

# RP04/05/06

DEVICE ROUTINE (MPG)  
MD-11-DTRPA-B

EP-DTRPA-B-DL-A NOV 1976  
COPYRIGHT © 1976  
FICHE 1 OF 1 MADE IN U.S.

This microfiche card contains a grid of frames. The frames are arranged in approximately 15 rows and 5 columns. Each frame contains a small, high-contrast image of a document page, likely a technical drawing or data table. The images are very small and difficult to read, but they appear to be organized in a structured manner. The card is otherwise blank.



45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56

.SBTTL REVISION HISTORY

- APR 76 DTRPA-B RELEASE
- FEB 76 CHANGED THE DEVICE TYPE OF THIS DEVICE ROUTINE  
JAN 76 TO THE RPO4/RPO5/RPO6 DISKS ON THE RH11/RH70 CONTROLLERS.  
ADDED FULL SUPPORT FOR I/O COMMANDS AND OTHER  
FUNCTIONS.
- AUG 75 DTRPA-A INITIAL RELEASE AS A MINIMUM SUPPORT  
DEVICE ROUTINE FOR THE RP11/RPO3 DISK.

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  
105  
106  
107  
108  
109  
110  
111  
112  
113

.SBTTL STANDARD DEVICE ROUTINE TABLE  
.TITLE MAINDEC-11-DTRPA-B RH11/RH70 - RPO4/RPO5/RPO6 DEVICE ROUTINE FOR MPG  
;REVISION 'B'  
;FILENAME OF "TRPA80.MPG" ON MPG/XXDP MEDIA  
;MACY11: DTRPA?,DTRPA?/CRF:SYM/DOC=DTRPA?.P11  
;LNKX11: DTRPA?.MPG/B:0+DTRPA?/E  
;PAPER TAPE: PUNCH DTRPA?.MPG/FILE:ELEV  
  
000000'  
  
000000' 014002  
000002' 000000  
100000  
002000  
001000  
  
000200  
000100  
000040  
000020  
000010  
000004  
000002  
000001  
000004' 000000  
000006' 000000  
000010' 000000  
000012' 000003  
000014' 000000  
000016' 000000  
000020' 000000  
000022' 000000  
000024' 176700  
000026' 000254  
000030' 000240  
000032' 000000  
000034' 002414  
000036' 002472  
000040' 003330  
000042' 002254  
000044' 003030  
000046' 000000  
000050' 000000  
000052' 003000  
000054' 000000  
000056' 000000

.CSECT RJP11  
.DSABL GBL  
  
;THE FOLLOWING TABLE IS IN THE STANDARDIZED FORMAT REQUIRED  
;TO INTERFACE WITH MPG.  
  
LOCZ: .WORD DVREND-.  
DFLGND: .WORD 0  
WAITMD= 100000  
CORFLG= 2000  
VVFLG= 1000  
  
ANYIOI= 200  
CMDISU= 100  
SMOVT0= 40  
SMOIER= 20  
SMOVER= 10  
UNLDIP= 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 176700  
IVCTAD: .WORD 254  
PSWD: .WORD 240  
  
CIOSBY: .WORD 0  
CUPGER: .WORD 0  
ULIST: .WORD 0  
CLIST: .WORD 0  
BINASC: .WORD 0

:DEVICE ROUT SIZE IN BYTES  
:DEVICE ROUT FLAGWORD  
:WAIT MC - 0 = WAIT  
:CORRECT MODE - 0 = CORON  
:VV MODE - 0 = VVON  
  
:ANY I/O HAS BEEN ISSUED  
:I/O COMMAND HAS BEEN ISSUED  
:TIMEOUT ON SWITCH OVER  
:ERROR ON INT FOR SWITCH OVER  
:ACQUIRING DISK - SWITCH OVER  
:UNLOAD CMD IN PROGRESS  
:PROCESS I/O TERMINATION  
:ERROR ON CURRENT I/O  
:CYLINDER # (0 THRU 410./814.)  
:HEAD # (0 THRU 18.)  
:SECTOR # (0 THRU 21./19.)  
:# 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  
:CMDN MODE PRINT ROUTINE BRANCH ADR  
:CONVERT BINARY TO ASCII ROUT BR ADR

# E01

114	000060'	000000	BTASLZ:	.WORD	0	: CONVERT BINARY TO DECIMAL ASCII BR ADR
115	000062'	000000	DECASC:	.WORD	00	: CONVERT PACKED DECIMAL TO ASCII BR ADR
116	000064'	000000	CSYSFW:	.WORD	00	: MPG SYSTEM FLAGWORD ADR
117	000066'	000000	SETVEC:	.WORD	00	: SET INT VECT ROUT BR ADR
118	000070'	000000	CLRVEC:	.WORD	00	: CLEAR INT VECTOR ROUT BR ADR
119	000072'	000000	TSTVEC:	.WORD	00	: TEST INT VECTOR ROUT BR ADR
120	000074'	000000	RTNINT:	.WORD	00	: RETURN FROM INT ROUT BR ADR
121	000076'	000000	GETBYT:	.WORD	00	: GET DATA BYTE ROUT BR ADR
122	000100'	000000	PUTBYT:	.WORD	00	: PUT DATA BYTE ROUT BR ADR
123	000102'	000014		.WORD	DVREGS-	: ADR OF DEVICE REGISTER NAMES
124	000104'	000216		.WORD	DVCMOS-	: ADR OF DEVICE FUNCTIONS
125	000106'	000452		.WORD	DVPKTE-	: ADR OF PACK TBL EXTENSION
126	000110'	001120		.WORD	DVMVTE-	: ADR OF MODEL VECTOR TBL EXTEN.
127	000112'	001342		.WORD	DVCPTI-	: ADR OF COMPILER TBL EXTEN.
128	000114'	001740		.WORD	DVIWST-	: ADR OF DEV INTERFACE WD SYM TBL

				.SBTTL COMPILER TABLES & CONSTANT AREAS			
130							
131							
132							
133	000116'	050122	030503	DVREGS:	.ASCII	/RPC1/	:VALID DEVICE REGISTER NAMES &
134	000122'	000000			.WORD	0	:THEIR POSITIONS RELATIVE TO
135	000124'	050122	041527		.ASCII	/RPWC/	:THE DEVICE REGISTERS BASE ADDRESS.
136	000130'	000002			.WORD	2	
137	000132'	050122	040502		.ASCII	/RPBA/	
138	000136'	000004			.WORD	4	
139	000140'	050122	040504		.ASCII	/RPDA/	
140	000144'	000006			.WORD	6	
141	000146'	050122	031103		.ASCII	/RPC2/	
142	000152'	000010			.WORD	10	
143	000154'	050122	051504		.ASCII	/RPOS/	
144	000160'	000012			.WORD	12	
145	000162'	050122	030505		.ASCII	/RPE1/	
146	000166'	000014			.WORD	14	
147	000170'	050122	051501		.ASCII	/RPAS/	
148	000174'	000016			.WORD	16	
149	000176'	050122	040514		.ASCII	/RPLA/	
150	000202'	000020			.WORD	20	
151	000204'	050122	041104		.ASCII	/RPDB/	
152	000210'	000022			.WORD	22	
153	000212'	050122	051115		.ASCII	/RPMR/	
154	000216'	000024			.WORD	24	
155	000220'	050122	052104		.ASCII	/RPDT/	
156	000224'	000026			.WORD	26	
157	000226'	050122	047123		.ASCII	/RPSN/	
158	000232'	000030			.WORD	30	
159	000234'	050122	043117		.ASCII	/RPOF/	
160	000240'	000032			.WORD	32	
161	000242'	050122	041504		.ASCII	/RPDC/	
162	000246'	000034			.WORD	34	
163	000250'	050122	041503		.ASCII	/RPCC/	
164	000254'	000036			.WORD	36	
165	000256'	050122	031105		.ASCII	/RPE2/	
166	000262'	000040			.WORD	40	
167	000264'	050122	031505		.ASCII	/RPE3/	
168	000270'	000042			.WORD	42	
169	000272'	050122	047520		.ASCII	/RPP0/	
170	000276'	000044			.WORD	44	
171	000300'	050122	040520		.ASCII	/RPPA/	
172	000304'	000046			.WORD	46	
173	000306'	050122	042501		.ASCII	/RPAE/	
174	000312'	000050			.WORD	50	
175	000314'	050122	031503		.ASCII	/RPC3/	
176	000320'	000052			.WORD	52	
177		000322'		DVREGS=	.		
178							
179							
180	000322'	120	211	DVCMDS:	.BYTE	120,211	:VALID DEVICE FUNCTIONS
181	000324'	004636			.WORD	READ-	:FLAG BYTE:
182	000326'	120	211		.BYTE	130,211	:BIT 7 = NPR DEV
183	000330'	004662			.WORD	WRITE-	:BIT 3 = MASSBUS DEV
184	000332'	376	000		.BYTE	376,0	:BIT 0 = 2 WORDS FOR ADR
185	000334'	003322			.WORD	NOWAIT-	:(18 BIT ADRS)

186	000336'	375	000	.BYTE	375,0
187	000340'	003276		.WORD	WAIT-
188	000342'	374	000	.BYTE	374,0
189	000344'	002164		.WORD	REPORT-
190	000346'	373	000	.BYTE	373,0
191	000350'	002160		.WORD	REPORT-
192	000352'	372	000	.BYTE	372,0
193	000354'	004750		.WORD	SEEK-
194	000356'	371	211	.BYTE	371,211
195	000360'	004736		.WORD	WRCKHD-
196	000362'	370	211	.BYTE	370,211
197	000364'	004702		.WORD	WRCK-
198	000366'	367	211	.BYTE	367,211
199	000370'	004652		.WORD	RDHD-
200	000372'	366	211	.BYTE	366,211
201	000374'	004660		.WORD	WRHD-
202	000376'	365	000	.BYTE	365,0
203	000400'	003466		.WORD	CRESET-
204	000402'	364	000	.BYTE	364,0
205	000404'	003532		.WORD	DRESET-
206	000406'	363	000	.BYTE	363,0
207	000410'	004736		.WORD	SEARCH-
208	000412'	362	000	.BYTE	362,0
209	000414'	003006		.WORD	STEPUP-
210	000416'	361	000	.BYTE	361,0
211	000420'	003150		.WORD	STEPDN-
212	000422'	360	000	.BYTE	360,0
213	000424'	004734		.WORD	OFFSET-
214	000426'	357	000	.BYTE	357,0
215	000430'	004742		.WORD	RETCTR-
216	000432'	356	000	.BYTE	356,0
217	000434'	003510		.WORD	UNLOAD-
218	000436'	355	000	.BYTE	355,0
219	000440'	004744		.WORD	RECAL-
220	000442'	354	000	.BYTE	354,0
221	000444'	003506		.WORD	PACKACK-
222	000446'	353	000	.BYTE	353,0
223	000450'	003510		.WORD	ROPSET-
224	000452'	352	000	.BYTE	352,0
225	000454'	003512		.WORD	REL-
226	000456'	351	000	.BYTE	351,0
227	000460'	003206		.WORD	APORT-
228	000462'	350	000	.BYTE	350,0
229	000464'	003212		.WORD	BPORT-
230	000466'	347	000	.BYTE	347,0
231	000470'	003356		.WORD	FMT22-
232	000472'	346	000	.BYTE	346,0
233	000474'	003362		.WORD	FMT20-
234	000476'	345	000	.BYTE	345,0
235	000500'	003206		.WORD	000-
236	000502'	344	000	.BYTE	344,0
237	000504'	003212		.WORD	EVEN-
238	000506'	343	000	.BYTE	343,0
239	000510'	003216		.WORD	ECION-
240	000512'	342	000	.BYTE	342,0
241	000514'	003222		.WORD	ECIOFF-

242	000516'	341	000			.BYTE	341,0	
243	000520'	003226				.WORD	HCION-	
244	000522'	340	000			.BYTE	340,0	
245	000524'	003232				.WORD	HCIOFF-	
246	000525'	337	000			.BYTE	337,0	
247	000530'	003236				.WORD	BAION-	
248	000532'	336	000			.BYTE	336,0	
249	000534'	003242				.WORD	BAIOFF-	
250	000536'	335	000			.BYTE	335,0	
251	000540'	003246				.WORD	CORON-	
252	000542'	334	000			.BYTE	334,0	
253	000544'	003252				.WORD	COROFF-	
254	000546'	333	000			.BYTE	333,0	
255	000550'	003256				.WORD	VVON-	
256	000552'	332	000			.BYTE	332,0	
257	000554'	003262				.WORD	VVOFF-	
258	000556'	177777				.WORD	177777	
259								;TABLE TERMINATOR
260	000560'	047516	040527	052111	DVPKTE:	.ASCII	/NOWAIT/	
261	000566'	376	000			.BYTE	376,0	;PACK TABLE EXTENSION
262	000570'	020040	040527	052111		.ASCII	/WAIT/	
263	000576'	375	000			.BYTE	375,0	
264	000600'	052123	052101	051525		.ASCII	/STATUS/	
265	000606'	374	000			.BYTE	374,0	
266	000610'	047503	047125	051524		.ASCII	/COUNTS/	
267	000616'	373	000			.BYTE	373,0	
268	000620'	020040	042523	045505		.ASCII	/SEEK/	
269	000626'	372	000			.BYTE	372,0	
270	000630'	051127	045503	042110		.ASCII	/WRCKHD/	
271	000636'	371	000			.BYTE	371,0	
272	000640'	020040	051127	045503		.ASCII	/WRCK/	
273	000646'	370	000			.BYTE	370,0	
274	000650'	020040	042122	042110		.ASCII	/RDHD/	
275	000656'	367	000			.BYTE	367,0	
276	000660'	020040	051127	042110		.ASCII	/WRHD/	
277	000666'	366	000			.BYTE	366,0	
278	000670'	051103	051505	052105		.ASCII	/CRSET/	
279	000676'	365	000			.BYTE	365,0	
280	000700'	051104	051505	052105		.ASCII	/DRSET/	
281	000706'	364	000			.BYTE	364,0	
282	000710'	042523	051101	044103		.ASCII	/SEARCH/	
283	000716'	363	000			.BYTE	363,0	
284	000720'	052123	050105	050125		.ASCII	/STEPUP/	
285	000726'	362	000			.BYTE	362,0	
286	000730'	052123	050105	047104		.ASCII	/STEPDN/	
287	000736'	361	000			.BYTE	361,0	
288	000740'	043117	051506	052105		.ASCII	/OFFSET/	
289	000746'	360	000			.BYTE	360,0	
290	000750'	042522	041524	051124		.ASCII	/RECTR/	
291	000756'	357	000			.BYTE	357,0	
292	000760'	047125	047514	042101		.ASCII	/UNLOAD/	
293	000766'	356	000			.BYTE	356,0	
294	000770'	051040	041505	046101		.ASCII	/RECAL/	
295	000776'	355	000			.BYTE	355,0	
296	001000'	040520	040513	045503		.ASCII	/PAKACK/	
297	001006'	354	000			.BYTE	354,0	

298	001010'	042122	051520	052105	.ASCII	/RDPSET/
299	001016'	353	000		.BYTE	353,0
300	001020'	020040	051040	046105	.ASCII	/REL/
301	001026'	352	000		.BYTE	352,0
302	001030'	040440	047520	052122	.ASCII	/APORT/
303	001036'	351	000		.BYTE	351,0
304	001040'	041040	047520	052122	.ASCII	/BPORT/
305	001046'	350	000		.BYTE	350,0
306	001050'	043040	052115	031062	.ASCII	/FMT22/
307	001056'	347	000		.BYTE	347,0
308	001060'	043040	052115	030062	.ASCII	/FMT20/
309	001066'	346	000		.BYTE	346,0
310	001070'	020040	047440	042104	.ASCII	/000/
311	001076'	345	000		.BYTE	345,0
312	001100'	020040	053105	047105	.ASCII	/EVEN/
313	001106'	344	000		.BYTE	344,0
314	001110'	042440	044503	047117	.ASCII	/ECION/
315	001116'	343	000		.BYTE	343,0
316	001120'	041505	047511	043106	.ASCII	/ECIOFF/
317	001126'	342	000		.BYTE	342,0
318	001130'	044040	044503	047117	.ASCII	/HCION/
319	001136'	341	000		.BYTE	341,0
320	001140'	041510	047511	043106	.ASCII	/HCIOFF/
321	001146'	340	000		.BYTE	340,0
322	001150'	041040	044501	047117	.ASCII	/BAION/
323	001156'	337	000		.BYTE	337,0
324	001160'	040502	047511	043106	.ASCII	/BAIOFF/
325	001166'	336	000		.BYTE	336,0
326	001170'	041440	051117	047117	.ASCII	/CORON/
327	001176'	335	000		.BYTE	335,0
328	001200'	047503	047522	043106	.ASCII	/COROFF/
329	001206'	334	000		.BYTE	334,0
330	001210'	020040	053126	047117	.ASCII	/VVON/
331	001216'	333	000		.BYTE	333,0
332	001220'	053040	047526	043106	.ASCII	/VVOFF/
333	001226'	332	000		.BYTE	332,0
334						
335	001230'	000376	002106		DVMVTE:	.WORD 376,MSFMT1-LOCZ
336	001234'	000375	002106			.WORD 375,MSFMT1-LOCZ
337	001240'	000374	002106			.WORD 374,MSFMT1-LOCZ
338	001244'	000373	002106			.WORD 373,MSFMT1-LOCZ
339	001250'	000372	002106			.WORD 372,MSFMT1-LOCZ
340	001254'	000371	002107			.WORD 371,MSFMT2-LOCZ
341	001260'	000370	002107			.WORD 370,MSFMT2-LOCZ
342	001264'	000367	002114			.WORD 367,MSFMT3-LOCZ
343	001270'	000366	002123			.WORD 366,MSFMT4-LOCZ
344	001274'	000365	002106			.WORD 365,MSFMT1-LOCZ
345	001300'	000364	002106			.WORD 364,MSFMT1-LOCZ
346	001304'	000363	002106			.WORD 363,MSFMT1-LOCZ
347	001310'	000362	002132			.WORD 362,MSFMT5-LOCZ
348	001314'	000361	002132			.WORD 361,MSFMT5-LOCZ
349	001320'	000360	002132			.WORD 360,MSFMT5-LOCZ
350	001324'	000357	002106			.WORD 357,MSFMT1-LOCZ
351	001330'	000356	002106			.WORD 356,MSFMT1-LOCZ
352	001334'	000355	002106			.WORD 355,MSFMT1-LOCZ
353	001340'	000354	002106			.WORD 354,MSFMT1-LOCZ

;MODEL VECTOR TABLE EXTEN.

