044	104006			BEQ	,+4 ;JSR OR RTS FAILED
2803	013046	104001			HLT	
2804	013050	104006			BVS	,+4 ;JSR OR RTS FAILED
					HLT	
					BCS	,+4 ;JSR OR RTS FAILED
					HLT	


```

2913 013454 004767 000012 JSR X7,ROTALL IGD TO COMPARE ROUTINE
2914 013460 026727 000116 177777 CMP REFF,#=1 ITEST ALL VALUES
2915 013466 001370 BNE TSROT INO TEST THEM ALL
2916 013470 000046 BR TSRTZA IWE ARE DONE
2917
2918 013472 016767 000104 000104 ROTALL: MOV REFF,TEST
2919 013500 004067 000100 ROR TEST
2920 013504 004067 000074 ROR TEST
2921 013510 004067 000070 ROR TEST
2922 013514 004167 000064 ROL TEST
2923 013520 004167 000060 ROL TEST
2924 013524 004167 000054 ROL TEST
2925 013530 100004 BPL ,+12
2926 013532 103007 BCC ,+20 IZ=1
2927 013534 102013 BVC ,+30 IZ=1, C=1
2928 013536 104006 HLT IZ=C, BUT V=1
2929 013540 000411 BR ,+24
2930 013542 103006 BCC ,+16 IZ=0
2931 013544 102407 BVS ,+20 IZ=0, C=1
2932 013546 104006 HLT IZ NOT EQUAL C, V=1
2933 013550 000405 BR ,+14
2934 013552 102404 BVS ,+12 IZ=1, C=0
2935 013554 104006 HLT IZ NOT EQUAL C, V=1
2936 013556 000402 BR ,+6
2937 013560 102001 BVC ,+4 IZ=0, C=0
2938 013562 104006 HLT IZ=C, BUT V=1
2939 013564 104400 SCOPE
2940 013566 026767 000012 000006 CMP TEST,REFF
2941 013574 001401 BEQ ,+4
2942 013576 104006 HLT
2943 013600 000207 RTS X7 INITIAL NOT EQUAL TO FINAL
2944 013602 000000 REFF: 0
2945 013604 000000 TEST: 0
2946 013602 013602 REF=REFF
2947
2948
2949 013606 012767 177777 177766 JTEST ROTATING BYTE EVEN/ODD, ALL NUMBERS
2950 013614 005267 177762 TSRTZA: MOV #=1,REFF
2951 013620 004767 000016 TSROT2: INC REFF
2952 013624 004767 000122 JSR X7,ROTBE
2953 013630 022767 177777 177744 JSR X7,ROTBO
2954 013636 001366 BNE TSROT2
2955 013640 000505 BR ROTEN1
2956 013652 016767 177734 177734 ROTBE: MOV REFF,TEST
2957 013656 100067 177730 RORB TEST JROTATE BYTE EVEN
2958 013654 100067 177724 RORB TEST
2959 013660 100067 177720 RORB TEST
2960 013664 100167 177714 ROLB TEST
2961 013670 100167 177710 ROLB TEST
2962 013674 100167 177704 ROLB TEST
2963 013700 100004 BPL ,+12
2964 013702 103007 BCC ,+20 IZ=1
2965 013704 102013 BVC ,+30 IZ=1, C=1
2966 013706 104006 HLT IZ=C, BUT V=1

```

```

2967 013710 000411 BR ,+24
2968 013712 103006 BCC ,+16 IZ=0
2969 013714 102407 BVS ,+20 IZ=0, C=1
2970 013716 104006 HLT IZ NOT EQUAL C, V=1
2971 013720 000405 BR ,+14
2972 013722 102404 BVS ,+12 IZ=1, C=0
2973 013724 104006 HLT IZ NOT EQUAL C, V=1
2974 013726 000402 BR ,+6
2975 013730 102001 BVC ,+4 IZ=0, C=0
2976 013732 104006 HLT IZ=C, BUT V=1
2977 013734 104400 SCOPE
2978 013736 026767 177642 177636 CMP TEST,REFF
2979 013744 001401 BEQ ,+4
2980 013746 104006 HLT
2981 013750 000207 RTS X7
2982 013752 100067 177627 ROTBO: RORB TEST+1 JROTATE BYTE ODD
2983 013756 100067 177623 RORB TEST+1
2984 013762 100067 177617 RORB TEST+1
2985 013766 100167 177613 ROLB TEST+1
2986 013772 100167 177607 ROLB TEST+1
2987 013776 100167 177603 ROLB TEST+1
2988 014002 100004 BPL ,+12
2989 014004 103007 BCC ,+20 IZ=1
2990 014006 102013 BVC ,+30 IZ=1, C=1
2991 014010 104006 HLT IZ=C, BUT V=1
2992 014012 000411 BR ,+24
2993 014014 103006 BCC ,+16 IZ=0
2994 014016 102407 BVS ,+20 IZ=0, C=1
2995 014020 104006 HLT IZ NOT EQUAL C, V=1
2996 014022 000405 BR ,+14
2997 014024 102404 BVS ,+12 IZ=1, C=0
2998 014026 104006 HLT IZ NOT EQUAL C, V=1
2999 014030 000402 BR ,+6
3000 014032 102001 BVC ,+4 IZ=0, C=0
3001 014034 104006 HLT IZ=C, BUT V=1
3002 014036 104400 SCOPE
3003 014040 026767 177540 177534 CMP TEST,REFF
3004 014044 001401 BEQ ,+4
3005 014050 104006 HLT
3006 014052 000207 RTS X7
3007 014054 104400 ROTEN1: SCOPE
3008
3009
3010 JADD AND SUBTRACT ALL NUMBERS AGAINST FIXED NUMBERS
3011 014056 011667 000072 JA+B=C, C=A+B, BF SHOULD EQUAL BI
3012 014062 012767 000001 177512 MOV #6,NUMA
3013 014070 005267 177506 ARITST: INC #1,REF
3014 014074 004767 000014 JSR X7,ADSUB
3015 014100 022767 177777 177474 CMP #=1,REFF
3016 014106 001370 BNE ARITST
3017 014110 000422 BR ARIEND
3018 014112 104400 SCOPE
3019 014114 016767 177462 177462 ADSUB: MOV REF,TEST
3020 014122 006767 177454 ADD NUMA,TEST

