

RC11

MULTI-DISK
MD-11-DZRCC-B

EP-DZRCC-B-DL-A

OCT 1976

COPYRIGHT © 1976

digital

FICHE 1 OF 1

Made in U.S.A.

This microfiche card contains a grid of frames. The frames are arranged in approximately 12 rows and 3 columns. Each frame contains a small, high-contrast image or data set, likely representing a page of a document or a specific data point. The frames are too small to read clearly, but they appear to contain text and possibly some graphical elements. The card is otherwise blank.

CONTENTS

1.	ABSTRACT
2.	REQUIREMENTS
2.1	EQUIPMENT
2.2	STORAGE
2.3	PRELIMINARY PROGRAMS
3.	LOADING PROCEDURE
4.	STARTING PROCEDURE
5.	OPERATING PROCEDURE
5.1	OPERATIONAL SWITCH SETTINGS
5.2	STOPPING THE PROGRAM
5.3	PROGRAM ABSTRACT
6.	ERRORS
7.	RESTRICTIONS
9.	MISCELLANEOUS
8.1	EXECUTION TIME
9.	PROGRAM DESCRIPTION

MAINDEC-11-DZRCC-B
DZRCCB.P11

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

MAINDEC-11 DZRCO-B
DESCRIPTION

RC11 MULTI DISK TEST

PAGE 3

1. ABSTRACT

MULTI DISK IS A HIGH SPEED CONFIDENCE TEST THAT ASSURES THE USER THAT HE CAN TRANSFER DATA CORRECTLY, WITHOUT DESTROYING THE DATA ON THE DISK. MULTI DISK USES ALL EXISTING MEMORY ON THE SYSTEM AS BUFFER AREAS.

2. REQUIREMENTS

2.1 EQUIPMENT

PDP-11 4K TO 28K OF MEMORY
RC11 AND RS64 PLUS UP TO THREE ADDITIONAL RS64'S

2.2 STORAGE

THE MAIN BODY OF THE PROGRAM OCCUPIES THE FIRST 5K OCTAL (BYTES) OF MEMORY. THREE DATA BUFFERS OCCUPY THE REST OF EXISTING MEMORY.

BUFFER ORDER

- OUT BUFFER = RANDOM DATA TO BE WRITTEN ON THE DISK.
- IN BUFFER = AREA FOR RANDOM DATA WHEN READ FROM THE DISK.
- SAVE BUFFER = AREA TO SAVE DISK WHEN EXERCISING.

2.3 PRELIMINARY PROGRAMS

MAINDEC-11-DZRCA-B AND DZRCB-B FORMALLY MAINDEC-11-DSJA

3. LOADING PROCEDURE

PROGRAM FORMAT ABSOLUTE

- A. VERIFY THE BOOT LOADER IS IN MEMORY.
- B. LOAD TAPE INTO READER.
- C. SET SWITCH REGISTER EQUAL TO *500

136
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
188
189

MAINDEC-11 DZRCB-B
DESCRIPTION

RC11 MULTI DISK TEST

PAGE 4

MEMORY SIZE*

4K	17
8K	37
12K	57
16K	77
20K	117
24K	137
28K	157

D. DEPRESS LOAD ADDRESS.

E. DEPRESS START.

4. STARTING PROCEDURE

A. SET SWITCH REGISTER EQUAL TO 200.

B. DEPRESS LOAD ADDRESS.

C. DEPRESS START.

D. THE PROGRAM AT THIS POINT WILL DETERMINE THE AMOUNT OF EXISTING MEMORY, AND SET UP THE WORD COUNT AND BUFFER AREAS FOR THE PROGRAM.

E. THE PROGRAM WILL NEXT REPORT THE NUMBER OF EXISTING DISK(S) WHICH ARE ON THE SYSTEM. THE NUMBER WILL RANGE FROM 1 - 4 OCTAL. THE DISK SELECTION LINES MUST BE IN SEQUENTIAL ORDER RANGING FROM 0 TO 3.

02 EXISTENT DISK(S)

5. OPERATING PROCEDURE

5.1 OPERATIONAL SWITCH SETTINGS

THE SWITCH REGISTER HAS NO CONTROL OVER THE OPERATION OF THIS PROGRAM.

5.2 STOPPING THE PROGRAM

MULTI DISK MUST ONLY BE STOPPED BY TYPING CONTROL C. IF THE USER STOPS THE PROGRAM BY DEPRESSING CONSOLE HALT, THE DISK SURFACE MAY NOT BE RESTORED.

190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243

MAINDEC-11 DZRCB-B
DESCRIPTION

RC11 MULTI DISK TEST

PAGE 5

5.3 PROGRAM ABSTRACT

MULTI DISK WAS DESIGNED TO INSURE THE USER THAT THE DISK SYSTEM IS CAPABLE OF TRANSFERRING DATA CORRECTLY WHILE NOT DESTROYING THE USERS PROGRAMS ON THE DISK SURFACE. THE PROGRAM FIRST READS FROM THE DISK, THE LENGTH OF THE TRANSFER IS DETERMINED BY THE SIZE OF MEMORY. IF AN ERROR OCCURS WHILE READING, THE PROGRAM WILL MAKE UP TO THREE ATTEMPTS AT READING THE DATA. IF THE ERROR STILL EXISTS, THE PROGRAM WILL THEN HALT. IF THE PROGRAM SUCCESSFULLY READS FROM THE DISK WITHIN THE THREE ATTEMPTS, IT WILL THEN GENERATE A RANDOM BUFFER, WRITE IT ON THE DISK, AND READ IT BACK AND VERIFY IT. AFTER COMPARING THE DATA, THE PROGRAM THEN WRITES THE ORIGINAL DATA BACK ON THE DISK, MAKING UP TO THREE ATTEMPTS TO TRANSFER IF AN ERROR IS ENCOUNTERED, BEFORE HALTING. IF THE DATA WAS SUCCESSFULLY TRANSFERRED, THE PROGRAM WILL GO TO THE NEXT DISK BUFFER UNTIL THE COMPLETE DISK SYSTEM IS EXERCISED.

NOTE: EACH WRITE IS FOLLOWED BY A WRITE CHECK.

6. ERRORS

6.1 ERROR HALTS

THE PROCESSOR HALTED AT LOC. 1514. CAUSE OF HALT: NO SELECTED DISK ON THE SYSTEM, OR DISKS NOT SELECTED IN SEQUENTIAL ORDER STARTING WITH DISK 0.

THE PROCESSOR HALTED AT LOC. 2062. THE CAUSE OF THE HALT: UNABLE TO READ FROM THE DISK FOR THE SAVE BUFFER. THREE ATTEMPTS WERE MADE.

THE PROCESSOR HALTED AT LOC. 2370. THE CAUSE OF THE HALT, UNABLE TO RESTORE THE DISK SURFACE. THREE ATTEMPTS WERE MADE.