DTRPA8.P11 COMPILER TABLES & CONSTANT AREAS

354	001344'	000353	002106	.WORD	353,MSFMT1-LOCZ
355	001350'	000352	002106	.WORD	352,MSFMT1-LOCZ
356	001354'	000351	002106	.WORD	351,MSFMT1-LOCZ
357	001360'	000350	002106	.WORD	350,MSFMT1-LOCZ
358	001364'	000347	002106	.WORD	347,MSFMT1-LOCZ
359	001370'	000346	002106	.WORD	346,MSFMT1-LOCZ
360	001374'	000345	002106	.WORD	345,MSFMT1-LOCZ
361	001400'	000344	002106	.WORD	344,MSFMT1-LOCZ
362	001404'	000343	002106	.WORD	343,MSFMT1-LOCZ
363	001410'	000342	002106	.WORD	342,MSFMT1-LOCZ
364	001414'	000341	002106	.WORD	341,MSFMT1-LOCZ
365	001420'	000340	002106	.WORD	340,MSFMT1-LOCZ
366	001424'	000337	002106	.WORD	337,MSFMT1-LOCZ
367	001430'	000336	002106	.WORD	336,MSFMT1-LOCZ
368	001434'	000335	002106	.WORD	335,MSFMT1-LOCZ
369	001440'	000334	002106	.WORD	334,MSFMT1-LOCZ
370	001444'	000333	002106	.WORD	333,MSFMT1-LOCZ
371	001450'	000332	002106	.WORD	332,MSFMT1-LOCZ

COMPILER TABLE EXTENSION

372					
373					
374					
375					
376	001454'	003	376	DVCPTC:	.BYTE 3,376 ;NO WAIT
377	001456'	004537	000012		.WORD 4537,10.
378	001462'	003	375		.BYTE 3,375 ;WAIT
379	001464'	004537	000012		.WORD 4537,10.
380	001470'	004	374		.BYTE 4,374 ;STATUS
381	001472'	004537	000012	001002	.WORD 4537,10.,1002
382	001500'	004	373		.BYTE 4,373 ;COUNTS
383	001502'	004537	000012	001001	.WORD 4537,10.,1001
384	001510'	003	372		.BYTE 3,372 ;SEEK
385	001512'	004537	000012		.WORD 4537,10.
386	001516'	006	371		.BYTE 6,371 ;WRITE CHECK HEADER & DATA
387	001520'	004537	000012	000000	.WORD 4537,10.,0,2,2
	001526'	000002	000002		
388	001532'	006	370		.BYTE 6,370 ;WRITE CHECK DATA
389	001534'	004537	000012	000000	.WORD 4537,10.,0,2,2
	001542'	000002	000002		
390	001546'	006	367		.BYTE 6,367 ;READ HEADER & DATA
391	001550'	004537	000012	000000	.WORD 4537,10.,0,2,2
	001556'	000002	000002		
392	001562'	006	366		.BYTE 6,366 ;WRITE HEADER & DATA
393	001564'	004537	000012	000000	.WORD 4537,10.,0,2,2
	001572'	000002	000002		
394	001576'	003	365		.BYTE 3,365 ;CONTROL RESET
395	001600'	004537	000012		.WORD 4537,10.
396	001604'	003	364		.BYTE 3,364 ;DRIVE RESET
397	001606'	004537	000012		.WORD 4537,10.
398	001612'	003	363		.BYTE 3,363 ;SEARCH
399	001614'	004537	000012		.WORD 4537,10.
400	001620'	004	362		.BYTE 4,362 ;STEP UP
401	001622'	004537	000012	000000	.WORD 4537,10.,0
402	001630'	004	361		.BYTE 4,361 ;STEP DOWN
403	001632'	004537	000012	000000	.WORD 4537,10.,0
404	001640'	004	360		.BYTE 4,360 ;OFFSET
405	001642'	004537	000012	000000	.WORD 4537,10.,0

406	001650'	003	357	.BYTE	3,357	;RETURN TO CENTERLINE
407	001652'	004537	000012	.WORD	4537,10.	
408	001656'	003	356	.BYTE	3,356	;UNLOAD
409	001660'	004537	000012	.WORD	4537,10.	
410	001664'	003	355	.BYTE	3,355	;RECALIBRATE
411	001666'	004537	000012	.WORD	4537,10.	
412	001672'	003	354	.BYTE	3,354	;PACK ACKNOWLEDGE
413	001674'	004537	000012	.WORD	4537,10.	
414	001700'	003	353	.BYTE	3,353	;READIN PRESET
415	001702'	004537	000012	.WORD	4537,10.	
416	001706'	003	352	.BYTE	3,352	;RELEASE
417	001710'	004537	000012	.WORD	4537,10.	
418	001714'	003	351	.BYTE	3,351	;A PORT
419	001716'	004537	000012	.WORD	4537,10.	
420	001722'	003	350	.BYTE	3,350	;B PORT
421	001724'	004537	000012	.WORD	4537,10.	
422	001730'	003	347	.BYTE	3,347	;FORMAT 22
423	001732'	004537	000012	.WORD	4537,10.	
424	001736'	003	346	.BYTE	3,346	;FORMAT 20
425	001740'	004537	000012	.WORD	4537,10.	
426	001744'	003	345	.BYTE	3,345	;ODD
427	001746'	004537	000012	.WORD	4537,10.	
428	001752'	003	344	.BYTE	3,344	;EVEN
429	001754'	004537	000012	.WORD	4537,10.	
430	001760'	003	343	.BYTE	3,343	;ECI ON
431	001762'	004537	000012	.WORD	4537,10.	
432	001766'	003	342	.BYTE	3,342	;ECI OFF
433	001770'	004537	000012	.WORD	4537,10.	
434	001774'	003	341	.BYTE	3,341	;HCI ON
435	001776'	004537	000012	.WORD	4537,10.	
436	002002'	003	340	.BYTE	3,340	;HCI OFF
437	002004'	004537	000012	.WORD	4537,10.	
438	002010'	003	337	.BYTE	3,337	;BAI ON
439	002012'	004537	000012	.WORD	4537,10.	
440	002016'	003	336	.BYTE	3,336	;BAI OFF
441	002020'	004537	000012	.WORD	4537,10.	
442	002024'	003	335	.BYTE	3,335	;CORRECTION ON
443	002026'	004537	000012	.WORD	4537,10.	
444	002032'	003	334	.BYTE	3,334	;CORRECTION OFF
445	002034'	004537	000012	.WORD	4537,10.	
446	002040'	003	333	.BYTE	3,333	;VOLUME VALID ON
447	002042'	004537	000012	.WORD	4537,10.	
448	002046'	003	332	.BYTE	3,332	;VOLUME VALID OFF
449	002050'	004537	000012	.WORD	4537,10.	

...  
DVIWST: DEVICE INTERFACE WORD SYMBOL TABLE

451						
452						
453						
454	002054'	054503	020114	.ASCII	/CYL /	
455	002060'	003004		.WORD	DEVIW1	
456	002062'	042510	042101	.ASCII	/HEAD/	
457	002066'	000006		.WORD	DEVIW2	
458	002070'	042523	052103	.ASCII	/SECT/	
459	002074'	000010		.WORD	DEVIW3	
460	002076'	052122	054522	.ASCII	/RTRY/	
461	002102'	000012		.WORD	DEVIW4	

```

462 002104' 177777 .WORD 177777 ;END OF TABLE
463
464
465 ;
466 ; MODEL STATEMENT TABLE EXTENSION
467 002106' 000 MSFMT1: .BYTE 0
468 002107' 377 052101 000377 MSFMT2: .ASCIZ <377>/AT/<377>
469 002114' 044777 052116 177517 MSFMT3: .ASCIZ <377>/INTO/<377>
    002122' 000
470 002123' 377 051106 046517 MSFMT4: .ASCIZ <377>/FROM/<377>
    002130' 000377
471 002132' 377 000 MSFMT5: .BYTE 377,0
472 .EVEN
473
474
475 ;DEVICE ROUTINE CONSTANTS & EQUATES
476
477
478 002134' HSKPST= .
479 002134' ISTAT= . ;STORAGE FOR DEV REG'S AT INT
480 002134' 000000 000000 000000 .WORD 0,0,0,0,0,0,0,0
    002142' 000000 000000 000000
    002150' 000000 000000 000000
481 002154' 000000 000000 000000 .WORD 0,0,0,0,0,0,0,0
    002162' 000000 000000 000000
    002170' 000000
482 002172' 000000 000000 000000 .WORD 0,0,0,0,0,0,0,0
    002200' 000000 000000 000000
    002206' 000000
483
484 002210' 000026 CSTAT: .BLKW 22. ;DEV REG CURRENT VALUES STORAGE
485 002264' COUNTS:
486 002264' 000000 BYRD: .WORD 0 ;BYTES READ COUNT
487 002266' 000000 .WORD 0
488 002270' 000000 BYWR: .WORD 0 ;BYTES WRITTEN COUNT
489 002272' 000000 .WORD 0
490 002274' 000000 BYCK: .WORD 0 ;BYTES CHECKED COUNT
491 002276' 000000 .WORD 0
492 002300' 000000 RDCNT: .WORD 0 ;READ CMND COUNT
493 002302' 000000 WRCNT: .WORD 0 ;WRITE CMND COUNT
494 002304' 000000 CKCNT: .WORD 0 ;CHECK CMND COUNT
495 002306' 000000 SKCNT: .WORD 0 ;SEEK/SEARCH CMND COUNT
496 002310' 000000 MISCNT: .WORD 0 ;MISC. CMND COUNT
497 002312' 000000 ERRCNT: .WORD 0 ;DEVICE ERRORS COUNT
498 002314' 000000 CECCER: .WORD 0 ;CORRECTABLE ECC ERRORS
499 002316' 000000 DATAER: .WORD 0 ;DATA ERRORS COUNT
500 002320' 000000 DLT CNT: .WORD 0 ;DATA LATE ERRORS
501 002322' 000000 DTECNT: .WORD 0 ;DRIVE TIMING ERRORS
502 002324' 000000 HRCNT: .WORD 0 ;HEADER CRC ERRORS
503 002326' 000000 FERCNT: .WORD 0 ;FORMAT ERRORS
504 002330' 000000 HCECNT: .WORD 0 ;HEADER COMPARE ERRORS
505 002332' 000000 DCKCNT: .WORD 0 ;DATA CHECK ERRORS
506 002334' 000000 WCECNT: .WORD 0 ;WRITE CHECK ERRORS
507 002336' 000000 RETRYS: .WORD 0 ;# OF RETRIES ON I/O CMNDS
508 002340' 000000 INTCNT: .WORD 0 ;INTERRUPTS COUNT
509

```

510	002342'	000000	ERRADR: .WORD	0	; CURR ADR IN USER PROG
511	002344'	000000	CNTADR: .WORD	0	; ADR OF BYTE COUNT TOTALS
512	002346'	000000	CURFLG: .WORD	0	; FLAG WORD OF CURR CMND
513	002350'	000000	CURCMD: .WORD	0	; CURR CMND CODE
514	002352'	000000	CURADR: .WORD	0	; CURR BUS ADDRESS
515	002354'	000000		0	
516	002356'	000000	CURCNT: .WORD	0	; NEG WORD CNT FOR CURR CMND
517	002360'	000000	CURPBC: .WORD	0	; POSITIVE BYTE CNT FOR CURR CMND
518	002362'	000000	FINCNT: .WORD	0	; FINAL WORD CNT (RPWC)
519	002364'	000000	CURRTY: .WORD	0	; CURR RETRY COUNT
520	002366'	000000	RTRYIP: .WORD	0	; RETRY IN PROGRESS FLAG
521	002370'	000000	CURPSW: .WORD	0	; PSW STORAGE AREA
522		002372'	HSKPEN=	.	
523					
524	002372'	000000	RPCS1V: .WORD	0	; BASE VALUE FOR RPCS1 REG
525		002000	PORT=	2000	; PORT SELECT BIT - 1 = PORT B
526					
527	002374'	000000	RPCS2V: .WORD	0	; BASE VALUE FOR RPCS2 REG
528		000010	BAI=	10	; BUS ADDRESS INCREMENT INHIBIT = 1
529		000020	PARITY=	20	; PARITY MODE - 1 = EVEN
530					
531	002376'	010000	RPOFV: .WORD	10000	; BASE VALUE FOR RPOF REG
532		002000	HCI=	2000	; HEADER COMPARE INHIBIT = 1
533		004000	ECI=	4000	; ECC INHIBIT = 1
534		010000	FMT228=	10000	; FORMAT 16 BIT WORDS = 1
535					
536		000000	XXXX=	0	; VALUE TO BE TAILORED BY DEV ROUT
537					
538		177776	PS=	177776	; PSW ADDRESS
539					
540		000026	REGNUM=	22	; # OF DEVICE REGISTERS
541		000027	CNTNUM=	23	; # OF STATISTICAL COUNT WORDS
542					
543		000001	MMVER=	1	; SYS FLGWD BIT DEFINITIONS
544		000002	USMTPS=	2	
545		000010	CPU70=	10	
546					
547		172350	KPAR4=	172350	; MEMORY MANAGEMENT EQUATES
548		172310	KPDR4=	172310	
549		100000	P4CONS=	100000	
550		077406	PDRCON=	077406	
551					
552		000000	RPCS1=	0	; EQUATES FOR DEVICE REGISTER NAMES
553		000002	RPWC=	2	
554		000004	RPBA=	4	
555		000006	RPDA=	6	
556		000010	RPCS2=	10	
557		000012	RPDS=	12	
558		000014	RPER1=	14	
559		000016	RPAS=	16	
560		000020	RPLA=	20	
561		000022	RPDB=	22	
562		000024	RPDR=	24	
563		000026	RPDT=	26	
564		000030	RPSN=	30	
565		000032	RPOF=	32	

566	000034	RPDC=	34
567	000036	RPCC=	36
568	000040	RPER2=	40
569	000042	RPER3=	42
570	000044	RPEC1=	44
571	000046	RPEC2=	46
572	000050	RPBAE=	50
573	000052	RPCS3=	52

;RPCS1 DEVICE BIT EQUATES

578	040000	TRE=	40000
579	020000	MCPE=	20000
580	004000	DVA=	4000

;RPCS2

584	010000	NED=	10000
-----	--------	------	-------

;RPDS

588	100000	ATA=	100000
589	040000	ERR=	40000
590	010000	MOL=	10000
591	000400	DPR=	400
592	000100	VV=	100

;RPER1

596	040000	UNS=	40000
-----	--------	------	-------

002400'

PATCH:	.REPT	20.	;PATCH AREA
	.WORD	0	
	.ENDR		

599  
600  
601

603  
604  
605  
606  
607  
608  
609  
610  
611  
612  
613  
614  
615  
616  
617  
618  
619  
620  
621  
622  
623  
624  
625  
626  
627  
628  
629  
630  
631  
632  
633  
634  
635  
636  
637  
638  
639  
640  
641  
642  
643  
644  
645  
646  
647  
648  
649  
650  
651  
652  
653  
654  
655  
656  
657  
658

.SBTTL RPO4/RPO5/RPO6 SUPPORT ROUTINES ENTERED FROM MPG

;DEVICE ROUTINE HOUSEKEEPING

```

;JSR   RS,MSKEEP      S/R CALL
;WORD  0 OR 1         0 = DO HSKP PER OPSW
;                               1 = UNCOND. DO HSKP
;R2 = PROG'S OPSW
;DESTROYS R0,R1
    
```

```

002450' 012767 000003 175334 MSKEEP: MOV   #3,RTRY      ;INIT # OF RETRY ATTEMPTS
002456' 005067 177710          CLR   RPCS1V     ;INITIALIZE RPCS1 VALUE
002462' 005067 177706          CLR   RPCS2V     ;INITIALIZE RPCS2 VALUE
002466' 012767 010000 177702 MOV   #10000,RPOFV ;INITIALIZE RPOF VALUE
002474' 005725          TST   (R5)+      ;UNCONDITIONALLY DO HSKP?
002476' 001003          BNE   10$        ;Y,N-10$
002500' 032702 000004          BIT   #MSKPEP,R2 ;OPSW SPECIFY EACH PASS HSKP?
002504' 001010          BNE   30$        ;Y,N-30$
002506' 010700          10$: MOV   PC,R0      ;SET UP FIRST WD ADR
002510' 062700 177424          ADD   #MSKPST-. ,R0
002514' 012701 000117          MOV   #MSKPEN-MSKPST/2,R1 ;SET UP # OF WORDS
002520' 005020          20$: CLR   (R0)+      ;MSKP ALL NECESSARY AREAS
002522' 005301          DEC   R1
002524' 001375          BNE   20$
002526' 000205          30$: RTN      ;EXIT IN-LINE
    
```

;RPO4/RPO5/RPO6 REPORT ROUTINE

```

;JSR   RS,REPORT      S/R CALL
;WORD  FLAGWD         FLAGWORD
;                               BIT 15 = CMND MODE CALL
;                               BIT  9 = PROG STMT CALL
;                               BIT  1 = DO STATUS REPORT
;                               BIT  0 = DO COUNTS REPORT
    
```

```

002530' 004067 006604          REPORT: JSR   R0,SAVREG ;SAVE REG'S R0 - R5
002534' 004767 006632          JSR   PC,SUPTAD ;SET UP PROG TBL ADR IN R3
002540' 011504          MOV   (R5),R4   ;GET FLAGWORD
002542' 032704 000002          BIT   #2,R4     ;GOING TO DO STATUS DISPLAY?
002546' 001403          BEQ   5$        ;Y,N-5$
002550' 004567 006636          JSR   RS,STSTAT ;GO STORE STATUS REG'S
002554' 177434          .WORD  CSTAT-
002556' 032704 177776          5$:  BIT   #177776,R4 ;DISPLAYING CNTS AT END OF
002562' 001012          BNE   15$      ;PROG PASS? (Y,N-15$)
002564' 010700          MOV   PC,R0     ;SET UP ADR OF CNTS
002566' 062700 177476          ADD   #COUNTS-. ,R0
002572' 012701 000027          10$: MOV   #CNTNUM,R1 ;GET # OF CNT WORDS
002576' 005720          TST   (R0)+    ;THIS CNT WORD = 0?
002600' 001003          BNE   15$      ;Y,N-15$
002602' 005301          DEC   R1        ;DECR WORD CNT
002604' 001374          BNE   10$      ;CK'ED ALL WORDS? (Y,N-10$)
002606' 000477          BR    DVREX     ;GO TO EXIT -- ALL CNTS ARE 0'S
002610' 004767 006706          15$: JSR   PC,DEVID ;DISPLAY DEVICE I.D.
002614' 032704 000002          BIT   #2,R4     ;DOING STATUS DISPLAY?
    
```