```

3021	014130	166767	000020	177446	SUB	NUMA,TEST	
3022	014136	026767	177440	177440	CMP	REF,TEST	
3023	014144	001401			BEQ	.*4	
3024	014146	104006			HLT		
3025	014150	104400			SCOPE		
3026	014152	000207			RTS	X7	
3027	014154	000000			NUMA:	0	
3028	014156	104400			ARIEND:	SCOPE	
3029							
3030							
3031	014160	005067	000610				JTEST COMPLIMENTING ALL NUMBERS
3032	014164	005067	000610		CLR	TEMP	JBASE DATA
3033	014170	005167	000600		CLR	TEMP+4	JBASE REFERENCE
3034	014174	005367	000600		TCOM1:	COM	JCOMPLIMENT DATA
3035	014200	026767	000570	000572	DEC	TEMP+4	JDECREMENT REFERENCE
3036	014206	001401			CMP	TEMP,TEMP+4	JCOMPARE
3037	014210	104006			BEQ	.*4	JTEST
3038	014212	005167	000556		HLT		JCOMPLIMENT OR DECREMENT FAILED
3039	014216	005267	000552		COM	TEMP	
3040	014222	001362			INC	TEMP	JINCREMENT AND TEST FOR DONE
3041	014224	104400			BNE	TCOM	JNOT FINISHED GO LOOP
3042					SCOPE		
3043							
3044	014226	005067	000542				JTEST COMB (EVEN BYTE)
3045	014232	005067	000542		CLR	TEMP	JBASE DATA
3046	014236	105167	000532		CLR	TEMP+4	JREFERENCE DATA
3047	014242	005367	000532		TCOM2:	COMB	
3048	014246	126767	000522	000524	DEC	TEMP+4	
3049	014254	001401			CMPB	TEMP,TEMP+4	JCOMPARE
3050	014256	104006			BEQ	.*4	
3051	014260	105167	000510		HLT		JCOMPLIMENT OR INCREMENT BYTE FAILED
3052	014264	105267	000504		COMB	TEMP	
3053	014270	001362			INCB	TEMP	
3054	014272	104400			BNE	TCOM2	
3055					SCOPE		
3056							
3057	014274	005067	000474				JTEST COMB (ODD BYTE)
3058	014300	005067	000474		CLR	TEMP	JBASE DATA
3059	014304	105167	000465		CLR	TEMP+4	JREFERENCE DATA
3060	014310	005367	000464		TCOM3:	COMB	JODD BYTE
3061	014314	126767	000455	000456	DEC	TEMP+4	
3062	014322	001401			CMPB	TEMP+1,TEMP+4	
3063	014324	104006			BEQ	.*4	
3064	014326	105167	000443		HLT		JCOMPLIMENT BYTE FAILED
3065	014332	105267	000437		COMB	TEMP+1	
3066	014336	001362			INCB	TEMP+1	
3067	014340	104400			BNE	TCOM3	
3068					SCOPE		
3069							
3070	014342	005067	000426				JTEST COMPARE ALL VALUE EVEN BYTE WITH ODD
3071	014346	126767	000422	000421	TCOMB:	CLR	JBASE VALUE
3072	014354	001401			CMPB	TEMP,TEMP+1	JCOMPARE
					BEQ	.*4	

3073	014356	104006			HLT		JCOMPARE FAILED
3074	014360	002001			BGE	.*4	
3075	014362	104006			HLT		JV IS NOT = TO N
3076	014364	001401			BLE	.*4	
3077	014366	104006			HLT		JV IS SET
3078	014370	022767	000401	000376	ADD	#401,TEMP	
3079	014376	022767	177777	000370	CMP	#-1,TEMP	
3080	014404	001360			BNE	TSOMB	
3081	014406	104400			SCOPE		
3082							
3083	014410	012767	000010	000316	MOV	#10,ICOUNT	
3084							
3085							
3086	014416	016767	163554	000026			JTEST TO SEE IF I/O DEVICES WERE SELECTED
3087	014424	005167	000022		MOV	SREG2,CKWAIT	
3088	014430	032767	000371	000014	COM	CKWAIT	
3089	014436	001406			BIT	#371,CKWAIT	
3090	014440	000001			BEQ	WAIT0	JBRANCH IF NO DEVICES SELECTED
3091	014442	000001			WAIT		JINTERRUPTS WILL OCCUR
3092	014444	000001			WAIT		JIF DEVICES ARE SELECTED
3093	014446	000001			WAIT		
3094	014450	000401			BR	.*4	
3095	014452	000000			CKWAIT:	0	
3096	014454	104400			WAIT4:	SCOPE	
3097	014456	012767	000400	000250	MOV	#400,ICOUNT	
3098							
3099							
3100	014464	012767	000200	177112			JTEST SWAB
3101	014472	000367	177106		MOV	#0200,TEST	
3102	014476	100001			SWAB	TEST	
3103	014500	104006			BPL	.*4	
3104	014502	001401			HLT		
3105	014504	104006			BEQ	.*4	
3106	014506	000367	177072		HLT		
3107	014512	100001			SWAB	TEST	
3108	014514	104006			BMI	.*4	
3109	014516	001001			HLT		
3110	014520	104006			BNE	.*4	
3111	014522	104400			HLT		
3112					SCOPE		
3113							
3114	014524	005067	177054				JTEST ALL COMBINATIONS OF SWAB
3115	014530	005067	177046		CLR	TEST	JNUMBER UNDER TEST
3116	014534	000367	177044		CLR	REF	JREFERENCE NUMBER
3117	014540	026767	177040	177034	SWAB:	SWAB	JOPERATION UNDER TEST
3118	014546	001401			CMP	TEST,REF	JTEST SWAB INSTRUCTION
3119	014550	104006			BEQ	.*4	
3120	014552	000367	177026		HLT		JSWAB FAILED
3121	014556	005267	177020		SWAB	TEST	
3122	014562	105267	177017		INC	REF	JINCREMENT REFERENCE NUMBER
3123	014566	001362			INCB	TEST+1	JINC TEST NUMBER
3124	014570	104400			BNE	SWABA	JLOOP TILL DONE
3125	014572	012767	004000	000134	SCOPE		
3126					MOV	#4000,ICOUNT	


```

3127
3128
3129
3130
3131
3132 014600 1040'0
3133 014602 000.40
3134
3135
3136 014604 000207
3137 014606 000277
3138 014610 000207
3139 014612 004767 177770
3140 014616 000207
3141 014620 004767 177766
3142 014674 000207
3143 014626 004767 177766
3144 014632 000207
3145 014634 004767 177766
3146 014640 000207
3147
3148
3149 014642 032767 002000 163326
3150 014650 001403
3151 014652 022626
3152 014654 000167 165416
3153 014660 032767 040000 163310
3154 014666 001012
3155 014670 032767 004000 163300
3156 014676 001011
3157 014700 026767 000032 000026
3158 014706 100005
3159 014710 005267 000022
3160 014714 016716 000020
3161 014720 000002
3162 014722 005067 000010
3163 014726 011667 000006
3164 014732 000002
3165 014734 000400
3166 014736 000000
3167 014740 000000
3168
3169
3170 014742 125252
3171 014744 014742
3172 014746 052525
3173 014752 014752
3174 014752 177777
3175 014754 014756
3176 014756 014756
3177 014756 125252
3178 014760 014762
3179 014762 052525
3180