6.2 ERROR REPORTS

STATUS ERROR

STATUS ERROR XXXXXX DAR XXXXXX DCS XXXXXX DAE
 A B C

A= THE DISK ADDRESS REGISTER
B= THE DISK CONTROL REGISTER
C= THE DISK ERROR REGISTER

245
246
247
248
249
250
251
252
253
254
255
256
257
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

MAINDEC-11 DZRCC-B
DESCRIPTION

RC11 MULTI DISK TEST

PAGE 6

BIT LAYOUT OF DCS REGISTER

BIT15= ERROR
BIT14= DATA ERROR
BIT13= ADDRESS ERROR
BIT12= WRITE LOCK ERROR
BIT11= NON-EXISTENT DISK
BIT10= WRITE CHECK ERROR
BIT9= INHIBIT INCREMENTING CA
BIT8= ABORT
BIT7= READY
BIT6= INTERRUPT ENABLE
BIT5= EXTENDED MEMORY 1 (XM1)
BIT4= EXTENDED MEMORY 0 (XM0)
BIT3= MAINTENANCE
BIT2-1= FUNCTION REGISTER

BIT 2	BIT 1	OPERATION
0	0	LOOK AHEAD
1	0	READ
0	1	WRITE
1	1	WRITE CHECK

BIT0= GO (WRITE ONLY BIT)

NOTE: IF AN ERROR OCCURS, THE FOLLOWING INFORMATION IS AVAILABLE TO THE USER IN THE DAE:

BIT4= ADDRESS NOT FOUND
BIT5= DISK OVERFLOW
BIT6= ADDRESS SYNC. ERROR
BIT7= ADDRESS PARITY ERROR
BIT8= B TRACK ERROR
BIT9= A TRACK ERROR
BIT12= NONEXISTENT MEMORY
BIT13= DATA SYNC. ERROR
BIT14= BLOCK CHECK ERROR
BIT15= DATA LATE

6.3 DATA ERRORS

DATA ERROR XXXXXX DAR XXXXXX GOOD DATA XXXXXX BAD DATA
A B C

A= THE DISK ADDRESS REGISTER
B= THE DATA WRITTEN ON THE DISK

H01

MAINDEC-11-DZRCB-B
DZRCB.P11

RC11 MULTI DISK

REPLACES DSBA-PB

MACY11 27(732) 10-SEP-76 15:10 PAGE 7

301

C= THE DATA READ FROM THE DISK

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
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357

MAINDEC-11 DZRCB-B
DESCRIPTION

RC11 MULTI DISK TEST

PAGE 7

- 6.4 DISK ADDRESS ERROR
DISK ADDRESS ERROR XXXXXX DAR
THE TERMINATING DISK ADDRESS AFTER THE TRANSFER WAS NOT CORRECT THE DAR SHOULD EQUAL WHAT WAS REPORTED.
- 6.5 PROCESSOR TIME OUT
PROCESSOR BACKGROUND TIMED OUT.
THIS MESSAGE WILL BE REPORTED IF THE DISK FAILS TO RAISE A BR REQUEST AFTER EXTENDED PERIOD OF TIME.
- 6.7 PARITY ERROR
THIS MESSAGE IS REPORTED IF THE PROGRAM DETECTS A MEMORY PARITY ERROR DURING PROGRAM EXECUTION.
- 6.8 END
END
THIS MESSAGE IS REPORTED AT THE END OF ONE COMPLETE PASS OF THE DISK SYSTEM.
- 7. RESTRICTIONS
THE DISKS MUST BE IN SEQUENTIAL ORDER RANGING FROM DISK SELECT 0 - 3.
- 8. MISCELLANEOUS
- 8.1 EXECUTION TIME
12 SEC. PER DISK
- 9. PROGRAM DESCRIPTION
MULTI DISK IS NOT A DIAGNOSTIC - IT IS MERELY A CONFIDENCE TEST. IF THE USER ENCOUNTERS ANY PROBLEMS WITH THE SYSTEM WHILE OPERATING MULTI DISK, HE SHOULD NOT TRY TO DIAGNOSE THE PROBLEM USING THIS PROGRAM. IF PROBLEMS ARE ENCOUNTERED, IT IS SUGGESTED THAT THE USER REVERTS TO DISK

J01

MAINDEC-11-DZRCB-B
DZRCB.P11

RC11 MULTI DISK

REPLACES DSBA-PB

MACY11 27(732) 10-SEP-76 15:10 PAGE 9

358

DATA AS A MEANS OF DIAGNOSING HIS PROBLEM.

```

359                                     %
360      .TITLE MAINDEC-11-DZRCC-B      RC11 MULTI DISK      REPLACES DSBA-PB
361      :COPYRIGHT 1971, 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS
362      ;PROGRAM BY BOB BRAIN/C. CASWELL
363
364      .ENABL ABS
365      BIT0= 1
366      BIT1= 2
367      BIT2= 4
368      BIT3= 10
369      BIT4= 20
370      BIT5= 40
371      BIT6= 100
372      BIT7= 200
373      BIT8= 400
374      BIT9= 1000
375      BIT10= 2000
376      BIT11= 4000
377      BIT12= 10000
378      BIT13= 20000
379      BIT14= 40000
380      BIT15= 100000
381
382      WRITE= TRAP+3
383      WRCHECK=TRAP+7
384      READ= TRAP+5
385
386      ;STATIC ROUTINES
387
388      ;RC11 DATA TEST
389      ;VECTORS USED IN PROGRAM
390      ;#1 LOC 204 DISK INTERRUPT
391      ;#2 LOC 30 EMT (TELETYPE OUTPUT)
392      ;#3 LOC 34 TRAP (DISK HANDLERS)
393      ;#4 LOC 14 TRACE TRAP (USED IN BACKGROUND TEST)
394      ;#5 LOC 20 IOT TRAP (USED IN CALLING BACKGROUND TEST)
395
396      000000      .= 0      ;TRAP CATCHER 0 - 776
397      000200      .= 200
398
399      000200 000167 000720      JMP START
400
401      001000      .= 1000
402
403      ;RC11 MULTIDISK
404      ;VECTORS USED IN PROGRAM
405      ;#1 LOC 204 DISK INTERRUPT
406      ;#2 LOC 30 EMT (TELETYPE OUTPUT)
407      ;#3 LOC 34 TRAP (DISK HANDLER)
408      ;#4 LOC 14 TRACE TRAP (USED IN BACKGROUND TEST)
409      ;#5 LOC 20 IOT TRAP (USED IN CALLING BACKGROUND TEST)