654	002620'	001432		BEQ	DISCNT		:Y N-DISCNT
660	002622'	010700		MOV	PC,R0		:SET UP ADR OF REG'S AT
661	002624'	062700	177310	ADD	#ISTAT-. ,R0		:LAST INT
662	002630'	012701	000026	MOV	#REGNUM,R1		:SET UP # OF REG'S
663	002634'	005720		20\$: TST	(R0)+		:ALL REG'S = 0?
664	002636'	001003		BNE	30\$		:N,Y-40\$
665	002640'	005301		DEC	R1		
666	002642'	001374		BNE	20\$		
667	002644'	000407		BR	40\$		
668	002646'	004567	007250	30\$: JSR	RS,PRINT		:ISSUE 'AT LAST INT' MSG
669	002652'	007371		.WORD	ATMSG-		
670	002654'	000014		.WORD	12.		
671	002656'	004567	006770	JSR	RS,DISPST		:GO DISPLAY STATUS AT LAST INT
672	002662'	177252		.WORD	ISTAT-		
673	002664'	004567	007232	40\$: JSR	RS,PRINT		:ISSUE 'CURRENTLY' MSG
674	002670'	007367		.WORD	CURMSG-		
675	002672'	000012		.WORD	10.		
676	002674'	004567	006752	JSR	RS,DISPST		:GO DISPLAY CURRENT STATUS
677	002700'	177310		.WORD	CSTAT-		
678	002702'	004767	007144	JSR	PC,PRTIWD		:GO DISPLAY INFO WORDS
679	002706'	032704	000001	DISCNT: BIT	#1,R4		:DISPLAY COUNTS?
680	002712'	001431		BEQ	RPTEND		:Y N-RPTEND
681	002714'	012700	000027	MOV	#CNTNUM,R0		:SET UP # OF WORDS
682	002720'	010701		MOV	PC,R1		:SET UP ADR OF CNTS
683	002722'	062701	177342	ADD	#COUNTS-. ,R1		
684	002726'	010702		MOV	PC,R2		:SET UP TBL ADR
685	002730'	062702	000066	ADD	#REPTBL-. ,R2		
686	002734'	012267	000012	RPTLP: MOV	(R2)+,RPTBAS		:MOV MSG ADR TO S/R LINKAGE
687	002740'	004067	006374	JSR	R0,SAVREG		:SAVE ALL REG'S
688	002744'	011100		MOV	(R1),R0		:GET CURRENT COUNT
689	002746'	004577	175104	JSR	RS,ABINASC		:CONVERT IT TO ASCII
690	002752'	000000		RPTBAS: .WORD	XXXX		
691	002754'	004067	006374	JSR	R0,RESREG		:RESTORE REG'S
692	002760'	005721		TST	(R1)+		:POINT AT NXT CNT
693	002762'	005300		DEC	R0		:DONE ALL WORDS?
694	002764'	001363		BNE	RPTLP		:Y N-RPTLP
695	002766'	004567	007130	JSR	RS,PRINT		:GO ISSUE COUNTS MSG
696	002772'	007416		.WORD	CNTSMG-		
697	002774'	000512		.WORD	CNTSEN-CNTSMG		
698	002776'	004567	007120	RPTEND: JSR	RS,PRINT		:ISSUE "END OF REPORT" MSG
699	003002'	007267		.WORD	RENDMG-		
700	003004'	177763		.WORD	-13.		
701	003006'	004067	006342	DVREX: JSR	R0,RESREG		:RESTORE REGISTERS
702	003012'	005725		TST	(R5)+		:SET UP RETURN POINT
703	003014'	000205		RTS	RS		:EXIT IN-LINE
704							
705							
706	003016'	007452		REPTBL: .WORD	BCMRD-RPTBAS		
707	003020'	007460		.WORD	BCMRD+6-RPTBAS		
708	003022'	007474		.WORD	BCMWR-RPTBAS		
709	003024'	007502		.WORD	BCMWR+6-RPTBAS		
710	003026'	007520		.WORD	BCMCK-RPTBAS		
711	003030'	007526		.WORD	BCMCK+6-RPTBAS		
712	003032'	007553		.WORD	CMDCRD-RPTBAS		
713	003034'	007566		.WORD	CMDCWR-RPTBAS		
714	003036'	007601		.WORD	CMDCCK-RPTBAS		



# E02

771	003266'	110064	000010			MOV B	R0, RPCS2(R4)	: MOVE UNIT # TO RH11
772	003272'	005714				TST	(R4)	: SELECT MY DRIVE
773	003274'	016400	000012			MOV	RPO5(R4), R0	: GET DRIVE'S STATUS
774	003300'	005100				COM	R0	: FLIP THE BITS
775	003302'	032700	010200			BIT	#10200, R0	: WERE MOL & DAY BOTH SET?
776	003306'	001005				BNE	40\$	: Y, N-40\$
777	003310'	042713	000010			BIC	#WT4IOT, (R3)	: RESET WAITING FOR I/O TERM FLAG
778	003314'	042767	000004	174460		BIC	#UNLDIP, DFLGWD	: RESET UNLOAD IN PROGRESS FLAG
779	003322'	012763	005670	00003C	40\$:	MOV	#3000, PTOCNT(R3)	: RESTORE 1 SECOND T/O COUNT
780	003330'	110164	000010			MOV B	R1, RPCS2(R4)	: RESTORE ORIG UNIT #
781	003334'	005714				TST	(R4)	: RESELECT ORIG DRIVE
782	003336'	032777	000002	174520		BIT	#USMTPS, @CSYSFW	: NEED TO USE MTPS INST?
783	003344'	001004				BNE	50\$	: N, Y-50\$
784	003346'	116737	177016	177776		MOV B	CURPSW, @#PS	: RESTORE ORIG PRIORITY
785	003354'	000402				BR	60\$	: GO TO EXIT
786	003356'	106467	177006		50\$:	MTPS	CURPSW	: RESTORE ORIG PRIORITY
787	003362'	004067	005766		60\$:	JSR	R0, RESREG	: RESTORE REGISTERS
788	003366'	000205				RTS	R5	: RETURN IN-LINE
789								
790								
791								: KILL USER PROGRAM ROUTINE
792								
793								: JSR R5, KILL S/R CALL
794								: R3 MUST CONTAIN PROG TBL ADR
795								: DESTROYS R0, R1
796								
797	003370'	016701	174430			KILL: MOV	DREGADR, R1	: GET DEV REG ADR
798	003374'	004567	004254			JSR	R5, TVECT	: DO I HAVE VECTOR CONTROL?
799	003400'	000407				BR	KILLEX	: BR IF I DON'T
800	003402'	132711	000100			BIT B	#100, (R1)	: IS INT ENABLE SET?
801	003406'	001402				BEQ	10\$	: Y, N-10\$
802	003410'	142711	000100			BIC B	#100, (R1)	: RESET INT ENABLE
803	003414'	004767	004210		10\$:	JSR	PC, RINTV	: RESET INT VECTOR INFO
804	003420'	000205				KILLEX: RTS	R5	: EXIT IN-LINE

```

806 .SBTTL RPO4/RPO5/RPO6 NON-I/O FUNCTION ROUTINES
807
808
809 ;"STEPUP" FUNCTION ROUTINE
810
811 ;JSR R5,STEPUP FUNCTION CALL
812 ;.WORD NBR INCREMENT FACTOR
813
814 003422' 004767 000060 STEPUP: JSR PC,STPCOM ;DO COMMON SETUP
815 003426' 062502 ADD (R5)+,R2 ;ADD INCR VALUE TO SECT #
816 003430' 020203 10$: CMP R2,R3 ;SECT # IN RANGE?
817 003432' 103403 BLO 15$ ;N,Y-15$
818 003434' 160302 SUB R3,R2 ;ADJ SECT # DOWNWARDS
819 003436' 005201 INC R1 ;ADD 1 TO HEAD #
820 003440' 000773 BR 10$ ;GO CK NEW SECT # VALUE
821 003442' 020127 000023 15$: CMP R1,#19. ;HEAD # IN RANGE?
822 003446' 103404 BLO 20$ ;N,Y-20$
823 003450' 162701 000023 SUB #19.,R1 ;ADJ HEAD # DOWNWARDS
824 003454' 005200 INC R0 ;ADD 1 TO CYL #
825 003456' 000771 BR 15$ ;GO CHECK NEW HEAD #
826 003460' 020004 20$: CMP R0,R4 ;CYL # IN RANGE?
827 003462' 103402 BLO STEPEX ;N,Y-STEPEX
828 003464' 160400 SUB R4,R0 ;ADJ CYL # DOWNWARDS
829 003466' 000774 BR 20$ ;GO CK IT NOW
830 003470' 010067 174310 STEPEX: MOV R0,CYL ;STORE NEW CYL,HEAD,SECT VALUES
831 003474' 010167 174306 MOV R1,HEAD
832 003500' 010267 174304 MOV R2,SECT
833 003504' 000205 RTS R5 ;EXIT TO USER PROG
834
835
836 003506' 016700 174272 STPCOM: MOV CYL,R0 ;GET CYL, HEAD, & SECT VALUES
837 003512' 016701 174270 MOV HEAD,R1 ;IN REGISTERS
838 003516' 016702 174266 MOV SECT,R2
839 003522' 004767 005644 JSR PC,SUPTAD ;GET PROG TBL ADR IN R3
840 003526' 012704 000633 MOV #411.,R4 ;PRESET # OF CYL'S TO RPO4/RPO5
841 003532' 032763 000040 000032 BIT #40,PMDLCD(R3) ;THIS AN RPO6?
842 003540' 001402 BEQ STPC10 ;Y,N-STPC10
843 003542' 012704 001457 MOV #815.,R4 ;SET UP # OF RPO6 CYL'S
844 003546' 012703 000026 STPC10: MOV #22.,R3 ;PRESET TO # OF 16 BIT WD SECTORS
845 003552' 032767 010000 176616 BIT #FMT22B,RPOFV ;USING 16 BIT WORD FORMAT?
846 003560' 001002 BNE STPCEX ;N,Y-STPCEX
847 003562' 012703 000024 STPCEX: MOV #20.,R3 ;SET UP # OF 18 BIT WD SECTORS
848 003566' 000207 RTS PC ;EXIT IN-LINE
849
850
851 ;"STEPDN" FUNCTION ROUTINE
852
853 ;JSR R5,STEPDN FUNCTION CALL
854 ;.WORD NBR DECREMENT FACTOR
855
856 003570' 004767 177712 STEPDN: JSR PC,STPCOM ;DO COMMON SETUP
857 003574' 162502 SUB (R5)+,R2 ;SUB DECR FACTOR FROM SECT #
858 003576' 020203 40$: CMP R2,R3 ;SECT # IN RANGE?
859 003600' 103403 BLO 45$ ;N,Y-45$
860 003602' 060302 ADD R3,R2 ;ADJ SECT # UPWARDS
861 003604' 005301 DEC R1 ;DECR HEAD # BY 1
    
```

```

862 003606' 000773          BR      40$          ;GO CK NEW SECT # VALUE
863 003610' 020127 000023 45$:  CMP      R1,#19.      ;HEAD # IN RANGE?
864 003614' 103404          BLO     50$          ;N Y-50$
865 003616' 062701 000023  ADD     #19.,R1      ;ADJ IT UPWARDS
866 003622' 005300          DEC     R0          ;DECR CYL # BY 1
867 003624' 000771          BR      45$          ;GO CHECK NEW HEAD #
868 003626' 020004          50$:  CMP     R0,R4      ;CYL # IN RANGE?
869 003630' 103717          BLO     STEPEX      ;N Y-STEPEX
870 003632' 060400          ADD     R4,R0      ;ADJ IT UPWARDS
871 003634' 000774          BR      50$          ;GO CK IT NOW
872
873
874          ;"WAIT" FUNCTION ROUTINE
875
876          ;JSR    RS,WAIT          FUNCTION CALL
877
878 003636' 042767 100000 174136 WAIT: BIC     #WAITMD,DFLGMD ;RESET THE "NOWAIT" FLAG
879 003644' 004767 003464          JSR     PC,CKDBSY   ;WAIT IF BUSY & DO TERMINATION
880 003650' 004767 003754          JSR     PC,RINTV    ;RESET THE INTERRUPT VECTOR
881 003654' 000205          RTS     R5          ;EXIT IN-LINE
882
883
884          ;"NOWAIT" FUNCTION ROUTINE
885
886          ;JSR    RS,NOWAIT        FUNCTION CALL
887
888 003656' 052767 100000 174116 NOWAIT: BIS     #WAITMD,DFLGMD ;SET THE "NOWAIT" FLAG
889 003664' 000205          RTS     R5          ;EXIT IN-LINE
890
891
892          ;"APORT" FUNCTION ROUTINE
893
894          ;JSR    RS,APORT         FUNCTION CALL
895
896 003666' 042767 002000 176476 APORT: BIC     #PORT,RPCS1V ;RESET THE PORT BIT
897 003674' 000205          RTS     R5          ;EXIT IN-LINE
898
899
900          ;"BPORT" FUNCTION ROUTINE
901
902          ;JSR    RS,BPORT        FUNCTION CALL
903
904 003676' 052767 002000 176466 BPORT: BIS     #PORT,RPCS1V ;SET THE PORT BIT TO B PORT
905 003704' 000205          RTS     R5          ;EXIT IN-LINE
906
907
908          ;"ODD" FUNCTION ROUTINE
909
910          ;JSR    RS,ODD          FUNCTION CALL
911
912 003706' 042767 000020 176460 ODD:  BIC     #PARITY,RPCS2V ;RESET THE PARITY BIT
913 003714' 000205          RTS     R5          ;EXIT IN-LINE

```

## H02

MAINDEC-11-DTRPA-B RH11/RH70 - RPO4/RPO5/RPO6 DEVICE ROUTINE FOR MPG MACY11 27(732) 24-SEP-76 14:21 PAGE 6-2  
 DTRPAB.P11 RPO4/RPO5/RPO6 NON-I/O FUNCTION ROUTINES

SEQ 0020

```

915 ;"EVEN" FUNCTION ROUTINE
916
917 ;JSR RS,EVEN FUNCTION CALL
918
919 003716' 052767 000020 176450 EVEN: BIS #PARITY,RPCS2V ;SET THE PARITY BIT FOR EVEN
920 003724' 000205 RTS RS ;EXIT IN-LINE
921
922
923 ;"ECION" FUNCTION ROUTINE
924
925 ;JSR RS,ECION FUNCTION CALL
926
927 003726' 052767 004000 176442 ECION: BIS #ECI,RPOFV ;SET THE ECI BIT
928 003734' 000205 RTS RS ;EXIT IN-LINE
929
930
931 ;"ECIOFF" FUNCTION ROUTINE
932
933 ;JSR RS,ECIOFF FUNCTION CALL
934
935 003736' 042767 004000 176432 ECIOFF: BIC #ECI,RPOFV ;RESET THE ECI BIT
936 003744' 000205 RTS RS ;EXIT IN-LINE
937
938
939 ;"HCION" FUNCTION ROUTINE
940
941 ;JSR RS,HCION FUNCTION CALL
942
943 003746' 052767 002000 176422 HCION: BIS #HCI,RPOFV ;SET THE HCI BIT
944 003754' 000205 RTS RS ;EXIT IN-LINE
945
946
947 ;"HCIOFF" FUNCTION ROUTINE
948
949 ;JSR RS,HCIOFF FUNCTION CALL
950
951 003756' 042767 002000 176412 HCIOFF: BIC #HCI,RPOFV ;RESET THE HCI BIT
952 003764' 000205 RTS RS ;EXIT IN-LINE
953
954
955 ;"BAION" FUNCTION ROUTINE
956
957 ;JSR RS,BAION FUNCTION CALL
958
959 003766' 052767 000010 176400 BAION: BIS #BAI,RPCS2V ;SET THE BAI BIT
960 003774' 000205 RTS RS ;EXIT IN-LINE
961
962
963 ;"BAIOFF" FUNCTION ROUTINE
964
965 ;JSR RS,BAIOFF FUNCTION CALL
966
967 003776' 042767 000010 176370 BAIOFF: BIC #BAI,RPCS2V ;RESET THE BAI BIT
968 004004' 000205 RTS RS ;EXIT IN-LINE

```

```

970          ;"CORON" FUNCTION ROUTINE
971
972          ;JSR    RS,CORON          FUNCTION CALL
973
974 004006' 042767 002000 173766 CORON: BIC    #CORFLG,DFLGWD  ;RESET THE COR INH FLAG
975 004014' 000205          RTS      RS          ;EXIT IN-LINE
976
977
978          ;"COROFF" FUNCTION ROUTINE
979
980          ;JSR    RS,COROFF        FUNCTION CALL
981
982 004016' 052767 002000 173756 COROFF: BIS    #CORFLG,DFLGWD  ;SET THE COR INH FLAG
983 004024' 000205          RTS      RS          ;EXIT IN-LINE
984
985
986          ;"VVON" FUNCTION ROUTINE
987
988          ;JSR    RS,VVON          FUNCTION CALL
989
990 004026' 042767 001000 173746 VVON:  BIC    #VVFLG,DFLGWD  ;RESET THE VV INH FLAG
991 004034' 000205          RTS      RS          ;EXIT IN-LINE
992
993
994          ;"VVOFF" FUNCTION ROUTINE
995
996          ;JSR    RS,VVOFF        FUNCTION CALL
997
998 004036' 052767 001000 173736 VVOFF: BIS    #VVFLG,DFLGWD  ;SET THE VV INH FLAG
999 004044' 000205          RTS      RS          ;EXIT IN-LINE
1000
1001
1002          ;"FMT22" FUNCTION ROUTINE
1003
1004          ;JSR    RS,FMT22        FUNCTION CALL
1005
1006 004046' 052767 010000 176322 FMT22: BIS    #FMT22B,RPOFV   ;SET THE FORMAT 22 BIT
1007 004054' 000205          RTS      RS          ;EXIT IN-LINE
1008
1009
1010          ;"FMT20" FUNCTION ROUTINE
1011
1012          ;JSR    RS,FMT20        FUNCTION CALL
1013
1014 004056' 042767 010000 176312 FMT20: BIC    #FMT22B,RPOFV   ;RESET THE FORMAT 22 BIT
1015 004064' 000205          RTS      RS          ;EXIT IN-LINE