```

 /END OF USER CODE IN BANK/
 /CALL KERNEL/
 /ALTERED IN CORE EXPANSION/
 DONE: EOB
 NOP

ITD ALLOW CORE EXPANSION TO PATCH IN JMP

IGROUP OF NESTED SUBROUTINES/
 SUBR1: RTS X7 /ONE INSTRUCTION
 SUBR2: SCC /ONE DEEP
 SUBR3: JSR X7,SUBR2 /TWO DEEP
 SUBR4: JSR X7,SUBR3 /THREE DEEP
 SUBR5: JSR X7,SUBR4 /FOUR DEEP
 SUBR6: JSR X7,SUBR5 /FIVE DEEP

ISCOPE AND/OR ITERATION LOOP FOR EACH TEST TIMES/
 SCOPEC: BIT #2000,SREG2 /INHIBIT PROCESSOR TESTS?
 BEQ .+10 /NO
 CMP (SP)+,(SP)+ /YES
 JMP MAIN /TEST SR FOR SCOPE
 BIT #4000,SREG2 /YES, SCOPE
 BNE SCOPEB /NO-TEST FOR ITERATION
 BIT #4000,SREG2 /INHIBIT ITERATION
 BNE SCOPEG /COMPARE CURRENT COUNT TO MAX NUMBER
 CMP SCOPEF,ICOUNT /EXIT-DONE
 BPL SCOPEG /INCREMENT COUNT
 INC SCOPEF
 SCOPEB: MOV RETURN,#SP
 RTI
 SCOPEG: CLR SCOPEF /CLEAR COUNT
 MOV #%,RETURN /SAVE SCOPE RETURN POINTER
 RTI /RETURN INLINE-NEXT TEST
 ICOUNT: 400 /ITERATION COUNT
 SCOPEF: 0 /COUNT LOCATION FOR ITERATION LOOP
 RETURN: 0 /ADDRESS OF LAST TEST

IFIXED VALUES FOR USE IN TEST/
 B: 125252
 B /ADDRESS OF B
 052525
 ,=B+10
 A: -1
 A+4
 ,=A+4
 125252
 A+10 /ADDRESS OF A+10
 052525

```

3181
3182 014764 000000
3183 014766 014764
3184 014774 014774
3185 014774 000000
3186 014776 014774
3187 015002 015002
3188 015002 015004
3189 015004 000000
3190
3191
3192 015006 010146
3193 015010 010246
3194 015012 010346
3195 015014 012701 000536
3196
3197 015020 012703 000010
3198 015024 012102
3199
3200 015026 005022
3201 015030 077302
3202 015032 020127 000544
3203 015036 003770
3204 015040 012603
3205 015042 012602
3206 015044 012601
3207 015046 000207
3208
3209
3210
3211 015050 162716 000002
3212 015054 006576 000000
3213 015060 012667 000022
3214 015064 062716 000022
3215 015070 105067 000013
3216 015074 062767 015110 000004
3217 015102 017707 000000
3218 015106 000000
3219 000000
3220 000000
3221 000000
3222 015110 000000
3223 015112 000000
3224 015114 000000
3225 015116 016164
3226 015120 015122
3227
3228
3229 015122 117737 163422 000177
3230 015130 032767 000001 163036
3231 015136 001507
3232 015140 004767 001410
3233 015144 042766 000020 000002
3234 015152 012737 000016 000014

```

IFOR STORAGE
 C: 0
 C /ADDRESS OF C
 ,=C+10
 TEMP: 0
 TEMP /ADDRESS OF TEMP
 ,=TEMP+6
 TEMP+10 /ADDRESS OF TEMP+10 OR "0"
 D: 0

ISUBROUTINE TO INITIALIZE ALL PAGES TO NR, BANK 0, 1 PAGE, UP/
 NRALL: MOV R1,=(R6) /SAVE REGISTERS
 MOV R2,=(R6)
 MOV R3,=(R6)
 MOV #IPDRTAB,R1 /R1 HOLDS ADDRESS OF CURRENT POSITION
 /IN TABLE OF ADDRESSES
 NRLOOP: MOV #0,R3 /R3 USED AS COUNTER
 MOV (R1)+,R2 /R2 CONTAINS ADDRESS OF PDR OR
 /PAR TO BE CLEARED
 CLR (R2)+ /CLEAR ALL ASR'S FOR THIS MODE
 SOB R3,=#2
 CMP R1,#IPDREND /CHECK FOR DONE
 BLE NRLOOP /CLEAR ALL IN NEXT MODE IF NOT DONE
 MOV (R6)+,R3
 MOV (R6)+,R2
 MOV (R6)+,R1
 RTS X7