```

```

410 ;I/O ADDRESS POINTERS
411
412 001000 177570 SWR: 177570 ; SWITCH REGISTER
413 001002 177776 PS: 177776 ; PROCESSOR STATUS REGISTER
414 001004 177566 TPB: 177566 ; TELETYPE REGISTERS
415 001006 177562 TKB: 177562
416 001010 177564 TPS: 177564
417 001012 177560 TKS: 177560
418
419 ;DISK I/O REGISTERS
420
421 001014 177446 RCCS: 177446 ; DISK CONTROL REGISTER
422 001016 177450 RCWC: 177450 ; WORD COUNT REGISTER
423 001020 177452 RCBA: 177452 ; CURRENT ADDRESS REGISTER
424 001022 177442 RCDA: 177442 ; LOWER 16 BITS OF DISK ADDRESS
425 001024 177444 RCER: 177444 ; EXTENSION ADDRESS REGISTER
426 001026 177456 RCDB: 177456 ; DATA BUFFER REGISTER
427 001030 177454 RCMR: 177454 ; RCMRINTENANCE REGISTER
428 001032 177440 RCLA: 177440 ; LOOK AHEAD REGISTER
429 001034 000210 VECTOR: 210 ; INTERRUPT VECTOR ADDRESS
430 001036 000212 STATUS: 212 ; DISK INTERRUPT STATUS
431 001040 000200 PRIORITY:BIT7 ; DISK PRIORITY LEVEL
432
433 ;RC11 DEDICATE REGISTERS (MEMORY)
434
435 001042 000000 FLAG: 0 ; INTERNAL PROGRAM FLAG
436 001044 146723 RANNU: 146723 ; RANDOM NUMBER PRIME
437 001046 000000 WRDCT: 0 ; WORKING WORD COUNT
438 001050 000000 DMA: 0 ; WORKING DATA BUFFER (OUT-IN)
439 001052 000000 PATNU: 0 ; DATA PATTERN INDEX
440 001054 000000 BUF: 0 ; WORKING DATA BUFFER (OUT-IN)
441 001056 000000 TWRDCT: 0 ; TEMP WORD COUNT
442 001060 000000 TDMA: 0 ; TEMP RCDA
443 001062 000000 SWRDCT: 0 ; STANDARD WORD COUNT
444 001064 000000 ERCOUNT:0 ; ERROR COUNT FOR MESSATES.
445 001066 000000 SAVE: 0
446 001070 000000 SAV1: 0
447 001072 000000 PASS: 0
448 001074 000000 DSKNOR: 0
449 001076 000000 HRDER: 0 ; POINTER FOR HARD ERROR
450 001100 000000 LIMIT: 0
451 001102 000000 TTRACK: 0 ; TEMP TRACK NO.
452 001104 000000 TDSKNO: 0 ; TEMP DISK NO.
453 001106 005062 OUTBUF: BUFFER
454 001110 000000 INBUF: 0
455 001112 000000 DISKNO: 0
456 001114 000000 SAVBUF: 0
457
458 ;RC11 WORK REGISTERS
459 ; (CAN BE CHANGED IN ANY ROUTINE)
460 001116 000000 WORK: 0
461 001120 000000 WORK1: 0
462 001122 000000 WORK2: 0

```

MO1

MAINDEC-11-DZRCB-B
DZRCB.P11

RC11 MULTI DISK

REPLACES D5BA-PB

MACY11 27(732) 10-SEP-76 15:10 PAGE 12

```

463 001124 000005          START:  RESET                ;CLEAR THE WORLD
464 001126 012706 001000      MOV      #1000,%6          ;SET UP STACK
465 001132 005777 177642      TST      @SWR
466 001136 100003          BPL      .+10
467 001140 052767 000020 176630  BIS      #20,177776
468 001146 012777 002450 177660  MOV      #DKINT,@VECTOR ;SET UP DISK POINTER
469 001154 012777 000340 177654  MOV      #340,@STATUS  ;LOCK UP INTERRUPTS
470 001162 012767 003662 176640  MOV      #EMTRP,30     ;SET UP TTY POINTER
471 001170 012767 000340 176634  MOV      #340,32      ;LOCK UP INTERRUPTS
472 001176 012767 002374 176630  MOV      #DISK,34      ;SET UP DISK HANDLER POINTER
473 001204 012767 000340 176624  MOV      #340,36      ;LOCK UP INTERRUPTS
474 001212 012777 000340 177562  MOV      #340,@PS     ;LOCK UP INTERRUPT LEVELS
475 001220 005067 177616      CLR      FLAG          ;CLEAR PROGRAM FLAG
476 001224 005067 177620      CLR      DMA           ;CLEAR RCDA REGISTERS
477 001230 005067 177616      CLR      PATNU        ;CLEAR PATTERN COUNT
478 001234 012767 003476 176556  MOV      #XWAIT,20
479 001242 005067 176554      CLR
480 001246 012767 004360 176570  MOV      #DOWN,24     ;SET UP PWR FAIL
481 001254 012767 000340 176544  MOV      #340,26      ;LOCK UP INTERRUPTS
482 001262 004767 002714      JSR      %7,MAMF      ;SET UP PARITY SWITCHES
483
484                                     ;THIS ROUTINE DETERMINES HOW MUCH MEMORY IS ON THE SYSTEM
485
486 001266 012777 000340 177506      MOV      #340,@PS     ;LOCK UP PRIORITY LEVELS
487 001274 013767 000042 177564      MOV      @#42,SAVE    ;GET MONITOR ADDRESS
488 001302 100433          BMI      LGMEM         ;GREATER THAN 16K
489 001304 001407          BEQ      1$           ;SKIP IF 0
490 001306 162767 000020 177552      SUB      #20,SAVE     ;DEC IT
491 001314 022767 005062 177544      CMP      #BUFFER,SAVE ;IS IT ACT11?
492 001322 100434          BMI      GOTMEM       ;NO - SKIP
493 001324 012767 001402 176452 1$:  MOV      #MAXREF,4    ;SET UP I/O BUS TRAP
494 001332 012767 000340 176446      MOV      #340,6
495 001340 012767 017446 177520      MOV      #17446,SAVE  ;SET UP REFERENCE TO 4K
496 001346 005777 177514      EXREF:  TST      @SAVE ;REFERENCE MEMORY
497 001352 022767 177446 177506      CMP      #177446,SAVE ;TEST FOR GREATER THAN 28K
498 001360 001410          BEQ      MAXREF
499 001362 062767 020000 177476      ADD      #20000,SAVE  ;SET UP FOR NEXT REFERENCE
500 001370 000766          BR       EXREF        ;REFERENCE NEXT BANK
501 001372 162767 000020 177466  LGMEM:  SUB      #20,SAVE ;DEC IT
502 001400 000405          BR       GOTMEM
503 001402 012706 001000      MAXREF: MOV      #1000,%6  ;RESET STACK
504 001406 162767 020000 177452      SUB      #20000,SAVE ;
505
506                                     ;ROUTINE TO REPORT THE NUMBER OF EXISTENT DISK(S)
507                                     ;ON THE SYSTEM.
508 001414 005067 177476      GOTMEM: CLR      WORK
509 001420 016777 177472 177374  INDSK:  MOV      WORK,@RCDA
510 001426 032777 004000 177360      BIT      #BIT11,@RCCS ;TEST FOR NON-EXISTENT DISK
511 001434 001010          BNE      NEDST        ;BRANCH IF NON EXISTENT DISK SET
512 001436 062767 004000 177452      ADD      #4000,WORK   ;SET UP NEXT DISK NO.
513 001444 022767 020000 177444      CMP      #20000,WORK  ;CHECK FOR 4 DISKS
514 001452 001401          BEQ      .+4         ;BRANCH IF 4 DISKS
515 001454 000761          BR       INDSK