```

```

1017 .SBTTL RPO4/RPO5/RPO6 NON-INTERRUPT TYPE I/O FUNCTION ROUTINES
1018
1019
1020 ;"CRESET" FUNCTION ROUTINE
1021
1022 ;JSR RS,CRESET FUNCTION CALL
1023
1024 004066' 004767 003242 CRESET: JSR PC,CKDBSY ;GO CK IF DEV IS BUSY
1025 004072' 005267 176212 INC MISCNT ;ADD 1 TO MISC. CMND CNT
1026 004076' 005067 173720 CLR ERR1 ;RESET THE ERROR INDICATOR
1027 004102' 005000 CLR R0 ;INITIALIZE TIME OUT COUNT
1028 004104' 052764 000040 000010 BIS #40,RPCS2(R4) ;SET THE CLR BIT IN RPCS2
1029 004112' 105714 10$: TSTB (R4) ;READY SET YET?
1030 004114' 100407 BMI 20$ ;N,Y-20$
1031 004116' 005300 DEC R0 ;TIMED OUT?
1032 004120' 100774 BMI 10$ ;Y,N-10$
1033 004122' 004567 003556 JSR RS,ERRCS ;GO ISSUE CRESET TIMEOUT ERROR
1034 004126' 003277 .WORD CRT0-ERMBAS
1035 004130' 000177 173714 JMP @CUPGER ;GO TO ERR RET POINT IN MPG
1036 004134' 000205 20$: RTS ;EXIT IN-LINE TO USER'S PROG
1037
1038
1039 ;"DRESET" FUNCTION ROUTINE
1040
1041 ;JSR RS,DRESET FUNCTION CALL
1042
1043 004136' 012702 000011 DRESET: MOV #011,R2 ;SET UP DRESET CMND CODE
1044 004142' 000414 BR NOICOM ;GO TO NO INT CMND COM PROCESSING
1045
1046
1047 ;"UNLOAD" FUNCTION ROUTINE
1048
1049 ;JSR RS,UNLOAD FUNCTION CALL
1050
1051 004144' 012702 000003 UNLOAD: MOV #003,R2 ;SET UP UNLOAD CMND CODE
1052 004150' 000411 BR NOICOM ;GO TO NO INT CMND COM PROCESSING
1053
1054
1055 ;"PAKACK" FUNCTION ROUTINE
1056
1057 ;JSR RS,PAKACK FUNCTION CALL
1058
1059 004152' 012702 000023 PAKACK: MOV #023,R2 ;SET UP PACK ACKNOWLEDGE CMND CODE
1060 004156' 000406 BR NOICOM ;GO TO NO INT CMND COM PROCESSING
1061
1062
1063 ;"RDPSET" FUNCTION ROUTINE
1064
1065 ;JSR RS,RDPSET FUNCTION CALL
1066
1067 004160' 012702 000021 RDPSET: MOV #021,R2 ;SET UP READ IN PRESET CMND CODE
1068 004164' 000403 BR NOICOM ;GO TO NO INT CMND COM PROCESSING
    
```

# K02

```

1070 ;"REL" FUNCTION ROUTINE
1071
1072 ;JSR RS,REL FUNCTION CALL
1073
1074 004166' 012702 000013 REL: MOV #013,R2 ;SET UP RELEASE CMND CODE
1075 004172' 000400 BR NOICOM ;GO TO NO INT CMND COM PROCESSING
1076
1077
1078 ;NON INTERRUPT I/O FUNCTION COMMON PROCESSING
1079
1080 ;RS = USER PROGRAM RETURN ADR
1081 ;R4 = RPCS1 ADR
1082 ;R3 = PROG TBL ADR
1083 ;R2 = FUNCTION'S COMMAND CODE
1084
1085 004174' 004767 003134 NOICOM: JSR PC,CKDBSY ;GO CK IF DEV IS BUSY
1086 004200' 005267 176104 INC MISCNT ;ADD 1 TO MISC. CMND CNT
1087 004204' 004767 000202 JSR PC,ACQHSK ;ACQUIRE & HOUSEKEEP THE DISK
1088 004210' 016764 176162 000032 MOV RPOFV,RPOF(R4) ;SET UP BITS IN RPOF
1089 004216' 020227 000003 CMP R2,#003 ;UNLOAD CMND?
1090 004222' 001403 BEQ 10$ ;N,Y-10$
1091 004224' 020227 000013 CMP R2,#013 ;RELEASE CMND?
1092 004230' 001012 BNE 20$ ;Y,N-20$
1093 004232' 032767 001000 173542 10$: BIT #VVFLG,DFLGWD ;VVON STATEMENT IN EFFECT?
1094 004240' 001006 BNE 20$ ;Y,N-20$
1095 004242' 032764 000100 000012 BIT #VV,RPDS(R4) ;IS THE VV BIT SET?
1096 004250' 001002 BNE 20$ ;N,Y-20$
1097 004252' 112714 000023 MOV# #023,(R4) ;ISSUE THE PACK ACKNOWLEDGE CMND
1098 004256' 056702 176110 20$: BIS RPCS1V,R2 ;SET PORT SELECT BIT INTO CMND WORD
1099 004262' 052767 000300 1735_2 BIS #CMDISU+ANYIOI,DFLGWD ;SET CMND ISSUED FLAGS
1100 004270' 010214 MOV R2,(R4) ;ISSUE SPECIFIED CMND
1101 004272' 012700 000012 MOV #10.,R0 ;SET UP DELAY CNT
1102 004276' 005300 30$: DEC R0 ;DELAY FOR A FEW MICROSECONDS
1103 004300' 001376 BNE 30$
1104 004302' 005714 TST (R4) ;IS 'SC' ERROR BIT SET?
1105 004304' 100021 BPL 50$ ;Y,N-50$
1106 004306' 032714 060000 BIT #TRE+MCPE,(R4) ;TRE OR MCPE ERR BITS SET?
1107 004312' 001011 BNE 40$ ;N,Y-40$
1108 004314' 032764 040000 000012 BIT #ERR,RPDS(R4) ;ERROR SUMMARY BIT SET?
1109 004322' 001005 BNE 40$ ;N,Y-40$
1110 004324' 016400 000016 MOV RPAS(R4),R0 ;GET ATA SUMMARY REG
1111 004330' 036700 000622 BIT MYATA,R0 ;IS IT MY ATA LINE?
1112 004334' 001405 BEQ 50$ ;Y,N-50$
1113 004336' 004567 003342 40$: JSR R5,ERRCS ;REPORT NON-INT TERM ERROR
1114 004342' 003473 .WORD NOITER-ERMBAS
1115 004344' 000177 173500 JMP @CUPGER ;GO TO MPG'S ERROR RETURN POINT
1116 004350' 120227 000003 50$: CMPB R2,#003 ;IS THIS THE UNLOAD CMND?
1117 004354' 001015 BNE 60$ ;Y,N-60$
1118 004356' 012763 005670 000030 MOV #3000,PTOCNT(R3) ;INITIALIZE 1 SECOND T/O CNT
1119 004364' 052767 000004 173410 BIS #UNLDIP,DFLGWD ;SET UNLOAD IN PROGRESS FLAG
1120 004372' 005767 173404 TST DFLGWD ;"NOWAIT" FLAG SET?
1121 004376' 100404 BMI 60$ ;N,Y-60$
1122 004400' 052713 000010 BIS #WT4IOT,(R3) ;SET WAITING FOR I/O TERM FLAG
1123 004404' 004577 173436 JSR R5,@CIOBSY ;RELEASE CNTRL UNTIL UNIT IS ON-LINE
1124 004410' 000205 60$: RTS R5 ;EXIT TO USER PROG
    
```

```

1126                                     ;ACQUIRE DISK AND HOUSEKEEP IT
1127
1128                                     ;JSR    PC,ACQHSK        S/R CALL
1129
1130                                     ;R5 = ADR AFTER USER PROG JSR
1131                                     ;R4 = RPCS1 ADR
1132                                     ;R3 = PROG TBL ADR
1133
1134                                     ;DESTROYS R0
1135
1136 004412' 010146 ACQHSK: MOV R1, -(SP) ;SAVE R1
1137 004414' 005063 000030 ACQRTY: CLR PTOCNT(R3) ;SET T/O CNT TO MAX VALUE
1138 004420' 042767 000174 173354 BIC #SWOVER+SWOIER+SWOVTO+UNLDIP+CMDISU,DFLGWD ;MSKP FLAG BITS
1139 004426' 116300 000035 MOV PCURDV(R3),R0 ;GET MY UNIT #
1140 004432' 020027 000007 CMP R0,#7 ;VALID UNIT #'
1141 004436' 101411 BLOS 10$ ;N,Y-10$
1142 004440' 012767 003663 000446 MOV #INVDVN-ERMBAS,ACQEAD ;SET UP ADR OF INV UNIT # ERR MSG
1143 004446' 005267 175644 INC DATAER ;ADJ ERROR COUNTS
1144 004452' 005367 175634 DEC ERRCNT
1145 004456' 000167 000422 JMP ACQERR ;GO REPORT THE ERROR
1146 004462' 010001 10$: MOV R0,R1 ;GET DISPLACEMENT INTO
1147 004464' 006301 ASL R1 ;THE ATA TABLE FOR
1148 004466' 060701 ADD PC,R1 ;THIS UNIT #
1149 004470' 062701 000446 ADD #ATATBL-. ,R1
1150 004474' 112167 000456 MOVB (R1)+,MYATA ;STORE ATA BIT MASKS FOR
1151 004500' 111167 000454 MOVB (R1),OTHATA ;THIS UNIT #
1152 004504' 056700 175664 BIS RPCS2V,R0 ;SET PAT & BAI BITS IN UNIT #
1153 004510' 005001 CLR R1 ;MSKP 1ST TIME FLAG
1154 004512' 010064 000010 12$: MOV R0,RPCS2(R4) ;MOVE UNIT # TO RH11
1155 004516' 005714 TST (R4) ;SELECT THE DRIVE
1156 004520' 032764 010000 000010 BIT #NED,RPCS2(R4) ;NON-EXISTENT DRIVE?
1157 004526' 001413 BEQ 20$ ;Y,N-20$
1158 004530' 005701 TST R1 ;FIRST TIME?
1159 004532' 001005 BNE 16$ ;Y,N-16$
1160 004534' 005201 INC R1 ;RESET FIRST TIME
1161 004536' 112764 000100 000001 MOVB #100,1(R4) ;DO RH11 ERROR CLEAR
1162 004544' 000762 BR 12$ ;GO LOAD UNIT # AGAIN
1163 004546' 012767 003360 000340 16$: MOV #NONEXD-ERMBAS,ACQEAD ;SET UP ADR OF NON-EXIST DRIVE ERR MSG
1164 004554' 000553 BR ACQERR ;GO REPORT THE ERROR
1165 004556' 005064 000012 20$: CLR RPODS(R4) ;WRITE TO A DRIVE REG
1166 004562' 032714 004000 BIT #DVA,(R4) ;DRIVE AVAILABLE SET?
1167 004566' 001035 BNE GOTDSK ;N,Y-GOTDSK
1168 004570' 112764 000100 000001 MOVB #100,1(R4) ;MSKP ANY RH11 ERRORS
1169 004576' 052713 000010 BIS #WT410T,(R3) ;SET WAITING FOR I/O TERM FLAG
1170 004602' 052767 000010 173172 BIS #SWOVER,DFLGWD ;SET WAITING FOR SWITCH OVER FLAG
1171 004610' 112714 000101 MOVB #101,(R4) ;ISSUE NOP & SET INT ENABLE
1172 004614' 004577 173226 JSR R5,#ACIOBSY ;RELEASE CONTROL UNTIL DISK IS SEIZED
1173 004620' 032767 000020 173154 BIT #SWOIER,DFLGWD ;ERROR IN INT DURING SWITCH OVER?
1174 004626' 001007 BNE 30$ ;N,Y-30$
1175 004630' 032767 000040 173144 BIT #SWOVTO,DFLGWD ;TIME OUT ERROR ON SWITCH OVER?
1176 004636' 001133 BNE ACQGDK ;N,Y-ACQGDK
1177 004640' 112714 000001 MOVB #1,(R4) ;RESET INT FNABLE
1178 004644' 000744 BR 20$ ;GO MAKE SURE WE STILL HAVE IT
1179 004646' 010146 30$: MOV R1,-(SP) ;SAVE R1 & R2
1180 004650' 010246 MOV R2,-(SP)
1181 004652' 004567 003052 JSR R5,ERRIS ;REPORT INT'S ERROR MSG
    
```

# M02

1182	004656'	003531			ACQIAD: .WORD	NOATA-ERMBAS	
1183	004660'	000516			BR	ACQERC	; GO TO COMMON ERROR RETURN
1184	004662'	042767	000030	173112	GOTDSK: BIC	#SWOVER+SWOIER,DFLGWD	; RESET SWITCH OVER FLAGS
1185	004670'	005063	000030		CLR	PTCNT(R3)	; HSKP T/O CNT
1186	004674'	032764	010000	000012	BIT	#MOL,RPDS(R4)	; DISK ON-LINE?
1187	004702'	001004			BNE	40\$	; N,Y-40\$
1188	004704'	012767	003436	000202	MOV	#OFFLIN-ERMBAS,ACQEAD	; SET UP ADR OF DISK OFFLINE ERR MSG
1189	004712'	000474			BR	ACQERR	; GO REPORT THE ERROR
1190	004714'	005001			40\$: CLR	R1	; RESET ERROR LOOP CNT
1191	004716'	005714			50\$: TST	(R4)	; IS 'SC' ERROR BIT SET?
1192	004720'	100062			BPL	100\$	; Y,N-100\$
1193	004722'	032764	040000	000014	BIT	#UNS,RPER1(R4)	; IS THERE AN UNSAFE ERROR?
1194	004730'	001033			BNE	80\$	; N,Y-80\$
1195	004732'	020127	000065		CMP	R1,#5	; 5TH TIME THRU ON THIS ERROR?
1196	004736'	001004			BNE	60\$	; Y,N-60\$
1197	004740'	012767	003412	000146	MOV	#INITDE-ERMBAS,ACQEAD	; SET UP ADR OF INITIATION ERR MSG
1198	004746'	000456			BR	ACQERR	; GO REPORT THE ERROR
1199	004750'	005201			60\$: INC	R1	; ADD 1 TO ERROR LOOP CNT
1200	004752'	016400	000016		MOV	RPAS(R4),R0	; GET ATA REG
1201	004756'	105700			TSTB	R0	; IS THERE AN ATTENTION?
1202	004760'	001432			BEQ	90\$	; Y,N-90\$
1203	004762'	036700	000172		BIT	OTHATA,R0	; MY ATA ONLY?
1204	004766'	001427			BEQ	90\$	; N,Y-90\$
1205	004770'	036700	000162		BIT	MYATA,R0	; MY ATA LINE ALSO?
1206	004774'	001006			BNE	70\$	; N,Y-70\$
1207	004776'	010064	000016		MOV	R0,RPAS(R4)	; RESET OTHER ATA LINES
1208	005002'	112764	000100	000001	MOVB	#100,1(R4)	; DO RH11 ERROR CLEAR
1209	005010'	000742			BR	50\$	; GO CK 'SC' AGAIN
1210	005012'	010064	000016		70\$: MOV	R0,RPAS(R4)	; RESET OTHER ATA LINES
1211	005016'	000413			BR	90\$	; GO CLEAR MY DRIVE
1212	005020'	005767	172776		80\$: TST	ERRI	; ERROR ON PREVIOUS I/O?
1213	005024'	001010			BNE	90\$	; N,Y-90\$
1214	005026'	032767	0002	172746	BIT	#ANYIOI,DFLGWD	; ANY I/O CMDS BEEN ISSUED?
1215	005034'	001404			BEQ	90\$	; Y,N-90\$
1216	005036'	012767	003403	000050	MOV	#INITUS-ERMBAS,ACQEAD	; SET UP ADR OF UNSAFE ERROR MSG
1217	005044'	000417			BR	ACQERR	; GO REPORT THE ERROR
1218	005046'	005067	172750		90\$: CLR	ERRI	; RESET PREV I/O ERR FLAG
1219	005052'	052767	000200	172722	BIS	#ANYIOI,DFLGWD	; SET FLAG TO PREVENT LOOP
1220	005060'	012714	040011		MOV	#TRE+011,(R4)	; DO RH11 & DRIVE CLEAR
1221	005064'	000714			BR	50\$	; GO CK 'SC' AGAIN
1222	005066'	005067	172730		100\$: CLR	ERRI	; HSKP ERROR INDICATOR
1223	005072'	042767	000001	172702	BIC	#IOERR,DFLGWD	; & FLAGS
1224	005100'	012601			MOV	(SP)+,R1	; RESTORE R1
1225	005102'	000207			RTS	PC	; EXIT IN-LINE
1226							
1227	005104'	010146			ACQERR: MOV	R1,-(SP)	; SAVE R1 & R2
1228	005106'	010246			MOV	R2,-(SP)	
1229	005110'	004567	002570		JSR	R5,ERRCS	; STORE CURR STATUS & REPORT
1230	005114'	003412			ACQEAD: .WORD	INITDE-ERMBAS	; THE ERROR
1231	005116'	012602			ACQERC: MOV	(SP)+,R2	; RESTORE R1 & R2
1232	005120'	012601			MOV	(SP)+,R1	
1233	005122'	004577	172722		JSR	R5,ACUPGER	; GO TO MPG'S ERROR RETURN POINT
1234	005126'	004767	002202		ACQGDK: JSR	PC,CKDBSY	; GO CK IF DEV IS BUSY NOW
1235	005132'	000167	177256		JMP	ACQRTY	; GO TRY AGAIN
1236							
1237							

1238	005136'	001	376	ATATBL:	.BYTE	001,376
1239	005140'	002	375		.BYTE	002,375
1240	005142'	004	373		.BYTE	004,373
1241	005144'	010	367		.BYTE	010,367
1242	005146'	020	357		.BYTE	020,357
1243	005150'	040	337		.BYTE	040,337
1244	005152'	100	277		.BYTE	100,277
1245	005154'	200	177		.BYTE	200,177
1246						
1247	005156'	000000		MYATA:	.WORD	0
1248	005160'	000000		OTHATA:	.WORD	0



```

1298                                     ;"WRHD" FUNCTION ROUTINE
1299                                     ;JSR    R5,WRHD      FUNCTION CALL
1300                                     ;..WORD ADR      DATA ADDRESS (BITS 16 - 21)
1301                                     ;..WORD ADR      DATA ADDRESS (BITS 0 - 15)
1302                                     ;..WORD CNT      BYTE COUNT
1303
1304
1305 005254' 012702 000163      WRHD:  MOV    #163,R2      ;SET UP WRHD CMD CODE
1306 005260' 012701 000236      MOV    #236,R1      ;SET UP CMD FLAG WORD
1307 005264' 000756      BR     WRCOM        ;GO TO WRITE COMMON PROCESSING
1308
1309
1310                                     ;"WRCK" FUNCTION ROUTINE
1311                                     ;JSR    R5,WRCK      FUNCTION CALL
1312                                     ;..WORD ADR      DATA ADDRESS (BITS 16 - 21)
1313                                     ;..WORD ADR      DATA ADDRESS (BITS 0 - 15)
1314                                     ;..WORD CNT      BYTE COUNT
1315
1316
1317 005266' 012702 000151      WRCK:  MOV    #151,R2      ;SET UP WRCK CMD CODE
1318 005272' 012701 000236      CKCOM: MOV    #236,R1      ;SET UP CMD FLAG WORD
1319 005276' 004767 002032      JSR    PC,CKDBSY    ;GO CK IF DEV IS BUSY
1320 005302' 005267 174776      INC    CKCNT        ;ADD 1 TO CHECK CMD COUNT
1321 005306' 010700      MOV    PC,R0        ;SET UP ADR OF BYTES
1322 005310' 062700 174766      ADD    #BYCK+2-.,R0 ;CHECKED COUNT
1323 005314' 000440      BR     CMDCOM       ;GO TO CMD COM PROCESSING
1324
1325
1326                                     ;"WRCKHD" FUNCTION ROUTINE
1327                                     ;JSR    R5,WRCKHD   FUNCTION CALL
1328                                     ;..WORD ADR      DATA ADDRESS (BITS 16 - 21)
1329                                     ;..WORD ADR      DATA ADDRESS (BITS 0 - 15)
1330                                     ;..WORD CNT      BYTE COUNT
1331
1332
1333 005316' 012702 000153      WRCKHD:MOV    #153,R2      ;SET UP WRCKHD CMD CODE
1334 005322' 000763      BR     CKCOM        ;GO TO CHECK CMD COM PROCESSING
1335
1336
1337                                     ;"SEEK" FUNCTION ROUTINE
1338                                     ;JSR    R5,SEEK     FUNCTION CALL
1339
1340
1341 005324' 012702 000105      SEEK:  MOV    #105,R2      ;SET UP SEEK CMD CODE
1342 005330' 012701 000060      MOV    #060,R1      ;SET UP CMD FLAG WORD
1343 005334' 004767 001774      SKCOM: JSR    PC,CKDBSY    ;GO CK IF DEV IS BUSY
1344 005340' 005267 174742      INC    SKCNT        ;ADD 1 TO SEEK CMD COUNT
1345 005344' 000424      BR     CMDCOM       ;GO TO CMD COMMON PROCESSING

