

CO1

5/5/76

.SBTTL REVISION HISTORY

; JUL 76 DTR3A-A INITIAL RELEASE

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
96
97
98
99
100
101
102
103
104

000000' 007402
000002' 000000
100000
000004
000002
000001
000004' 000000
000006' 000000
000010' 000000
000012' 000003
000014' 000000
000016' 000000
000020' 000001
000022' 000000
000024' 176710
000026' 000254
000030' 000240
000032' 000000
000034' 001330
000036' 001400
000040' 002014
000042' 001254
000044' 001730
000046' 000000
000050' 000000
000052' 000000
000054' 000000
000056' 000000
000060' 000000
000062' 000000
000064' 000000
000066' 000000
000070' 000000
000072' 000000
000074' 000000

.SBTTL STANDARD DEVICE ROUTINE TABLE
.TITLE MAINDEC-11-DTR3A-A RPO2/RPO3 DEVICE ROUTINE FOR MPG
;REVISION 'A'
;FILENAME OF "TR3AAD.MPG" ON MPG/XXDP MEDIA
;MACY11: DTR3A',DTR3A'/CRF:SYM/DOC=DTR3A'.P11
;LNX11: DTR3A'.MPG/B:0=DTR3A'/E
;PAPER TAPE: PUNCH DTR3A'.MPG/FILE:ELEV

.CSECT RP11
.DSABL GBL

;THE FOLLOWING TABLE IS IN THE STANDARDIZED FORMAT REQUIRED
;TO INTERFACE WITH MPG.

LOCZ: .WORD DVREND-
DFLGMD: .WORD 0
WAITMD= 100000
WAITTN= 4
DOTERM= 2
IOERR= 1
CYL: .WORD 0
HEAD: .WORD 0
SECT: .WORD 0
RTRY: .WORD 3
SIZE: .WORD 0
ERR1: .WORD 0
DREGAD: .WORD 176710
IVCTAD: .WORD 254
PSMD: .WORD 240
HSKEEP-
REPORT-
KILL-
DATAER-
TOUTER-
CIOSBY: .WORD 0
CUPGER: .WORD 0
ULIST: .WORD 0
CLIST: .WORD 0
BINASC: .WORD 0
BTASLZ: .WORD 0
DECASC: .WORD 0
CSYSFW: .WORD 0
SETVEC: .WORD 0
CLRVEC: .WORD 0
TSTVEC: .WORD 0
RTNINT: .WORD 0

: DEVICE ROUT SIZE IN BYTES
: DEVICE ROUT FLAGWORD
: WAIT MODE - 0 = WAIT
: WAITING FOR ATTN INT
: PROCESS I/O TERMINATION
: ERROR ON CURRENT I/O
: CYLINDER # (0 THRU 202./405.)
: HEAD # (0 THRU 19.)
: SECTOR # (0 THRU 9.)
: # OF RETRY ATTEMPTS
: INTERFACE WORD # 5 (NOT USED)
: INTERFACE WORD # 6 (NOT USED)
: # OF BYTES TRANSFERRED / UNIMAP FLG
: ERROR ON LAST I/O INDICATOR
: FIRST DEVICE REGISTER ADR
: INTERRUPT VECTOR ADR
: INT PROC STATUS WORD (BR 5)
: NOT USED
: HOUSEKEEPING ROUT REL ADR
: REPORT ROUT REL ADR
: KILL ROUT REL ADR
: DATA ERROR COUNTER REL ADR
: TIME OUT ERROR ROUT REL ADR
: I/O BUSY BRANCH ADR
: DEVICE ERROR BRANCH ADR
: USER MODE PRINT ROUTINE BRANCH ADR
: CMD MODE PRINT ROUTINE BRANCH ADR
: CONVERT BINARY TO ASCII ROUT BR ADR
: CONVERT BINARY TO DECIMAL ASCII BR ADR
: CONVERT PACKED DECIMAL TO ASCII BR ADR
: MPG SYSTEM FLAGWORD ADR
: SET INT VECT ROUT BR ADR
: CLEAR INT VECTOR ROUT BR ADR
: TEST INT VECTOR ROUT BR ADR
: RETURN FROM INT ROUT BR ADR

105	000076'	000000	GETBYT:	.WORD	0	:GET DATA BYTE ROUT BR ADR
106	000100'	000000	PUTBYT:	.WORD	0	:PUT DATA BYTE ROUT BR ADR
107	000102'	000014		.WORD	DVREGS--	:ADR OF DEVICE REGISTER NAMES
108	000104'	000122		.WORD	DVCHDS--	:ADR OF DEVICE FUNCTIONS
109	000106'	000242		.WORD	DVPKTE--	:ADR OF PACK TBL EXTENSION
110	000110'	000460		.WORD	DVMYTE--	:ADR OF MODEL VECTOR TBL EXTEN.
111	000112'	000566		.WORD	DVCPTE--	:ADR OF COMPILER TBL EXTEN.
112	000114'	000772		.WORD	DVINST--	:ADR OF DEV INTERFACE WD SYM TBL

.SBTTL COMPILER TABLES & CONSTANT AREAS

```

114
115
116
117 000116' 050122 051504
118 000122' 000000
119 000124' 050122 051105
120 000130' 000002
121 000132' 000002 051503
122 000136' 000004
123 000140' 050122 041527
124 000144' 000006
125 000146' 050122 040502
126 000152' 000010
127 000154' 050122 040503
128 000160' 000012
129 000162' 050122 040504
130 000166' 000014
131 000170' 050122 030515
132 000174' 000016
133 000176' 050122 031115
134 000202' 000020
135 000204' 050122 031515
136 000210' 000022
137 000212' 052523 040503
138 000216' 000024
139 000220' 044523 047514
140 000224' 000026
141 000226'
142
143
144 000226' 120 201
145 000230' 002232
146 000232' 130 201
147 000234' 002256
148 000236' 376 000
149 000240' 002072
150 000242' 375 000
151 000244' 002046
152 000246' 374 000
153 000250' 001166
154 000252' 373 000
155 000254' 001162
156 000256' 372 000
157 000260' 001632
158 000262' 371 000
159 000264' 001760
160 000266' 370 201
161 000270' 002252
162 000272' 367 201
163 000274' 002260
164 000276' 366 201
165 000300' 002266
166 000302' 365 000
167 000304' 002312
168 000306' 364 000
169 000310' 002330

```

```

DVREGS: .ASCII /RPOS/
         .WORD 0
         .ASCII /RPER/
         .WORD 2
         .ASCII /RPCS/
         .WORD 4
         .ASCII /RPMC/
         .WORD 6
         .ASCII /RPBA/
         .WORD 10
         .ASCII /RPCA/
         .WORD 12
         .ASCII /RPOA/
         .WORD 14
         .ASCII /RPM1/
         .WORD 16
         .ASCII /RPM2/
         .WORD 20
         .ASCII /RPM3/
         .WORD 22
         .ASCII /SUCA/
         .WORD 24
         .ASCII /SILO/
         .WORD 26
DVREGS= .

```

```

: VALID DEVICE REGISTER NAMES &
: THEIR POSITIONS RELATIVE TO
: THE DEVICE REGISTERS BASE ADDRESS.

```

```

DVCMDS: .BYTE 120,201
         .WORD READ-
         .BYTE 130,201
         .WORD WRITE-
         .BYTE 376,0
         .WORD NOWAIT-
         .BYTE 375,0
         .WORD WAIT-
         .BYTE 374,0
         .WORD REPORT-
         .BYTE 373,0
         .WORD REPORT-
         .BYTE 372,0
         .WORD STEPUP-
         .BYTE 371,0
         .WORD STEPON-
         .BYTE 370,201
         .WORD RDNOSK-
         .BYTE 367,201
         .WORD WRNOSK-
         .BYTE 366,201
         .WORD WRCK-
         .BYTE 365,0
         .WORD SEEK-
         .BYTE 364,0
         .WORD HOMESK-

```

```

: VALID DEVICE FUNCTIONS
: FLAG BYTE:
: BIT 7 = NPR DEV
: BIT 3 = MASSBUS DEV
: BIT 0 = 2 WORDS FOR ADR
: (18 BIT ADRS)

```

170	000312'	363	000			.BYTE	363,0
171	000314'	002324				.WORD	RECAL-
172	000316'	362	000			.BYTE	362,0
173	000320'	002062				.WORD	IDLE-
174	000322'	361	000			.BYTE	361,0
175	000324'	002056				.WORD	CRESET-
176	000326'	360	000			.BYTE	360,0
177	000330'	002012				.WORD	HDRON-
178	000332'	357	000			.BYTE	357,0
179	000334'	002016				.WORD	HDROFF-
180	000336'	356	000			.BYTE	356,0
181	000340'	002022				.WORD	MODE11-
182	000342'	355	000			.BYTE	355,0
183	000344'	002026				.WORD	MODE10-
184	000346'	177777				.WORD	177777
185							
186	000350'	047516	040527	052111	DVPKTE:	.ASCII	/NOWAIT/
187	000356'	376	000			.BYTE	376,0
188	000360'	020040	040527	052111		.ASCII	/ WAIT/
189	000366'	375	000			.BYTE	375,0
190	000370'	052123	052101	051525		.ASCII	/STATUS/
191	000376'	374	000			.BYTE	374,0
192	000400'	047503	047125	051524		.ASCII	/COUNTS/
193	000406'	373	000			.BYTE	373,0
194	000410'	052123	050105	050125		.ASCII	/STEPUP/
195	000416'	372	000			.BYTE	372,0
196	000420'	052123	050105	047104		.ASCII	/STEPDN/
197	000426'	371	000			.BYTE	371,0
198	000430'	042122	047516	045523		.ASCII	/RDNOSK/
199	000436'	370	000			.BYTE	370,0
200	000440'	051127	047516	045523		.ASCII	/WRNOSK/
201	000446'	367	000			.BYTE	367,0
202	000450'	020040	051127	045503		.ASCII	/WRCK/
203	000456'	366	000			.BYTE	366,0
204	000460'	020040	042523	045505		.ASCII	/SEEK/
205	000466'	365	000			.BYTE	365,0
206	000470'	047510	042515	045523		.ASCII	/HOMESK/
207	000476'	364	000			.BYTE	364,0
208	000500'	051040	041505	046101		.ASCII	/RECAL/
209	000506'	363	000			.BYTE	363,0
210	000510'	020040	042111	042514		.ASCII	/IDLE/
211	000516'	362	000			.BYTE	362,0
212	000520'	051103	051505	052105		.ASCII	/CRESET/
213	000526'	361	000			.BYTE	361,0
214	000530'	044040	051104	047117		.ASCII	/HDRON/
215	000536'	360	000			.BYTE	360,0
216	000540'	042110	047522	043106		.ASCII	/HDROFF/
217	000546'	357	000			.BYTE	357,0
218	000550'	047515	042504	030461		.ASCII	/MODE11/
219	000556'	356	000			.BYTE	356,0
220	000560'	047515	042504	030011		.ASCII	/MODE10/
221	000566'	355	000			.BYTE	355,0
222							
223	000570'	000376	001140		DVMVTE:	.WORD	376,MSFMT1-LOC2
224	000574'	000375	001140			.WORD	375,MSFMT1-LOC2
225	000600'	000374	001140			.WORD	374,MSFMT1-LOC2

;TABLE TERMINATOR

;PACK TABLE EXTENSION

;MODEL VECTOR TABLE EXTEN.

000604	000373	001140	.WORD	373,MSFMT1-LOCZ
000610	000372	001164	.WORD	372,MSFMT5-LOCZ
000614	000371	001164	.WORD	371,MSFMT5-LOCZ
000620	000370	001146	.WORD	370,MSFMT3-LOCZ
000624	000367	001155	.WORD	367,MSFMT4-LOCZ
000630	000366	001141	.WORD	366,MSFMT2-LOCZ
000634	000365	001140	.WORD	365,MSFMT1-LOCZ
000640	000364	001140	.WORD	364,MSFMT1-LOCZ
000644	000363	001140	.WORD	363,MSFMT1-LOCZ
000650	000362	001140	.WORD	362,MSFMT1-LOCZ
000654	000361	001140	.WORD	361,MSFMT1-LOCZ
000660	000360	001140	.WORD	360,MSFMT1-LOCZ
000664	000357	001140	.WORD	357,MSFMT1-LOCZ
000670	000356	001140	.WORD	356,MSFMT1-LOCZ
000674	000355	001140	.WORD	355,MSFMT1-LOCZ

COMPILER TABLE EXTENSION

000700	003	376	DVCPTC:	.BYTE	3,376	;NO WAIT
000702	004537	000012		.WORD	4537,10.	
000706	003	375		.BYTE	3,375	;WAIT
000710	004537	000012		.WORD	4537,10.	
000714	004	374		.BYTE	4,374	;STATUS
000716	004537	000012	001002	.WORD	4537,10.,1002	
000724	004	373		.BYTE	4,373	;COUNTS
000726	004537	000012	001001	.WORD	4537,10.,1001	
000734	004	372		.BYTE	4,372	;STEP UP
000736	004537	000012	000000	.WORD	4537,10.,0	
000744	004	371		.BYTE	4,371	;STEP DOWN
000746	004537	000012	000000	.WORD	4537,10.,0	
000754	006	370		.BYTE	6,370	;READ NO SEEK
000756	004537	000012	000000	.WORD	4537,10.,0,2,2	
000764	000002	000002				
000770	006	367		.BYTE	6,367	;WRITE NO SEEK
000772	004537	000012	000000	.WORD	4537,10.,0,2,2	
001000	000002	000002				
001004	006	366		.BYTE	6,366	;WRITE CHECK
001006	004537	000012	000000	.WORD	4537,10.,0,2,2	
001014	000002	000002				
001020	003	365		.BYTE	3,365	;SEEK
001022	004537	000012		.WORD	4537,10.	
001026	003	364		.BYTE	3,364	;HOME SEEK
001030	004537	000012		.WORD	4537,10.	
001034	003	363		.BYTE	3,363	;RECAL
001036	004537	000012		.WORD	4537,10.	
001042	003	362		.BYTE	3,362	;IDLE
001044	004537	000012		.WORD	4537,10.	
001050	003	361		.BYTE	3,361	;CRESET
001052	004537	000012		.WORD	4537,10.	
001056	003	360		.BYTE	3,360	;HEADER ON
001060	004537	000012		.WORD	4537,10.	
001064	003	357		.BYTE	3,357	;HEADER OFF
001066	004537	000012		.WORD	4537,10.	
001072	003	356		.BYTE	3,356	;MODE 11
001074	004537	000012		.WORD	4537,10.	


```

279 001100' 003 355 .BYTE 3,355 ;MODE 10
280 001102' 004537 000012 .WORD 4537,10.
281
282
283
284
285 001106' 054503 020114
286 001112' 000004
287 001114' 042510 04210
288 001120' 000006
289 001122' 042523 052103
290 001126' 000010
291 001130' 052102 054522
292 001134' 000012
293 001136' 177777
294
295
296
297
298
299
300 001140' 000
001141' 377 052101 000377
001146' 044777 052116 177517
001154' 000
301 001155' 377 051106 046517
001162' 000377
001164' 377 000
302
303
304
305
306
307
308
309
310 001166' 001166'
001166' 000000 000000 000000
311 001174' 000000 000000 000000
001202' 000000 000000 000000
312 001210' 000000 000000
313
314 001214' 000013
315
316 001242' 000000
317 001244' 000000
318 001246' 000000
319 001250' 000000
320 001252' 000000
321 001254' 000000
322 001256' 000000
323 001260' 000000
324 001262' 000000
325 001264' 000000
326
327 001266'
328 001266' 000000
329 001270' 000000
330 001272' 000000