EMT HANDLER/
 IFIRST 3 CALLS LEFT OPEN IN TABLE FOR EASY PATCHES/
 EMTSRV: SUB #2,#SP /GET CALL
 MFPI 0(SP)
 MOV (SP)+,EPC
 ADD #2,#SP
 CLRB EPC+1 /SAVE OFFSET ONLY
 ADD #EMTAB,EPC /POINT TO TABLE OF ADDRESSES
 MOV #EPC,PC /JUMP TO DESIRED ROUTINE
 EPC: 0
 PATCH1=0
 PATCH2=0
 PATCH3=0
 ENTAB: PATCH1 /PATCH IN ADDRESS OF ROUTINE
 PATCH2
 PATCH3
 PRINT /ERROR PRINTOUT
 EOBRSV /END OF BANK

END OF BANK SERVICE
 EOBRSV: MOV #SRH,#SREG2+1 /READ SWITCHES AGAIN
 BIT #1,MMOPT /MEMORY MANAG. INHIBITED?
 BEQ EOB2 /NO - CONTINUE
 JSR X7,BELL /SIGNAL END OF PASS
 BIC #20,2(SP) /CLEAR TRACE BIT OF STATUS ON STACK
 MOV #16,#14 /SETUP TRACE RETURN TO CAUSE HALT

3235	015160	005037	000016		CLR	#016		IF A TRACE TRAP OCCURS
3236	015164	032737	010000	000176	BIT	#10000, #SREG2		INHIBIT TRACE TRAPPING?
3237	015172	001011			EOR1A			YES - BRANCH
3238	015174	005167	163402		COM	TRPB		SWITCH TRACE FLAG
3239	015200	100006			BPL	EOR1A		IF NOT SET, LEAVE TRACE OFF
3240	015202	052766	000020	000002	BIS	#20, 2(SP)		IF SET, SET TRACE BIT OF STATUS ON STACK
3241	015210	012737	016162	000014	MOV	#TRTRP, #014		
3242	015216	032737	000040	000174	EOR1A: BIT	#40, #MMOPT		CORE EXPANSION INHIBITED?
3243	015224	001051			BNE	EOR1C		YES, SKIP
3244	015226	013701	000176		MOV	#SREG2, R1		
3245	015232	032767	000002	163344	BIT	#2, MEM0		
3246	015240	001402			BEQ	DSW1		
3247	015242	010137	020176		MOV	R1, #SREG2+20000		
3248	015246	032767	000004	163330	BIT	#4, MEM0		
3249	015254	001402			BEQ	DSW2		
3250	015256	010137	040176		MOV	R1, #SREG2+40000		
3251	015262	032767	000010	163314	BIT	#10, MEM0		
3252	015270	001402			BEQ	DSW3		
3253	015272	010137	060176		MOV	R1, #SREG2+60000		
3254	015276	032767	000020	163300	BIT	#20, MEM0		
3255	015304	001402			BEQ	DSW4		
3256	015306	010137	100176		MOV	R1, #SREG2+100000		
3257	015312	032767	000040	163264	BIT	#40, MEM0		
3258	015320	001402			BEQ	DSW5		
3259	015322	010137	120176		MOV	R1, #SREG2+120000		
3260	015326	032767	000100	163250	BIT	#100, MEM0		
3261	015334	001402			BEQ	EOR1B		
3262	015336	010137	140176		MOV	R1, #SREG2+140000		
3263	015342	012716	005420		EOR1B: RTI	#BEGINX, (SP)		
3264	015346	000002			MOV	EOR1C:		#BEGIN, (SP)
3265	015350	012716	005452		RTI			
3266	015354	000002			MOV	EOR2:		#340, #PSR
3267	015356	042737	000340	177776	BIC	#2, MMOPT		USER/KERNEL INHIBITED?
3268	015364	032767	000002	162602	BIT	EOR1		YES - SET PC AND RETURN
3269	015372	001262			BNE	#4, MMOPT		NO--INHIBIT 4K AS 32K?
3270	015374	032767	000004	162572	BIT	EOR1		YES - SET PC AND RETURN
3271	015402	001256			BNE	EOR1		LAST USER ASR DONE?
3272	015404	026767	163164	163102	MOV	CURPAR, UPAR7		YES - GO FIND NEXT BANK
3273	015412	001444			NXTBNK			UPDATE SCOPE VECTOR ADDRESS IN BANK 0
3274	015414	062737	020000	000034	ADD	#20000, #034		UPDATE BANK START TO REFERENCE CURRENT ASR
3275	015422	062767	020000	163150	ADD	#20000, BNKSTR		
3276	015430	016716	163144		MOV	BNKSTR, (SP)		
3277	015434	026767	163050	163132	MOV	UPAR0, CURPAR		
3278	015442	001402			BEQ	NXTSEG		
3279	015444	005077	163124		CLR	CURPAR		SET PREVIOUS ASR TO NR, BANK 0
3280	015450	005077	163122		CLR	CURPDR		
3281	015454	062767	000002	163112	NXTSEG: ADD	#2, CURPAR		UPDATE POINTERS TO NEXT SEGMENT
3282	015462	062767	000002	163106	ADD	#2, CURPDR		
3283	015470	012777	077406	163100	MOV	#77406, CURPDR		SET NEXT SEGMENT RW, 4K
3284	015476	016777	163066	163070	MOV	CURBNK, CURPAR		MAP NEXT SEGMENT TO CURRENT BANK
3285	015504	052737	030000	177776	BIS	#30000, #PSR		SET PREVIOUS MODE TO USER
3286	015512	006506			MFPI	R6		PICK UP USER STACK POINTER
3287	015514	062716	020000		ADD	#20000, #R6		MAP IT TO NEXT ASR
3288	015520	006606			MTP1	R6		PUT IT BACK