```

```

1347                                     ;"SEARCH" FUNCTION ROUTINE
1348
1349                                     ;JSR   RS,SEARCH           FUNCTION CALL
1350
1351 005346' 012702 000131   SEARCH: MOV   #131,R2           ;SET UP SRCH CMND CODE
1352 005352' 012701 000060   MOV   #060,R1           ;SET UP CMND FLAG WORD
1353 005356' 000766   BR    SKCOM             ;GO TO SEEK CMND COM PROCESSING
1354
1355                                     ;"OFFSET" FUNCTION ROUTINE
1356
1357                                     ;JSR   RS,OFFSET           FUNCTION CALL
1358                                     ;.WORD VALUE           BINARY OFFSET VALUE
1359
1360
1361 005360' 012702 000115   OFFSET: MOV  #115,R2      ;SET UP OFFSET CMND CODE
1362 005364' 012701 000140   MOV  #140,R1            ;SET UP CMND FLAG WORD
1363 005370' 000761   BR    SKCOM             ;GO TO SEEK CMND COM PROCESSING
1364
1365                                     ;"RETCTR" FUNCTION ROUTINE
1366
1367                                     ;JSR   RS,RETCTR          FUNCTION CALL
1368
1369
1370 005372' 012702 000117   RETCTR: MOV  #117,R2     ;SET UP RETURN TO CENTERLINE CMND CODE
1371 005376' 012701 000040   MOV  #040,R1           ;SET UP CMND FLAG WORD
1372 005402' 000754   BR    SKCOM             ;GO TO SEEK CMND COM PROCESSING
1373
1374                                     ;"RECAL" FUNCTION ROUTINE
1375
1376                                     ;JSR   RS,RECAL          FUNCTION CALL
1377
1378
1379 005404' 012702 000107   RECAL: MOV  #107,R2     ;SET UP RECALIBRATE CMND CODE
1380 005410' 012701 000040   MOV  #040,R1           ;SET UP CMND FLAG WORD
1381 005414' 000747   BR    SKCOM             ;GO TO SEEK CMND COM PROCESSING

```



# F03

1439	005600'	012767	003503	001770		MOV	#IOTERM-EP#BAS, INTEAD	; INIT TERMINATION ERROR MSG	
1440	005606'	052767	000002	172166		BIS	#OOTERM,DFLGWD	; SET THE "PROCESS TERMINATION" FLAG	
1441	005614'	052713	000010			BIS	#WT4IOT,(R3)	; SET WAITING FOR I/O TERM FLAG	
1442	005620'	052767	000300	172154		BIS	#CMDISU+ANYIOI,DFLGWD	; SET CMND ISSUED FLAG	
1443	005626'	010214				MOV	R2,(R4)	; ISSUE SPECIFIED CMND	
1444	005630'	005767	172146			TST	DFLGWD	; "NOWAIT" BIT - "	
1445	005634'	100003				BPL	40\$	; Y,N-40\$	
1446	005636'	042713	000010			BIC	#WT4IOT,(R3)	; RESET WAITING FOR I/O TERM	
1447	005642'	000404				BR	50\$	; GO TO EXIT	
1448	005644'	004577	172176		40\$:	JSR	R5,PCIOBSY	; WAIT FOR I/O TO COMPLETE	
1449	005650'	004767	001626			JSR	PC,PROCTM	; GO PROCESS TERMINATION	
1450	005654'	000205			50\$:	RTS	R5	; EXIT IN-LINE TO USER PROG	
1451									
1452									
1453									
1454									
1455									
1456									
1457									
1458									
1459									
1460									
1461									
1462									
1463									
1464									
1465									
1466	005656'	032701	000020			SUIORG:	BIT	#20,R1	; NEED TO SET UP CYL/HEAD/SECT?
1467	005662'	001412					BEQ	10\$	; Y,N-10\$
1468	005664'	016764	172114	000034		MOV	CYL,RPDC(R4)	; LOAD CYL #	
1469	005672'	016746	172110			MOV	HEAD,-(SP)	; GET HEAD #	
1470	005676'	000316				SWAB	(SP)	; PUT IN CORRECT BIT POSITION	
1471	005700'	116716	172104			MOV	SECT,(SP)	; SET IN SECT #	
1472	005704'	012664	000006			MOV	(SP)+,RPDA(R4)	; LOAD HEAD & SECT #'S	
1473	005710'	032701	000004		10\$:	BIT	#4,R1	; DATA XFER CMND?	
1474	005714'	001423				BEQ	30\$	; Y,N-30\$	
1475	005716'	016700	174430			MOV	CURADR,R0	; GET HIGH BITS OF ADR	
1476	005722'	042700	177774			BIC	#177774,R0	; RESET BITS ABOVE A17	
1477	005726'	000300				SWAB	R0	; ALIGN BITS A16 & A17	
1478	005730'	050002				BIS	R0,R2	; SET THEM INTO CMND CODE WORD	
1479	005732'	016764	174416	000004		MOV	CURADR+2,RPBA(R4)	; LOAD BITS 0-15 OF ADR	
1480	005740'	032777	000010	172116		BIT	#CPU70,SCSYFW	; RUNNING ON AN 11/70?	
1481	005746'	001403				BEQ	20\$	; Y,N-20\$	
1482	005750'	016764	174376	000050		MOV	CURADR,RPBAE(R4)	; MOVE BITS A16-A21 TO ADR EXT	
1483	005756'	016764	174374	000002	20\$:	MOV	CURCNT,RPWC(R4)	; LOAD WORD COUNT	
1484	005764'	000207			30\$:	RTS	PC	; EXIT IN-LINE	

;SET UP DEVICE REGS FOR I/O

```

;JSR    PC,SUIORG      S/R CALL
;
;R4 = RPCS1 ADR
;R3 = PROG TBL ADR
;R2 = CMND CODE
;R1 = CMND FLAGWORD
;DESTROYS R0
  
```

```

;NEED TO SET UP CYL/HEAD/SECT?
;Y,N-10$
;LOAD CYL #
;GET HEAD #
;PUT IN CORRECT BIT POSITION
;SET IN SECT #
;LOAD HEAD & SECT #'S
;DATA XFER CMND?
;Y,N-30$
;GET HIGH BITS OF ADR
;RESET BITS ABOVE A17
;ALIGN BITS A16 & A17
;SET THEM INTO CMND CODE WORD
;LOAD BITS 0-15 OF ADR
;RUNNING ON AN 11/70?
;Y,N-20$
;MOVE BITS A16-A21 TO ADR EXT
;LOAD WORD COUNT
;EXIT IN-LINE
  
```

```

1486          .SBTTL  RPO4/RPO5/RPO6 INTERRUPT SERVICE ROUTINE
1487
1488
1489      005766' 004067 003346      RHPINT: JSR      RD, SAVREG      ;SAVE ALL REGISTERS
1490      005772' 004567 003414      JSR      RS, STSTAT    ;GO TO GET ALL DEV REG'S
1491      005776' 174136      .WORD    ISTAT-
1492      006000' 005267 174334      INC      INT CNT      ;ADD 1 TO INTERRUPT CNT
1493      006004' 004767 003362      JSR      PC, SUPTAD    ;SET UP PROG TBL & RPCS1 ADR'S
1494      006010' 032767 000010 171764  BIT      #SWOVR, DFLGWD ;WAITING FOR SWITCH OVER INT?
1495      006016' 001457      BEQ      TRMINT        ;Y, N-TRMINT
1496      006020' 016400 000016      MOV      RPAR(R4), RD  ;GET ATA REG CONTENTS
1497      006024' 105700      TSTB     RD            ;ANY ATA LINES SET?
1498      006026' 001011      BNE     10$           ;N, Y-10$
1499      006030' 012767 003531 176620  MOV      #NOATA-ERMBAS, ACQIAD ;SET UP ADR OF NO ATA ON INT ERR
1500      006036' 042713 000010 5$:     BIC      #WT4IOT, (R3)  ;RESET WAITING FOR I/O TERM FLAG
1501      006042' 052767 000020 171732  BIS      #SWOIER, DFLGWD ;SET ERROR ON SWITCH OVER INT
1502      006050' 000470      BR       INTEX         ;GO TO INT EXIT
1503      006052' 036700 177100 10$:     BIT      MYATA, RD      ;IS IT MY ATA?
1504      006056' 001003      BNE     15$           ;N, Y-15$
1505      006060' 010064 000016      MOV      RD, RPAR(R4)  ;RESET OTHER ATA
1506      006064' 000431      BR       25$          ;GO SET INT ENABLE
1507      006066' 032764 100000 000012 15$:     BIT      #ATA, RPAR(R4) ;DO I HAVE THE DISK?
1508      006074' 001004      BNE     17$           ;N, Y-17$
1509      006076' 116764 177054 000016      MOVB     MYATA, RPAR(R4) ;RESET MY ATA LINE
1510      006104' 000421      BR       25$          ;GO SET INT ENB & EXIT
1511      006106' 036700 177046 17$:     BIT      OTHATA, RD    ;OTHER ATA LINES SET?
1512      006112' 001402      BEQ     18$           ;Y, N-18$
1513      006114' 010064 000016      MOV      RD, RPAR(R4)  ;RESET OTHER ATA
1514      006120' 032764 000400 000012 18$:     BIT      #DPR, RPAR(R4) ;DRIVE PRESENT SET?
1515      006126' 001004      BNE     20$           ;N, Y-20$
1516      006130' 012767 003457 176520  MOV      #DPNSET-ERMBAS, ACQIAD ;SET UP ADR OF DRV NOT PRES ERR MSG
1517      006136' 000737      BR       5$           ;GO CLEAR FLAGS
1518      006140' 112714 000011 20$:     MOVB     #011, (R4)    ;RESET MY ATA WITH DRIVE CLEAR
1519      006144' 042713 000010      BIC      #WT4IOT, (R3)  ;RESET WAITING FOR I/O TERM FLAG
1520      006150' 112714 000011 25$:     MOVB     #101, (R4)    ;SET INT ENABLE WITH NOP CMND
1521      006154' 000426      BR       INTEX         ;GO TO INT EXIT
1522
1523      006156' 016701 174164      TRMINT: MOV      CURFLG, R1 ;GET THIS CMND'S FLGWD
1524      006162' 005714      TST      (R4)         ;IS 'SC' BIT SET?
1525      006164' 100426      BMI     CKSC          ;N, Y-CKSC
1526      006166' 032701 000040      BIT      #40, R1      ;CMND SUPPOSED TO SET ATA?
1527      006172' 001406      BEQ     CLRWTF        ;Y, N-CLRWTF
1528      006174' 012767 003531 001374  MOV      #NOATA-ERMBAS, INTEAD ;SET UP NO ATA ERR MSG ADR
1529      006202' 052767 000001 171572  SETERR: BIS      #IOERR, DFLGWD ;SET THE TERMINATION I/O ERR FLAG
1530      006210' 042713 000010  CLRWTF: BIC      #WT4IOT, (R3)  ;RESET WAITING FOR I/O TERM
1531      006214' 032767 000004 174124  BIT      #4, CURFLG    ;THIS A DATA TRANSFER CMND?
1532      006222' 001403      BEQ     INTEX         ;Y, N-INTEX
1533      006224' 016467 000002 174130  MOV      RPAR(R4), FINCNT ;SAVE FINAL WORD COUNT
1534      006232' 004067 003116      INTEX: JSR      RD, RESREG ;RESTORE ALL REGISTERS
1535      006236' 000177 171632      JMP      @RTNINT      ;EXIT FROM INTERRUPT
1536
1537      006242' 032714 020000      CKSC:   BIT      #MCPE, (R4) ;MCPE ERROR SET?
1538      006246' 001402      BEQ     28$           ;N, Y-HARDER
1539      006250' 000167 000400      JMP      HARDER
1540      006254' 032714 040000      28$:   BIT      #TRE, (R4)  ;TRE ERROR BIT SET?
1541      006260' 001031      BNE     ERRFND        ;N, Y-ERRFND
    
```

# H03

1542	006262'	032764	040000	000012		BIT	#ERR, RPOS(R4)	: ERROR SUMMARY BIT SET?
1543	006270'	001025				BNE	ERRFND	: N, Y-ERRFND
1544	006272'	016400	000016			MOV	RPAS(R4), RO	: GET ATA REG
1545	006276'	036700	176654			BIT	MYATA, RO	: MY ATA BIT SET?
1546	006302'	001011				BNE	40\$	: N, Y-40\$
1547	006304'	010064	000016			MOV	RO, RPAS(R4)	: RESET OTHER ATA BITS
1548	006310'	032701	000004			BIT	#4, R1	: DATA XFER CMND?
1549	006314'	001335				BNE	CLRWTF	: N, Y-CLRWTF
1550	006316'	036764	176634	000016	30\$:	BIT	MYATA, RPAS(R4)	: MY ATA SET?
1551	006324'	001774				BEQ	30\$	: Y, N-30\$
1552	006326'	032701	000040		40\$:	BIT	#40, R1	: CMND SUPPOSED TO SET ATA?
1553	006332'	001326				BNE	CLRWTF	: N, Y-CLRWTF
1554	006334'	012767	003551	001234		MOV	#UXPATA-ERMBAS, INTEAD	: SET UP UNEXP ATA ERR MSG ADR
1555	006342'	000717				BR	SETERR	: GO TO ERROR EXIT
1556								
1557	006344'	032764	037400	000010	ERRFND:	BIT	#037400, RPCS2(R4)	: UPE/NED/NEP/PGE/MXF/MOPE
1558	006352'	001140				BNE	HARDER	: IN RPCS2? (N, Y-HARDER)
1559	006354'	032764	067057	000014		BIT	#067057, RPER1(R4)	: UNS/OPI/MLE/IAE/AOE/WCF/PAR/RMR/
1560	006362'	001134				BNE	HARDER	: ILR/ILF IN RPER1? (N, Y-HARDER)
1561	006364'	032764	137777	000040		BIT	#137777, RPER2(R4)	: ACU/PLU/30VU/IXE/NHS/MHS/WRU/ABS/
1562								: FEN/TUF/TDF/RAW/MSE/CSU/MSU/CSF/
1563	006372'	001130				BNE	HARDER	: WCU IN RPER2? (N, Y-HARDER)
1564	006374'	032764	160153	000042		BIT	#160153, RPER3(R4)	: OCYL/SKI/OPE/DCL/ACL/UMR/WAO/VUF/
1565	006402'	001124				BNE	HARDER	: DCU/PSU IN RPER3? (N, Y-HARDER)
1566	006404'	032764	140000	000010		BIT	#140000, RPCS2(R4)	: DLT OR WCE IN RPCS2?
1567	006412'	001004				BNE	CKRTRY	: N, Y-CKRTRY
1568	006414'	032764	010720	000014		BIT	#010720, RPER1(R4)	: DTE/HCRC/HCE/ECH/FER IN RPER1?
1569	006422'	001524				BEQ	CKCORR	: Y, N-CKCORR
1570	006424'	005767	173736		CKRTRY:	TST	RTRYIP	: ALREADY DONE RETRIES ON THIS CMND?
1571	006430'	001063				BNE	55\$	: N, Y-55\$
1572	006432'	005767	173726			TST	CURRTY	: ARE RETRIES SPECIFIED?
1573	006436'	001506				BEQ	HARDER	: Y, N-HARDER
1574	006440'	032764	100000	000010		BIT	#100000, RPCS2(R4)	: DLT ERROR?
1575	006446'	001403				BEQ	42\$	: Y, N-42\$
1576	006450'	005267	173644			INC	DLTCNT	: ADD 1 TO DLT COUNT
1577	006454'	000451				BR	55\$	: GO CK RETRY COUNT
1578	006456'	032764	010000	000014	42\$:	BIT	#10000, RPER1(R4)	: DTE ERROR?
1579	006464'	001403				BEQ	44\$	: Y, N-44\$
1580	006466'	005267	173630			INC	DTECNT	: ADD 1 TO DTE COUNT
1581	006472'	000442				BR	55\$	: GO CK RETRY COUNT
1582	006474'	032764	000400	000014	44\$:	BIT	#400, RPER1(R4)	: HCRC ERROR?
1583	006502'	001403				BEQ	46\$	: Y, N-46\$
1584	006504'	005267	173614			INC	HCRCNT	: ADD 1 TO HCRC COUNT
1585	006510'	000433				BR	55\$	: GO CK RETRY COUNT
1586	006512'	032764	000020	000014	46\$:	BIT	#20, RPER1(R4)	: FER ERROR?
1587	006520'	001403				BEQ	48\$	: Y, N-48\$
1588	006522'	005267	173600			INC	FERCNT	: ADD 1 TO FER COUNT
1589	006526'	000424				BR	55\$	: GO CK RETRY COUNT
1590	006530'	032764	000200	000014	48\$:	BIT	#200, RPER1(R4)	: HCE ERROR?
1591	006536'	001403				BEQ	50\$	: Y, N-50\$
1592	006540'	005267	173564			INC	HCECNT	: ADD 1 TO HCE COUNT
1593	006544'	000415				BR	55\$	: GO CK RETRY COUNT
1594	006546'	032764	100000	000014	50\$:	BIT	#100000, RPER1(R4)	: DCK ERROR?
1595	006554'	001403				BEQ	52\$	: Y, N-52\$
1596	006556'	005267	173550			INC	DCKCNT	: ADD 1 TO DCK COUNT
1597	006562'	000406				BR	55\$	: GO CK RETRY COUNT