```

... DEVICE INTERFACE WORD SYMBOL TABLE

```

DVIWST: .ASCII /CYL /
.WORD DEVIW1
.ASCII /HEAD/
.WORD DEVIW2
.ASCII /SECT/
.WORD DEVIW3
.ASCII /RTRY/
.WORD DEVIW4
.WORD 177777 ;END OF TABLE

```

... MODEL STATEMENT TABLE EXTENSION

```

MSFMT1: .BYTE 0
MSFMT2: .ASCIZ <377>/AT/<377>
MSFMT3: .ASCIZ <377>/INT0/<377>
MSFMT4: .ASCIZ <377>/FROM/<377>
MSFMT5: .BYTE 377,0
.EVEN

```

;DEVICE ROUTINE CONSTANTS & EQUATES

```

MSKPST= .
ISTAT= . ;STORAGE FOR DEV REG'S AT INT
.WORD 0,0,0,0,0,0
.WORD 0,0,0,0,0
CSTAT: .BLKW 11. ;DEV REG CURRENT VALUES STORAGE
OBJADR: .WORD 0 ;CURR ADR IN USER PROG
CNTADR: .WORD 0 ;ADR OF BYTE COUNT TOTALS
CURFLG: .WORD 0 ;FLAG WORD OF CURR CMND
CURCMD: .WORD 0 ;CURR CMND CODE
CURADR: .WORD 0 ;CURR BUS ADDRESS
CURCNT: .WORD 0 ;NEG WORD CNT FOR CURR CMND
FINCNT: .WORD 0 ;FINAL WORD CNT (RPWC)
CURRTY: .WORD 0 ;CURR RETRY COUNT
RTRYIP: .WORD 0 ;RETRY IN PROGRESS FLAG
COUNTS:
BYRD: .WORD 0 ;BYTES READ COUNT
BYWR: .WORD 0 ;BYTES WRITTEN COUNT

```

```

331 001274' 000000          .WORD 0
332 001276' 000000          BYCK: .WORD 0 ;BYTES CHECKED COUNT
333 001500' 000000          .WORD 0
334 001302' 000000          RDCNT: .WORD 0 ;READ CMD COUNT
335 001304' 000000          WRCNT: .WORD 0 ;WRITE CMD COUNT
336 001306' 000000          CKCNT: .WORD 0 ;CHECK CMD COUNT
337 001310' 000000          SKCNT: .WORD 0 ;SEEK CMD COUNT
338 001312' 000000          MISCNT: .WORD 0 ;MISC. CMD COUNT
339 001314' 000000          ERRCNT: .WORD 0 ;DEVICE ERRORS COUNT
340 001316' 000000          DATAER: .WORD 0 ;DATA ERRORS COUNT
341 001320' 000000          TIECNT: .WORD 0 ;TIMING ERRORS (DLT)
342 001322' 000000          CSECNT: .WORD 0 ;CHECKSUM ERRORS
343 001324' 000000          WPECNT: .WORD 0 ;WORD PARITY ERRORS
344 001326' 000000          LPECNT: .WORD 0 ;LONGITUDINAL PARITY ERRORS
345 001330' 000000          WCECNT: .WORD 0 ;WRITE CHECK ERRORS
346 001332' 000000          RETRYS: .WORD 0 ;# OF RETRIES ON I/O CMDS
347 001334' 000000          INTCNT: .WORD 0 ;INTERRUPTS COUNT
348
349          001336'          HSKPEN= .
350
351 001336' 000000          RPCSV: .WORD 0 ;BASE VALUE FOR RPCS REG
352          010000          MODE= 10000 ; MODE SELECT BIT - 1 = PDP-10/15
353          004000          HEADER= 4000 ; HEADER BIT - 1 = HEADER OPERATION
354
355          ;RPCS          DEVICE BIT EQUATES
356
357          100000          ERR= 100000
358          040000          HE= 40000
359
360          000013          REGNUM= 11 ;# OF DEVICE REGISTERS
361          000024          CNTNUM= HSKPEN-COUNTS/2 ;# OF STATISTICAL COUNT WORDS
362
363          000001          MMVER= 1 ;SYS FLGMD BIT DEFINITIONS
364
365          000000          XXXX= 0 ;VALUE TO BE TAILORED BY DEV ROUT
366
367          177774          RPDS= -4 ;EQUATES FOR DEVICE REGISTER NAMES
368          177776          RPER= -2 ;RELATIVE TO R4'S CONTENTS
369          000000          RPCS= 0
370          000002          RPWC= +2
371          000004          RPBA= +4
372          000006          RPCA= +6
373          000010          RPDA= +10
374          000012          RPM1= +12
375          000014          RPM2= +14
376          000016          RPM3= +16
377          000020          SUCA= +20
378          000022          SILO= +22
379
380
381
382 001340'          PATCH: .REPT 10. ;PATCH AREA
383          .WORD 0
384          .ENDR

```

386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441

.SBTTL RP02/RP03 SUPPORT ROUTINES ENTERED FROM MPG

;DEVICE ROUTINE HOUSEKEEPING

```

:JSR   R5,HSKEEP           S/R CALL
:WORD  0 OR 1              0 = DO HSKP PER OPSW
:                               1 = UNCOND. DO HSKP
:R2 = PROG'S OPSW
:DESTROYS R0,R1
:INIT # OF RETRY ATTEMPTS
:INITIALIZE RPCS VALUE
:SET UP FIRST WD ADR
:SET UP # OF WORDS
:UNCONDITIONALLY DO HSKP?
:N,Y-10$
:OPSW SPECIFY DON'T HSKP COUNTS?
:Y,N-10$
:REMOVE THEM FROM LOOP COUNT
:MSKP ALL NECESSARY AREAS
:EXIT IN-LINE
    
```

```

001364' 012767 000003 176420 HSKEEP: MOV
001372' 005067 177740 CLR
001376' 010700 MOV
001400' 062700 177566 ADD
001404' 012701 000064 MOV
001410' 005725 TST
001412' 001005 BNE
001414' 032702 000004 BIT
001420' 001402 BEQ
001422' 162701 000024 SUB
001426' 005020 10$: CLR
001430' 005301 DEC
001432' 001375 BNE
001434' 000205 RTS
    
```

;RP02/RP03 REPORT ROUTINE

```

:JSR   R5,REPORT          S/R CALL
:WORD  FLAGWD             FLAGWORD
:                               BIT 15 = CMD MODE CALL
:                               BIT 9 = PROG STMT CALL
:                               BIT 1 = DO STATUS REPORT
:                               BIT 0 = DO COUNTS REPORT
    
```

```

001436' 004067 003704 REPORT: JSR
001442' 004767 003732 JSR
001446' 011504 MOV
001450' 032704 000002 BIT
001454' 001403 BEQ
001456' 004567 003742 JSR
001462' 177532 WORD
001464' 032704 177776 5$: BIT
001470' 001012 BNE
001472' 010700 MOV
001474' 062700 177572 ADD
001500' 012701 000024 MOV
001504' 005720 10$: TST
001506' 001003 BNE
001510' 005301 DEC
001512' 001374 BNE
001514' 000477 BR
001516' 004767 003726 15$: JSR
001522' 032704 000002 BIT
    
```

```

:SAVE REG'S R0 - R5
:SET UP PROG TBL ADR IN R3
:GET FLAGWORD
:GOING TO DO STATUS DISPLAY?
:Y,N-5$
:GO STORE STATUS REG'S
:DISPLAYING CNTS AT END OF
:PROG PASS? (Y,N-15$)
:SET UP ADR OF CNTS
:GET # OF CNT WORDS
:THIS CNT WORD = 0?
:Y,N-15$
:DECR WORD CNT
:CK'ED ALL WORDS? (Y,N-10$)
:GO TO EXIT -- ALL CNTS ARE 0'S
:DISPLAY DEVICE I.D.
:DOING STATUS DISPLAY?
    
```

```

442 001526' 001432      BEQ      DISCNT      ;Y N-DISCNT
443 001530' 010700      MOV      PC,R0       ;SET UP ADR OF REG'S AT
444 001532' 062700      ADD      #1,STAT-. ,R0 ;LAST INT
445 001536' 012701      MOV      #REGNUM,R1  ;SET UP # OF REG'S
446 001542' 005720      20S:   TST      (R0)+      ;ALL REG'S = 0?
447 001544' 001003      BNE     30S         ;N,Y-40S
448 001546' 005301      DEC     R1
449 001550' 001374      BNE     20S
450 001552' 000407      BR      40S
451 001554' 004567      30S:   JSR     RS,PRINT ;ISSUE 'AT LAST INT' MSG
452 001560' 004361      .WORD  ATMSG-.
453 001562' 000014      .WORD  12.
454 001564' 004567      JSR     RS,DISPST   ;GO DISPLAY STATUS AT LAST INT
455 001570' 177376      .WORD  ISTAT-.
456 001572' 004567      40S:   JSR     RS,PRINT   ;ISSUE 'CURRENTLY' MSG
457 001576' 004357      .WORD  CURMSG-.
458 001600' 000012      .WORD  10.
459 001602' 004367      JSR     RS,DISPST   ;GO DISPLAY CURRENT STATUS
460 001606' 177406      .WORD  CSTAT-.
461 001610' 004767      JSR     PC,PRTIND   ;GO DISPLAY INFO WORDS
462 001614' 032704      DISCNT: BIT     #1,R4   ;DISPLAY COUNTS?
463 001620' 001431      BEQ     RPTEND      ;Y N-RPTEND
464 001622' 012700      MOV     #CNTNUM,R0  ;SET UP # OF WORDS
465 001626' 010701      MOV     PC,R1       ;SET UP ADR OF CNTS
466 001630' 062701      ADD     #COUNTS-. ,R1
467 001634' 010702      MOV     PC,R2       ;SET UP TBL ADR
468 001636' 062702      ADD     #RPTBL-. ,R2
469 001642' 012267      RPTLP: MOV     (R2)+,RPTBAS ;MOV MSG ADR TO S/R LINKAGE
470 001646' 004067      JSR     R0,SAVEREG  ;SAVE ALL REG'S
471 001652' 011100      MOV     (R1),R0     ;GET CURRENT COUNT
472 001654' 004577      JSR     RS,2BINASC  ;CONVERT IT TO ASCII
473 001660' 000000      RPTBAS: .WORD  XXXX
474 001662' 004067      JSR     R0,RESREG   ;RESTORE REG'S
475 001666' 005721      TST     (R1)+       ;POINT AT NXT CNT
476 001670' 005300      DEC     R0          ;DONE ALL WORDS?
477 001672' 001363      BNE     RPTLP       ;Y N-RPTLP
478 001674' 004567      JSR     RS,PRINT    ;GO ISSUE COUNTS MSG
479 001700' 004406      .WORD  CNTSMG-.
480 001702' 000440      .WORD  CNTSEN-CNTSMG
481 001704' 004567      RPTEND: JSR     RS,PRINT   ;ISSUE "END OF REPORT" MSG
482 001710' 004257      .WORD  RENDMG-.
483 001712' 177763      .WORD  -13.
484 001714' 004067      DVREX: JSR     R0,RESREG ;RESTORE REGISTERS
485 001720' 005725      TST     (R5)+       ;SET UP RETURN POINT
486 001722' 000205      RTS     RS          ;EXIT IN-LINE
487
488
489 001724' 004442      REPTBL: .WORD  BCMRD-RPTBAS
490 001726' 004450      .WORD  BCMRD+6-RPTBA
491 001730' 004464      .WORD  BCMWR-RPTBAS
492 001732' 004472      .WORD  BCMWR+6-RPTBA
493 001734' 004510      .WORD  BCMCK-RPTBAS
494 001736' 004516      .WORD  BCMCK+6-RPTBAS
495 001740' 004543      .WORD  CMDCRD-RPTBAS
496 001742' 004556      .WORD  CMDCWR-RPTBAS
497 001744' 004571      .WORD  CMDCK-RPTBAS

```

```

498 001746' 004607 .WORD CMDCSK-RPTBAS
499 001750' 004624 .WORD CMDCMS-RPTBAS
500 001752' 004652 .WORD CNTERR-RPTBAS
501 001754' 004667 .WORD CNTDER-RPTBAS
502 001756' 004717 .WORD CNTTIE-RPTBAS
503 001760' 004734 .WORD CM'CSE-RPTBAS
504 001762' 004750 .WORD CNTWPE-RPTBAS
505 001764' 004767 .WORD CNTLPE-RPTBAS
506 001766' 005003 .WORD CNTWCE-RPTBAS
507 001770' 005032 .WORD CNTRTY-RPTBAS
508 001772' 005060 .WORD CNTINT-RPTBAS
509
510
511
512
513 ;TIMEOUT ERROR ROUTINE
514
515 ;JSR RS,TOUTER S/R CALL
516
517 001774' 004067 003346 TOUTER: JSR R0,SAVREG ;SAVE ALL REGISTERS
518 002000' 004767 003374 JSR PC,SUPTAD ;SET UP RPCS1 & PROG TBL ADR'S
519 002004' 004567 003414 JSR R5,STSTAT ;STORE CURRENT STATUS
520 002010' 177204 .WORD CSTAT-
521 002012' 004567 002332 JSR R5,TVECT ;CK IF I HAVE VECTOR CONTROL
522 002016' 000404 BR 10$ ;BR IF I DON'T
523 002020' 042714 020100 BIC #20100,(R4) ;RESET BOTH INT ENABLES
524 002024' 004767 002274 JSR PC,RINTV ;RESET THE INTERRUPT VECTOR
525 002030' 042713 000010 10$: BIC #21410T,(R3) ;RESET WAITING FOR I/O FLG
526 002034' 004567 002346 JSR R5,ERRCSI ;ISSUE I/O TIMEOUT ERROR MSG
527 002040' 002441 .WORD IOTO-ERMBAS
528 002042' 004067 003314 JSR R0,RESREG ;RESTORE REGISTERS
529 002046' 012605 MOV (SP)+,R5 ;REMOVE RETURN ADR
530 002050' 000177 175774 JMP @CUPGER ;GO TO ERROR EXIT
531
532
533 ;KILL USER PROGRAM ROUTINE
534
535 ;JSR RS,KILL S/R CALL
536
537 ;R3 = PROG TBL ADR
538
539 ;DESTROYS R0,R1
540
541 002054' 016701 175744 KILL: MOV DREGAD,R1 ;GET DEV REG ADR
542 002060' 062701 000004 ADD #4,R1 ;POINT AT RPCS REG
543 002064' 004567 002260 JSR R5,TVECT ;DO I HAVE VECTOR CONTROL?
544 002070' 000407 BR KILLEX ;BR IF I DON'T
545 002072' 032711 020100 BIT #20100,(R1) ;ARE INT ENABLES SET?
546 002076' 001402 BEQ 10$ ;Y,N-10$
547 002100' 042711 020100 BIC #20100,(R1) ;RESET INT ENABLES
548 002104' 004767 002214 10$: JSR PC,RINTV ;RESET INT VECTOR INFO
549 002110' 000205 KILLEX: RTS R5 ;EXIT IN-LINE