3289	015522	000002			RTI			GO BACK TO MAINLINE
3290	015524	005327	000000		NXTBNK: DEC	#0		STALL SO DOUBLE BELL NOTED
3291	015530	001375			BNE	-4		
3292	015532	004767	001016		JSR	#7, BELL		
3293	015536	012746	000400		MOV	#UBUFF, -(SP)		
3294	015542	052737	030000	177776	BIS	#30000, #PSR		
3295	015550	006606			MTP1	R6		
3296	015552	013737	000570	000572	MOV	#CURBNK, #OLDBNK		SAVE PREV BANK ADDRESS
3297	015560	062767	000200	163002	BNKTST: ADD	#200, CURBNK		
3298	015566	006367	163016		ASL	COREPT		
3299	015572	103006			BCC	13		
3300	015574	012767	000001	163006	MOV	#1, COREPT		
3301	015602	012767	000006	163002	MOV	#MEM1, MEMUT		
3302	015610	022767	007600	162752	13: CMP	#7600, CURBNK		CHECK FOR EXTERNAL BANK
3303	015616	001067			BNE	EOR3		IF NOT, TEST FOR ITS PRESENCE
3304	015620	012767	000000	162742	MOV	#0, CURBNK		START OVER, TESTING BANK 0
3305	015626	012767	000001	162734	MOV	#1, COREPT		
3306	015634	012767	000004	162750	MOV	#MEM0, MEMUT		
3307	015642	013701	000042		LOGIC: MOV	#42, R1		
3308	015646	001412			BEQ	BNKT		
3309	015650	000005			RESET			
3310	015652	005646			CLR	-(SP)		CLEAR T8TT VIA RTI
3311	015654	012746	015662		MOV	#LOGICAL, -(SP)		
3312	015660	000002			RTI			
3313	015662	004711			LOGICAL: JSR	#7, #R1		
3314	015664	000240			NOP			
3315	015666	000240			NOP			
3316	015670	000240			NOP			
3317	015672	000000			HALT			
3318	015674	032737	000001	000176	BNKT: BIT	#1, #SREG2		TTY OUT SELECTED
3319	015702	001410			BEQ	BNKT1		YES, NO ASTERISK
3320	015704	004767	000672		JSR	#7, CRLF		
3321	015710	105777	162474		TSR0	#TCSR		WAIT FOR TELETYPE
3322	015714	100375			BPL	-4		
3323	015716	012777	000252	162466	MOV	#252, #TDBR		OUTPUT ASTERISK TO SIGNAL END OF PASS
3324	015724	042766	000000	000006	BNKT1: BIC	#20, 6(SP)		CLEAR TRACE BIT OF STATUS ON STACK
3325	015732	012737	000016	000014	MOV	#16, #014		
3326	015740	005037	000016		CLR	#016		
3327	015744	032777	010000	162574	BIT	#10000, #SR		
3328	015752	001011			BNE	EOR3		
3329	015754	005167	162622		COM	TRPB		
3330	015760	100006			BPL	EOR3		
3331	015762	052766	000020	000006	BIS	#20, 6(SP)		
3332	015770	012737	016162	000014	MOV	#TRTRP, #014		
3333	015776	016777	162566	162526	EOR3: MOV	CURBNK, #KPAR2		MAP KERNEL SEGMENT 2 TO BANK BEING LOOKED FOR
3334	016004	012777	077406	162510	MOV	#77406, #KPAR2		
3335	016012	036777	162572	162572	BIT	COREPT, #MEMUT		
3336	016020	001657			BEQ	BNKST		
3337	016022	042737	160000	000034	BIC	#160000, #034		INITIALIZE SCOPE VECTOR ADDRESS
3338	016030	005001			CLR	R1		ADDRESSES BANK 0 THRU KERNEL ASR0
3339	016032	012702	040000		MOV	#40000, R2		ADDRESSES NEW BANK THRU KERNEL ASR2
3340	016036	012703	015004		MOV	#0, R3		
3341	016042	006203			ASR	R3		
3342	016044	012122			CORMOV: MOV	(R1)+, (R2)+		

```

3343 016046 077302          SOB      R3,CORMOV
3344 016050 016767 162434 162516      MOV      UPAR0,CURPAR      ;FIRST ASR CHECKED IS USER ASR0
3345 016056 016767 162420 162512      MOV      UPDR0,CURPDR
3346 016064 016777 162500 162502      MOV      CURBNK,#CURPAR
3347 016072 012777 077406 162476      MOV      #77406,#CURPDR
3348 016100 005077 162410          CLR      #UPAR7
3349 016104 005077 162376          CLR      #UPDR7
3350 016110 026727 162456 000000      CMP      OLDBNK,#0          ;PREV BANK = 0
3351 016116 001414          BEQ      E0B6              ;YES, DO NOT CLEAR
3352 016120 016777 162446 162404      MOV      OLDBNK,#KPAR2
3353 016126 012777 077406 162366      MOV      #77406,#KPAR2
3354 016134 012701 040000          MOV      #40000,R1
3355 016140 012703 007630          MOV      #7630,R3
3356 016144 005021          DNKLP1 CLR      (R1)+
3357 016146 077302          SOB      R3,BNKLP
3358 01615 012716 005452          E0B6:  MOV      #BEGIN,(SP)
3359 016154 011667 162420          MOV      (SP),BNKSTR
3360 016160 000002          RTI
3361
3362
3363 016162 000006          ;RTT EXECUTED WHEN TRACE IS ON/
3364          TRTRP: RTT
3365
3366          ;ENTERED WITH SYSTEM TRAP CALL (HLT)
3367 016164 005767 000162          ;PRINT OUT THE ERROR PC+2, STATUS REGISTER, AND LOCATION IN BACKGROUND
3368 016170 001401          PRINT: TST      PRTON          ;CHECK PRINT ON FLAG
3369 016172 000002          BEQ      .+4
3370 016174 005267 000152          RTI
3371 016200 012767 000340 161570      INC      PRTON          ;IF ANOTHER HALT IS BEING PRINTED, SKIP THIS ONE
3372 016206 037727 162334 020000      MOV      #340,PSR          ;SET PRIORITY TO 7
3373 016214 001401          BIT      #SR,#20000        ;TEST FOR INHIBIT PRINT OUT
3374 016216 000444          BEQ      .+4              ;BRANCH TO PRINT
3375 016220 012667 000122          BR      CK                ;INHIBIT, CHECK FOR HALT
3376 016224 012667 000120          MOV      (6)+,SAVPC        ;PC OF FAILING ROUTINE
3377 016230 024646          MOV      (6)+,SAVPSR       ;PSR OF ERROR CONDITION
3378 016232 012767 000200 161536      CMP      -(6),-(6)        ;RESTORE STACK
3379 016240 004767 000336          MOV      #200,PSR
3380 016244 016767 000076 000260      JSR      X7,CRLF          ;OUTPUT CARRIAGE RETURN AND LINE FEED
3381 016252 004767 000076          MOV      SAVPC,PTEMP1     ;LOAD WITH FAILING PC+2
3382 016256 004767 000254          JSR      X7,PROCT         ;PRINT FAILING PC+2
3383 016262 016767 000062 000242      JSR      X7,SPACE
3384 016270 004767 000060          MOV      SAVPSR,PTEMP1    ;LOAD PROCESSOR STATUS
3385 016274 004767 000236          JSR      X7,PROCT         ;PRINT PROCESSOR STATUS
3386 016300 016767 162264 000224      JSR      X7,SPACE
3387 016306 004767 000042          MOV      CURBNK,PTEMP1
3388 016312 004767 000220          JSR      X7,PROCT
3389 016316 016767 176416 000206      JSR      X7,SPACE
3390 016324 004767 000024          MOV      RETURN,PTEMP1
3391 016330 005777 162212          JSR      X7,PROCT
3392 016334 100001          CK:    TST      #SR          ;CHECK SR FOR HALT SWITCH
3393 016336 000000          BPL      .+4              ;BRANCH IF NOT SET
3394 016340 005067 000006          HALT
3395 016344 000002          CLR      PRTON           ;ROUTINE DONE - CLEAR FLAG
3396 016346 000000          RTI                      ;RETURN TO MAIN LINE
3397
3398
3399
3400          SAVPSR: 0
3401 016354 012727 000006 016360          PRTON: 0
3402 016362 005067 000142          ;SUBROUTINE TO PRINT OUT OCTAL NUMBER/
3403 016366 012767 000260 000140      PROCT: MOV      #6,#PTEMP3    ;CLEAR R4 FOR COUNTING CHARACTERS OUTPUT
3404 016374 005767 000132          PTEMP3: -2
3405 016400 100002          CLR      PRFLG           ;INITIALIZE CARRY FLAG FOR ROTATES
3406 016402 005267 000126          MOV      #260,PTEMP2     ;SETUP R3
3407 016406 006167 000120          TST      PTEMP1          ;CHECK BIT 15 OF NUMBER
3408 016412 006167 000114          BPL      .+6              ;BRANCH IF ZERO
3409 016416 005567 000106          INC      PTEMP2          ;INCREMENT R3 IF ONE
3410 016422 105777 161762          ROL      PTEMP1          ;ROTATE LEFT MOST OCTAL TO RIGHT END
3411 016426 100375          ADC      PRFLG           ;STORE CARRY
3412 016430 016777 000100 161754      P_WAIT: TSTB      #TC3R    ;WAIT FOR TTY READY
3413 016436 005367 177716          BPL      P_WAIT
3414 016442 000207          MOV      PTEMP2,#TDBR    ;OUTPUT NEXT CHARACTER
3415 016444 000207          DEC      PTEMP3          ;COUNT
3416 016446 000207          BNE      P_CNT1         ;BRANCH IF NOT DONE
3417 016450 005767 000054          RTS
3418 016454 001403          P_CNT1: CLC              ;CLEAR CARRY
3419 016456 005067 000046          TST      PRFLG           ;CHECK FOR PREVIOUS CARRY
3420 016462 000261          BEQ      .+10            ;BRANCH IF PREVIOUSLY ZERO
3421 016464 006167 000042          CLR      PRFLG          ;INITIALIZE FLAG
3422 016470 006167 000036          SEC
3423 016474 006167 000032          ROL      PTEMP1         ;SET CARRY
3424 016476 006167 000024          ROL      PTEMP1         ;ROTATE NEXT CHARACTER INTO RIGHT END OF REGISTER
3425 016500 005567 000022          ADC      PRFLG           ;STORE CARRY
3426 016512 042767 177770 020014      MOV      PTEMP1,PTEMP2   ;LOAD DATA INTO R3
3427 016520 052767 000260 000006      BIC      #17770,PTEMP2   ;CLEAR ALL BUT LOWEST OCTAL DIGIT
3428 016526 000735          BIS      #260,PTEMP2    ;SET TO ASCII EQUIVALENT
3429 016530 000000          BR      P_WAIT          ;LOOP
3430 016532 000000          PRFLG: 0
3431 016534 000000          PTEMP1: 0
3432 016536 000000          PTEMP2: 0
3433
3434          ;SUBROUTINE TO ISSUE SPACE/
3435 016542 100375 161646          SPACE: TSTB      #TC3R    ;WAIT FOR TTY READY
3436 016544 012777 000240 161640      BPL      .+4
3437 016546 000207          MOV      #240,#TDBR     ;OUTPUT A SPACE
3438 016552 000207          RTS
3439
3440          ;BELL ON PASS COMPLETE
3441 016554 032737 000001 000176      BELL:  BIT      #1,#PSREG2
3442 016562 001406          BEQ      18
3443 016564 105777 161620          TSTB      #TC3R
3444 016570 100375          BPL      .+4
3445 016572 012777 000207 161612      MOV      #207,#TDBR
3446 016600 000207          RTS
3447
3448          ;SUBROUTINE TO OUTPUT CARRIAGE RETURN AND LINEFEED/
3449 016602 105777 161602          CRLF:  TSTB      #TC3R    ;WAIT FOR TTY READY
3450 016606 100375          BPL      .+4
  