1598	006564'	032764	040000	000010	52\$:	BIT	#40000,RPCS2(R4)	;WCE ERROR?
1599	006572'	001402				BEQ	55\$	;Y,N-55\$
1600	006574'	005267	173534			INC	WCECNT	;ADD 1 TO WCE COUNT
1601	006580'	003367	173560		55\$:	DEC	CURRTY	;DECREMENT RETRY COUNT
1602	006604'	100004				BPL	60\$	;CNT EXHAUSTED? (Y,N-60\$)
1603	006606'	012767	003570	000762		MOV	#RTYEXH-ERMBAS,INTEAD	;SET UP EXHAUSTED RETRIES ERR MSG ADR
1604	006614'	000425				BR	JSETER	;GO TO ERROR EXIT
1605	006616'	005267	173514		60\$:	INC	RETRY5	;ADD 1 TO RETRY TOTAL CNT
1606	006622'	005267	173540			INC	RTRYIP	;SET RETRY IN PROGRESS FLAG
1607	006626'	012714	040011			MOV	#40011,(R4)	;ISSUE RH11 & DRIVE CLEAR
1608	006632'	016702	173512			MOV	CURCMD,R2	;GET CURR CMND IN R2
1609	006636'	004767	177014			JSR	PC,SUIORG	;SET UP DEV REGS
1610	006642'	005063	000030			CLR	PTOCNT(R3)	;HSKIP T/O COUNT
1611	006646'	010214				MOV	R2,(R4)	;RE-ISSUE THE ORIG CMND
1612	006650'	000167	177356		JINTEX:	JMP	INTEX	;GO TO INT EXIT
1613								
1614	006654'	032764	040000	000014	HARDER:	BIT	#UNS,RPER1(R4)	;IS THE UNSAFE BIT SET?
1615	006662'	001402				BEQ	JSETER	;Y,N-SETERR
1616	006664'	112714	000011			MOVB	#011,(R4)	;ISSUE DRIVE CLEAR FOR UNSAFE COND
1617	006670'	000167	177306		JSETER:	JMP	SETERR	;GO TO ERROR EXIT
1618								
1619	006674'	032764	100000	000014	CKCORR:	BIT	#100000,RPER1(R4)	;DCK SHD BE SET - IS IT?
1620	006702'	001764				BEQ	HARDER	;Y,N-HARDER (NO ERROR BITS SET)
1621	006704'	032767	002000	171070		BIT	#CORFLG,DFLGWD	;CORON STATEMENT IN EFFECT?
1622	006712'	001244				BNE	CKRTRY	;Y,N-CKRTRY
1623	006714'	032764	010000	000032		BIT	#10000,RPOF(R4)	;IN 18 BIT WORD MODE?
1624	006722'	001640				BEQ	CKRTRY	;N,Y-CKRTRY
1625	006724'	032764	004000	000032		BIT	#4000,RPOF(R4)	;IS THE ECI BIT SET?
1626	006732'	001234				BNE	CKRTRY	;N,Y-CKRTRY
1627	006734'	005764	000046			TST	RPEC2(R4)	;ECC BIT PATTERN = 0?
1628	006740'	001004				BNE	70\$	;Y,N-70\$
1629	006742'	012767	003612	000626		MOV	#INVPAT-ERMBAS,INTEAD	;SET UP INV BIT PATTERN ERR MSG ADR
1630	006750'	000747				BR	JSETER	;GO TO ERROR EXIT
1631	006752'	026427	000044	010041	70\$:	CMP	RPEC1(R4),#10041	;ECC BIT POSITION TOO BIG?
1632	006760'	101404				BLOS	80\$	;Y,N-80\$
1633	006762'	012767	003636	000606		MOV	#INVPOS-ERMBAS,INTEAD	;SET UP INV BIT POSITION ERR MSG ADR
1634	006770'	000737				BR	JSETER	;GO TO ERROR EXIT
1635	006772'	016405	000002		80\$:	MOV	RPWC(R4),R5	;GET REMAINING NEG WORD CNT
1636	006776'	006305				ASL	R5	;MAKE IT A BYTE CNT
1637	007000'	066705	173354			ADD	CURPBC,R5	;ADD IN ORIG POS BYTE CNT
1638	007004'	001607				BEQ	CKRTRY	;ANY BYTES READ? (Y,N-CKRTRY)
1639	007006'	005001				CLR	R1	;RESET DATA START DISPL VALUE
1640	007010'	012746	001000			MOV	#512,-(SP)	;INITIALIZE BLOCK LENGTH
1641	007014'	032767	001000	173324		BIT	#1000,CURFLG	;READ HEADER AND DATA CMND?
1642	007022'	001402				BEQ	85\$	;Y,N-85\$
1643	007024'	062716	000010			ADD	#8,(SP)	;ALLOW FOR HEADER WORDS
1644	007030'	020516			85\$:	CMP	R5,(SP)	;REACHED START OF BAD BLOCK?
1645	007032'	101403				BLOS	90\$	;N,Y-90\$
1646	007034'	061601				ADD	(SP),R1	;ADD BLK LNGTH TO DATA DISPL
1647	007036'	161605				SUB	(SP),R5	;REDUCE BYTE READ CNT
1648	007040'	000773				BR	85\$	;GO CK IT NOW
1649	007042'	162716	001000		90\$:	SUB	#512,(SP)	;REMOVE CNT FOR DATA WORDS
1650	007046'	062601				ADD	(SP)+R1	;ALLOW FOR HEADER WDS IF NEEDED
1651	007050'	016400	000044			MOV	RPEC1(R4),R0	;GET ECC BIT POSITION VALUE
1652	007054'	005300				DEC	R0	;ADJ IT FOR SHIFTING
1653	007056'	010002				MOV	R0,R2	;SET UP A SHIFT COUNT

```

1654 007060' 042702 177760      BIC      #177760,R2      ; ISOLATE SHIFT CNT BITS
1655 007064' 040200      BIC      R2,R0        ; CLEAR SHIFT CNT BITS IN BIT POSITION
1656 007066' 006200      ASR      R0           ; CONVERT BIT POSITION TO A WORD
1657 007070' 006200      ASR      R0           ; DISPL INTO THE DATA IN THIS
1658 007072' 006200      ASR      R0           ; BLOCK
1659 007074' 060001      ADD      R0,R1        ; ADD IT TO DATA DISPL VALUE
1660 007076' 020167 173256      CMP      R1,CURPBC    ; CORRECTION WITHIN USER'S DATA?
1661 007102' 103026      BHIS    110$         ; Y,N-110$
1662 007104' 005267 173204      INC      CECCER      ; ADD 1 TO CORRECTABLE ECC ERROR CNT
1663 007110' 016405 000046      MOV      RPEC2(R4),R5 ; GET ECC BIT PATTERN
1664 007114' 005046      CLR      -(SP)       ; CLEAR SHIFT INTO WORD
1665 007116' 005302      95$:   DEC      R2      ; DECREMENT SHIFT COUNT
1666 007120' 002403      BLT     100$        ; FINISHED SHIFTING? (N,Y-100$)
1667 007122' 006305      ASL     R5          ; DO A BIT SHIFT ON 2 WORDS
1668 007124' 006116      ROL     (SP)
1669 007126' 000773      BR      95$
1670 007130' 004767 000052      100$:  JSR     PC,DOCORR   ; GO CK SHIFT CNT
1671 007134' 012605      MOV     (SP)+,R5     ; CORRECT 1ST WORD OF PAIR
1672 007136' 062701 000002      ADD     #2,R1        ; GET 2ND CORRECTION WORD
1673 007142' 020167 173212      CMP     R1,CURPBC    ; INCR DATA DISPL VALUE
1674 007146' 103002      BHIS    105$        ; STILL WITHIN HIS DATA?
1675 007150' 004767 000032      JSR     PC,DOCORR   ; Y,N-105$
1676 007154' 004767 002212      105$:  JSR     PC,SUPTAD    ; CORRECT 2ND WORD OF PAIR
1677 007160' 005764 000002      110$:  TST     RPLC(R4)    ; SET UP R3 & R4 AGAIN
1678 007164' 001002      BNE     115$        ; ANY MORE DATA TO READ?
1679 007166' 000167 177016      JMP     CLRWTF       ; Y,N-115$
1680 007172' 112714 000011      115$:  MOVB   #011,(R4)    ; GO CLEAR WAIT FLG & EXIT
1681 007176' 116714 173146      MOVB   CURCMD,(R4)  ; ISSUE DRIVE CLEAR CMND
1682 007202' 000167 177024      JMP     INTX        ; RESUME ORIG CMND
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693 007206' 016703 173140      DOCORR: MOV    CURADR,R3      ; DO ECC DATA CORRECTION
1694 007212' 016704 173136      MOV    CURADR+2,R4   ; JSR PC,DOCORR S/R CALL
1695 007216' 060104      ADD    R1,R4        ; R1 = DATA WORD DISPLACEMENT
1696 007220' 005503      ADC    R3           ; R5 = BIT PATTERN
1697 007222' 032777 000001 170634      BIT    #MMVER,2CSYSFW ; GET DATA'S ABS STARTING ADR
1698 007230' 001424      SEQ    20$         ; ADD DATA DISPL INTO IT
1699 007232' 010400      MOV    R4,R0        ; RUNNING UNDER MEM MGMT?
1700 007234' 042704 177700      BIC    #177700,R4    ; Y,N-20$
1701 007240' 052704 100000      BIS    #P4CONS,R4   ; LOW 16 BITS TO A WORK REG
1702 007244' 012702 000006      MOV    #6,R2        ; ISOLATE UP TO 32 WORD OFFSET
1703 007250' 006203      10$:  ASR    R3           ; SET ADR TO SELECT PAGE 4
1704 007252' 006000      ROR    R0           ; SET UP SHIFT CNT
1705 007254' 005302      DEC    R2           ; SHIFT ADR 1 BIT
1706 007256' 001374      BNE    105$        ; GOT HI 16 BITS IN 1 WORD?
1707 007260' 013746 172350      MOV    @#KPAR4,-(SP) ; Y,N-105$
1708 007264' 013746 172310      MOV    @#KPR4,-(SP) ; SAVE PAGE 4 STUFF
1709 007270' 010037 172350      MOV    R0,@#KPAR4   ; SET PAGE 4 TO USER'S ARREA
    
```

# K03

1710	007274'	012737	077406	172310		MOV	#PDRCON,@#KPD4	
1711	007302'	011400			20\$:	MOV	(R4),R0	:GET DATA WORD READ
1712	007304'	040514				BIC	R5,(R4)	:RESET PATTERN'S BITS IN WORD READ
1713	007306'	040005				BIC	R0,R5	:RESET WORD READ BITS IN PAT WORD
1714	007310'	050514				BIS	R5,(R4)	:SET REMAINING SINGLE BITS BACK IN
1715	007312'	032777	000001	170544		BIT	#MAVER,@CSYSFW	:RUNNING UNDER MEM MGMT?
1716	007320'	001404				BEQ	30\$	:Y N-30\$
1717	007322'	012637	172310			MOV	(SP)+,@#KPD4	:RESTORE PAGE 4'S REGS
1718	007326'	012637	172350			MOV	(SP)+,@#KPAR4	
1719	007332'	000207			30\$:	RTS	PC	:EXIT IN-LINE

```

1721 .SBTTL SUBROUTINES FOR RPO4/RPO5/RPO6 FUNCTION ROUTINES
1722
1723
1724 ;CHECK IF DEVICE IS BUSY AND WAIT IF IT IS
1725
1726 ;JSR PC,CKDBSY S/R CALL
1727
1728 ;DESTROYS R0,R3,R4
1729 ;ON EXIT: R3 = PROG TBL ADR
1730 ; R4 = RPCS1 ADR
1731
1732 007334' 004767 002032 CKDBSY: JSR PC,SUPTAD ;SET UP PROG TBL & RPCS1 ADR'S
1733 007340' 016400 000010 10$: MOV RPCS2(R4),R0 ;GET CURR UNIT #
1734 007344' 010046 MOV R0,-(SP) ;SAVE IT
1735 007346' 042700 177747 BIC #177747,R0 ;RESET UNIT # & OTHER BITS
1736 007352' 156300 000035 BISB PCURDV(R3),R0 ;SET IN MY UNIT #
1737 007356' 110064 000010 MOVB R0,RPCS2(R4) ;SELECT MY DRIVE
1738 007362' 011400 MOV (R4),R0 ;GET RPCS1 REG
1739 007364' 112664 000010 MOVB (SP)+,RPCS2(R4) ;RESTORE ORIG UNIT #
1740 007370' 032700 000100 BIT #100,R0 ;INT ENABLE ON?
1741 007374' 001403 BEQ 20$ ;Y,N-20$
1742 007376' 004577 170444 15$: JSR R5,@CIOBSY ;RELEASE CONTROL
1743 007402' 000756 BR 10$ ;GO CK AGAIN
1744 007404' 032767 000004 170370 20$: BIT #UNLDIP,DFLGWD ;DO I HAVE AN UNLOAD IN PROGRESS?
1745 007412' 001403 BEQ 25$ ;Y,N-25$
1746 007414' 052713 000010 BIS #WT4IOT,(R3) ;SET WAITING FOR I/O TERM
1747 007420' 000766 BR 15$ ;GO RELEASE CONTROL
1748 007422' 032767 000002 170352 25$: BIT #DOTERM,DFLGWD ;HAVE TO PROCESS PREV TERMINATION?
1749 007430' 001403 BEQ 30$ ;Y,N-30$
1750 007432' 004767 000044 JSR PC,PROCTM ;GO PROCESS TERMINATION
1751 007436' 000740 BR 10$ ;GO CK INT ENABLE AGAIN
1752 007440' 016767 170362 000012 30$: MOV IVCTAD,40$ ;STORE INT VECTOR ADR
1753 007446' 016767 170356 000006 MOV PSWD,45$ ;STORE PROC STATUS WORD
1754 007454' 004577 170406 JSR R5,@SETVEC ;GO SET UP INTERRUPT VECTOR
1755 007460' 000000 40$: .WORD XXXX ;INT VECTOR ADR
1756 007462' 000000 45$: .WORD XXXX ;PSW
1757 007464' 176302 .WORD RHPINT- ;REL INT ROUT ADR
1758 007466' 010567 172650 STMADR: MOV R5,ERRADR ;SAVE CURR USER STMT ADR
1759 007472' 162767 000004 172642 SUB #4,ERRADR
1760 007500' 000207 RTS PC ;EXIT IN-LINE
1761
1762
1763 ;PROCESS TERMINATION OF PREVIOUS I/O FUNCTION
1764
1765 ;JSR PC,PROCTM S/R CALL
1766
1767 ;R3 = PROG TABLE ADR
1768
1769 ;DESTROYS R0
1770
1771 007502' 010146 PROCTM: MOV R1,-(SP) ;SAVE R1 & R2
1772 007504' 010246 MOV R2,-(SP)
1773 007506' 042767 000002 170266 BIC #DOTERM,DFLGWD ;RESET PROCESS TERMINATION FLAG
1774 007514' 032767 000010 172624 BIT #10,CURFLG ;INCR BYTE COUNT?
1775 007522' 001417 BEQ 20$ ;Y,N-20$
1776 007524' 016700 172626 MOV CURCNT,R0 ;GET INITIAL WORD CNT
    
```

```

1777 007530' 005400          NEG      RO          ;MAKE IT POSITIVE AGAIN
1778 007532' 016701 172624  MOV      FINCNT,R1  ;GET FINAL WORD CNT
1779 007536' 100001          BPL      10$         ;IS IT NEGATIVE? (Y,N-10$)
1780 007540' 005401          NEG      R1          ;MAKE IT POSITIVE
1781 007542' 160100          10$: SUB      R1,RO        ;SUB REMAINING CNT FROM INITIAL CNT
1782 007544' 006300          ASL      RO          ;MAKE IT A BYTE CNT
1783 007546' 010067 170246  MOV      RO,SIZE    ;STORE # OF BYTES ACTUALLY XFERRED
1784 007552' 016701 172566  MOV      CNTADR,R1  ;GET ADR OF BYTE CNT TOTALS
1785 007556' 060011          ADD      RO,(R1)    ;ADD IN THIS CNT
1786 007560' 005541          ADC      -(R1)     ;UPDATE MOST SIGNF WORD OF CNT
1787 007562' 032767 000001 170212 20$: BIT      #IOERR,DFLGWD ;WAS THERE AN ERROR?
1788 007570' 001412          BEQ      PROCEX      ;Y N-PROCEX
1789 007572' 004567 000132          JSR      R5,ERRIS   ;GO ISSUE I/O TERMINATION
1790 007576' 003503          INTEAD: .WORD  IOTERM-ERMBAS ;ERROR MSG
1791 007600' 004767 000024          JSR      PC,RINTV  ;RESET THE INT VECTOR
1792 007604' 012602          MOV      (SP)+,R2  ;RESTORE R1 & R2
1793 007606' 012601          MOV      (SP)+,R1
1794 007610' 004577 170234          JSR      R5,ACUPGER ;GO TO MPG ERR RETN POINT
1795 007614' 000207          RTS      PC        ;RETURN IN-LINE
1796 007616' 004767 000006          PROCEX: JSR      PC,RINTV ;GO RESET INT VECTOR
1797 007622' 012602          MOV      (SP)+,R2  ;RESTORE R1 & R2
1798 007624' 012601          MOV      (SP)+,R1
1799 007626' 000207          RTS      PC        ;EXIT IN-LINE
1800
1801
1802          ;RESET INTERRUPT VECTOR S/R
1803
1804          ;JSP      PC,RINTV      S/R CALL
1805          ;R3 MUST CONTAIN PROG TBL ADR
1806          ;DESTROYS RO
1807
1808 007630' 004567 000020          RINTV: JSR      R5,TVECT ;GO CK IF I HAVE VECTOR CONTROL
1809 007634' 000406          BR      RINTEX      ;BR IF I DON'T
1810 007636' 016767 170164 000004  MOV      IVCTAD,10$ ;GET CURR INT VECT ADR
1811 007644' 004577 170220          JSR      R5,ACLAVEC ;GO HAVE MPG CLEAR IT
1812 007650' 000000          10$: .WORD  XXXX
1813 007652' 000207          RINTEX: RTS      PC        ;EXIT IN-LINE
1814
1815          ;TEST INTERRUPT VECTOR S/R
1816
1817
1818          ;JSR      R5,TVECT      S/R CALL
1819          ;BR      LABEL          EXECUTED IF NOT SAME
1820          ;R3 MUST CONTAIN PROG TBL ADR
1821          ;DESTROYS RO
1822
1823 007654' 016767 170146 000010  TVECT: MOV      IVCTAD,20$ ;GET CURR INT VECT ADR
1824 007662' 016346 000004          MOV      PFWADR(R3),-(SP) ;STORE FLGWD ADR TO IDENTIFY ME
1825 007666' 004577 170200          JSR      R5,ATSTVEC ;DO I HAVE VECTOR CONTROL?
1826 007672' 000000          20$: .WORD  XXXX      ;MPG WILL TELL ME SINCE I CAN'T
1827 007674' 176072          .WORD  RHPINT-    ;GET AT LOWER MEM IF MEM MGMT i
1828 007676' 000401          BR      TVECTX      ;BR IF I DONT'T HAVE CNTRL
1829 007700' 005725          TST      (R5)+     ;BYPASS BR INST IN S/R CALL
1830 007702' 000205          TVECTX: RTS      R5        ;EXIT IN-LINE