```

.SBTTL RPO2/RPO3 NON-I/O & NON-INTERRUPT FUNCTION ROUTINES

;"STEPUP" FUNCTION ROUTINE

```

551                                     ;JSR   R5,STEPUP      FUNCTION CALL
552                                     ;.WORD NBR          INCREMENT FACTOR
553
554                                     ;"STEPUP" FUNCTION ROUTINE
555
556                                     ;JSR   R5,STEPUP      FUNCTION CALL
557                                     ;.WORD NBR          INCREMENT FACTOR
558
559 002112' 004767 000060   STEPUP: JSR   PC,STPCOM      ;DO COMMON SETUP
560 002116' 062502                ADD   (R5)+,R2      ;ADD INCR VALUE TO SECT #
561 002120' 020203   10$:  CMP   R2,R3          ;SECT # IN RANGE?
562 002122' 103403                BLO  15$           ;N,Y-15$
563 002124' 160302                SUB   R3,R2        ;ADJ SECT # DOWNWARDS
564 002126' 001201                INC   R1           ;ADD 1 TO HEAD #
565 002130' 000773                BR   10$          ;GO CK NEW SECT # VALUE
566 002132' 020127 000024   15$:  CMP   R1,#20.    ;HEAD # IN RANGE?
567 002136' 103404                BLO  20$           ;N,Y-20$
568 002140' 162701 000024   SUB   #20.,R1      ;ADJ HEAD # DOWNWARDS
569 002144' 005200                INC   R0           ;ADD 1 TO CYL #
570 002146' 000771                BR   15$          ;GO CHECK NEW HEAD #
571 002150' 020004   20$:  CMP   R0,R4          ;CYL # IN RANGE?
572 002152' 103402                BLO  STEPEX       ;N,Y-STEPEX
573 002154' 160400                SUB   R4,R0        ;ADJ CYL # DOWNWARDS
574 002156' 000774                BR   20$          ;GO CK IT NOW
575 002160' 010067 175620   STEPEX: MOV  R0,CYL    ;STORE NEW CYL,HEAD,SECT VALUES
576 002164' 010167 175616     MOV  R1,HEAD
577 002170' 010267 175614     MOV  R2,SECT
578 002174' 000205                RTS
579
580                                     ;EXIT TO USER PROG
581
581 002176' 016700 175602   STPCOM: MOV  CYL,R0      ;GET CYL, HEAD, & SECT VALUES
582 002202' 016701 175600     MOV  HEAD,R1        ;IN REGISTERS
583 002206' 016702 175576     MOV  SECT,R2
584 002212' 004767 003162     JSR  PC,SUPTAD      ;GET PROG TBL ADR IN R3
585 002216' 012704 000313     MOV  #203.,R4      ;PRESET # OF CYL'S TO RPO2
586 002222' 032763 000020 000032  BIT  #20,PMDLCD(R3) ;THIS AN RPO3?
587 002230' 001402                BEQ  STPC10        ;Y,N-STPC10
588 002232' 012704 000626     MOV  #406.,R4      ;SET UP # OF RPO3 CYL'S
589 002236' 012703 000012   STPC10: MOV  #10.,R3   ;SET UP # OF SECTORS
590 002242' 000207                RTS                ;EXIT IN-LINE
591
592
593                                     ;"STEPDN" FUNCTION ROUTINE
594
595                                     ;JSR   R5,STEPDN      FUNCTION CALL
596                                     ;.WORD NBR          DECREMENT FACTOR
597
598 002244' 004767 177726   STEPDN: JSR  PC,STPCOM ;DO COMMON SETUP
599 002250' 162502                SUB  (R5)+,R2      ;SUB DECR FACTOR FROM SECT #
600 002252' 020203   40$:  CMP  R2,R3          ;SECT # IN RANGE?
601 002254' 103403                BLO  45$           ;N,Y-45$
602 002256' 060302                ADD  R3,R2        ;ADJ SECT # UPWARDS
603 002260' 005301                DEC  R1           ;DECR HEAD # BY 1
604 002262' 000773                BR   40$          ;GO CK NEW SECT # VALUE
605 002264' 020127 000024   45$:  CMP  R1,#20.    ;HEAD # IN RANGE?
606 002270' 103404                BLO  50$           ;N,Y-50$
    
```



```

607 002272' 062701 000024      ADD      #20.,R1      ;ADJ IT UPWARDS
608 0 76' 007300      DEC      R0          ;DECR CYL # BY 1
609 0 00' 000771      BR       45$        ;GO CHECK NEW HEAD #
610 002302' 020004      SOS:    CMP      R0,R4      ;CYL # IN RANGE?
611 002304' 103725      BLO     STEPEX     ;N.Y-STEPEX
612 002306' 060400      ADD     R4,R0      ;ADJ IT UPWARDS
613 002310' 000774      BR      SOS        ;GO CK IT NOW
614
615
616 ;"WAIT" FUNCTION ROUTINE
617
618 ;JSR RS,WAIT      FUNCTION CALL
619
620 002312' 042767 100000 175462 WAIT: BIC      #WAITMD,DFLGMD ;RESET THE "NOWAIT" FLAG
621 002320' 004767 001552      JSR     PC,CKD85Y  ;WAIT IF BUSY & DO TERMINATION
622 002324' 004767 001774      JSR     PC,RINTV   ;RESET THE INTERRUPT VECTOR
623 002330' 000205      RTS     RS         ;EXIT IN-LINE
624
625 ;"NOWAIT" FUNCTION ROUTINE
626
627 ;JSR RS,NOWAIT   FUNCTION CALL
628
629 002332' 052767 100000 175442 NOWAIT: BIS      #WAITMD,DFLGMD ;SET THE "NOWAIT" FLAG
630 002340' 000205      RTS     RS         ;EXIT IN-LINE
631
632 ;"HDRON" FUNCTION ROUTINE
633
634 ;JSR RS,HDRON   FUNCTION CALL
635
636 002342' 052767 014000 176766 HDRON: BIS      #HEADER+MODE,RPCSV ;SET THE HEADER & MODE BITS
637 002350' 000205      RTS     RS         ;EXIT IN-LINE
638
639 ;"HDROFF" FUNCTION ROUTINE
640
641 ;JSR RS,HDROFF  FUNCTION CALL
642
643 002352' 042767 014000 176756 HDROFF: BIC     #HEADER+MODE,RPCSV ;RESET THE HEADER & MODE BITS
644 002360' 000205      RTS     RS         ;EXIT IN-LINE
645
646 ;"MODE11" FUNCTION ROUTINE
647
648 ;JSR RS,MODE11  FUNCTION CALL
649
650 002362' 042767 010000 176746 MODE11: BIC     #MODE,RPCSV ;RESET THE MODE BIT
651 002370' 000205      RTS     RS         ;EXIT IN-LINE
    
```

```

657 ;"MODE10" FUNCTION ROUTINE
658
659 ;JSR RS,MODE10 FUNCTION CALL
660
661 002372' 052767 010000 176736 MODE10: BIS #MODE,RPCSV ;SET THE MODE BIT
662 002400' 000205 RTS RS ;EXIT IN-LINE
663
664 ;"IDLE" & "CRESET" FUNCTION ROUTINES
665
666 ;JSR RS,IDLE FUNCTION CALL
667 ;JSR RS,CRESET FUNCTION CALL
668
669
670 002402' IDLE:
671 002402' 004767 001470 CRESET: JSR PC,CKOBSY ;GO CK IF DEV IS BUSY
672 002406' 005767 176700 INC MISCNT ;ADD 1 TO MISC. CHND CNT
673 002412' 005567 175404 CLR ERRI ;RESET THE ERROR INDICATOR
674 002416' 012714 000001 MOV #1,(R4) ;ISSUE THE IDLE FUNCTION CODE
675 002422' 012700 000012 MOV #10.,R0 ;SET UP DELAY COUNT
676 002425' 005300 5S: DEC R0 ;DELAY FOR A FEW MICROSECONDS
677 002430' 001376 BNE 5S
678 002432' 012714 000001 MOV #1,(R4) ;ISSUE THE IDLE FUNCTION CODE
679 002435' 105714 10S: TSTB (R4) ;READY SET YET?
680 002440' 100407 BMI 20S ;N Y-20S
681 002442' 005300 DEC R0 ;TIMED OUT?
682 002444' 100774 BMI 10S ;Y N-10S
683 002446' 004567 001726 JSR RS,ERRCS ;GO ISSUE CRESET TIMEOUT ERROR
684 002452' 002416 .WORD CRT0-ERMB.CC
685 002454' 000177 JMP @CUPGER
686 002460' 000205 20S: RTS RS ;GO TO ERR RET POINT IN MPG
;EXIT IN-LINE TO USER'S PROG
    
```

.SBTTL RPO2/RPO3 INTERRUPT TYPE I/O FUNCTION ROUTINES

;"READ" FUNCTION ROUTINE

688										
689										
690										
691										
692										
693										
694										
695										
696										
697										
698										
699	002462'	012702	000105	READ:	MOV	#105,R2				:SET UP READ FUNCT CODE
700	002466'	012701	000275		MOV	#275,R1				:SET UP CMD FLAG WORD
701	002472'	004767	001400	RDCOM:	JSR	PC CKDBSY				:GO CK IF DEV IS BUSY
702	002476'	005267	176600		INC	RDCNT				:ADD 1 TO READ CMD CNT
703	002502'	010700			MOV	PC,R0				:SET UP ADR OF BYTES READ CNT
704	002504'	062700	176564		ADD	#BYRD+2-.,R0				
705	002510'	000456			BR	CHDCOM				:GO TO CMD COMMON PROCESSING

;"WRITE" FUNCTION ROUTINE

706										
707										
708										
709										
710										
711										
712										
713										
714										
715										
716	002512'	012702	000103	WRITE:	MOV	#103,R2				:SET UP WRITE FUNCT CODE
717	002516'	012701	000275		MOV	#275,R1				:SET UP CMD FLAG WORD
718	002522'	004767	001350	WRCOM:	JSR	PC CKDBSY				:GO CK IF DEV IS BUSY
719	002526'	005267	176552		INC	WRCNT				:ADD 1 TO WRITE CMD CNT
720	002532'	010700			MOV	PC,R0				:SET UP ADR OF BYTES WRITTEN CNT
721	002534'	062700	176540		ADD	#BYWR+2-.,R0				
722	002540'	000442			BR	CHDCOM				:GO TO CMD COMMON PROCESSING

;"RDNOSK" FUNCTION ROUTINE

723										
724										
725										
726										
727										
728										
729										
730										
731										
732	002542'	012702	000117	RDNOSK:	MOV	#117,R2				:SET UP READ (NO SEEK) FUNCT CODE
733	002546'	012701	000256		MOV	#256,R1				:SET UP CMD FLAG WORD
734	002552'	000747			BR	RDCOM				:GO TO COMMON READ PROCESSING

```

736                                     ;"WRNOSK" FUNCTION ROUTINE
737
738                                     :JSR      RS,WRNOSK      FUNCTION CALL
739                                     :.WORD   ADR              DATA ADDRESS (BITS 16 - 17)
740                                     :.WORD   ADR              DATA ADDRESS (BITS 0 - 15)
741                                     :.WORD   CNT              BYTE COUNT
742
743 002554' 012702 000113      WRNOSK: MOV      #113,R2      ;SET UP WRITE (NO SEEK) FUNCT CODE
744 002560' 012701 000256      MOV      #256,R1      ;SET UP CHND FLAG WORD
745 002564' 000756      BR        WRNCOM      ;GO TO WRITE COMMON PROCESSING
746
747
748                                     ;"WRCK" FUNCTION ROUTINE
749
750                                     :JSR      RS,WRCK       FUNCTION CALL
751                                     :.WORD   ADR              DATA ADDRESS (BITS 16 - 17)
752                                     :.WORD   ADR              DATA ADDRESS (BITS 0 - 15)
753                                     :.WORD   CNT              BYTE COUNT
754
755 002566' 012702 000107      WRCK:   MOV      #107,R2      ;SET UP WRITE CHECK FUNCT CODE
756 002572' 012701 000276      MOV      #276,R1      ;SET UP CHND FLAG WORD
757 002576' 004767 001274      JSR      PC,CKDBSY     ;GO CK IF DEV IS BUSY
758 002602' 005267 176500      INC      CKCNT         ;ADD 1 TO CHECK CHND COUNT
759 002606' 010700      MOV      PC,RO         ;SET UP ADR OF BYTES
760 002610' 062700 176470      ADD      #BYCK+2--,RO  ;CHECKED COUNT
761 002614' 000414      BR        CHDCOM      ;GO TO CHND COM PROCESSING
762
763
764                                     ;"SEEK" FUNCTION ROUTINE
765
766                                     :JSR      RS,SEEK      FUNCTION CALL
767
768 002616' 012702 000111      SEEK:   MOV      #111,R2      ;SET UP SEEK FUNCT CODE
769 002622' 012701 000160      SKCOM:  MOV      #160,R1      ;SET UP CHND FLAG WORD
770 002626' 004767 001244      JSR      PC,CKDBSY     ;GO CK IF DEV IS BUSY
771 002632' 005267 176452      INC      SKCNT         ;ADD 1 TO SEEK CHND COUNT
772 002636' 000403      BR        CHDCOM      ;GO TO CHND COMMON PROCESSING
773
774
775                                     ;"HOMESK" & "RECAL" FUNCTION ROUTINES
776
777                                     :JSR      RS,HOMESK    FUNCTION CALL
778                                     :JSR      RS,RECAL     FUNCTION CALL
779
780 002640'      HOMESK:
781 002640' 012702 000115      RECAL:  MOV      #115,R2      ;SET UP HOME SEEK FUNCT CODE
782 002644' 000766      BR        SKCOM        ;GO TO COMMON SEEK PROCESSING

```

; INTERRUPT TYPE I/O FUNCTION COMMON PROCESSING ROUTINE

; R4 = ADR OF RPCS DEV REG
; R3 = PROG TBL ADR
; R2 = FUNCTION CODE
; R1 = COMMAND FLAG WORD
; R0 = ADR OF BYTE COUNT, IF APPLICABLE

; CMD FLAGWORD FORMAT:

; BIT 7 = 200 = PERFORM RETRIES ON CMD
; BIT 6 = 100 = TWO INTS & TERMINATES WITH ATTN
; BIT 5 = 040 = SET UP SECT #
; BIT 4 = 020 = SET UP CYL AND HD #
; BIT 3 = 010 = INCREMENT BYTE COUNTS
; BIT 2 = 004 = DATA TRANSFER CMD
; BIT 1 = 002 = 3 ARGUMENT CMD
; BIT 0 = 001 = 4 ARGUMENT CMD