```

```

3397 016350 000000          SAVPSR: 0
3398 016352 000000          PRTON: 0
3399
3400
3401 016354 012727 000006 016360          ;SUBROUTINE TO PRINT OUT OCTAL NUMBER/
3402 016362 005067 000142          PROCT: MOV      #6,#PTEMP3    ;CLEAR R4 FOR COUNTING CHARACTERS OUTPUT
3403 016366 012767 000260 000140      PTEMP3: -2
3404 016374 005767 000132          CLR      PRFLG           ;INITIALIZE CARRY FLAG FOR ROTATES
3405 016400 100002          MOV      #260,PTEMP2     ;SETUP R3
3406 016402 005267 000126          TST      PTEMP1          ;CHECK BIT 15 OF NUMBER
3407 016406 006167 000120          BPL      .+6              ;BRANCH IF ZERO
3408 016412 006167 000114          INC      PTEMP2          ;INCREMENT R3 IF ONE
3409 016416 005567 000106          ROL      PTEMP1          ;ROTATE LEFT MOST OCTAL TO RIGHT END
3410 016422 105777 161762          ADC      PRFLG           ;STORE CARRY
3411 016426 100375          P_WAIT: TSTB      #TC3R    ;WAIT FOR TTY READY
3412 016430 016777 000100 161754      BPL      P_WAIT
3413 016436 005367 177716          MOV      PTEMP2,#TDBR    ;OUTPUT NEXT CHARACTER
3414 016442 000207          DEC      PTEMP3          ;COUNT
3415 016444 000207          BNE      P_CNT1         ;BRANCH IF NOT DONE
3416 016446 000207          RTS
3417 016450 005767 000054          P_CNT1: CLC              ;CLEAR CARRY
3418 016454 001403          TST      PRFLG           ;CHECK FOR PREVIOUS CARRY
3419 016456 005067 000046          BEQ      .+10            ;BRANCH IF PREVIOUSLY ZERO
3420 016462 000261          CLR      PRFLG          ;INITIALIZE FLAG
3421 016464 006167 000042          SEC
3422 016470 006167 000036          ROL      PTEMP1         ;SET CARRY
3423 016474 006167 000032          ROL      PTEMP1         ;ROTATE NEXT CHARACTER INTO RIGHT END OF REGISTER
3424 016476 006167 000024          ADC      PRFLG           ;STORE CARRY
3425 016500 005567 000022          MOV      PTEMP1,PTEMP2   ;LOAD DATA INTO R3
3426 016512 042767 177770 020014      BIC      #17770,PTEMP2   ;CLEAR ALL BUT LOWEST OCTAL DIGIT
3427 016520 052767 000260 000006      BIS      #260,PTEMP2    ;SET TO ASCII EQUIVALENT
3428 016526 000735          BR      P_WAIT          ;LOOP
3429 016530 000000          PRFLG: 0
3430 016532 000000          PTEMP1: 0
3431 016534 000000          PTEMP2: 0
3432 016536 000000
3433
3434          ;SUBROUTINE TO ISSUE SPACE/
3435 016542 100375 161646          SPACE: TSTB      #TC3R    ;WAIT FOR TTY READY
3436 016544 012777 000240 161640      BPL      .+4
3437 016546 000207          MOV      #240,#TDBR     ;OUTPUT A SPACE
3438 016552 000207          RTS
3439
3440          ;BELL ON PASS COMPLETE
3441 016554 032737 000001 000176      BELL:  BIT      #1,#PSREG2
3442 016562 001406          BEQ      18
3443 016564 105777 161620          TSTB      #TC3R
3444 016570 100375          BPL      .+4
3445 016572 012777 000207 161612      MOV      #207,#TDBR
3446 016600 000207          RTS
3447
3448          ;SUBROUTINE TO OUTPUT CARRIAGE RETURN AND LINEFEED/
3449 016602 105777 161602          CRLF:  TSTB      #TC3R    ;WAIT FOR TTY READY
3450 016606 100375          BPL      .+4
  