```

516	001456	000367	177434		NEDST:	SWAB	WORK	
517	001462	105067	177431			CLRB	WORK+1	
518	001466	000241				CLC		
519	001470	006067	177422			ROR	WORK	
520	001474	006067	177416			ROR	WORK	
521	001500	006067	177412			ROR	WORK	
522	001504	032767	000007	177404		BIT	#7,WORK	
523	001512	001002				BNE	.+6	
524	001514	000000				HALT		
525	001516	000777				BR		
526	001520	004567	002376			JSR	%5,CONV	
527	001524	001116				WORK		
528	001526	004715				MES11		
529	001530	000002				2		
530	001532	104000				EMT	+0	
531	001534	004547				HED5A		
532	001536	004715				MES11		
533	001540	177777				-1		
534	001542	005367	177350			DEC	WORK	
535	001546	006167	177344			ROL	WORK	
536	001552	006167	177340			ROL	WORK	
537	001556	006167	177334			ROL	WORK	
538	001562	000367	177330			SWAB	WORK	
539	001566	042767	163777	177322		BIC	#163777,WORK	
540	001574	016767	177316	177272		MOV	WORK,DSKNOR	
541								
542								
543								
544	001602	005067	177304			CLR	DISKNO	
545	001606	166767	177274	177252		SUB	OUTBUF,SAVE	
546	001614	005067	177276			CLR	WORK	
547	001620	162767	000003	177240	SETBUF:	SUB	#3,SAVE	;DIVIDE MEMORY BY 3
548	001626	022767	000003	177232		CMP	#3,SAVE	
549	001634	101003				BHI	OVERFLO	;BRANCH IF NO REMAINDER
550	001636	005267	177254			INC	WORK	;+1 COUNT
551	001642	000766				BR	SETBUF	
552	001644	042767	000001	177244	OVERFLO:	BIC	#BIT0,WORK	
553	001652	016767	177230	177230		MOV	OUTBUF,INBUF	;SET UP IN-BUFFER
554	001660	066767	177232	177222		ADD	WORK,INBUF	
555	001666	016767	177216	177220		MOV	INBUF,SAVBUF	;SET UP SAVE BUFFER
556	001674	066767	177216	177212		ADD	WORK,SAVBUF	
557	001702	016767	177210	177152		MOV	WORK,SWRDCT	;SET UP WORD COUNT
558	001710	000241				CLC		
559	001712	006067	177144			ROR	SWRDCT	;DIVIDE BYTES X 2
560	001716	042767	000037	177136		BIC	#37,SWRDCT	
561	001724	012767	003626	176126		MOV	#CNTLC,60	;SET UP TTY VECTOR
562	001732	012767	000340	176122		MOV	#340,62	
563	001740	052777	000100	177044		BIS	#BIT6,DTKS	;ENABLE TTY INTERRUPT
564	001746	016767	177110	177072	DATAT:	MOV	SWRDCT,WRDCT	;SET UP WORD COUNT
565	001754	012767	001746	177114		MOV	#DATAT,HRDR	;SET UP FOR HARD ERROR
566	001762	005067	177062			CLR	DMA	
567	001766	012777	000340	177006	RDSAV:	MOV	#340,@PS	;LOCK UP PROCESSOR PRIORITY
568	001774	016767	177114	177052		MOV	SAVBUF,BUF	;SET UP CURRENT ADDR
569	002002	042767	000003	177032		BIC	#3,FLAG	

;ROUTINE TO SET UP WRITE/READ BUFFERS

;DIVIDE MEMORY BY 3

;BRANCH IF NO REMAINDER

;+1 COUNT

;SET UP IN-BUFFER

;SET UP SAVE BUFFER

;SET UP WORD COUNT

;DIVIDE BYTES X 2

;SET UP TTY VECTOR

;ENABLE TTY INTERRUPT

;SET UP WORD COUNT

;SET UP FOR HARD ERROR

;LOCK UP PROCESSOR PRIORITY

;SET UP CURRENT ADDR

570	002010	042767	001000	177024	LDAT:	BIC	#BIT9,FLAG	
571	002016	005267	177020			INC	FLAG	
572	002022	104505				READ	+100	
573	002024	000004				IOT		:WAIT FOR FLAG IN BACKGROUND TEST
574	002026	032767	001000	177006		BIT	#BIT9,FLAG	:CHECK FOR ERROR
575	002034	001414				BEQ	WRDAT	:WRITE RANDOM DATA
576	002036	016767	177000	177052		MOV	FLAG,WORK	:TRY TO READ 3 TIMES
577	002044	042767	177774	177044		BIC	#177774,WORK	
578	002052	022767	000003	177036		CMP	#3,WORK	
579	002060	001353				BNE	LDAT	
580	002062	000000				HALT		:UNABLE TO READ FROM
581	002054	000777				BR		:DISK
582	002066	012767	002206	177002	WRDAT:	MOV	#WRSV,HRDR	:SET UP FOR HARD ERROR
583	002074	016767	177006	176752		MOV	OUTBUF,BUF	:SET UP RANDOM BUFFER
584	002102	104503				WRITE	+100	
585	002104	000004				IOT		:WAIT FOR FLAG IN BACK GROUND
586	002106	042767	000003	176726		BIC	#3,FLAG	
587	002114	016767	176770	176732		MOV	INBUF,BUF	:SET UP CORRENT ADDR
588	002122	042767	001000	176712	RAND:	BIC	#BIT9,FLAG	
589	002130	005267	176706			INC	FLAG	
590	002134	104505				READ	+100	
591	002136	000004				IOT		:WAIT FOR FLAG IN BACKGROUND
592	002140	004567	001126			JSR	%5,COMPARE	:COMPARE DATA
593	002144	032767	001000	176670		BIT	#BIT9,FLAG	:CHECK FOR ERROR
594	002152	001415				BEQ	WRSV	:RESTORE DISK BUFFER
595	002154	016767	176662	176734		MOV	FLAG,WORK	:CHECK FOR 3 RE-READS
596	002162	042767	177774	176726		BIC	#177774,WORK	
597	002170	022767	000003	176720		CMP	#3,WORK	
598	002176	001351				BNE	RAND	:GC RE-READ DATE
599	002200	042767	000003	176634		BIC	#3,FLAG	
600	002206	042767	001000	176626	WRSV:	BIC	#BIT9,FLAG	:CLEAR PROGRAM ERROR FLAG
601	002214	016767	176674	176632		MOV	SAVBUF,BUF	:SET UP CURRENT ADDRESS
602	002222	104503				WRITE	+100	
603	002224	000004				IOT		:WAIT IN BACKGROUND TEST
604	002226	104507				WRCHECK	+100	
605	002230	000004				IOT		:WAIT FOR FLAG
606	002232	032767	001000	176602		BIT	#BIT9,FLAG	:CHECK FOR ERROR
607	002240	001413				BEQ	STORBU	
608	002242	016767	176574	176646		MOV	FLAG,WORK	
609	002250	042767	177774	176640		BIC	#177774,WORK	
610	002256	022767	000003	176632		CMP	#3,WORK	
611	002264	001350				BNE	WRSV	
612	002266	000000				HALT		