784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839

002646 010067 176372
002652 010167 176370
002656 032701 000003
002662 001416
002664 012567 176362
002670 012567 176360
002674 012500
002676 000241
002700 000000
002702 005400
002704 010067 176346
002710 032701 003001
002714 001401
002716 005725
002720 116300 000035
002724 020027 000007
002730 101411
002732 004567 001450
002736 002634
002740 005267 176352
002744 005367 176344
002750 000177 175074
002754 000300
002756 050002
002760 056702 176352
002764 010267 176260
002770 000300
002772 110064 000001
002776 032764 040000 177774
003004 001004
003006 004567 001366
003012 002601
003014 000755
003016 005764 177774
003022 100404

CHDCOM: MOV R0, CNTADR
MOV R1, CURFLG
BIT #3, R1
BEQ 10\$
MOV (R5)+, CURADR
MOV (R5)+, CURADR+2
MOV (R5)+, R0
CLC
ROR R0
NEG R0
MOV R0, CURCNT
BIT #1, R1
PEQ 10\$
TST (R5)+
10\$: MOVB PCURDV(R3), R0
CMP R0, #7
BLOS 20\$
JSR R5, ERRCS1
WORD INVDVN-ERMBAS
INC DATAER
DEC ERRCNT
15\$: JMP @CUPGER
20\$: SWAB R0
BIS R0, R2
BIS RPCS, R2
MOV R2, CURCMD
SWAB R0
MOVB R0, 1(R4)
BIT #4000, RPS(R4)
BNE 30\$
JSR R5, ERRCS
WORD UNOFFL-ERMBAS
BR 15\$
30\$: TST RPS(R4)
BMI 40\$

; SAVE ADR OF BYTE COUNT
; SAVE FLAGWD FOR TERMINATION
; THIS CMD HAVE BUS ADR & HD CNT?
; Y, N-10\$
; STORE 2 WORD BUS ADR
; GET BYTE COUNT
; MAKE IT A WORD COUNT
; MAKE IT NEGATIVE
; SAVE IT
; THERE A 4TH WORD?
; Y, N-10\$
; BYPASS IT
; GET CURRENT UNIT #
; IS IT A VALID UNIT #?
; N, Y-20\$
; ISSUE INV UNIT # ERROR MSG
; ADJUST ERROR COUNTS
; GO TO MPG ERR RETURN POINT
; ALIGN UNIT # BITS
; SET UNIT # INTO CMD CODE WORD
; SET HEADER & MODE BITS IN CMD CODE
; SAVE CURR CMD CODE
; MOVE UNIT # TO HIGH BYTE
; OF RPCS TO SELECT THE DRIVE
; IS THE UNIT ON-LINE?
; N, Y-30\$
; ISSUE UNIT OFF-LINE ERROR MSG
; GO TO ERROR EXIT
; IS UNIT READY?
; N, Y-40\$

```

840 003024' 004567 001350 JSR RS,ERRCS ;ISSUE UNIT NOT RDY ERROR MSG
841 003030' 002517 .WORD UNRDY-ERMBAS
842 003032' 000746 BR 15$ ;GO TO ERROR EXIT
843 003034' 032764 001000 177774 40$: BIT @1000,RPOS(R4) ;IS THE UNIT UNSAFE?
844 003042' 001404 BEQ 50$ ;Y,N-50$
845 003044' 004567 001330 JSR RS,ERRCS ;ISSUE UNSAFE ON INIT ERROR MSG
846 003050' 002460 .WORD INITUS-ERMBAS
847 003052' 000736 BR 15$ ;GO TO ERROR EXIT
848 003054' 006300 50$: ASL R0 ;USING THE UNIT #, INDEX INTO
849 003056' 060700 ADD PC,R0 ;THE ATTN TBL FOR BITS FOR
850 003060' 062700 000136 ADD @ATATBL-,R0 ;THIS UNIT #
851 003064' 112067 000146 MOVB (R0)+,MYATA ;STORE UNIT'S ATTN BIT
852 003070' 111067 000144 MOVB (R0),OTHATA ;STORE OTHER ATTN BITS
853 003074' 016400 177774 MOV RPOS(R4),R0 ;GET THE ATTN BITS
854 003100' 136700 000132 BITB MYATA,R0 ;IS MY ATTN ALREADY SET?
855 003104' 001403 BEQ 55$ ;Y,N-55$
856 003106' 116764 000124 177774 MOVB MYATA,RPOS(R4) ;KNOCK IT DOWN
857 003114' 004767 000122 55$: JSR PC,SUIORG ;GO SET UP REGS FOR I/O
858 003120' 016767 174666 176134 MOV RTRY,CURRTY ;INITIALIZE RETRY COUNT
859 003126' 007067 176132 CLR RTRYIP ;CLEAR RETRY IN PROGRESS FLAG
860 003132' 012767 002504 001132 MOV @IOTERM-ERMBAS,INTEAD ;INIT TERMINATION ERROR MSG
861 003140' 042767 000005 174634 BIC @IOERR+MYATTN,DFLGMD ;HSKIP INT FLAGS
862 003146' 02767 000002 174626 BIS @DOTERM,DFLGMD ;SET THE "PROCESS TERMINATION" FLAG
863 003154' 022713 000010 BIS @WT4IOT,(R3) ;SET WAITING FOR I/O TERM FLAG
864 003160' 116764 176065 000001 MOVB CURCMD+1,1(R4) ;LOAD HIGH BYTE OF RPCS REG
865 003166' 110214 MOVB R2,(R4) ;ISSUE SPECIFIED CMD
866 003170' 005767 174606 TST DFLGMD ;"NOWAIT" BIT SET?
867 003174' 107203 BPL 60$ ;Y,N-60$
868 003176' 042713 000010 BIC @WT4IOT,(R3) ;RESET WAITING FOR I/O TERM
869 003202' 000404 BR 70$ ;GO TO EXIT
870 003204' 004577 174636 60$: JSR RS,@CIOBSY ;WAIT FOR I/O TO COMPLETE
871 003210' 004767 000762 JSR PC,PROCTH ;GO PROCESS TERMINATION
872 003214' 000205 70$: RTS ;EXIT IN-LINE TO USER PROG
873
874
875 003216' 001 376 ATATBL: .BYTE 001,376
876 003220' 002 375 .BYTE 002,375
877 003222' 004 373 .BYTE 004,373
878 003224' 010 367 .BYTE 010,367
879 003226' 020 357 .BYTE 020,357
880 003230' 040 337 .BYTE 040,337
881 003232' 100 277 .BYTE 100,277
882 003234' 200 177 .BYTE 200,177
883
884 003236' 000000 MYATA: .WORD 0
885 003240' 000000 OTHATA: .WORD 0

```



```

887 ;SET UP DEVICE REGS FOR I/O
888
889 ;JSR PC,SUIORG S/R CALL
890
891 ;R4 = RPCS ADR
892 ;R3 = PROG TBL ADR
893 ;R2 = CMND CODE
894 ;R1 = CMND FLAGWORD
895
896 ;DESTROYS R0
897
898 003242' 032701 000060 SUIORG: BIT #60,R1 ;NEED TO SET UP CYL/HEAD/SECT?
899 003246' 001414 BEQ 10$ ;Y,N-10$
900 003250' 116764 174534 000010 MOVB SECT,RPDA(R4) ;STORE SECT #
901 003256' 032701 000020 BIT #20,R1 ;NEED CYL & HEAD #?
902 003262' 001406 BEQ 10$ ;Y,N-10$
903 003264' 116764 174516 000011 MOVB HEAD,RPDA+1(R4) ;LOAD HEAD #
904 003272' 016764 174506 000006 MOV CYL,RPCA(R4) ;LOAD CYL #
905 003300' 032701 000004 10$: BIT #4,R1 ;DATA XFER CMND?
906 003304' 001417 BEQ 20$ ;Y,N-20$
907 003306' 016700 175740 MOV CURADR,R0 ;GET HIGH BITS OF ADR
908 003312' 042700 177774 BIC #177774,R0 ;RESET BITS ABOVE A17
909 003316' 006300 ASL R0 ;ALIGN BITS A16 & A17
910 003320' 006300 ASL R0 ;TO THEIR POSITIONS IN RPCS
911 003322' 006300 ASL R0
912 003324' 006300 ASL R0
913 003326' 050020 BIS R0,R2 ;SET THEM INTO CMND CODE WORD
914 003330' 016764 175720 000004 MOV CURADR+2,RPBA(R4) ;LOAD BITS 0-15 OF ADR
915 003336' 016764 175714 000002 MOV CURCNT,RPWC(R4) ;LOAD WORD COUNT
916 003344' 012763 072460 000030 20$: MOV #30000,PTOCNT(R3) ;INITIALIZE 10 SEC TIMEOUT CNT
917 003352' 000207 RTS PC ;EXIT IN-LINE

```

```

919          .SBTTL  RPO2/RPO3 INTERRUPT SERVICE ROUTINE
920
921 003354' 004067 001766          RPINT: JSR    R0, SAVREG          ;SAVE ALL REGISTERS
922 003360' 004567 002040          JSR    RS, STSTAT        ;GO STORE ALL DEV REG'S
923 003364' 175602          WORD  ISTAT-
924 003366' 005267 175742          INC    INTCNT           ;ADD 1 TO INTERRUPT CNT
925 003372' 004767 002002          JSR    PC, SUPTAD       ;SET UP PROG TBL & RPCS1 ADR'S
926 003376' 016701 175644          MOV    CURFLG, R1      ;GET THIS CMND'S FLGWD
927 003402' 032767 000004 174372  BIT    #MYATTN, DFLGWD  ;WAITING FOR ATTN INT?
928 003410' 001032          BNE    ATNINT          ;N, Y-ATTNINT
929 003412' 042714 000100          BIC    #100, (R4)      ;RESET INTERRUPT ENABLE
930 003416' 016467 000002 175634  MOV    RPNCR(R4), FINCNT ;STORE FINAL WORD COUNT
931 003424' 032714 140000          BIT    #140000, (R4)  ;'ERR' OR 'HE' SET?
932 003430' 001055          BNE    ERRFND         ;N, Y-ERRFND
933 003432' 032701 000100          BIT    #100, R1       ;CMND RESULT IN TWO INTERRUPTS?
934 003436' 001411          BEQ    CLRWTF         ;Y, N-CLRWTF
935 003440' 052767 000004 174334  BIS    #MYATTN, DFLGWD  ;SET 'WAITING FOR ATTN' FLAG
936 003446' 052714 020000          BIS    #20000, (R4)   ;SET ATTN INTERRUPT ENABLE
937 003452' 000405          BR     INTEX          ;GO TO INTERRUPT EXIT
938 003454' 052767 000001 174320  SETERR: BIS    #IOERR, DFLGWD ;SET THE TERMINATION I/O ERR FLAG
939 003462' 042713 000010          CLRWTF: BIC    #MYIOT, (R3) ;RESET WAITING FOR I/O TERM
940 003466' 004067 001670          INTEX: JSR    R0, RESREG ;RESTORE ALL REGISTERS
941 003472' 000177 174376          JMP    #RTNINT        ;EXIT FROM INTERRUPT
942
943          ATNINT: BIC    #20000, (R4)   ;RESET ATTN INT ENABLE
944 003476' 042714 020000          BIT    #140000, (R4)  ;'ERR' OR 'HE' SET?
945 003502' 032714 140000          BNE    SETERR         ;N, Y-SETERR
946 003506' 001362          BITB   MYATA, RPO5(R4) ;IS MY ATTN BIT SET?
947 003510' 136764 177522 177774  BEQ    10S            ;Y, N-10S
948 003516' 001404          MOVB   MYATA, RPO5(R4) ;RESET MY ATTN BIT
949 003520' 116764 177512 177774  BR     CLRWTF         ;GO TO INT EXIT
950 003526' 000755          BITB   0THATA, RPO5(R4) ;IS ANOTHER ATTN BIT SET?
951 003530' 136764 177504 177774  BEQ    20S            ;Y, N-20S
952 003536' 001406          MOVB   0THATA, RPO5(R4) ;RESET OTHER ATTN BITS
953 003540' 116764 177474 177774  BIS    #20000, (R4)   ;SET ATTN INT ENABLE AGAIN
954 003546' 052714 020000          BR     INTEX          ;GO TO INT EXIT
955 003552' 000745          MOV    #NOATA-ERMBAS, INTEAD ;SET UP 'ATTN NOT SET' ERR MSG ADR
956 003554' 012767 002532 000510 20S:  BR     SETERR         ;GO SET ERROR FLAG & EXIT
957 003562' 000734
958
959          ERRFND: BIT    #177407, RPER(R4) ;ANY HARD ERROR BITS IN RPER?
960 003564' 032764 177407 177776  BNE    SETERR         ;N, Y-SETERR
961 003572' 001330          MOV    RPO5(R4), R0   ;GET THE RPO5 REG
962 003574' 016400 177774          BIT    #15000, R0     ;ANY HARD ERROR BITS IN RPO5?
963 003600' 032700 015000          BNE    SETERR         ;N, Y-SETERR
964 003604' 001323          COM    R0             ;FLIP RPO5 BITS
965 003606' 005100          BIT    #140000, R0   ;STILL ON-LINE & READY?
966 003610' 032700 140000          BNE    SETERR         ;Y, N-SETERR
967 003614' 001317          MOV    RPER(R4), R0  ;GET THE RPER REG
968 003616' 016400 177776          BIT    #370, R0      ;RETRYABLE ERROR?
969 003622' 032700 000370          BEQ    SETERR         ;Y, N-SETERR
970 003630' 001712          BIT    #200, R1      ;DO RETRIES ON THIS CMND?
971 003634' 032701 000200          BEQ    SETERR         ;Y, N-SETERR
972
973          CKRTRY: TST    RTRYIP       ;ALREADY DONE RETRIES ON THIS CMND?
974 003636' 005767 175422          BNE    55S           ;N, Y-55S