```

3451	016610	012777	000215	161574	MOV	#215,0TDBR	OUTPUT CARRIAGE RETURN
3452	016616	105777	161566	TSTB	0TCSR	WAIT FOR TTY READY	
3453	016622	100375		BPL	,=-4		
3454	016624	012777	000212	161560	MOV	#212,0TDBR	OUTPUT LINEFEED
3455	016632	000207		RTB	X7	RETURN	
3456							
3457							
3458	016634	013746	000024		JENTER HERE ON POWER FAIL/ PFAIL: MOV #024,-(6)	STORE STACK POSITION	
3459	016640	010667	000010		MOV %6,SAVR6		
3460	016644	012737	016656	000024	MOV #RESTR,0#24		
3461	016652	000000			HALT	HALT ON POWER DOWN NORMAL	
3462	016654	000000		SAVR6: 0	RESTR: 0	STACK IS SAVED HERE	
3463	016656	016706	177772		MOV SAVR6,%6	RESTORE STACK WHEN POWERING UP	
3464	016662	012637	000024		MOV (6)+,#24		
3465	016666	022626			CHP (SP)+,(SP)+	RESTORE STACK	
3466	016670	104006			HLT	POWER FAIL OCCURRED	
3467	016672	000167	161720		JMP RSTR	RETURN TO MAIN LINE	
3468							
3469							
3470	016676	000207			USER: RTS X7	OVERLAY USER ROUTINE HERE IF 4KW	
3471						USE BANK1 IF 8KW	
3472		017760			KSTACK: 0,=17760		
3473	017760	000000			0,END		
3474		000001					