613	002270	005767	176546		STORBU:	TST	FLAG	
614	002274	100435				BMI	RESTORE	:RESTORE DISK AND EXIT
615	002276	004767	000342			JSR	%7,DISBUF	:SET UP NEXT DISK BUFFER
616	002302	000167	177460			JMP	RDSAV	:READ NEXT BUFFER
617	002306	026767	176608	176560		CMP	DISKNO,DSKNOR	
618	002314	001405				BEG	STEM	
619	002316	062767	004000	176566		ADD	#4000,DISKNO	
620	002324	000167	177436			JMP	RDSAV	
621	002330	016767	176526	176510	STEM:	MOV	SWROCT,WRDCT	
622	002336	005067	176550			CLR	DISKNO	
623	002342	104001				EMT	+1	:REPORT END.
624	002344	005054				END		
625	002346	013700	000042			MOV	#42,%0	:GET MONITOR ADDRESS
626	002352	001404				BEQ	LXIT	:SKIP IF NO HOOK
627	002354	004710			LOGICAL:	JSR	7,(0)	:GO TO MONITOR
628	002356	000240				NOP		
629	002360	000240				NOP		
630	002362	000240				NOP		
631	002364	000167	177356		LXIT:	JMP	DATAT	:RECYCLE TEST
632								
633	002370	000000			RESTORE:	HALT		
634	002372	000777				BR		:DO NOT RESTORE MEM
635								
636								
637								
638								
639	002374	012705	001024					
640	002400	016755	176444					
641	002404	056775	176502	000000	DISK:	MOV	#RCR,%5	:SET UP TO LOAD DISK REG
642	002412	016755	176436			MOV	DMA,%(5)	:LOAD WORD ADDRESS
643	002416	016755	176424			BIS	DISKNO,%(5)	:GET DISK NO.
644	002422	005475	000000			MOV	BUF,%(5)	:SET UP CURRENT ADDRESS
645	002426	011604				MOV	WRDCT,%(5)	:LOAD WORD COUNT
646	002430	014467	176462			NEG	%(5)	:SET UP TWO'S COMPLEMENT
647	002434	042767	177600	176454		MOV	%(5),%4	
648	002442	016755	176450			MOV	-(4),WORK	
649	002446	000002				BIC	#177600,WORK	:MASK FUNCTION BITS
						MOV	WORK,%(5)	:LOAD FUNCTION REG.
						RTI		:RETURN FROM TRAP


```

650          :RC11 DISK INTERRUPT HANDLER
651          :ROUTINE CONTINUES ON ERRORS
652 002450 005046          DKINT: CLR      -(6)          ;CLEAR STACK
653 002452 012746 002460  MOV      #1$,-(6)          ;SET RETURN
654 002456 000002          RTI              ;CLEAR T BIT
655 002460 005777 176330  1$:  TST      @RCCS          ;TEST FOR ERROR
656 002464 100064          BPL      INTEXT        ;BRANCH IF NO ERROR
657 002466 052767 001000 176346  BIS      @BIT9,FLAG      ;SET ERROR BIT
658 002474 104001          EMT+1
659 002476 004455          HED2
660 002500 017767 176316 176410 DELMES: MOV      @RCD,WORK      ;SET UP LOWER 16 BITS OF ADDR.
661 002506 032777 074000 176306  BIT      @74000,@RCD    ;HARD ERROR?
662 002514 001002          SNE      .+6
663 002516 005367 176374          DEC      WORK
664 002522 004567 001374          JSR      %5,CONV        ;CONVERT TO ASCII
665 002526 001116          WORK
666 002530 004551          MES1
667 002532 000006          6
668 002534 104001          EMT+1
669 002536 004551          MES1
670 002540 017767 176250 176350  MOV      @RCCS,WORK      ;SET UP STATUS
671 002546 004567 001350          JSR      %5,CONV
672 002552 001116          WORK
673 002554 004566          MES2
674 002556 000006          6
675 002560 104001          EMT+1
676 002562 004566          MES2
677 002564 017767 176234 176324  MOV      @RCER,WORK      ;FETCH ERROR EXT. BITS
678 002572 004567 001324          JSR      %5,CONV        ;CONVERT TO ASCII
679 002576 001116          WORK
680 002600 004603          MES2A
681 002602 000006          6
682 002604 104001          EMT+1
683 002606 004603          MES2A
684 002610 012706 001000          MOV      #1000,%6
685 002614 000177 176256          JMP      @RDR
686 002620 005777 176172          SOFTER: TST      @RCWC
687 002624 001404          BEQ      INTEXT
688 002626 052777 000001 176160  BIS      @BIT0,@RCCS
689 002634 000002          RTI
690 002636 012706 000774          INTEXT: MOV      @774,%6
691 002642 000002          RTI

```

```

692          ;ROUTINE TO SETUP DISK BUFFERS
693          ;ADD WORD COUNT TO STARTING DISK ADDRESSES
694          ;COMPARE CALCULATED ADDRESS TO TERMINATING ADDRESS
695
696 002644 016767 176176 176244 DISBUF: MOV      WRDCT,WORK
697 002652 032767 000037 176236          BIT      #37,WORK
698 002660 001403          SEQ      .+10
699 002662 062767 000040 176226          ADD      #40,WORK
700 002670 042767 000037 176220          BIC      #37,WORK
701 002676 000241          CLC
702 002700 006067 176212          ROR      WORK
703 002704 006067 176206          ROR      WORK
704 002710 006067 176202          ROR      WORK
705 002714 006067 176176          ROR      WORK
706 002720 006067 176172          ROR      WORK
707 002724 066767 176166 176116          ADD      WORK,DMA          ;ADD WORD COUNT TO LOWER 16 BITS
708 002732 016767 176154 176162          MOV      DISKNO,WORK2
709 002740 066767 176104 176154          ADD      DMA,WORK2
710 002746 042767 020000 176146          BIC      #2000,WORK2
711 002754 026777 176142 176040 COMDAR: CMP      WORK2,ARCDA          ;COMPARE LOWER 16 BITS
712 002762 001047          BNE      ERADR
713
714 002764 105767 176052          CMDA:  TSTB     FLAG          ;CHECK FOR LAST DISK BUFFER
715 002770 100013          BPL      EXTCME
716 002772 005067 176052          CLR      DMA          ;CLEAR LOWER 16 BITS
717 002776 042767 000200 176036          BIC      #200,FLAG
718 003004 062716 000004          ADD      #4,(6)          ;INC STOCK POINTER
719 003010 016767 176046 176030          MOV      SWRDCT,WRDCT
720 003016 000207          RTS      %7
721
722 003020 017767 175776 176070 EXTCME: MOV      ARCA,WORK          ;FETCH LOWER 16 BITS OF ADDRESS
723 003026 000241          CLC
724 003030 006167 176062          ROL      WORK
725 003034 006167 176056          ROL      WORK
726 003040 006167 176052          ROL      WORK
727 003044 006167 176046          ROL      WORK
728 003050 006167 176042          ROL      WORK
729 003054 066767 175766 176034          ADD      WRDCT,WORK          ;WILL DISK OVERFLOW
730 003062 103006          BCC      AKH
731 003064 052767 000200 175750          BIS      #200,FLAG
732 003072 166767 176020 175746          SUB      WORK,WRDCT
733 003100 000207          AKH:  RTS      %7          ;EXIT
734 003102 004567 001014          ERADR: JSR      %5,CONV          ;CONVERT DMA REG COUNT TO ASCII
735 003106 001122          WORK2
736 003110 004551          MES1
737 003112 000006          6
738 003114 104000          EMT      +0          ;REPORT ERROR
739 003116 004522          HED4
740 003120 004551          MES1
741 003122 177777          -1
742 003124 000000          HALT
743 003126 000777          BR