```

975	003644'	005767	175412		TST	CURRTY	:ARE RETRIES SPECIFIED?
976	003650'	001701			BEQ	SETERR	:Y,N-SETERR
977	003652'	032700	000020		BIT	#20,RO	:TIMING ERROR?
978	003656'	001411			BEQ	42\$:Y,N-42\$
979	003660'	005267	175434		INC	TIECNT	:ADD 1 TO TIMING ERROR COUNT
980	003664'	012767	046524	003434	MOV	#TH,RTYID	:SET RETRY I.D. TO "TME"
981	003672'	012767	042505	003430	MOV	#EE,RTYID+2	
982	003700'	000451			BR	55\$:GO CK RETRY COUNT
983	003702'	032700	000040	42\$:	BIT	#40,RO	:CHECKSUM ERROR?
984	003706'	001411			BEQ	44\$:Y,N-44\$
985	003710'	005267	175406		INC	CSECNT	:ADD 1 TO CHECKSUM ERROR COUNT
986	003714'	012767	051503	003404	MOV	#CS,RTYID	:SET RETRY I.D. TO "CSME"
987	003722'	012767	042515	003400	MOV	#ME,RTYID+2	
988	003730'	000435			BR	55\$:GO CK RETRY COUNT
989	003732'	032700	000100	44\$:	BIT	#100,RO	:WORD PARITY ERROR?
990	003736'	001406			BEQ	46\$:Y,N-46\$
991	003740'	005267	175360		INC	WPECNT	:ADD 1 TO WORD PARITY ERROR COUNT
992	003744'	012767	050127	003354	MOV	#WP,RTYID	:SET RETRY I.D. TO "WPE"
993	003752'	000421			BR	50\$:GO CK RETRY COUNT
994	003754'	032700	000200	46\$:	BIT	#200,RO	:LONGITUDINAL PARITY ERROR?
995	003760'	001406			BEQ	48\$:Y,N-48\$
996	003762'	005267	175340		INC	LPECNT	:ADD 1 TO LONGITUDINAL PARITY COUNT
997	003766'	012767	050114	003332	MOV	#LP,RTYID	:SET RETRY I.D. TO "LPE"
998	003774'	000410			BR	50\$:GO CK RETRY COUNT
999	003776'	032700	000010	48\$:	BIT	#10,RO	:WRITE CHECK ERROR?
1000	004002'	001410			BEQ	55\$:Y,N-55\$
1001	004004'	005267	175320		INC	WCECNT	:ADD 1 TO WRITE CHECK ERROR COUNT
1002	004010'	012767	041527	003310	MOV	#WC,RTYID	:SET RETRY I.D. TO "WCE"
1003	004016'	012767	020105	003304	MOV	#E,RTYID+2	
1004	004024'	005367	175232	50\$:	DEC	CURRTY	:DECREMENT RETRY COUNT
1005	004030'	100004		55\$:	BPL	60\$:CNT EXHAUSTED? (Y,N-60\$)
1006	004032'	012767	002547	000232	MOV	#RTYEXH-ERMBAS,INTEAD	:SET UP EXHAUSTED RETRIES ERR MSG ADDR
1007	004040'	000605			BR	SETERR	:GO TO ERR R EXIT
1008	004042'	005267	175264	60\$:	INC	RETRY\$:ADD 1 TO RETRY TOTAL CNT
1009	004046'	005267	175212		INC	RTRYIP	:SET RETRY IN PROGRESS FLAG
1010	004052'	016702	175172		MOV	CURCMD,R2	:GET CURR CMD IN R2
1011	004056'	004767	177160		JSR	PC,SUIORG	:SET UP DEV REGS
1012	004062'	116764	175163	000001	MOVB	CURCMD+1,1(R4)	:LOAD HIGH BYTE OF RPCS
1013	004070'	110214			MOVB	R2,(R4)	:RE-ISSUE THE ORIG CMD
1014	004072'	000167	177370		JMP	INTEX	:GO TO INT EXIT

```

1016 .SBTTL SUBROUTINES FOR RP02/RP03 FUNCTION ROUTINES
1017
1018
1019 ;CHECK IF DEVICE IS BUSY AND WAIT IF IT IS
1020
1021 ;JSR PC,CKDBSY S/R CALL
1022
1023 ;DESTROYS R0,R3,R4
1024
1025 ;ON EXIT: R3 = PROG TBL ADR
1026 ; R4 = RPCS1 ADR
1027
1028 004076' 004767 001276 CKDBSY: JSR PC,SUPTAD ;SET UP PROG TBL & RPCS1 ADR'S
1029 004102' 032714 020100 10S: BIT #20100,(R4) ;EITHER INT ENABLE ON?
1030 004106' 001403 BEQ 20S ;Y,N-20S
1031 004110' 004577 173732 JSR R5,DCIOBSY ;RELEASE CONTROL
1032 004114' 000772 BR 10S ;GO CK AGAIN
1033 004116' 032767 000002 173656 20S: BIT #00TERM,DFLGWD ;HAVE TO PROCESS PREV TERMINATION?
1034 004124' 001403 BEQ 30S ;Y,N-30S
1035 004126' 004767 000044 JSR PC,PROCTM ;GO PROCESS TERMINATION
1036 004132' 000763 BR 10S ;GO CK INT ENABLE AGAIN
1037 004134' 016767 173666 000012 30S: MOV IVCTAD,40S ;STORE INT VECTOR ADR
1038 004142' 016767 173662 000006 MOV PSWD,45S ;STORE PROC STATUS WORD
1039 004150' 004577 173712 JSR R5,ASETVEC ;GO SET UP INTERRUPT VECTOR
1040 004154' 000000 40S: .WORD XXXX ;INT VECTOR ADR
1041 004156' 000000 45S: .WORD XXXX ;PSW
1042 004160' 177174 .WORD RPINT- ;REL INT ROUT ADR
1043 004162' 010567 175054 STMADR: MOV R5,OBJADR ;SAVE CURR USER STMT ADR
1044 004166' 162767 000004 175046 SUB #4,OBJADR
1045 004174' 000207 RTS PC ;EXIT IN-LINE
1046
1047
1048 ;PROCESS TERMINATION OF PREVIOUS I/O FUNCTION
1049
1050 ;JSR PC,PROCTM S/R CALL
1051
1052 ;R3 = PROG TABLE ADR
1053
1054 ;DESTROYS R0
1055
1056 004176' 010146 PROCTM: MOV R1,-(SP) ;SAVE R1 & R2
1057 004200' 010246 MOV R2,-(SP)
1058 004202' 042767 000002 173572 BIC #00TERM,DFLGWD ;RESET PROCESS TERMINATION FLAG
1059 004210' 032767 000010 175030 BIT #10,CURFLG ;INCR BYTE COUNT?
1060 004216' 001417 BEQ 20S ;Y,N-20S
1061 004220' 016700 175032 MOV CURCNT,R0 ;GET INITIAL WORD CNT
1062 004224' 005400 NEG R0 ;MAKE IT POSITIVE AGAIN
1063 004226' 016701 175026 MOV FINCNT,R1 ;GET FINAL WORD CNT
1064 004232' 100001 BPL 10S ;IS IT NEGATIVE? (Y,N-10S)
1065 004234' 005401 NEG R1 ;MAKE IT POSITIVE
1066 004236' 160100 10S: SUB R1,R0 ;SUB REMAINING CNT FROM INITIAL CNT
1067 004240' 006300 ASL R0 ;MAKE IT A BYTE CNT
1068 004242' 010067 173552 MOV R0,SIZE ;STORE # OF BYTES ACTUALLY XFERRD
1069 004246' 016701 174772 MOV CNTADR,R1 ;GET ADR OF BYTE CNT TOTALS
1070 004252' 060011 ADD R0,(R1) ;ADD IN THIS CNT
1071 004254' 005541 ADC -(R1) ;UPDATE MOST SIGNF WORD OF CNT
    
```

```

1072 004256' 032767 000001 173516 20S: BIT      #IOERR,DFLGWD      ;WAS THERE AN ERROR?
1073 004264' 001412          BEQ      PROCEX          ;Y,N-PROCEX
1074 004266' 004557 000132          JSR      RS,EPRIS        ;GO ISSUE I/O TERMINATION
1075 004272' 002504          INTEAD: .WORD      IOTERM-ERMBAS ;ERROR MSG
1076 004274' 004767 000024          JSR      PC,RINTV       ;RESET THE INT VECTOR
1077 004300' 012602          MOV      (SP)+,R2       ;RESTORE R1 & R2
1078 004302' 012601          MOV      (SP)+,R1
1079 004304' 004577 173540          JSR      RS,@CUPGER     ;GO TO MPG ERR RETN POINT
1080 004310' 000207          RTS      PC            ;RETURN IN-LINE
1081 004312' 004767 000006          PROCEX: JSR      PC,RINTV ;GO RESET INT VECTOR
1082 004316' 012602          MOV      (SP)+,R2       ;RESTORE R1 & R2
1083 004320' 012601          MOV      (SP)+,R1
1084 004322' 000207          RTS      PC            ;EXIT IN-LINE
1085
1086
1087          ;RESET INTERRUPT VECTOR S/R
1088
1089          ;JSR      PC,RINTV      S/R CALL
1090
1091          ;R3 = PROG TBL ADR
1092
1093          ;DESTROYS R0
1094
1095 004324' 004567 000020          RINTV: JSR      RS,TVECT   ;GO CK IF I HAVE VECTOR CONTROL
1096 004330' 000406          BR      RINTEX         ;BR IF I DON'T
1097 004332' 016767 173470 000004          MOV      IVCTAD,IOS    ;GET CURR INT VECT ADR
1098 004340' 004577 173524          JSR      RS,@CLRVEC    ;GO HAVE MPG CLEAR IT
1099 004344' 000600          10S:   .WORD      XXXX
1100 004346' 000207          RINTEX: RTS      PC      ;EXIT IN-LINE
1101
1102
1103          ;TEST INTERRUPT VECTOR S/R
1104
1105          ;JSR      RS,TVECT   S/R CALL
1106          ;BR      LABEL      EXECUTED IF NOT SAME
1107
1108          ;R3 = PROG TBL ADR
1109
1110          ;DESTROYS R0
1111
1112 004350' 016767 173452 000010          TVECT: MOV      IVCTAD,20S ;GET CURR INT VECT ADR
1113 004356' 016346 000004          MOV      PFWADR(R3),-(SP) ;STORE FLGWD ADR TO IDENTIFY ME
1114 004362' 004577 173504          JSR      RS,@STVECT   ;DO I HAVE VECTOR CONTROL?
1115 004366' 000000          20S:   .WORD      XXXX ;MPG WILL TELL ME SINCE I CAN'T
1116 004370' 176764          .WORD      RPINT-    ;GET AT LOWER MEM IF MEM MGMT
1117 004372' 000401          BR      TVECTX        ;BR IF I DONT'T HAVE CNTRL
1118 004374' 005725          TST      (RS)+        ;BYPASS BR INST IN S/R CALL
1119 004376' 000205          TVECTX: RTS      RS    ;EXIT IN-LINE
    
```

```

;ERROR INFORMATION DISPLAY S/R
1121
1122
1123      ;JSR      RS,ERRCS      S/R CALL FOR CURR STATUS
1124      ;          ,ERRCS1    S/R CALL FOR CURR STATUS W/O STORING
1125      ;          ,ERRIS    S/R CALL FOR INT STATUS
1126      ;.WORD  MSGADR-ERMBAS REL ADR OF ERROR MSG
1127
1128      ;R3 = PROG TABLE ADR
1129      ;DESTROYS R0,R1,R2
1130
1131 004400' 004567 001020      ERRCS: JSR      RS,STSTAT      ;STORE CURR STATUS
1132 004404' 174610      ;.WORD  CSTAT-
1133 004406' 012767 174246 000332      ERRCS1: MOV     #CSTAT-ERSTAD,ERSTAD      ;STORE ADR OF CURR STATUS
1134 004414' 012767 174436 000144      MOV     #CSTAT+6-EBSBAS,EBSTAT
1135 004422' 004406      BR      ERRCOM      ;GO TO COMMON POINT
1136 004424' 012767 174220 000314      ERRIS: MOV     #ISTAT-ERSTAD,ERSTAD      ;STORE ADR OF LAST INT STATUS
1137 004432' 012767 174410 000126      MOV     #ISTAT+6-EBSBAS,EBSTAT
1138 004440' 012567 000066      ERRCOM: MOV    (RS)+,ERMBAS      ;STORE MSG ADR
1139 004444' 005267 174644      INC     ERRCNT      ;ADD 1 TO ERROR CNT
1140 004450' 012767 000001 173344      MOV     #1,ERRPRI      ;SET THE ERROR INDICATOR
1141 004456' 032763 020400 000002      BIT     #DOEPRK+PRONER,POPSW(R3) ;ERROR CHECKING OR PRINTING INHIBITED?
1142 004464' 001163      BNE     ERREX      ;Y N-ERREX
1143 004466' 010446      MOV     R4,-(SP)      ;SAVE R4 & RS
1144 004470' 010546      MOV     RS,-(SP)
1145 004472' 005304      CLR     R4      ;SET USER MODE PRINT FLAG
1146 004474' 004767 000750      JSR     PC,DEVID      ;DISPLAY DEVICE I.D.
1147 004500' 010700      MOV     PC,R0      ;GET START ADR OF ERROR MSG
1148 004502' 062700 000030      ADD     #ERMBAS-.,R0
1149 004506' 061000      ADD     (R0),R0
1150 004510' 012701 177777      MOV     #-1,R1
1151 004514' 005201      10$: INC     R1      ;INITIALIZE MSG LENGTH
1152 004516' 105720      TSTB   (R0)+      ;ADD 1 TO MSG LENGTH
1153 004520' 001375      BNE     10$      ;MSG TERMINATOR?
1154 004522' 010167 000006      MOV     R1,ERMBAS+2      ;Y N-10$
1155 004526' 004567 001266      JSR     RS,PRINT      ;STORE MSG LENGTH
1156 004532' 000000      ERMBAS: .WORD  XXXX      ;PRINT ERROR MSG SPECIFIED
1157 004534' 000000      .WORD  XXXX
1158 004536' 026727 177770 002634      CMP     ERMBAS,#INVDVN-ERMBAS      ;INVALID UNIT # MSG OR HIGHER?
1159 004544' 103103      BHIS   ERRSNM      ;N,Y-ERRSNM
1160 004546' 010701      MOV     PC,R1      ;GET ADR OF CODE AREA IN ERR MSG
1161 004550' 062701 002300      ADD     #COOFLD-.,R1
1162 004554' 010700      MOV     PC,R0      ;SET UP ADR OF ERROR CODE TBL
1163 004556' 062700 000260      ADD     #ERCOTB-.,R0
1164 004562' 010702      MOV     PC,R2      ;SET UP ADR OF STORED DEV REG'S
1165 004564' 062702      EBSBAS: ADD    (PC)+,R2
1166 004566' 174436      EBSTAT: .WORD  CSTAT+6-EBSBAS
1167 004570' 012767 000015 000142      MOV     #13.,70$      ;INITIALIZE MSG LENGTH
1168 004576' 012746 000100      MOV     #64,-(SP)      ;INITIALIZE CODE FIELD CNT
1169 004602' 014205      15$: MOV     -(R2),R5      ;GET NEXT DEV REG WORD
1170 004604' 000305      17$: SWAB   R5      ;GET DESIRED BYTE IN LOW BYTE
1171 004606' 112004      20$: MOV     (R0)+,R4      ;GET FLAG & LENGTH BYTE
1172 004610' 001444      BEQ    60$      ;END OF CODE TBL? (N,Y-60$)
1173 004612' 122704 000377      CMPB   #377,R4      ;GO TO NXT DEV REG WORD?
1174 004616' 001771      BEQ    15$      ;N,Y-15$
1175 004620' 122704 000376      CMPB   #376,R4      ;GO TO NXT BYTE IN DEV REG WORD?
1176 004624' 001767      BEQ    17$      ;N,Y-17$
    
```