A	014752	ADSUB	014114	ARIEND	014156	ARITST	014070
B	014742	BEGIN	005452	BEGINX	005420	BELL	016554
BNKLP	016144	BNKSTR	000600	BNKT	015674	BNKTST	015560
BNKT1	015724	C	014764	CK	016330	CKWAIT	014452
CLINCT	003250	CMPI	013414	CMPI	013436	COMPAR	013410
COREPT	000610	CORMOV	016044	CRLF	016602	CURBNK	000570
CURPAR	000574	CURPAT	003160	CURPOR	000576	D	015004
DATA2	002766	DET1	002334	DET2	002452	DET3	002750
DET4	002366	DO	000001	DONE	014600	DSW1	015246
DSW2	015262	DSW3	015276	DSW4	015312	DSW5	015326
EIGHT	002404	EMTAB	015110	EMTSRV	015050	EOB	104010
EOBSRV	015122	EOB1	015140	EOB1A	015216	EOB1B	015342
EOB1C	015350	EOB2	015356	EOB3	015776	EOB6	016150
EPC	015106	ERR0K	003536	ERRSET	003532	FENDZ	003612
FEND1	003634	HLT	104006	ICOUNT	014734	IE	000500
IPDREN	000544	IPDRTA	000536	IRF	003472	IRK	003376
KPAR0	000526	KPAR1	000530	KPAR2	000532	KPAR7	000534
KPDR0	000516	KPDR1	000520	KPDR2	000522	KPDR7	000524
KSTACK	017760	KWLST	000424	KWLVC	000422	LKCSR	000426
LK1	003034	LK2	003054	LK3	003056	LK4	003074
LLIMIT	003600	LOGIC	015642	LOGICA	015662	LPCSR	000430
LPDR	000432	LPINTR	003200	LPST	000436	LPVC	000434
LP1	003154	LP2	003162	LP3	003270	LP4	003300
LP5	003336	LP6	003236	LP7	003234	MAIN	002276
MAP1	001074	MAP2	001164	MEMUT	000612	MEM0	000604
MEM1	000606	MFOPT	000174	MODE	002226	MOD12	002704
MOD16	002670	MOD20	002654	MOD24	002640	MOD4	002734
MOD8	002720	MOVE	002464	MOVEPT	001134	N	000001
MODEV	002314	NOP	000240	NRALL	015006	NRLOOP	015020
YUMA	014154	NXTBNK	015524	NXTSEG	015454	OLDBNK	000572
-ATCH1	000000	PATCH2	000000	PATYCH3	000000	PC	000000
PFAIL	016634	PRFL6	016530	PRINT	016164	PROCT	016354
PRTON	016352	PSR	177776	PTEMP1	016532	PTEMP2	016534
PTEMP3	016360	P,CNT1	016446	P,WAIT	016422	R	004000
RB	000002	RO	000004	REF	013602	REFF	013602
REND2	004010	REND1	004040	RESTR	016656	RETURN	014740
RFCAR	000446	RFCSR	000450	RFCSRH	000452	RFOAE	000440
RFDAR	000442	RFFUNC	003576	RFST	000456	RFSTAR	003442
RFVC	000454	RFWC	000444	RF1	003446	RKBAR	000466
RKCSR	000470	RKCSRH	000472	RKDAE	000462	RKDAH	000460
RKJUNC	003440	RKST	000476	RKSTAR	003346	RKVC	000474
RKWC	000464	RK1	003352	ROTALL	013472	ROTBE	013642
ROB0	013752	ROTEN1	014054	RSTR	000616	R0	000000
R1	=X000001	R2	=X000002	R3	=X000003	R4	=X000004
R5	=X000005	R6	=X000006	SAVPC	016346	SAVPSR	016350
SAVR6	016654	SCOPE	104400	SCOPEB	014714	SCOPEC	014642
SCOPEF	014736	SCOPEG	014722	SEG1	002074	SETEX	002054
SETSEG	002220	SETUSE	002124	SOLPAT	003156	SP	000000
SPACE	016336	SR	000546	SREG2	000176	SRH	000550
SR0	000500	START	000640	STARTX	000756	START1	000764
STRY28	002550	STRY4	002504	ST3	001362	ST3A	001336
ST4	001452	ST5	001516	ST6	001604	ST7	001644
ST8	001754	SUBR1	014604	SUBR2	014606	SUBR3	014612
SUBR4	014620	SUBR5	014626	SUBR6	014634	SWABA	014534

SWREG 000176	TBANK 000614	TCBA 000562	TCCM 000552
TCDT 000556	TCEXPE 003610	TCFIRS 003604	TCF1 003664
TCF1A 003656	TCF2 003716	TCF3 003732	TCF4 004000
TCIV 000564	TCLAST 003606	TCOM 014170	TCOM2 014236
TCOM3 014304	TCRBK 004270	TCRBUF 004420	TCR31 004326
TCR1 004136	TCRIA 004172	TCR2 004200	TCR3 004214
TCR4 004262	TCRSR 000410	TCST 000554	TCSTA 000566
TCWBK 004056	TCWBUF 004420	TCW01 004110	TCWC 000560
TC1 004354	TC2 004374	TDBR 000412	TEMP 014774
TEST 013604	TIME 003152	TJSR1 012766	TJSR2 012770
TJSR3 013002	THEMEY 001124	TRCSR 000406	TRPB 000602
TRTRP 016162	TSCOMB 014346	TSRDT 013450	TSRDT2 013614
TSRT2A 013606	TTCSR 000410	TTDBR 000412	TTPST 000416
TTPVC 000414	TTSAY 000420	TYOUT 002764	TYOUTR 003000
TYOUT1 002770	UBUFF 000400	UPAR0 000510	UPAR1 000512
UPART 000514	UPDR0 000502	UPDR1 000504	UPDR7 000506
USEALL 002146	USER 016676	WAIT4 014454	WD 000014
WORDCT 003602	XFENDZ 000006	XFER12 002616	XFER16 002606
XFER20 002576	XFER24 002566	XFER28 002556	XFER8 002626
.	017762		

ERRORS DETECTED: 0

*DFKTGA,DFKTGA,DFKTGA.SRC/SOL
 RUN-TIME: 9 19 0 SECONDS
 CORE USED: 5K