```

```

744          ;ROUTINE TO FILL BUFFERS WITH RANDOM DATA
745
746          ;ENTER FROM JSR %7,RANEX
747
748 003130 016767 175712 175760 RANEX: MOV     WRDCT,WORK      ;SET UP WORK
749 003136 012705 001106          MOV     #OUTBUF,%5      ;LOC. OF OUTBUFFER
750
751          ;RANDOM DATA GENERATOR SUBROUTINE
752
753 003142 016700 000120 DAYGEN: MOV     LONUM,%0      ;SET UP R0 WITH 5 DIGITS LOW
754 003146 016701 000116          MOV     HINUM,%1      ;SET UP R1 WITH 5 DIGITS HIGH
755 003152 012703 000007          MOV     #7,%3        ;SET UP SHIFT COUNT
756 003156 005002          CLR     %2          ;CLEAR R2
757 003160 006300          SHIFT: ASL    %0      ;SHIFT R0 LEFT AND
758 003162 006101          ROL    %1          ;ROTATE CARRY INTO LSB OF R1 INTO
759 003164 006102          ROL    %2          ;ROTATE CARRY OUT OF R1 INTO R2
760 003166 005303          DEC    %3        ;DECREMENT R3
761 003170 001373          BNE    SHIFT      ;CONTINUE SHIFT LOOP
762 003172 066700 000070          ADD    LONUM,%0    ;ADDN IN NUMBER TO MAKE X 129
763 003176 005501          ADC    %1          ;PROPOGATE CARRY
764 003200 066701 000064          ADD    HINUM,%1    ;ADDN IN NUMBER TO MAKE X 129
765 003204 005502          ADC    %2          ;PROPOGATE CARRY
766 003206 062700 001057          ADD    #1057,%0    ;ADDN LOW CONSTANT
767 003212 005501          ADC    %1          ;PROPOGATE CARRIES
768 003214 005502          ADC    %2          ;PROPOGATE AGAIN
769 003216 062701 047401          ADD    #47401,%1   ;ADDN HIGH CONSTANT
770 003222 005502          ADC    %2          ;PROPOGATE CARRY
771 003224 062702 000006          ADD    #6,%2       ;ADDN HIGHEST CONSTANT
772 003230 060200          ADD    %2,%0       ;REPRIME R0 WITH HIGH DIGIT
773 003232 005501          ADC    %1          ;PROPOGATE CARRY
774 003234 010067 000026          MOV    %0,LONUM    ;PUT R0 BACK IN LONUM
775
776 003240 010025          MOV    %0,(5)+     ;HOLD LONUM FOR PROGRAM
777 003242 005367 175650          DEC    WORK
778 003246 001406          BEQ    EXGEN
779 003250 010167 000014          MOV    %1,HINUM    ;PUT R1 BACK IN HINUM
780 003254 010125          MOV    %1,(5)+     ;HOLD HINUM FOR PROGRAM
781 003256 005367 175634          DEC    WORK
782 003262 001327          BNE    DAYGEN
783 003264 000207          EXGEN: RTS        ;RETURN TO PROGRAM
784 003266 000000          LONUM: 0
785 003270 000000          HINUM: 0

```

```

786                                     :DATA COMPARISON ROUTINE
787                                     :IF AN ERROR OCCURS BETWEEN THE OUT-BUFFER AND
788                                     :THE IN-BUFFER AN ERROR WILL BE REPORTED IN THE
789                                     :FOLLOWING MANNER
790                                     :DATA ERROR XXXXXX RCDA XXXXXX GOOD DATA XXXXXX BAD DATA
791
792 003272 005067 175624 COMPARE: CLR      WORK2          ;WORD COUNT
793 003276 016767 175604 175562      MOV      OUTBUF,SAVE ;SET UP OUTBUFFER POINTER
794 003304 016767 175600 175556      MOV      INBUF,SAVI ;SET UP IN BUFFER POINTER
795 003312 027777 175550 175550 WRDCMP: CMP      @SAVE,@SAVI ;COMPARE BUFFERS
796 003320 001016      BNE      WDERR          ;WORD IN ERROR
797 003322 005267 175574      WRDINC: INC     WORK2          ;+1 WORD COUNT
798 003326 026767 175514 175566      CMP      WRDCT,WORK2 ;IS COMPLETE BUFFER CHECKED
799 003334 001407      BEQ      ADAM            ;EXIT ROUTINE
800 003336 062767 000002 175522      ADD     #2,SAVE
801 003344 062767 000002 175516      ADD     #2,SAVI
802 003352 000757      BR      WRDCMP
803 003354 000205      ADAM:  RTS      %S          ;COMPARE NEXT WORD
804                                     ;EXIT THIS ROUTINE
805 003356 052767 001000 175456 WDERR: BIS      #BIT9,FLAG ;SET ERROR BIT
806 003364 016767 175460 175524      MOV     DMA,WORK        ;FETCH STARTING DISK ADDR
807 003372 066767 175514 175516      ADD     DISKNO,WORK
808 003400 066767 175516 175510      ADD     WORK2,WORK
809 003406 004567 000510      JSR    %S,CONV          ;CALCULATE FAILING ADDR
810 003412 001116      WORK          ;CONVERT WORD ADDR TO ASCII
811 003414 004551      MES1
812 003416 000006      6
813 003420 017767 175442 175470      MOV     @SAVE,WORK        ;FETCH GOOD DATA
814 003426 004567 000470      JSR    %S,CONV          ;CONVERT GOOD DATA TO ASCII
815 003432 001116      WORK
816 003434 004660      MESS
817 003436 000006      6
818 003440 017767 175424 175450      MOV     @SAVI,WORK        ;FETCH BAD DATA
819 003446 004567 000450      JSR    %S,CONV          ;CONVERT TO ASCII
820 003452 001116      WORK
821 003454 004674      MES6
822 003456 000006      6
823 003460 104000      EMT      +0            ;PRINT MESSAGE
824 003462 004440      HED1
825 003464 004551      MES1
826 003466 004660      MESS
827 003470 004674      MES6
828 003472 177777      -1
829 003474 000712      BR      WRDINC          ;GO COMPARE NEXT WORD

```

```

830          ;BACKGROUND TEST FOR INTERRUPTS
831
832 003476 012767 003624 174310 XWAIT: MOV #RTIX,14 ;SET UP TRACE TRAP
833 003504 005067 174306          CLR 16
834 003510 012737 003532 000010 MOV #15,2#10 ;SET ILLEGAL INST
835 003516 006727 000000          SXT #0 ;TEST FOR 11/45
836 003522 012767 000006 000074 MOV #6,RTIX ;MAKE IT AN RTT
837 003530 000401          BR 25 ;SKIP JUNK
838 003532 022626          15: CMP (6)+,(6)+ ;CLEAR STACK
839 003534 012737 000012 000010 25: MOV #12,2#10 ;RESET I/O
840 003542 005067 175324          CLR PASS ;SET UP TIME BASE
841 003546 012746 000020          MOV #20,-(6) ;SET T BIT ON STACK
842 003552 012746 003560          MOV #.+6,-(6) ;SET RETURN ADDRESS
843 003556 000002          RTI
844 003560 005027 000000          CLR #0
845 003564 005267 177772 XINCW: INC XINCW-2
846 003570 105767 177766          TSTB XINCW-2
847 003574 100373          BPL XINCW
848 003576 005267 175270          INC PASS
849 003602 001401          BEQ .+4
850 003604 000765          BR XINCW-4
851          ;REPORT BACKGROUND TEST TIMED OUT
852 003606 005046          CLR -(6) ;CLEAR PS ON STACK
853 003610 012746 003616          MOV #15,-(6) ;SET RETURN
854 003614 000002          RTI ;CLEAR TRACE BIT
855 003616 104001          15: EMT+1
856 003620 004745          TIMO
857 003622 000000          HALT
858 003624 000002          RTIX: RTI
859
860          ;TEST FOR CONTROL "C"
861          ;UPON RECEIPT OF CNTLC SET FLAG BIT
862
863 003626 017767 175154 175262 CNTLC: MOV @TKB,WORK ;FETCH KEYBOARD BUFFER
864 003634 042767 000200 175254          BIC #BIT7,WORK ;MASK FOR 33
865 003642 022767 000003 175246          CMP #3,WORK ;TEST FOR CNTLC
866 003650 001003          BNE .+10 ;NOT CNTLC EXIT
867 003652 052767 100000 175162          BIS #BIT15,FLAG ;SET FLAG BIT
868 003660 000002          RTI ;EXIT ROUTINE