1177	004626'	105704		TSTB	R4		;BIT VALUE OF 0 = AN ERROR CONDITION?
1178	004630'	100003		BPL	30\$;Y,N-30\$
1179	004632'	131005		BITB	(R0),R5		;THIS BIT RESET IN DEV REG BYTE?
1180	004634'	001407		BEQ	40\$;N,Y-40\$
1181	004636'	000402		BR	35\$;GO TO NXT TBL ENTRY
1182	004640'	131005	30\$:	BITB	(R0),R5		;THIS ERROR BIT SET IN DEV REG BYTE?
1183	004642'	001004		BNE	40\$;N,Y-40\$
1184	004644'	042704	177770	35\$:	BIC	#177770,R4	;ISOLATE ENTRY LENGTH
1185	004650'	060400		ADD	R4,R0		;POINT AT NXT CODE TBL ENTRY
1186	004652'	000755		BR	20\$;GO CK FOR NXT CODE
1187	004654'	042704	177770	40\$:	BIC	#177770,R4	;ISOLATE I.D. NAME LENGTH + 1
1188	004660'	020416		CMP	R4,(SP)		;ENOUGH ROOM FOR NAME?
1189	004662'	101017		BHI	60\$;Y,N-60\$
1190	004664'	060467	000050	ADD	R4,70\$;ADJ MSG LENGTH FOR NAME
1191	004670'	005304		DEC	R4		;ADJ FOR BIT MASK CHAR
1192	004672'	005200		INC	R0		;POINT PAST BIT MASK
1193	004674'	021627	000100	CMP	(SP),#64.		;FIRST ERROR CODE IN MSG?
1194	004700'	001403		BEQ	50\$;N,Y-50\$
1195	004702'	112721	000054	MOVB	#',(R1)+		;MOVE COMMA TO MSG
1196	004706'	005316		DEC	(SP)		;ADJ REMAINING ROOM IN MSG
1197	004710'	112021		50\$:	MOVB	(R0)+,(R1)+	;MOVE ERROR CODE TO MSG
1198	004712'	005316		DEC	(SP)		;ADJ REMAINING ROOM IN MSG
1199	004714'	005304		DEC	R4		;MOVED ALL NAME CHARS?
1200	004716'	001374		BNE	50\$;Y,N-50\$
1201	004720'	000732		BR	20\$;GO CK FOR MORE ERROR BITS
1202	004722'	005004		60\$:	CLR	R4	;SET USER MODE PRINT
1203	004724'	022627	000100	CMP	(SP)+,#64.		;ANY ERROR CODES PUT IN MSG?
1204	004730'	001404		BEQ	80\$;Y,N-80\$
1205	004732'	004567	001062	JSR	R5,PRINT		;GO ISSUE ERROR BITS MSG
1206	004736'	002074		.WORD	ERBMMSG-		
1207	004740'	000116		70\$:	.WORD	78.	
1208	004742'	004567	000616	80\$:	JSR	R5,DISPST	;DISPLAY DEVICE REG'S
1209	004746'	000000		ERSTAD:	.WORD	XXXX	
1210	004750'	004767	000774	JSR	PC,PTIWD		;DISPLAY CYL HEAD SECT VALUES
1211	004754'	016300	000022	ERRSNM:	MOV	PSRCST(R3),R0	;GET ADR OF SRC STMNTS
1212	004760'	111001		110\$:	MOVB	(R0),R1	;SAVE STMT LENGTH
1213	004762'	026067	000004 174252	CMP	4(R0),OBJADR		;ERROR OCCUR ON THIS STMNT?
1214	004770'	001402		BEQ	120\$;N,Y-120\$
1215	004772'	060100		ADD	R1,R0		;POINT AT NXT STMNT
1216	004774'	000771		BR	110\$;GO CK NXT STMNT
1217	004776'	005720		120\$:	TST	(R0)+	;SET UP ADR OF STMNT # DATA
1218	005000'	010701		MOV	PC,R1		;SET UP DATA OUTPUT ADR
1219	005002'	062701	002022	ADD	#STMNUM-,R1		
1220	005006'	004577	173050	JSR	R5,DECSA		;CONVERT IT TO ASCII
1221	005012'	012767	020040 002010	MOV	#20040,STMNUM+4		;SET 2 LOW DIGITS TO SPACES
1222	005020'	004567	000774	JSR	R5,PRINT		;ISSUE STMNT # MSG
1223	005024'	001770		.WORD	STMMG-		
1224	005026'	177762		.WORD	-14.		
1225	005030'	012605		MOV	(SP)+,R5		;RESTORE R5 & R4
1226	005032'	012604		MOV	(SP)+,R4		
1227	005034'	000205		ERREX:	RTS	R5	;EXIT IN-LINE

1274	005301'	006	040420	052124	.ASCII	<006><020>/ATTM/	
	005306'	032116					
1275	005310'	004006	052101	047124	.ASCII	<006><010>/ATTN3/	
	005316'	063					
1276	005317'	006	040404	052124	.ASCII	<006><004>/ATTN2/	
	005324'	031116					
1277	005326'	001006	052101	047124	.ASCII	<006><002>/ATTN1/	
	005334'	061					
1278	005335'	006	040401	052124	.ASCII	<006><001>/ATTN0/	
	005342'	030116					
1279	005344'	000			.BYTE	0	;TABLE TERMINATOR
1280		005346'			.EVEN		

```

1282 .SBTTL SUBROUTINES FOR RPO2/RPO3 DEVICE ROUTINE
1283
1284
1285
1286 ;SAVE REGISTERS R0 THRU R5
1287
1288 ;JSR R0,SAVREG S/R CALL
1289
1290 SAVREG: MOV R1,-(SP) ;SAVE R0 THRU R5
1291 MOV R2,-(SP)
1292 MOV R3,-(SP)
1293 MOV R4,-(SP)
1294 MOV R5,-(SP)
1295 MOV R0,PC ;EXIT IN-LINE
1296
1297
1298 ;RESTORE REGISTERS R0 THRU R5
1299
1300 ;JSR R0,RESREG S/R CALL
1301
1302 RESREG: TST (SP)+ ;RESTORE R5 THRU R0
1303 MOV (SP)+,R5
1304 MOV (SP)+,R4
1305 MOV (SP)+,R3
1306 MOV (SP)+,R2
1307 MOV (SP)+,R1
1308 RTS R0 ;EXIT IN-LINE
1309
1310
1311 ;SET PROGRAM'S PROG TABLE ADR IN R3 & RPCS ADR IN R4
1312
1313 ;JSR PC,SUPTAD S/R CALL
1314
1315 SUPTAD: MOV PC,R3 ;SET UP LOCATION ZERO ADR
1316 ADD #LOC2-,R3
1317 SUB -2(R3),R3 ;SUBTRACT PROG TBL LENGTH
1318 MOV DREGAD,R4 ;GET DEV REG BASE ADR
1319 ADD #4,R4 ;POINT AT RPCS REGISTER
1320 RTS PC ;EXIT IN-LINE
1321
1322
1323 ;STORE DEVICE'S STATUS REGISTERS
1324
1325 ;JSR R5,STSTAT S/R CALL
1326 ;WORD STADR- REL STORAGE ADR
1327 ;DESTROYS R0,R1
1328
1329 STSTAT: MOV R5,R1 ;GET REL STORAGE ADR & MAKE
1330 ADD (R5)+,R1 ;IT ABSOLUTE
1331 MOV DREGAD,R0 ;GET DEV REG ADR
1332 MOV #REGNUM,R2 ;SET UP # OF REGS
1333 10$: MOV (R0)+,(R1)+ ;STORE DEV REG
1334 DEC R2 ;DONE ALL REGS?
1335 BNE 10$ ;Y,N-10$
1336 RTS R5 ;EXIT IN-LINE
1337

```

```

1338
1339
1340
1341
1342
1343
1344
1345
1346
1347 005450 012700 031060 DEVID: MOV #02,RO ;INITIALIZE TO RPO2
1348 005454 032763 000020 000032 BIT #20,PHOLCD(R3) ;IS THIS AN RPO2?
1349 005462 001402 BEQ 105 ;N,Y-105
1350 005464 012700 031460 MOV #03,RO ;SET UP FOR AN RPO3
1351 005470 010067 000516 105: MOV RO,UNITMG+6 ;TAILOR DEV I.D. MSG
1352 005474 012767 000026 000056 MOV #22,DEVIML ;INITIALIZE TO NORMAL MSG LNGTH
1353 005508 116300 000035 MOV#B PCURV(R3),RO ;GET CURR UNIT #
1354 005506 020027 000007 CMP RO,#7 ;VALID UNIT #?
1355 005512 101007 BHI DEVIIV ;Y N-DEVIIV
1356 005514 004577 172340 JSR RS,28TASLZ ;CONVERT # TO DECIMAL ASCII
1357 005520 000510 .WORD UNASCI-
1358 005522 016767 000506 000510 MOV UNASCI+4,UNASCI ;MOVE ASCII # TO 1ST TWO DIGITS
1359 005530 000410 BR DEVIIV ;GO ISSUE MSG
1360 005540 012767 000032 000020 DEVIIV: MOV #26,DEVIML ;SET ERR COND MSG LNGTH
1361 005540 042700 177400 BIC #177400,RO ;RESID HIGH BYTE
1362 005544 004577 172306 JSR RS,28BINASC ;CONVERT BINARY # TO ASCII
1363 005550 000460 .WORD UNASCI-
1364 005552 004567 000242 DEVIIV: JSR RS,PRINT ;GO ISSUE UNIT # MSG
1365 005556 000426 .WORD UNITMG-
1366 005560 000026 DEVIIV: .WORD 22
1367 005562 000207 RTS ;EXIT IN-LINE
1368
1369
1370 ;TAILOR STATUS MSG & PRINT IT
1371
1372
1373
1374
1375
1376 005564 010346 DISPST: MOV R3,-(SP) ;SAVE R3
1377 005566 010503 MOV RS,R3 ;GET REL DATA ADR
1378 005570 062503 ADD (RS)+,R3 ;MAKE IT ABS
1379 005572 010546 MOV RS,-(SP) ;SAVE RS
1380 005574 010705 MOV PC,RS ;SET UP ADR OF REG NAMES IN ASCII
1381 005576 062705 172320 ADD #0VREGS-,RS
1382 005602 012746 000013 MOV #11,-(SP) ;SET UP # OF REGISTERS TO DISPLAY
1383 005606 012700 000003 105: MOV #3,RO ;SET UP 3 REG LOOP CNT
1384 005612 010701 MOV PC,R1 ;POINT AT REG NAME IN MSG
1385 005614 062701 000422 ADD #0VRMG-,R1
1386 005620 012521 155: MOV (RS)+,(R1)+ ;MOVE REG NAME TO MSG
1387 005622 012521 MOV (RS)+,(R1)+
1388 005624 005725 TST (RS)+ ;POINT TO NEXT NAME
1389 005626 062701 000012 ADD #10.,R1 ;POINT TO NEXT FIELD IN MSG
1390 005632 005300 DEC RO ;DONE 3 REGS?
1391 005634 001371 BNE 155 ;Y N-155
1392 005636 012300 MOV (R3)+,RO ;CONVERT OCTAL REGISTER CONTENTS
1393 005640 004577 172212 JSR RS,28NASC ;FOR 3 REGISTERS TO ASCII