```

```

1832                                     ;ERROR INFORMATION DISPLAY S/R
1833
1834                                     :JSR   RS,ERRCS           S/R CALL FOR CURR STATUS
1835                                     :JSR   RS,ERRCS1        S/R CALL FOR CURR STATUS W/O STORING
1836                                     :JSR   RS,ERRIS        S/R CALL FOR INT STATUS
1837                                     :.WORD MSGADR-ERMBAS   REL ADR OF ERROR MSG
1838
1839                                     :R3 = PROG TABLE ADR
1840                                     :DESTROYS R0,R1,R2
1841
1842 007704' 004567 001502      ERRCS: JSR   RS,STSTAT           ;STOPE CURR STATUS
1843 007710' 172300                                     .WORD CSTAT-
1844 007712' 012767 171606 000462  ERRCS1: MOV   #CSTAT-ERSTAD,ERSTAD   ;STORE ADR OF CURR STATUS
1845 007720' 012767 172052 000212      MOV   #CSTAT-EBSBAS,EBSTAT
1846 007726' 000406                                     BR    ERRCOM
1847 007730' 012767 171532 000444  ERRIS: MOV   #ISTAT-ERSTAD,ERSTAD   ;GO TO COMMON POINT
1848 007736' 012767 171776 000174      MOV   #ISTAT-EBSBAS,EBSTAT   ;STORE ADR OF LAST INT STATUS
1849 007744' 012567 000134      ERRCOM: M.   (RS)+,ERMBAS
1850 007750' 005267 172336                                     .WORD ERRCNT
1851 007754' 012767 000001 170040      MOV   #1,ERRI
1852 007762' 032763 000400 000002      MOV   #DOERCK,POPSW(R3)
1853 007770' 001004      BNE    2$
1854 007772' 032763 020000 000002      BIT   #PRONER,POPSW(R3)
1855 010000' 001402      BEQ    4$
1856 010002' 000167 000462      2$: JMP   ERREX
1857 010006' 010446      4$: MOV   R4,-(SP)
1858 010010' 010546      MOV   R5,-(SP)
1859 010012' 005004      CLR   R4
1860 010014' 004767 001502      JSR   PC,DEVID
1861 010020' 032767 000100 167754      BIT   #CMDISU,DFLGWD
1862 010026' 001005      BNE    6$
1863 010030' 004567 002066      JSR   R5,PRINT
1864 010034' 003270      .WORD BEFIO-
1865 010036' 000030      .WORD 24
1866 010040' 000404      BR    8$
1867 010042' 004567 002054      6$: JSR   R5,PRINT
1868 010046' 003306      .WORD AFTIO-
1869 010050' 000027      .WORD 23
1870 010052' 010700      8$: MOV   PC,R0
1871 010054' 062700 000030      ADD   #ERMBAS--,R0
1872 010060' 061000      ADD   (R0),R0
1873 010062' 012701 177777      MOV   #-1,R1
1874 010066' 005201      10$: INC   R1
1875 010070' 105720      TSTB (R0)+
1876 010072' 001375      BNE   10$
1877 010074' 010167 000006      MOV   R1,ERMBAS+2
1878 010100' 004567 002016      JSR   R5,PRINT
1879 010104' 000000      ERMBAS: .WORD XXXX
1880 010106' 000000      .WORD XXXX
1881 010110' 026727 177770 003663      CMP   ERMBAS,#INVDVN-ERMBAS
1882 010116' 103134      BHS   ERRSNM
1883 010120' 010701      MOV   PC,R1
1884 010122' 062701 003102      ADD   #CODFLD--,R1
1885 010126' 010700      MOV   PC,R0
1886 010130' 062700 000342      ADD   #ERCOTB--,R0
1887 010134' 010702      MOV   PC,R2
    
```

1898	010136	062702			EBSBAS: ADD	(PC)+,R2	
1899	010140	172052			EBSTAT: .WORD	CSTAT-EBSBAS	
1890	010142	012767	000015	000224	MOV	#13,70\$	: INITIALIZE MSG LENGTH
1891	010150	012746	000100		MOV	#64,-(SP)	: INITIALIZE CODE FIELD CNT
1892	010154	012205			15\$: MOV	(R2)+,R5	: GET NEXT DEV REG WORD
1893	010156	000305			17\$: SWAB	R5	: GET DESIRED BYTE IN LOW BYTE
1894	010160	112004			20\$: MOVB	(R0)+,R4	: GET FLAG & LENGTH BYTE
1895	010162	005704			TST	R4	: END OF THE CODE TBL?
1896	010164	001474			BEQ	60\$	: N,Y-60\$
1897	010166	122704	000377		CMPB	#377,R4	: GO TO NXT DEV REG WORD?
1898	010172	001770			BEQ	15\$	: N,Y-15\$
1899	010174	122704	000376		CMPB	#376,R4	: GO TO NXT BYTE IN DEV REG WORD?
1900	010200	001766			BEQ	17\$	: N,Y-17\$
1901	010202	032704	000040		BIT	#40,R4	: THIS AN 11/70 ONLY ERROR BIT?
1902	010206	001405			BEQ	22\$	: Y,N-22\$
1903	010210	032777	000010	167646	BIT	#CPU70,2CSYSFW	: RUNNING ON AN 11/70?
1904	010216	001430			BEQ	35\$	: Y,N-35\$
1905	010220	000417			BR	26\$	: GO CK ERROR BIT
1906	010222	032704	000010	22\$:	BIT	#10,R4	: THIS AN RPO4 ONLY ERROR BIT?
1907	010226	001405			BEQ	24\$	: Y,N-24\$
1908	010230	032763	000060	000032	BIT	#60,PMDCD(R3)	: THIS AN RPO4?
1909	010236	001020			BNE	35\$	: Y,N-35\$
1910	010240	000407			BR	26\$	: GO CK ERROR BIT
1911	010242	032704	000020	24\$:	BIT	#20,R4	: THIS AN RPO5/RPO6 ONLY ERROR BIT?
1912	010246	001404			BEQ	26\$	: Y,N-26\$
1913	010250	032763	000060	000032	BIT	#60,PMDCD(R3)	: THIS AN RPO5/RPO6?
1914	010256	001410			BEQ	35\$	: Y,N-35\$
1915	010260	032704	000100	26\$:	BIT	#100,R4	: BIT VALUE OF 0 = AN ERROR CONDITION?
1916	010264	001403			BEQ	30\$	: Y,N-30\$
1917	010266	131005			BITB	(R0),R5	: THIS BIT RESET IN DEV REG BYTE?
1918	010270	001407			BEQ	40\$	: N,Y-40\$
1919	010272	000402			BR	35\$	: GO TO NXT TBL ENTRY
1920	010274	131005		30\$:	BITB	(R0),R5	: THIS ERROR BIT SET IN DEV REG BYTE?
1921	010276	001004			BNE	40\$	: N,Y-40\$
1922	010300	042704	177770	35\$:	BIC	#177770,R4	: ISOLATE ENTRY LENGTH
1923	010304	060400			ADD	R4,R0	: POINT AT NXT CODE TBL ENTRY
1924	010306	000724			BR	20\$	: GO CK FOR NXT CODE
1925	010310	042704	177770	40\$:	BIC	#177770,R4	: ISOLATE I.D. NAME LENGTH + 1
1926	010314	020416			CMP	R4,(SP)	: ENOUGH ROOM FOR NAME?
1927	010316	101017			BHI	60\$	: Y,N-60\$
1928	010320	060467	000050		ADD	R4,70\$	: ADJ MSG LENGTH FOR NAME
1929	010324	005304			DEC	R4	: ADJ FOR BIT MASK CHAR
1930	010326	005200			INC	R0	: POINT PAST BIT MASK
1931	010330	021627	000100		CMP	(SP),#64.	: FIRST ERROR CODE IN MSG?
1932	010334	001403			BEQ	50\$	: N,Y-50\$
1933	010336	112721	000054		MOVB	#,(R1)+	: MOVE COMMA TO MSG
1934	010342	005316			DEC	(SP)	: ADJ REMAINING ROOM IN MSG
1935	010344	112021		50\$:	MOVB	(R0)+,(R1)+	: MOVE ERROR CODE TO MSG
1936	010346	005316			DEC	(SP)	: ADJ REMAINING ROOM IN MSG
1937	010350	005304			DEC	R4	: MOVED ALL NAME CHARS?
1938	010352	001374			BNE	50\$	: Y,N-50\$
1939	010354	000701			BR	20\$	: GO CK FOR MORE ERROR BITS
1940	010356	005004		60\$:	CLR	R4	: SET USER MODE PRINT
1941	010360	022627	000100		CMP	(SP)+,#64.	: ANY ERROR CODES PUT IN MSG?
1942	010364	001404			BEQ	80\$	: Y,N-80\$
1943	010366	004567	001530		JSR	R5,PRINT	: GO ISSUE ERROR BITS MSG

```

1944 010372' 002614          .WORD  DKEMSG-.
1945 010374' 000116          70$:  .WORD  78.
1946 010376' 004567 001250  80$:  JSR    RS,DISPST      ;DISPLAY DEVICE REG'S
1947 010402' 000000          ERSTAD: .WORD  XXXX
1948 010404' 004767 001442  JSR    PC,PRTIWD          ;DISPLAY CYL,HEAD,SECT VALUES
1949 010410' 016300 000022  ERRSNM: MOV   PSACST(R3),R0 ;GET ADR OF SRC STMTS
1950 010414' 111001          110$: MOVB  (R0),R1         ;SAVE STMT LENGTH
1951 010416' 026067 000004 171716  CMP   4(R0),ERRADR       ;ERROR OCCUR ON THIS STMT?
1952 010424' 001402          BEQ   120$                ;N Y-120$
1953 010426' 060100          ADD   R1,R0              ;POINT AT NXT STMT
1954 010430' 000771          BR   110$                ;GO CK NXT STMT
1955 010432' 005720          120$: TST  (R0)+           ;SET UP ADR OF STMT # DATA
1956 010434' 010701          MOV   PC,R1              ;SET UP DATA OUTPUT ADR
1957 010436' 062701 002542  ADD   #STNUM-,R1
1958 010442' 004577 167414  JSR   RS,DECSA          ;CONVERT IT TO ASCII
1959 010446' 012767 020040 002530  MOV   #20040,STNUM+4     ;SET 2 LOW DIGITS TO SPACES
1960 010454' 004567 001442  JSR   RS,PRINT          ;ISSUE STMT # MSG
1961 010460' 002510          .WORD  STNUMG-.
1962 010462' 177762          .WORD  -14.
1963 010464' 012605          MOV   (SP)+,R5           ;RESTORE R5 & R4
1964 010466' 012604          MOV   (SP)+,R4
1965 010470' 000205          ERREX: RTS              ;EXIT IN-LINE

```

;ERROR MESSAGE CODE TABLE

```

;377 = GO TO NEXT DEVICE REGISTER WORD
;376 = GO TO NEXT DEVICE REGISTER BYTE
;BYTE 0 CONTAINS FLAG BITS & I.D. NAME LENGTH
;BITS 0-2 = LENGTH OF BIT MASK + I.D. NAME
;BIT 3 = RPO4 ONLY ERROR BIT
;BIT 4 = RPO5/RPO6 ONLY ERROR BIT
;BIT 5 = 11/70 ONLY ERROR BIT
;BIT 6 = BIT = 0 IS AN ERROR CONDITION
;BYTE 1 IS THE BIT MASK
;BYTES 2 THRU 7 ARE THE BIT'S ASCII I.D.

```

```

1985 010472' 100003 041523          ERCDTB: .ASCII <003><200>/SC/      ;RPCS1 - BYTE 1
1986 010476' 040004 051124          .ASCII <004><100>/TRE/
1987 010503' 005 046440 050103      .ASCII <005><040>/MCPE/
1988 010511' 104 042010 040526      .ASCII <104><010>/DVA/
1989 010516' 377          .BYTE 377
1990 010517' 377          .BYTE 377
1991 010520' 377          .BYTE 377
1992 010521' 377          .BYTE 377
1993 010522' 100004 046104          .ASCII <004><200>/DLT/      ;RPCS2 - BYTE 1
1994 010527' 004 053500 042503      .ASCII <004><100>/MCE/
1995 010534' 020004 050125          .ASCII <004><040>/UPE/
1996 010541' 004 047020 042105      .ASCII <004><020>/NEO/
1997 010546' 004004 042516          .ASCII <004><010>/NEM/
1998 010553' 004 050004 042507      .ASCII <004><004>/PGE/

```

1999	010560'	001004	054115	106	.ASCII	<004><002>/MXF/	
2000	010565'	005	046401	050104	.ASCII	<005><001>/MOPE/	
	010572'	105					
2001	010573'	377			.BYTE	377	
2002	010574'	100004	052101	101	.ASCII	<004><200>/ATA/	;RPOS - BYTE 1
2003	010601'	004	042500	051122	.ASCII	<004><100>/ERR/	
2004	010606'	020004	044520	120	.ASCII	<004><040>/PIP/	
2005	010613'	104	046420	046117	.ASCII	<104><020>/MOL/	
2006	010620'	000504	050104	122	.ASCII	<104><001>/DPR/	
2007	010625'	376			.BYTE	376	
2008	010626'	100104	051104	131	.ASCII	<104><200>/DRY/	;RPOS - BYTE 0
2009	010633'	103	053100	126	.ASCII	<103><100>/VV/	
2010	010637'	377			.BYTE	377	
2011	010640'	100004	041504	113	.ASCII	<004><200>/DCK/	;RPER1 - BYTE 1
2012	010645'	004	052500	051516	.ASCII	<004><100>/UNS/	
2013	010652'	020004	0 3117	111	.ASCII	<004><040>/OPI/	
2014	010657'	004	0 2020	042524	.ASCII	<004><020>/DTE/	
2015	010664'	004004	046127	105	.ASCII	<004><010>/MLE/	
2016	010671'	004	044404	042501	.ASCII	<004><004>/IAE/	
2017	010676'	001004	047501	105	.ASCII	<004><002>/AOE/	
2018	010703'	005	044001	051103	.ASCII	<005><001>/HCRC/	
	010710'	103					
2019	010711'	376			.BYTE	376	
2020	010712'	100004	041510	105	.ASCII	<004><200>/ACE/	;RPER1 - BYTE 0
2021	010717'	004	042500	044103	.ASCII	<004><100>/ECH/	
2022	010724'	020004	041527	106	.ASCII	<004><040>/MCF/	
2023	010731'	004	043020	051105	.ASCII	<004><020>/FER/	
2024	010736'	004004	040520	122	.ASCII	<004><010>/PAR/	
2025	010743'	004	051004	051115	.ASCII	<004><004>/RMR/	
2026	010750'	001004	046111	122	.ASCII	<004><002>/ILR/	
2027	010755'	004	044401	043114	.ASCII	<004><001>/ILF/	
2028	010762'	377			.BYTE	377	
2029	010763'	376			.BYTE	376	
2030	010764'	100005	052101	033501	.ASCII	<005><200>/ATA7/	;RPA5 - BYTE 0
2031	010772'	040005	052101	033101	.ASCII	<005><100>/ATA6/	
2032	011000'	020005	052101	032501	.ASCII	<005><040>/ATA5/	
2033	011006'	010005	052101	032101	.ASCII	<005><020>/ATA4/	
2034	011014'	004005	052101	031501	.ASCII	<005><010>/ATA3/	
2035	011022'	002005	052101	031101	.ASCII	<005><004>/ATA2/	
2036	011030'	001005	052101	030501	.ASCII	<005><002>/ATA1/	
2037	011036'	000405	052101	030101	.ASCII	<005><001>/ATA0/	
2038	011044'	377			.BYTE	377	
2039	011045'	377			.BYTE	377	
2040	011046'	377			.BYTE	377	
2041	011047'	377			.BYTE	377	
2042	011050'	377			.BYTE	377	
2043	011051'	377			.BYTE	377	
2044	011052'	377			.BYTE	377	
2045	011053'	377			.BYTE	377	
2046	011054'	377			.BYTE	377	
2047	011055'	014	040600	052503	.ASCII	<014><200>/ACU/	;RPER2 - BYTE 1
2048	011062'	020004	046120	125	.ASCII	<004><040>/PLU/	
2049	011067'	015	031420	053060	.ASCII	<015><020>/30VU/	
	011074'	125					
2050	011075'	004	044410	042530	.ASCII	<004><010>/IXE/	
2051	011102'	002004	044116	123	.ASCII	<004><004>/NHS/	