```

```

869      ;ROUTINE TO DECODE EMT CALLS
870      ;EMT+1=TYPE ONE LINE OF TEXT
871      ;EMT+0=TYPE A SERIES OF LINES
872
873      003662 011600      EMTRP:  MOV      (6),%0
874      003664 022740 104001  CMP      #EMT+1,-(0)      ;WAS THE CALL EMT+1
875      003670 001077      BNE      TYP5            ;NO! TYPE A SERIES
876      003672 000400      BR       TYP             ;TYPE ONE LINE
877
878      003674 011600      ;SUBROUTINE TO OUTPUT ASCII MESSAGE ON TELETYPE PRINTER.
879      003676 062716 000002  TYP:    MOV      @%6,%0      ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
880      003702 011000      ADD      #2,@%6          ;SET UP EXIT.
881      003704 112067 000154      MOV      @%0,%0          ;ADDRESS OF MESSAGE TO RO.
882      003710 122767 000100 000146  TYPA:  MOVVB   (0)+,TYPDAT    ;GET CHARACTER
883      003716 001005      CMPB    #100,TYPDAT      ;CHECK FOR "0" CHARACTER
884      003720 005067 000140      BNE      TYPC            ;CRANCH IF NOT "0".
885      003724 004067 000030      CLR     TYPDAT          ;OUTPUT NULL TO
886      003730 000002      JSR     %7,TYPD         ;CLEAR BUFFER
887      003732 122767 000045 000124  TYPC:  RTI              ;TERMINATOR CHAR. DONE. EXIT.
888      003740 001436      CMPB    #45,TYPDAT      ;CHECK FOR "%".
889      003742 122767 000042 000114  BEQ     TYPF            ;BRANCH IF "%".
890      003750 001437      CMPB    #42,TYPDAT      ;NOT "%". CHECK FOR "#".
891      003752 004767 000002      BEQ     TYPG            ;BRANCH IF "#".
892      003756 000752      JSR     %7,TYPD         ;TYPE CHAR IN TYPDAT
893      003760 116777 000100 175016  TYPD:  BR       TYPA          ;OUTPUT CHARACTER TO PRINTER
894      003766 105777 175016      MOVVB   TYPDAT,@TPB     ;WAIT FOR DONE FLAG.
895      003772 100375      TSTB   @TPS
896      003774 122767 000015 000062  BPL     -4
897      004002 001003      CMPB    #15,TYPDAT      ;CHECK FOR CR
898      004004 012767 000011 000054  BNE     IS              ;NO - SKIP
899      004012 005767 000050      MOV     #9,NULL         ;SET NULL COUNTER
900      004016 001406      TST     NULL            ;TEST COUNTER
901      004020 005367 000042      BEQ     TYEXIT          ;ZERO - EXIT
902      004024 112767 000000 000032  DEC     NULL            ;DECREMENT
903      004032 000752      MOVVB   #0,TYPDAT       ;ZERO OUTPUT
904      004034 000207      BR      TYPD            ;OUTPUT NULL
905      004036 112767 000015 000020  TYEXIT:RTS             %7          ;EXIT
906      004044 004767 177710      TYPF:  MOVVB   #15,TYPDAT ;MOVE CARRIAGE RETURN CODE TO TYPDAT
907      004050 112767 000012 000006  JSR     %7,TYPD         ;GO TYPE CHAR.
908      004056 004767 177676      TYPG:  MOVVB   #12,TYPDAT ;MOVE LF CODE TO TYPDAT.
909      004062 000710      JSR     %7,TYPD         ;GO TYPE CHAR.
910      004064 000000      BR      TYPA
911      004066 000000      TYPDAT: 0
912      NULL: 0
913      ;SUBROUTINE TO OUTPUT A SERIES OF ASCII MESSAGES ON TELETYPE PRINTER
914      TYP5:  MOV      @%6,%0      ;GET ADDRESS THAT CONTAINS MESSAGE ADDRESS
915      004076 011067 000014      ADD      #2,@%6          ;UPDATE TO NEXT MESSAGE ADDRESS
916      004102 022767 177777 000006  MOV      @%0,TYPSB      ;ADDRESS OF MESSAGE TO TYPSB
917      004110 001001      CMP     #-1,TYPSB       ;CHECK FOR TERMINATOR
918      004112 000002      BNE     TYP5A          ;BRANCH IF NOT TERMINATOR.
919
920      004114 104001      RTI              ;TERMINATOR. EXIT
921
922      004116 000000      TYP5A:  EMT      +1          ;CALL ON TYP SUB TO TYPE MESSAGE
923      004120 000763      TYP5B:  0              ;ADDRESS OF MESSAGE GOES HERE
924
925      BR      TYP5          ;GO PROCESS NEXT MESSAGE

```

```

923 ;OCTAL TO ASCII CONVERT ROUTINE
924
925 ;ENTER ROUTINE AS FOLLOWS
926 ;JSR%5,CONV
927 ;ADDR#=ADDRESS OF NUMBER TO BE CONVERTED
928 ;ADDR BYTE=LSB OF WHERE ASCII IS GOING
929 ;ASCII#=THE NUMBER OF ASCII CHAR. TO BE CONVERTED
930
931 004122 013567 000310 CONV: MOV @5+,ACNVX ;VALUE OF # TO BE CONVERTED
932 004126 012501 MOV (5)+,%1 ;ASCII ADDR
933 004130 012502 MOV (5)+,%2 ;# OF ASCII CHAR
934 004132 060201 ADD %2,%1
935 004134 016703 000276 ACVN: MOV ACNVX,%3
936 004140 042703 177770 BIC #177770,%3 ;ISOLATE LEAST SIGNIFICANT OCTAL#
937 004144 062703 000060 ADD #60,%3 ;SET UP ASCII#
938 004150 110341 MOV#B %3,-(1) ;STORE ASCII CHAR
939 004152 042767 000007 000256 BIC #7,ACNVX
940 004160 006067 000252 ROR ACNVX ;ROTATE OCTAL#
941 004164 006067 000246 ROR ACNVX
942 004170 006067 000242 ROR ACNVX
943 004174 005302 DEC %2 ;-1 FROM ASCII CHAR COUNT
944 004176 001356 BNE ACVN
945 004200 000205 RTS %5 ;EXIT # CONVERTED
946
947 ;ROUTINE TO SET ACTION ENABLE ON MA/MF PARITY MEMORIES
948 ;CALL JSR PC, MAMF
949
950 172100 PARCSR= 172100
951 000114 PARVEC= 114
952 000004 ERRVEC= 4
953 000006 SP= %6
954
955 004202 012737 004274 000114 MAMF: MOV #PARSRV,@#PARVEC ;SET PARITY INTERRUPT VECTOR
956 004210 012737 000340 000116 MOV #340,@#PARVEC+2 ;AND PRIORITY LEVEL 7 ON INTERRUPT
957 004216 013746 000004 MOV @#ERRVEC,-(SP) ;SAVE CURRENT ERROR VECTOR
958 004222 013746 000006 MOV @#ERRVEC+2,-(SP) ;AND PRIORITY LEVEL
959 004226 012737 000006 000004 MOV #ERRVEC+2,@#ERRVEC
960 004234 012737 000002 000006 MOV #RTI,@#ERRVEC+2
961 004242 012700 172100 MOV #PARCSR,%0 ;GET FIRST CSR ADDRESS
962 004246 012702 000001 MOV #1,%2
963 004252 012720 000001 1$: MOV #1,(0)+ ;SET ACTION ENABLE IF AVAILABLE
964 004256 006302 ASL %2 ;SHIFT AVAILABILITY INDICATOR
965 004260 103374 BCC 1$
966 004262 012637 000006 MOV (SP)+,@#ERRVEC+2 ;RESTORE ERROR VECTOR
967 004266 012637 000004 MOV (SP)+,@#ERRVEC ;PRIORITY LEVEL AND INTERRUPT VECTOR
968 004272 000207 RTS %7
969
970 ;PARITY ERROR SERVICE ROUTINE
971 ;WHEN A PARITY ERROR IS DETECTED THE ROUTINE SCANS
972 ;MEMORY FOR THE PARITY ERROR & WHEN THE ERROR
973 ;IS DETECTED THE PROGRAM HALTS WITH THE ADDRESS
974 ;CAUSING THE ERROR IN RD
975 ;TO CONTINUE PRESS CONTINUE
976 004274 104001 PARSRV: EMT+1
977 004276 005005 PARERR
978 004300 012737 004326 000114 MOV #2$,@#PARVEC ;REPOSITION PARITY ERROR INT.