```

```

1394 005644' 000400 .WORD DVRT1- ;AND PLACE IN THE MSG
1395 005646' 012300 MOV (R3)+,R0
1396 005648' 004577 172202 JSR RS,28,NASC
1397 005650' 000406 .WORD DVRT2-
1398 005652' 012300 MOV (R3)+,R0
1399 005654' 004577 172172 JSR RS,28,NASC
1400 005656' 000414 .WORD DVRT3-
1401 005658' 012767 000050 000034 MOV #40,30$ ;INITIALIZE MSG LENGTH TO 3 REGS
1402 005660' 162716 000003 SUB #3,(SP) ;DECR REGISTER CNT
1403 005700' 100005 BPL 25$ ;< 3 REGS? (Y,N-25$)
1404 005702' 162767 000016 000020 20$: SUB #14,30$ ;SHORTEN MSG LENGTH BY 1 REG
1405 005710' 005216 INC (SP) ;INCR NEG REG CNT
1406 005712' 100773 BMI 20$ ;CNT BACK TO 0? (Y,N-20$)
1407 005714' 010346 25$: MOV R3,-(SP) ;SAVE REG DATA PNTR
1408 005716' 016603 000006 MOV 6(SP),R3 ;RESTORE PROG TBL ADR
1409 005722' 004567 000072 JSR RS,PRINT ;GO PRINT THE MSG
1410 005726' 000310 .WORD DVRGMG-
1411 005730' 000050 30$: .WORD 40
1412 005732' 012603 MOV (SP)+,R3 ;RESTORE REG DATA PNTR
1413 005734' 005716 TST (SP) ;MORE REGS TO GO?
1414 005736' 001323 BNE 10$ ;N Y-10$
1415 005740' 005726 TST (SP)+ ;REMOVE CNT FROM STACK
1416 005742' 012605 MOV (SP)+,RS ;RESTORE RS & R3
1417 005744' 012603 MOV (SP)+,R3
1418 005746' 000205 RTS ;EXIT IN-LINE
1419
1420
1421 ;DISPLAY CYL/HEAD/SECT WORDS' VALUES
1422
1423 ;JSR PC,PRTIWD S/R CALL
1424 ;DESTROYS R0,R1,R2
1425
1426 005750' 016700 172030 PRTIWD: MOV CYL,R0 ;GET CYL VALUE
1427 005754' 004577 172076 JSR RS,28,NASC ;CONVERT ITS VALUE TO ASCII
1428 005760' 000773 .WORD IFCYL-
1429 005762' 016700 172020 MOV HEAD,R0 ;GET & CONVERT HEAD VALUE
1430 005766' 004577 172064 JSR RS,28,NASC
1431 005772' 000776 .WORD IFHEAD-
1432 005774' 016700 172010 MOV SECT,R0 ;GET & CONVERT SECT VALUE
1433 006000' 004577 172052 JSR RS,28,NASC
1434 006004' 001001 .WORD IFSECT-
1435 006006' 004567 000006 JSR RS,PRINT ;PRINT MSG WITH THEIR VALUES
1436 006012' 000734 .WORD INFOG-
1437 006014' 000045 .WORD 37
1438 006016' 000207 RTS PC ;EXIT IN-LINE
    
```



```

1440                                     ;ISSUE MSG TO LIST DEVICE SUBROUTINE
1441
1442                                     ;JSR   RS,PRINT          S/R CALL
1443                                     ;.WORD MSGADR-.        REL ADR OF MSG
1444                                     ;.WORD BYTCNT          MSG BYTE CNT (IF NEGATIVE,
1445                                     ;                       RESET PRT DEV DEDICATED.)
1446                                     ;R3 = PROG TBL ADR
1447                                     ;R4 = FLAGWORD -- IF NEGATIVE, USE CHND MODE PRINT
1448                                     ;DESTROYS R0,R1,R2
1449
1450 006020' 010500          PRINT:  MOV   RS,R0          ;GET MSG ADR & MAKE IT ABS
1451 006022' 062500          ADD    (RS)+,R0
1452 006024' 012501          MOV   (RS)+,R1          ;GET BYTE COUNT
1453 006026' 005704          TST   R4              ;USE CHND MODE PRINT?
1454 006030' 100030          BPL   40$              ;Y,N-40$
1455 006032' 010702          MOV   PC,R2          ;SET UP LINK INFO ADR
1456 006034' 062702 000040  ADD    #20$,R2
1457 006040' 160200          SUB   R2,R0          ;MAKE MSG ADR REL
1458 006042' 010022          MOV   R0,(R2)+      ;STORE MSG ADR
1459 006044' 010112          MOV   R1,(R2)      ;STORE MSG'S BYTE COUNT
1460 006046' 100001          BPL   10$              ;CNT NEG? (Y,N-10$)
1461 006050' 005412          NEG   (R2)          ;MAKE IT POSITIVE
1462 006052' 016367 000006 000056 10$:  MOV   PASCIN(R3),PROG# ;STORE PROG'S # IN MSG
1463 006060' 004577 171770  JSR   RS,ACLIST      ;ISSUE PROG #
1464 006064' 000050          .WORD PNTMSG-.
1465 006066' 000005          .WORD 5
1466 006070' 004577 171760  JSR   RS,ACLIST      ;ISSUE MSG SPECIFIED
1467 006074' 000000          .WORD XXXX
1468 006076' 000000          .WORD XXXX
1469 006100' 004577 171750  JSR   RS,ACLIST      ;ISSUE A <CR> & <LF>
1470 006104' 000294          .WORD CRLF-.
1471 006106' 000002          .WORD 2
1472 006110' 000410          BR    PRTEX          ;GO TO EXIT
1473 006112' 010067 000010 40$:  MOV   R0,50$        ;STORE MSG'S ABS ADR
1474 006116' 010167 000006  MOV   R1,60$        ;STORE ITS BYTE CNT
1475 006122' 004577 171724  JSR   RS,ACLIST      ;GO TO MPG TO ISSUE THE MSG
1476 006126' 000000          .WORD XXXX
1477 006130' 000000          .WORD XXXX
1478 006132' 000205  PRTEX: RTS          ;EXIT IN-LINE
    
```

```

1480 .SBTTL RPO2/RPO3 MESSAGE STORAGE AREA
1481
1482
1483 .NLIST BEX
1484
1485 .EVEN
1486 006134' 021520 PNMMSG: .ASCII /P# /
1487 006136' 054130 011 PRGNM: .ASCII /XX/<011>
1488 006141' 101 020124 040514 R1MSG: .ASCII /AT LAST INT:/
1489 006155' 103 051125 042522 CUPMSG: .ASCII /CURRENTLY:/
1490 006167' 105 042116 047440 RENDMG: .ASCII /END OF REPORT/
1491
1492 006204' 025052 025052 050122 UNITMG: .ASCII /###RPIX DISK UNIT: /
1493 006230' 054130 054130 054130 UNASCI: .ASCII /XXXXXX/
1494
1495 006236' 054130 054130 020075 DVRMG: .ASCII /XXXX= /
1496 006241' 054130 054130 054130 DVJDT1: .ASCII /XXXXXX XXXX= /
1497 006242' 054130 054130 054130 DVJDT2: .ASCII /XXXXXX XXXX= /
1498 006243' 054130 054130 054130 DVJDT3: .ASCII /XXXXXX/
1499 006244' 054130 054130 054130 CNTSMG: .ASCII /BYTES: RD= /
1500 006245' 054130 054130 054130 BCHKD: .ASCII /XXXXXXXXXXXXXXXXX WR= /
1501 006246' 054130 054130 054130 BCHKR: .ASCII /XXXXXXXXXXXXXXXXX/
1502 006247' 054130 054130 054130 CRLF: .ASCII (015)<012>
1503 006248' 054130 054130 054130 .ASCII (011)<011>/CK= /
1504 006249' 054130 054130 054130 BCHK: .ASCII /XXXXXXXXXXXXXXXXX/<015><012><011>/CMDS: RD= /
1505 006250' 054130 054130 054130 CXCRO: .ASCII /XXXXXX WR= /
1506 006251' 054130 054130 054130 CXCNR: .ASCII /XXXXXX CK= /
1507 006252' 054130 054130 054130 CXCCK: .ASCII /XXXXXX/<015><012><011><011>/SK= /
1508 006253' 054130 054130 054130 CXCCK: .ASCII /XXXXXX MISC= /
1509 006254' 054130 054130 054130 CXCCK: .ASCII /XXXXXX/<015><012><011>/ERRORS: DEV= /
1510 006255' 054130 054130 054130 CXCCK: .ASCII /XXXXXX DATA= /
1511 006256' 054130 054130 054130 CNTLDR: .ASCII /XXXXXX/<015><012><011>/RETRY: TIME= /
1512 006257' 054130 054130 054130 CNTTIE: .ASCII /XXXXXX CSME= /
1513 006258' 054130 054130 054130 CNTCSE: .ASCII /XXXXXX MPE= /
1514 006259' 054130 054130 054130 CNTWPE: .ASCII /XXXXXX/<015><012><011><011>/LPE= /
1515 006260' 054130 054130 054130 CNTLPE: .ASCII /XXXXXX MCE= /
1516 006261' 054130 054130 054130 CNTWCE: .ASCII /XXXXXX/<015><012><011>/TOTAL RETRY: /
1517 006262' 054130 054130 054130 CNTRTY: .ASCII /XXXXXX/<015><012><011>/INTERRUPTS: /
1518 006263' 054130 054130 054130 CNTINT: .ASCII /XXXXXX/
1519 006264' 054130 054130 054130 CNTSEN: .ASCII /
1520 006746' 054503 025514 040 INFOG: .ASCII /CYL= /
1521 006753' 054130 054130 054130 IFCYL: .ASCII /XXXXXX HEAD= /
1522 006770' 054130 054130 054130 IFHEAD: .ASCII /XXXXXX SECT= /
1523 007005' 054130 054130 054130 IFSECT: .ASCII /XXXXXX/
1524
1525 007014' 052123 047115 020124 STNMG: .ASCII /STMT # /
1526 007024' 054130 054130 054130 STNMJN: .ASCII /XXXXXX/
1527 007032' 051105 047522 020122 EFBMSG: .ASCII /ERROR BITS: /<015><012><011>
1528 007050' 000100 CODFLD: .BLKB 64.
1529 007150' 027524 020117 047117 CRT0: .ASCIZ 'T/O ON IDLE/RESET'
1530 007173' 124 046511 047505 IOTO: .ASCIZ 'TIMEOUT ON I/O'
1531 007212' 047125 040523 042506 INITUS: .ASCII 'UNSAFE ON INITIATION'
1532 007236' 027511 020117 042524 IOTERM: .ASCIZ 'I/O TERMINATION ERROR'
1533 007264' 052101 047124 047040 NOATA: .ASCIZ 'ATTN NOT SET'
1534
1535 007301' 105 044130 052501 RTYEXH: .ASCII 'EXHAUSTED RETRIES ON '

```

1536	007326'	054130	054130	000	RTYID:	.ASCIZ	'XXXX'
1537	007333'	125	044516	020124	UNOFFL:	.ASCIZ	'UNIT OFF-LINE'
1538	007351'	125	044516	020124	UNRDY:	.ASCIZ	'UNIT NOT RDY'
1539	007366'	047111	020126	047125	INVDVN:	.ASCIZ	'INV UNIT #'
1540		007402'				.EVEN	
1541						.LIST	BEX
1542							
1543		007402'			DVREND=	.	

```

1545          .SBTTL FORMATS FOR PROGRAM & DEVICE ROUTINE TABLES
1546
1547          ; PROGRAM TABLE FORMAT
1548
1549          000242      PTLGTH= 162.      ;PROGRAM TABLE LENGTH - NON MEM MGMT VERSION OF MPG
1550
1551          ;(PTLGTH= 212. ;PROGRAM TABLE LENGTH - MEM MGMT VERSION OF MPG)
1552
1553          000000      PFLGWD= +0.      ;PROGRAM FLAG WORD - 1 WORD
1554
1555          000002      URSTOP= 2        ; 1 = USER HAS STOPPED THIS PROGRAM
1556          000004      ERSTOP= 4        ; 1 = AN ERROR HAS STOPPED THIS PROGRAM
1557          000010      WTYIOT= 10       ; 1 = WAITING FOR I/O TERMINATION
1558          000020      CIPRIO= 20       ; 1 = CONSOLE OR PRINTER I/O IN PROGRESS
1559          000040      SETDED= 40       ; 1 = THIS PROG SET THE PRT DEV DEDICATED FLAG
1560          000100      OCPRES= 100      ; 1 = OBJ CODE IS PRESENT
1561          000200      USEUBM= 200     ; 1 = THIS PROG USES THE UNIBUS MAP (MEM MGMT ONLY)
1562          100000      ACTIVE= 100000  ; 1 = PROGRAM IS ACTIVE (SPECIFIED FOR EXECUTION)
1563
1564          000002      POPSW= +2.      ;PROGRAM'S OPERATION SWITCHES - 1 WORD
1565
1566          100000      STONER= 100000   ; 1 = STOP PROG EXECUTION UPON ERROR
1567          040000      CYCPRG= 4000    ; 1 = CYCLE PROGRAM (ON CURRENT DEVICE)
1568          020000      PRNER= 2000    ; 1 = DO NOT PRINT ON ERROR
1569          010000      BIT12= 1000    ; 0 = NOT USED
1570          004000      BIT11= 4000    ; 0 = NOT USED
1571          002000      CYCDVL= 2000   ; 1 = CYCLE THE DEVICE LIST
1572          001000      GTNXTD= 1000   ; 1 = CYCLE ON SAME DEVICE UPON ERROR
1573          000400      DOERCK= 400    ; 1 = DON'T DO ERROR CHECKING
1574          000200      SOPER= 200    ; 1 = DEVICE SPECIAL OPERATION
1575          000100      BIT6= 100      ; 0 = NOT USED
1576          000040      DOIOT= 40      ; 1 = DO NOT PERFORM I/O TIMEOUT
1577          000020      AUTORP= 20     ; 1 = DO NOT AUTOMATICALLY DISPLAY COUNTS
1578          000010      AURPEP= 10    ; 1 = AUTO DISPLAY COUNTS AT END OF FINAL PASS ONLY
1579          000004      HSKPEP= 4     ; 1 = HOUSEKEEP COUNTS ONLY AT RUN COMMAND
1580          000002      PFBBOV= 2     ; 1 = PRINT FIRST BAD BYTE ONLY ON VERIFY
1581          000001      NOCOMP= 1     ; 1 = DO NOT PRINT PROG COMPLETED MSG
1582
1583          000004      PFWADR= +4.      ;*;PROGRAM FLAGWORD ADDRESS - 1 WORD
1584
1585          000006      PASCIN= +6.      ;PROGRAM'S NUMBER IN ASCII - 1 WORD
1586
1587          000010      PNAME= +8.      ;PROGRAM'S NAME IN ASCII - 6 BYTES
1588
1589          000016      PRDIOA= +14.     ;ADDRESS OF READ I/O AREA - 1 WORD
1590
1591          000020      PWRIOA= +16.     ;ADDRESS OF WRITE I/O AREA - 1 WORD
1592
1593          000022      PSRCST= +18.     ;SOURCE STATEMENTS START ADDRESS - 1 WORD
1594
1595          000024      POBJST= +20.     ;OBJECT CODE START ADDRESS - 1 WORD
1596
1597          000026      PLNGTH= +22.     ;PROG AREA LENGTH (OBJ END MINUS PROG TBL START) - 1 WORD
1598
1599          000030      PTOCNT= +24.     ;I/O TIMEOUT COUNT - 1 WORD
1600
    
```

1601	000032	PMOLCD= +26.	;DEV ROUT MODEL # CODE - 1 WORD
1602			
1603	000034	PDPNTR= +28.	;CURRENT DEVICE NUMBER POINTER - 1 BYTE
1604			
1605	000035	PCURDV= +29.	;CURRENT DEVICE # - 1 BYTE
1606			
1607	000036	PDNUMS= +30.	;DEVICE NUMBERS - 16 BYTES
1608			
1609	000056	PTE. 7= +46.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1610			
1611	000060	PTEM1= +48.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1612			
1613	000062	PTEM2= +50.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1614			
1615	000064	PTEM3= +52.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1616			
1617	000066	PTEM4= +54.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1618			
1619	000070	PTEM5= +56.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1620			
1621	000072	PTEM6= +58.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1622			
1623	000074	PTEM7= +60.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1624			
1625	000076	PTEM8= +62.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1626			
1627	000100	PTEM9= +64.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1628			
1629	000102	PTEM10= +66.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1630			
1631	000104	PTEM11= +68.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1632			
1633	000106	PTEM12= +70.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1634			
1635	000110	PTEM13= +72.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1636			
1637	000112	PTEM14= +74.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1638			
1639	000114	PTEM15= +76.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
1640			
1641	000116	PNBR= +78.	;NUMBER OF BYTES TO TRANSFER ON MOVE (NBR) - 1 WORD
1642			
1643	000120	PSRC= +80.	;DATA SOURCE ADDRESS ON MOVE (SRC) - 1 WORD
1644			
1645	000122	PDST= +82.	;DATA DESTINATION ADDRESS ON MOVE (DST) - 1 WORD
1646			
1647	000124	PSTKCT= +84.	;# OF WORDS (X 2) SAVED OFF STACK - 1 WORD
1648			
1649	000126	PSTKSV= +86.	;STACK WORDS STORAGE AREA - 30 WORDS
1650			
1651	000222	PSVREG= +146.	;USER'S R0 THRU R5 REGISTERS STORAGE AREA - 6 WORDS
1652			
1653	000236	PUSRPC= +158.	;USER'S CURRENT PROGRAM COUNTER - 1 WORD
1654			

L03

1656
 1657
 1658
 1659
 1660
 1661
 1662
 1663
 1664
 1665
 1666
 1667
 1668
 1669
 1670
 1671
 1672
 1673
 1674
 1675
 1676
 1677
 1678
 1679
 1680

;FOLLOWING ENTRIES (PROIOX THRU PUBMAP) ARE ONLY IN MEM MGMT VERSION

;(PROIOX= +160. ;18/22 BIT ABSOLUTE ADDRESS OF READ I/O AREA - 2 WORDS)
 ;(PROIOV= +164. ;18 BIT VIRTUAL ADDRESS OF READ I/O AREA - 2 WORDS)
 ;(PWRIOX= +168. ;18/22 BIT ABSOLUTE ADDRESS OF WRITE I/O AREA - 2 WORDS)
 ;(PWRIOV= +172. ;18 BIT VIRTUAL ADDRESS OF WRITE I/O AREA - 2 WORDS)
 ;(PUPARS= +176. ;STORAGE AREA FOR USER'S PAR'S 0 THRU 7 - 8 WORDS)
 ;(PUPDRS= +192. ;STORAGE AREA FOR USER'S PDR'S 0 THRU 7 - 8 WORDS)
 ;(PUBMAP= +208. ;1ST UNIBUS MAP REG # AND # OF RIGS USED - 1 WORD)

;END OF MEM MGMT ONLY ENTRIES

000240

PTSIZE= +160. ;PROGRAM TABLE SIZE IN BYTES - 1 WORD - NON MEM MGMT
 ;(PTSIZE= +210. ;PROGRAM TABLE SIZE IN BYTES - 1 WORD - MEM MGMT VERSION)

000242

PTEND= +162. ;END OF PROGRAM TABLE - NON MEM MGMT VERSION
 ;(PTEND= +212. ;END OF PROGRAM TABLE - MEM MGMT VERSION)

```

1682           ;      DEVICE ROUTINE TABLE
1683
1684
1685           000116      DRTLTH= 78.      ;DEVICE ROUTINE TABLE LENGTH
1686           ;
1687           ;
1688           000000      DEVRSZ= +0.      ;DEVICE ROUTINE SIZE IN BYTES - 1 WORD
1689
1690           000002      DEVFWD= +2.      ;DEVICE ROUTINE FLAGWORD - 1 WORD
1691
1692           000004      DEVIW1= +4.      ;DEVICE INTERFACE WORD # 1 - 1 WORD
1693
1694           000006      DEVIW2= +6.      ;DEVICE INTERFACE WORD # 2 - 1 WORD
1695
1696           000010      DEVIW3= +8.      ;DEVICE INTERFACE WORD # 3 - 1 WORD
1697
1698           000012      DEVIW4= +10.     ;DEVICE INTERFACE WORD # 4 - 1 WORD
1699
1700           000014      DEVIW5= +12.     ;DEVICE INTERFACE WORD # 5 - 1 WORD
1701
1702           000016      DEVIW6= +14.     ;DEVICE INTERFACE WORD # 6 - 1 WORD
1703
1704           000020      DEVIW7= +16.     ;DEVICE INTERFACE WORD # 7 - 1 WORD (SIZE)
1705
1706           000022      DEVIW8= +18.     ;DEVICE INTERFACE WORD # 8 - 1 WORD (ERR)
1707
1708           000024      DEVORA= +20.     ;DEVICE REGISTERS ADDRESS - 1 WORD
1709
1710           000026      DEVIVA= +22.     ;DEVICE INTERRUPT VECTOR ADDRESS - 1 WORD
1711
1712           000030      DEVRPS= +24.     ;DEVICE READ PROCESSOR STATUS WORD (BUS REQ) - 1 WORD
1713
1714           000032      DEVWPS= +26.     ;DEVICE WRITE PROC STATUS WORD (BUS REQ) - 1 WORD
1715
1716           000034      DHKPAD= +28.     ;DEVICE ROUT HOUSEKEEPING ROUT REL ENTRY ADR - 1 WORD
1717
1718           000036      DERPAD= +30.     ;DEVICE ROUT REPORT ROUT REL ENTRY ADR - 1 WORD
1719
1720           000040      DKILAD= +32.     ;DEVICE ROUT KILL ROUTINE REL ENTRY ADR - 1 WORD
1721
1722           000042      DECTAD= +34.     ;DEVICE ROUT ERROR COUNTER REL ADR - 1 WORD
1723
1724           000044      DTOEAD= +36.     ;DEVICE ROUT TIMEOUT ERR ROUT REL ENTRY ADR - 1 WORD
1725
1726           000046      DEVI08= +38.     ;DEVICE I/O BUSY BRANCH ADDRESS (CIOBSY) - 1 WORD
1727
1728           000050      DEVDER= +40.     ;DEVICE ERROR BRANCH ADDRESS (CUPGER) - 1 WORD
1729
1730           000052      DVUPRT= +42.     ;USER MODE PRINT BRANCH ADDRESS (ULIST) - 1 WORD
1731
1732           000054      DVCPRN= +44.     ;CMND MODE PRINT BRANCH ADDRESS (CLIST) - 1 WORD
1733
1734           000056      DEVBTA= +46.     ;CONVERT BINARY TO ASCII BR ADR (BINASC) - 1 WORD
1735
1736           000060      DVBTDA= +48.     ;CONVERT BINARY TO DECIMAL ASCII BR ADR (BTASLZ) - 1 WORD
1737

```

NO3

MAINDEC-11-DTR3A-A RPO2/RPO3 DEVICE ROUTINE FOR MPG MACY11 27(732) 24-SEP-76 14:11 PAGE 12-4
 DTR3AA.P11 FORMATS FOR PROGRAM & DEVICE ROUTINE TABLES

SEQ 0110

1738	000062	DVPOTA= +50.	; CONVERT PACKED DECIMAL TO ASCII BR ADR (DECASC) - 1 WORD
1739			
1740	000064	DVSFWD= +52.	; MPG SYSTEM FLAGWORD ADDRESS (CSYSFW) - 1 WORD
1741			
1742	000066	DVSVEC= +54.	; SET INTERRUPT VECTOR BR ADR (SETVEC) - 1 WORD
1743			
1744	000070	DVCVEC= +56.	; CLEAR INTERRUPT VECTOR BR ADR (CLRVEC) - 1 WORD
1745			
1746	000072	DVTVEC= +58.	; TEST INTERRUPT VECTOR BR ADR (TSTVEC) - 1 WORD
1747			
1748	000074	DVRINT= +60.	; RETURN FROM INTERRUPT BR ADR (RTNINT) - 1 WORD
1749			
1750	000076	DVGETB= +62.	; GET DATA BYTE BR ADR (GETBYT) - 1 WORD
1751			
1752	000100	DVPUTB= +64.	; PUT DATA BYTE BR ADR (PUTBYT) - 1 WORD
1753			
1754	000102	DEVSTP= +66.	; DEVICE ROUT REL SYMBOL TABLE POINTER - 1 WORD
1755			
1756	000104	DEVETP= +68.	; DEVICE ROUT REL ENTRY TABLE POINTER - 1 WORD
1757			
1758	000106	DVPTEP= +70.	; PACK TABLE EXTEN. REL POINTER - 1 WORD
1759			
1760	000110	DVVTEP= +72.	; VECTOR TABLE EXTEN. REL POINTER - 1 WORD
1761			
1762	000112	DVCTEP= +74.	; COMPILER TBL EXTEN. REL POINTER - 1 WORD
1763			
1764	000114	DVIWSP= +76.	; DEVICE INTERFACE WORD SYMBOL TBL REL POINTER - 1 WORD
1765			
1766	000116	DRTEND= +78.	; END OF DEVICE ROUTINE TABLE
1767			
1768			
1769	000001	.END	

SYMBOL TABLE

ACTIVE=	100000		CURADR	001252R	002	DVCVEC=	000070		IDLE	002402R	002	PRO10A=	000016	
ATATBL	003216R	002	CURCHD	001250R	002	DVGETB=	000076		IFCYL	006753R	002	PRINT	006020R	002
ATI7SG	006141R	002	CURCNT	001255R	002	DVILSP=	000114		IFHEAD	006770R	002	PROCEX	004312R	002
ATNINT	003476R	002	CURFLG	001246R	002	DVILST	001106R	002	IFSECT	007005R	002	PROCTH	004176R	002
AURPEP=	000010		CURMSC	006155R	002	DVMVTE	000570R	002	INFOMC	006746R	002	PROGPH	006136R	002
AUTORP=	000020		CURRTY	001252R	002	DVPDTR	000062		INITUS	007212R	002	PRONER=	020000	
BCMCK	006370R	002	CYCOVL=	002000		DVPKTE	000350R	002	INTCNT	001334R	002	PRTEX	006132R	002
BCMFD	006322R	002	CYCPRG=	040000		DVPTEP=	000106		INTEAD	004272R	002	PRTIND	005750R	002
BCMNR	006344R	002	CYL	000004R	002	DVPUTB=	000100		INTEX	003466R	002	PSRC	000120	
BINASC	000056R	002	DATAER	001316R	002	DVROT1	006244R	002	INVDVN	007366R	002	PSRCST=	000022	
BIT11 =	004000		DECASC	000062R	002	DVROT2	006262R	002	IOERR =	000001		PSTKCT=	000124	
BIT12 =	010000		DECTAD=	000042		DVROT3	006300R	002	IOTERM	007236R	002	PSTKSV=	000126	
BIT6 =	000100		DEPAD=	000036		DVREG1=	000226R	002	IOIO	007173R	002	PSVREG=	000222	
BITSLZ	000060R	002	DEVBA=	007056		DVREGS	000116R	002	ISTAT =	001166R	002	PSWD	000030R	002
BYCK	001276R	002	DEVDER=	000050		DVREND=	007402R	002	IVCTAD	000026R	002	PTEAD	000056	
BYRD	001266R	002	DEVDR=	000024		DVREX	001714R	002	KILL	002054R	002	PTEH1	000060	
BYMR	001272R	002	DEVTP=	000104		DVRGMC	006236R	002	KILLEX	002110R	002	PTEH10	000102	
CI0BSY	000046R	002	DEVFMD=	000002		DVRINT=	000074	002	LOCZ	000000R	002	PTEH11	000104	
CKCNT	001306R	002	DEVID	005450R	002	DVSFLD=	000064		LPECNT	001326R	002	PTEH12	000106	
CK0BSY	004076R	002	DEVIIV	005532R	002	DVSVEC=	000066		MISCNT	001312R	002	PTEH13	000110	
CKRTY	003636R	002	DEVIML	005560R	002	DVTEC=	000072		MVER =	000001		PTEH14	000112	
CLIST	000054R	002	DEVIOB=	000046		DVUPRT=	000052		MODE =	010000		PTEH15	000114	
CLVEC	000070R	002	DEVIPR	005552R	002	DVUTEP=	000110		MODE10	002372R	002	PTEH2	000062	
CLPATF	003462R	002	DEVIVR	000026		EBSBAS	004564R	002	MODE11	002362R	002	PTEH3	000064	
CHDOCK	006451R	002	DEVIMI=	000004		EBSTAT	004566R	002	MSFHT1	001140R	002	PTEH4	000066	
CHDOHS	006504R	002	DEVIM2=	000006		ERMSG	007032R	002	MSFHT2	001141R	002	PTEH5	000070	
CHDCOM	002646R	002	DEVIM3=	000010		ERCOTB	005036R	002	MSFHT3	001146R	002	PTEH6	000072	
CHDCRD	006423R	002	DEVIM4=	000012		ERMBAS	004532R	002	MSFHT4	001155R	002	PTEH7	000074	
CHDCSK	006467R	002	DEVIM5=	000014		ERR	100000		MSFHT5	001164R	002	PTEH8	000076	
CHDCUR	006436R	002	DEVIM6=	000016		ERRCNT	001314R	002	MYATA	003236R	002	PTEH9	000100	
CNTADR	001244R	002	DEVIM7=	000020		ERRCOM	00440R	002	NOATA	007264R	002	PTEAD	000242	
CNTCSE	006514R	002	DEVIM8=	000022		ERRCS	00440R	002	NOCMP=	000001		PTLGTH=	000242	
CNTDER	006547R	002	DEVIP1=	000030		ERRCS1	004406R	002	NOWAIT	002332R	002	PTOCHT=	000030	
CNTERR	006532R	002	DEVRS2=	000000		ERRCX	005034R	002	OBJADR	001242R	002	PTSIZE=	000240	
CNTINT	006740R	002	DEVSTP=	000102		ERRFND	003564R	002	OCPRES=	000100		PUSAPC=	000236	
CNTLPE	006647R	002	DEVVPS=	000032		ERRI	000022R	002	OTHATA	003240R	002	PUTBYT	000100R	002
CNTNUM=	000024		DPLGMD	000002R	002	ERRIS	004424R	002	PASCIN=	000006		PWR10A=	000020	
CNTRTY	006712R	002	DNDPAD=	000034	002	ERRSM	004754R	002	PATCH	001340R	002	ROCNT	001302R	002
CNTSEN=	006746R	002	DISCNT	001614R	002	ERSTAD	004746R	002	PC =	000007		ROCOM	002472R	002
CNTSMC	006306R	002	DISPST	005564R	002	ERSTOP	000004		PCURDV=	000035		RONOSK	002542R	002
CNTTIE	006577R	002	DKILAD	000040		FINCNT	001260R	002	PONUMS=	000036		READ	002462R	002
CNTWCE	006663R	002	DOERCK=	000400		GETBYT	000076R	002	PPNTR=	000034		RECAL	002640R	002
CNTWPE	006630R	002	DOTOT =	000040		GTNKT=	001000		POST =	000122		REGNUM=	000013	
CODFLD	007050R	002	DOTERM=	000002		HDROFF	002352R	002	PFB90V=	000002		RENOMG	006167R	002
COUNTS	001266R	002	DREGAD	000024R	002	HDRON	002342R	002	PFLGMD=	000000		REPORT	001436R	002
CRESET	002402R	002	DRTEND=	000116		HE	040000		PFWAOR=	000004		REPTBL	001724R	002
CRLF	006360R	002	DRTLTH=	000116		HEAD	000006R	002	PLNGTH=	000026		RESREG	005362R	002
CRTC	007150R	002	DTOEAD=	000044		HEADER=	004000		PHDLCD=	000032		RETRY5	001332R	002
CSECNT	001322R	002	DVBTDA=	000060		HOMESK	002640R	002	PNAME =	000010		RINTEX	004346R	002
CSTAT	001214R	002	DVCHDS	000226R	002	HSKEEP	001364R	002	PNER =	000116		RINTV	004324R	002
CSYSFW	000064R	002	DVCPR1=	000054		HSKPEP=	001336R	002	PNUMSG	006134R	002	RPBA =	000004	
CTPRIO=	000020		DVCPT2=	000700R	002	HSKPEP=	000004		POB1ST=	000024		RPCA =	000006	
CUPGER	000050R	002	DVCTEP=	000112		HSPST=	001166R	002	POPSH =	000002		RPCS =	000000	

DTR3AA.P11 SYMBOL TABLE

RPCSV	001336R	002	RTYEXH	007301R	002	SIZE	000020R	002	SUCA	= 000020		WAIT	002312R	002
RPOA	= 000010		RTYID	007326R	002	SKCNT	001310R	002	SUIORG	003242R	002	WAITMD	= 100000	
RPOS	= 177774		R0	=X000000		SKCOM	002622R	002	SUPTAD	005400R	002	WCECNT	001330R	002
RPER	= 177776		R1	=X000001		SP	=X000006		TIECNT	001320R	002	WPECNT	001324R	002
RPINT	003354R	002	R2	=X000002		SPOPER	= 000200		TOUTER	001774R	002	WRCK	002586R	002
RFM1	= 000012		R3	=X000003		STEPON	002244R	002	TSTVEC	000072R	002	WRCNT	001304R	002
RFM2	= 000014		R4	=X000004		STEPEX	002160R	002	TVECT	004350R	002	WRCOM	002522R	002
RFM3	= 000016		RS	=X000005		STEPUP	002112R	002	TVECTX	004376R	002	WRITE	002512R	002
RPTBAS	001660R	002	SAVREG	005346R	002	STMAOR	004162R	002	ULIST	000052R	002	WRNOSK	002554R	002
RPTEND	001704R	002	SECT	000010R	002	STNMG	007014R	002	UNASCI	006230R	002	WT410T	= 000010	
RPTLP	001642R	002	SEEK	002616R	002	STNUM	007024R	002	UNITMG	006204R	002	WYATTN	= 000004	
RPMC	= 000002		SETDED	= 000040		STONER	= 100000		UNRDY	007351R	002	XXXX	= 000000	
RTNINT	000074R	002	SETERR	003454R	002	STPCOM	002176R	002	UNOFFL	007333R	002	.	= 007402R	002
. ABS.	000000	000												
	000000	001												
RP11	007402	002												

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

*, DTR3AA/NL: TOC/DOC=DTR3AA.P11
RUN-TIME: 5 11 1 SECONDS
RUN-TIME RATIO: 21/17=1.2
CORE USED: 5K (9 PAGES)

DOCUMENT PAGES: 41