```

```

979 004306 012737 004354 000004      MOV      #4$, @#ERRVEC      ;SET TIME OUT TRAP
980 004314 005037 000006              CLR      @#ERRVEC+2
981 004320 005000              CLR      %0
982 004322 005720              1$:    TST      (0)+          ;SCAN MEMORY
983 004324 000776              BR       1$
984 004326 000000              2$:    HALT                ;PARITY ERROR-ADDRESS
985                                ;CAUSING ERROR IS IN REGISTER 0
986 004330 000005              3$:    RESET
987 004332 012737 004274 000114      MOV      #PARSRV, @#PARVEC ;RESOTRE PARITY VECTOR
988 004340 012737 000006 000004      MOV      #ERRVEC+2, @#ERRVEC ;RESOTRE TIME OUT HALT
989 004346 004767 177630      JSR      %7, MAMF
990 004352 000002              RTI
991 004354 000000              4$:    HALT                ;ERROR - PARITY ERROR NOT DETECTED ON SCAN
992 004356 000764              BR       3$                ;4(SP) CONTAINS PC WHERE
993                                ;PARITY ERROR WAS ORGINALLY DETECTED
994 004360 012767 004376 173436  DOWN:  MOV      #UP, 24          ;SET RETURN
995 004366 052777 000400 174420      BIS      #BIT8, @RCCS      ;ABORT
996 004374 000000              HALT
997 004376 012767 004360 173420  UP:    MOV      #DOWN, 24        ;RESET DOWN
998 004404 012706 001000              MOV      #1000, %6         ;RESET STACK
999 004410 012767 000454 174500      MOV      #300., WORK
1000 004416 000005  TIMCNT: RESET
1001 004420 005267 174472              INC      WORK              ;TIMEOUT
1002 004424 001374              BNE     TIMCNT
1003 004426 104001              EMT      +1                ;PRINT MSG
1004 004430 005032
1005 004432 000167 174466              FWRF
                                      JMP      START

```



```

1006 004436 000000          ACNVX: 0                      ;WORK REGISTER
1007
1008
1009          ;ERROR MESSAGE HEADERS
1010
1011          .EVEN
1012 004440 042045 052101 020101 HED1: .ASCII /%DATA ERROR @/
1013 004446 051105 047522 020122
1014 004454          100
1015 004455          045 052123 052101 HED2: .ASCII /%STATUS ERROR @/
1016 004462 051525 042440 051122
1017 004470 051117 040040
1018 004474 047445 042526 043122 HED3: .ASCII /%OVERFLO X-FER ERROR @/
1019 004502 047514 054040 043055
1020 004510 051105 042440 051122
1021 004516 051117 040040
1022 004522 042045 051511 020113 HED4: .ASCII /%DISK ADDRESS ERROR @/
1023 004530 042101 051104 051505
1024 004536 020123 051105 047522
1025 004544 020122          100
1026 004547          045          100 HED5A: .ASCII /%@/
1027
1028          ;MESSAGE TRAILERS
1029
1030 004551          040 020040 020040 MES1: .ASCII /          RCDA @/
1031 004556 020040 041522 040504
1032 004564 040040
1033 004566 020040 020040 020040 MES2: .ASCII /          RCCS @/
1034 004574 051040 041503 020123
1035 004602          100
1036 004603          040 020040 020040 MES2A: .ASCII /          RCER@/
1037 004610 020040 041522 051105
1038 004616          100
1039 004617          040 020040 020040 MES3: .ASCII /          WORD COUNT@/
1040 004624 047527 042122 041440
1041 004632 052517 052116          100
1042 004637          040 020040 020040 MES4: .ASCII /          WORD ADR.@/
1043 004644 020040 047527 042122
1044 004652 040440 051104 040056
1045 004660 043411 047517 020104 MES5: .ASCII /          GOOD DATA @/
1046 004666 040504 040524 040040
1047 004674 020040 020040 020040 MES6: .ASCII /          BAD DATA @/
1048 004702 041040 042101 042040
1049 004710 052101 020101          100
1050 004715          040 020040 054105 MES11: .ASCII /          EXISTENT DISK(S).@/
1051 004722 051511 042524 052116
1052 004730 043040 051511 024113
1053 004736 024523 040056
1054 004742 020040          100
1055 004745          045 051120 041517 MES12: .ASCII / @/
1056 004752 051505 047523 020122 TIMO: .ASCII /%PROCESSOR BACKGROUND TIMED OUT@/
1057 004760 040502 045503 051107
1058 004766 052517 042116 052040
1059 004774 046511 042105 047440
1060 005002 052125          100
1061 005005          045 042515 047515 PARERR: .ASCII /%MEMORY PARITY ERROR@/

```

M02

MAINDEC-11-DZRCO-B
DZRCOB.P11

RC11 MULTI DISK

REPLACES 058A-PB

MACY11 27(732) 10-SEP-76 15:10 PAGE 25

1062	005012	054522	050040	051101	
1063	005020	052111	020131	051105	
1064	005026	047522	040122		
1065	005032	050045	053517	051105	PWRF: .ASCII /%POWER HAS FAILED@/
1066	005040	044040	051501	043040	
1067	005046	044501	042514	040104	
1068					
1069					
1070	005054	042445	042116	040177	END: .ASCII /%END/<177>/@/ .EVEN
1071					
1072	005062	000000			BUFFER: 0
1073		000001			.END

E03

MAINDEC-11-DZRCO-B RC11 MULTI DISK REPLACES DSBA-PB
DZRCOB.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

MACY11 27(732) 10-SEP-76 15:10 PAGE 32

.END	1073	
.EVEN	1011	1071
.LIST	361	397
.NLIST	361	397
.REM	1	
.REPT	397	
.TITLE	360	

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*DZRCOB, DZRCOB, SEQ/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DZRCOB.P11
RUN-TIME: 3 6 1 SECONDS
RUN-TIME RATIO: 32/12=2.7
CORE USED: 7K (13 PAGES)