# E04

2052	011107'	004	046402	051510	.ASCII	<004><002>/MHS/	
2053	011114'	000404	051127	125	.ASCII	<004><001>/MRU/	
2054	011121'	376			.BYTE	376	
2055	011122'	100024	041101	123	.ASCII	<024><200>/ABS/	;RPER2 - BYTE 0
2056	011127'	014	043200	047105	.ASCII	<014><200>/FEN/	
2057	011134'	040004	052524	106	.ASCII	<004><100>/TUF/	
2058	011141'	004	052040	043104	.ASCII	<004><040>/TDF/	
2059	011146'	010024	040522	127	.ASCII	<024><020>/RAW/	
2060	011153'	014	046420	042523	.ASCII	<014><020>/MSE/	
2061	011160'	004004	051503	125	.ASCII	<004><010>/CSU/	
2062	011165'	004	053404	052523	.ASCII	<004><004>/WSU/	
2063	011172'	001004	051503	106	.ASCII	<004><002>/CSF/	
2064	011177'	004	053401	052503	.ASCII	<004><001>/WCU/	
2065	011204'	377			.BYTE	377	
2066	011205'	005	047600	054503	.ASCII	<005><200>/OCYL/	;RPER3 - BYTE 1
	011212'	114					
2067	011213'	004	051500	044513	.ASCII	<004><100>/SKI/	
2068	011220'	020024	050117	105	.ASCII	<024><040>/OPE/	
2069	011225'	376			.BYTE	376	
2070	011226'	040004	041504	114	.ASCII	<004><100>/DCL/	;RPER3 - BYTE 0
2071	011233'	004	040440	046103	.ASCII	<004><040>/ACL/	
2072	011240'	004014	053525	122	.ASCII	<014><010>/UMR/	
2073	011245'	024	053402	047501	.ASCII	<024><002>/WAO/	
2074	011252'	001014	052526	106	.ASCII	<014><002>/VUF/	
2075	011257'	024	042001	052503	.ASCII	<024><001>/DCU/	
2076	011264'	000414	051520	125	.ASCII	<014><001>/PSU/	
2077	011271'	377			.BYTE	377	
2078	011272'	377			.BYTE	377	
2079	011273'	377			.BYTE	377	
2080	011274'	377			.BYTE	377	
2081	011275'	044	040600	042520	.ASCII	<044><200>/APE/	;RPCS3 - BYTE 1 (11/70 ONLY)
2082	011302'	040046	050104	047505	.ASCII	<046><100>/DPEOW/	
	011310'	127					
2083	011311'	046	042040	042520	.ASCII	<046><040>/DPEEW/	
	011316'	053505					
2084	011320'	010046	041527	047505	.ASCII	<046><020>/WCEOW/	
	011326'	127					
2085	011327'	046	053410	042503	.ASCII	<046><010>/WCEEW/	
	011334'	053505					
2086	011336'	000			.BYTE	0	;TABLE TERMINATOR
2087	011340'				.EVEN		

```

2089          .SBTTL  SUBROUTINES FOR RPO4/RPO5/RPO6 DEVICE ROUTINE
2090
2091
2092
2093          ;SAVE REGISTERS R0 THRU R5
2094
2095          ;JSR    R0,SAVREG      S/R CALL
2096
2097          SAVREG: MOV    R1,-(SP)      ;SAVE R0 THRU R5
2098          MOV    R2,-(SP)
2099          MOV    R3,-(SP)
2100          MOV    R4,-(SP)
2101          MOV    R5,-(SP)
2102          MOV    R0,PC              ;EXIT IN-LINE
2103
2104
2105          ;RESTORE REGISTERS R0 THRU R5
2106
2107          ;JSR    R0,RESREG      S/R CALL
2108
2109          RESREG: TST    (SP)+        ;RESTORE R5 THRU R0
2110          MOV    (SP)+,R5
2111          MOV    (SP)+,R4
2112          MOV    (SP)+,R3
2113          MOV    (SP)+,R2
2114          MOV    (SP)+,R1
2115          RTS    R0                ;EXIT IN-LINE
2116
2117
2118          ;SET PROGRAM'S PROG TABLE ADR IN R3 & RPCS1 ADR IN R4
2119
2120          ;JSR    PC,SUPTAD      S/R CALL
2121
2122          SUPTAD: MOV    PC,R3        ;SET UP LOCATION ZERO ADR
2123          ADD    #LOCZ-,R3
2124          SUB    -2(R3),R3          ;SUBTRACT PROG TBL LENGTH
2125          MOV    DREGAD,R4        ;GET DEV REG BASE ADR
2126          RTS    PC              ;EXIT IN-LINE
2127
2128
2129          ;STORE DEVICE'S STATUS REGISTERS
2130
2131          ;JSR    R5,STSTAT      S/R CALL
2132          WORD  STADR-          REL STORAGE ADR
2133          ;DESTROYS R0,R1,R2
2134
2135          STSTAT: MOV    R5,R1        ;GET REL STORAGE ADR & MAKE
2136          ADD    (R5)+,R1          ;IT ABSOLUTE
2137          MOV    DREGAD,R0        ;GET DEV REG ADR
2138          MOV    R0,-(SP)        ;SET UP ADR OF RPCS2 REG
2139          ADD    #RPCS2,(SP)      ;FOR LATER USE
2140          MOV    #5011,R2        ;SET UP TWO LOOP COUNTS
2141          BIT    #CPU70,ACSYSFW   ;RUNNING ON AN 11/70?
2142          BEQ    10$,N-10$      ;Y,N-10$
2143          ADD    #1000,R2        ;ALLOW FOR 2 MORE REGS
2144          MOV    (R0)+,(R1)+     ;STORE DEV REG
    
```

```

2145 011452' 105302          DECB      R2          ;FINISHED WITH THIS GROUP OF REGS?
2146 011454' 001375          BNE     10$         ;Y,N-10$
2147 011456' 000302          SWAB    R2          ;SET UP NEXT LOOP CNT
2148 011460' 001417          BEQ     30$         ;DONE 2 PASSES? (N,Y-30$)
2149 011462' 010746          MOV     PC, -(SP)   ;SET UP CURRENT STATUS
2150 011464' 062716 170524   ADD     #CSTAT-., (SP) ;STORAGE ADR
2151 011470' 020126          CMP     R1, (SP)+  ;STORING STATUS FOR INTERRUPT?
2152 011472' 101005          BHI     15$         ;Y,N-15$
2153 011474' 032736 000200   BIT     #200, 2(SP)+ ;OUTPUT READY SET IN RPCS2?
2154 011500' 001403          BEQ     20$         ;Y,N-20$
2155 011502' 012021          MOV     (R0)+, (R1)+ ;STORE RPOB'S CONTENTS
2156 011504' 000761          BR      10$         ;GO DO SECOND PASS
2157 011506' 005726          15$:  TST     (SP)+    ;TAKE UNUSED ADR OFF STACK
2158 011510' 062700 000002   20$:  ADD     #2, R0   ;BYPASS READ C RPOB
2159 011514' 005021          CLR     (R1)+      ;SET ITS STORAGE TO 0'S
2160 011516' 000754          BR      10$         ;GO DO SECOND PASS
2161 011520' 000205          30$:  RTS     R5      ;EXIT IN-LINE
2162
2163
2164          ;DISPLAY DEVICE I.D. & UNIT #
2165
2166          ;JSR PC,DEVID S/R CALL
2167
2168          ;R3 MUST CONTAIN PROG TBL ADR
2169          ;DESTROYS R0,R1,R2
2170
2171 011522' 012700 032060   DEVID: MOV     #*04, R0 ;INITIALIZE TO RPO4
2172 011526' 032763 000060 000032   BIT     #60, PMDLCD(R3) ;IS THIS AN RPO4?
2173 011534' 001410          BEQ     10$         ;N,Y-10$
2174 011536' 012700 032460   MOV     #*05, R0   ;SET UP FOR AN RPO5
2175 011542' 032763 000020 000032   BIT     #20, PMDLCD(R3) ;IS IT AN RPO5?
2176 011550' 001002          BNE     10$         ;N,Y-10$
2177 011552' 012700 033060   MOV     #*06, R0   ;MUST BE AN RPO6
2178 011556' 010067 000532 10$:  MOV     R0, UNITMG+6 ;TAILOR DEV I.D. MSG
2179 011562' 012767 000026 000056   MOV     #22., DEVIML ;INITIALIZE TO NORMAL MSG LNGTH
2180 011570' 116300 000035   MOV     PCURDV(R3), R0 ;GET CURR UNIT #
2181 011574' 020027 000007   MOV     R0, #7     ;VALID UNIT #?
2182 011600' 101007          BHI     DEVIIV      ;Y,N-DEVIIV
2183 011602' 004577 166252   JSR     R5, 28TASLZ ;CONVERT # TO DECIMAL ASCII
2184 011606' 000524          .WORD  UNASCII-
2185 011610' 016767 000522 000514   MOV     UNASCII+4, UNASCII ;MOVE ASCII # TO 1ST TWO DIGITS
2186 011616' 000410          BR      DEVIPR      ;GO ISSUE MSG
2187 011620' 012767 000032 000020   DEVIIV: MOV     #26., DEVIML ;SET UP ERR COND MSG LNGTH
2188 011626' 042700 177400   BIC     #177400, R0 ;RESET HIGH BYTE
2189 011632' 004577 166220   JSR     R5, 28INASC ;CONVERT BINARY # TO ASCII
2190 011636' 000474          .WORD  UNASCII-
2191 011640' 004567 000256   DEVIPR: JSR     R5, PRINT ;GO ISSUE UNIT # MSG
2192 011644' 000442          .WORD  UNITMG-.
2193 011646' 000026          DEVIIML: .WORD  22.
2194 011650' 000207          RTS     PC         ;EXIT IN-LINE
    
```

```

;TAILOR STATUS MSG & PRINT IT
2196
2197
2198
2199
2200
2201
2202 011652' 010346          DISPST: MOV    R3,-(SP)          ;SAVE R3
2203 011654' 010503          MOV    R5,R3          ;GET REL DATA ADR
2204 011656' 062503          ADD    (R5)+,R3       ;MAKE IT ABS
2205 011660' 010546          MOV    R5,-(SP)       ;SAVE R5
2206 011662' 010705          MOV    PC,R5          ;SET UP ADR OF REG NAMES IN ASCII
2207 011664' 062705 166232  ADD    #DVRGMS-,R5
2208 011670' 012746 000024  MOV    #20,-(SP)      ;SET UP # OF REGISTERS TO DISPLAY
2209 011674' 032777 000010 166162 BIT    #CPU70,#CSYSFW ;RUNNING ON AN 11/70?
2210 011702' 001402          BEQ    10$
2211 011704' 062716 000002  ADD    #2,(SP)        ;MAKE IT 22 REGISTERS
2212 011710' 012700 000003 10$:  MOV    #3,R0          ;SET UP 3 REG LOOP CNT
2213 011714' 010701          MOV    PC,R1          ;POINT AT REG NAME IN MSG
2214 011716' 062701 000422  ADD    #DVRGMG-,R1
2215 011722' 012521 15$:  MOV    (R5)+,(R1)+    ;MOVE REG NAME TO MSG
2216 011724' 012521          MOV    (R5)+,(R1)+
2217 011726' 005725          TST   (R5)+
2218 011730' 062701 000012  ADD    #10-,R1
2219 011734' 005300          DEC   R0
2220 011736' 001371          BNE   15$
2221 011740' 012300          MOV   (R3)+,R0
2222 011742' 004577 166110  JSR   R5,#BINASC
2223 011746' 000400          .WORD DVRDT1-
2224 011750' 012300          MOV   (R3)+,R0
2225 011752' 004577 166100  JSR   R5,#BINASC
2226 011756' 000406          .WORD DVRDT2-
2227 011760' 012300          MOV   (R3)+,R0
2228 011762' 004577 166070  JSR   R5,#BINASC
2229 011766' 000414          .WORD DVRDT3-
2230 011770' 012767 000050 000034 MOV   #40,30$
2231 011776' 162716 000003  SUB   #3,(SP)
2232 012002' 100005          BPL   25$
2233 012004' 162767 000016 000020 20$:  SUB   #14,30$
2234 012012' 005216          INC   (SP)
2235 012014' 100773          BMI   20$
2236 012016' 010346 25$:  MOV   R3,-(SP)
2237 012020' 016603 000006  MOV   6(SP),R3
2238 012024' 004567 000072  JSR   R5,PRINT
2239 012030' 000310          .WORD DVRGMG-
2240 012032' 000050 30$:  .WORD 40
2241 012034' 012603          MOV   (SP)+,R3
2242 012036' 005716          TST   (SP)
2243 012040' 001323          BNE   10$
2244 012042' 005726          TST   (SP)+
2245 012044' 012605          MOV   (SP)+,R5
2246 012046' 012603          MOV   (SP)+,R3
2247 012050' 000205          RTS   R5
;EXIT IN-LINE
;RESTORE REG DATA PNTR
;MORE REGS TO GO?
;N,Y-10$
;REMOVE CNT FROM STACK
;RESTORE R5 & R3
;GO PRINT THE MSG
;RESTORE PROG TBL ADR
;SAVE REG DATA PNTR
;CNT BACK TO 0? (Y,N-20$)
;INCR NEG REG CNT
;SHORTEN MSG LENGTH BY 1 REG
;< 3 REGS? (Y,N-25$)
;DECR REGISTER CNT
;INITIALIZE MSG LENGTH TO 3 REGS
;AND PLACE IN THE MSG
;FOR 3 REGISTERS TO ASCII
;CONVERT OCTAL REGISTER CONTENTS
;Y,N-15$
;DONE 3 REGS?
;POINT TO NEXT FIELD IN MSG
;POINT TO NEXT NAME
;MOVE REG NAME TO MSG
;SET UP 3 REG LOOP CNT
;MAKE IT 22 REGISTERS
;Y,N-10$
;RUNNING ON AN 11/70?
;SET UP # OF REGISTERS TO DISPLAY
;SET UP ADR OF REG NAMES IN ASCII
;SAVE R5
;MAKE IT ABS
;GET REL DATA ADR
;SAVE R3
;S/R CALL
;REL ADR OF STATUS DATA
;DESTROYS R0,R1,R2
;WORD STATAOR-
;JSR R5,DISPST

```

DTRPAB.P11

SUBROUTINES FOR RPO4/RPO5/RPO6 DEVICE ROUTINE

```

2249                                     ;DISPLAY CYL/HEAD/SECT WORDS' VALUES
2250
2251                                     ;JSR   PC,PRTIWD       S/R CALL
2252                                     ;DESTROYS R0,R1,R2
2253
2254 012052' 016700 165726      PRTIWD: MOV   CYL,R0       ;GET CYL VALUE
2255 012056' 004577 165774      JSR   RS,28INASC    ;CONVERT ITS VALUE TO ASCII
2256 012062' 001045                .WORD IFCYL-
2257 012064' 016700 165716      MOV   HEAD,R0      ;GET & CONVERT HEAD VALUE
2258 012070' 004577 165762      JSR   RS,28INASC
2259 012074' 001050                .WORD IFHEAD-
2260 012076' 016700 165706      MOV   SECT,R0     ;GET & CONVERT SECT VALUE
2261 012102' 004577 165750      JSR   RS,28INASC
2262 012106' 001053                .WORD IFSECT-
2263 012110' 004567 000006      JSR   RS,PRINT     ;PRINT MSG WITH THEIR VALUES
2264 012114' 001006                .WORD INFOG-
2265 012116' 000045                .WORD 37.
2266 012120' 000207      RTS   PC           ;EXIT IN-LINE
2267
2268
2269                                     ;ISSUE MSG TO LIST DEVICE SUBROUTINE
2270
2271                                     ;JSR   RS,PRINT       S/R CALL
2272                                     ;.WORD MSGADR-      REL ADR OF MSG
2273                                     ;.WORD BYTCNT      MSG BYTE CNT (IF NEGATIVE,
2274                                     ;                  RESET PRT DEV DEDICATED.)
2275                                     ;R3 = PROG TBL ADR
2276                                     ;R4 = FLAGWORD -- IF NEGATIVE, USE CMND MODE PRINT
2277                                     ;DESTROYS R0,R1,R2
2278
2279 012122' 010500      PRINT: MOV   RS,R0       ;GET MSG ADR & MAKE IT ABS
2280 012124' 062500      ADD   (RS)+,R0
2281 012126' 012501      MOV   (RS)+,R1    ;GET BYTE COUNT
2282 012130' 005704      TST   R4         ;USE CMND MODE PRINT?
2283 012132' 100030      BPL   40$        ;Y,N-40$
2284 012134' 010702      MOV   PC,R2      ;SET UP LINK INFO ADR
2285 012136' 062702 000040      ADD   #20$-,R2
2286 012142' 160200      SUB   R2,R0      ;MAKE MSG ADR REL
2287 012144' 010022      MOV   R0,(R2)+  ;STORE MSG ADR
2288 012146' 010112      MOV   R1,(R2)   ;STORE MSG'S BYTE COUNT
2289 012150' 100001      BPL   10$        ;CNT NEG? (Y,N-10$)
2290 012152' 005412      NEG   (R2)       ;MAKE IT POSITIVE
2291 012154' 016367 000006 000056 10$: MOV   PASCIN(R3),PROG# ;STORE PROG'S # IN MSG
2292 012162' 004577 165666      JSR   RS,2CLIST  ;ISSUE PROG #
2293 012166' 000050      .WORD PNMMSG-
2294 012170' 000005      .WORD 5
2295 012172' 004577 165656      JSR   RS,2CLIST  ;ISSUE MSG SPECIFIED
2296 012176' 000000      .WORD XXXX
2297 012200' 000000      .WORD XXXX
2298 012202' 004577 165646      JSR   RS,2CLIST  ;ISSUE A <CR> & <LF>
2299 012206' 000254      .WORD CRLF-
2300 012210' 000002      .WORD 2
2301 012212' 000410      BR    PRTEX      ;GO TO EXIT
2302 012214' 010067 000010 40$: MOV   R0,50$     ;STORE MSG'S ABS ADR
2303 012220' 010167 000006      MOV   R1,60$     ;STORE ITS BYTE CNT
2304 012224' 004577 165622      JSR   RS,2ULIST  ;GO TO MPG TO ISSUE THE MSG

```

J04

2305 012230' 000000  
2306 012232' 000000  
2307 012234' 000205

50\$: .WORD XXXX  
60\$: .WORD XXXX  
PRTEX: RTS R5

;EXIT IN-LINE

```

2309 .SBTTL RPO4/RPO5/RPO6 MESSAGE STORAGE AREA
2310
2311
2312 .NLIST BEX
2313
2314 .EVEN
2315 012236' 021520 PNMMSG: .ASCII /P# /
2316 012240' 054130 011 PROGMM: .ASCII /XX/<011>
2317 012243' 101 020124 040514 ATIMSG: .ASCII /AT LAST INT:/
2318 012257' 103 051125 042522 CURMSG: .ASCII /CURRENTLY:/
2319 012271' 105 042116 047440 RENDMG: .ASCII /END OF REPORT/
2320 .EVEN
2321 012306' 025052 025052 050122 UNITMG: .ASCII /***RPXX DISK UNIT: /
2322 012332' 054130 054130 054130 UNASCI: .ASCII /XXXXXX/
2323 .EVEN
2324 012340' 054130 054130 020075 DVRCMG: .ASCII /XXXX= /
2325 012346' 054130 054130 054130 DVRO1: .ASCII /XXXXXX XXXX= /
2326 012364' 054130 054130 054130 DVRO2: .ASCII /XXXXXX XXXX= /
2327 012402' 054130 054130 054130 DVRO3: .ASCII /XXXXXX/
2328 012410' 054502 042524 035123 CNTSMG: .ASCII /BYTES: RD= /
2329 012424' 054130 054130 054130 BCMRD: .ASCII /XXXXXXXXXXXXXX WR= /
2330 012446' 054130 054130 054130 BCMWR: .ASCII /XXXXXXXXXXXXXX/
2331 012462' 005015 CRLF: .ASCII <015><012>
2332 012464' 004411 045503 020075 .ASCII <011><011>/CK= /
2333 012472' 054130 054130 054130 BCMCK: .ASCII /XXXXXXXXXXXXXX/<015><012><011>/CMNDS: RD= /
2334 012525' 130 054130 054130 CMDCRD: .ASCII /XXXXXX WR= /
2335 012540' 054130 054130 054130 CMDCAR: .ASCII /XXXXXX CK= /
2336 012553' 130 054130 054130 CMDCCK: .ASCII /XXXXXX/<015><012><011><011>/SK= /
2337 012571' 130 054130 054130 CMDCSK: .ASCII /XXXXXX MISC= /
2338 012606' 054130 054130 054130 CMDCMS: .ASCII /XXXXXX/<015><012><011>/ERRORS: DEV= /
2339 012634' 054130 054130 054130 CNTERR: .ASCII /XXXXXX CORR ECC= /
2340 012655' 130 054130 054130 CNTCEC: .ASCII /XXXXXX DATA= /
2341 012672' 054130 054130 054130 CNTDER: .ASCII /XXXXXX/<015><012><011>/RETRY: DLT= /
2342 012720' 054130 054130 054130 CNTDLT: .ASCII /XXXXXX DTE= /
2343 012734' 054130 054130 054130 CNTDTE: .ASCII /XXXXXX HCRC= /
2344 012751' 130 054130 054130 CNTHCR: .ASCII /XXXXXX/<015><012><011><011>/FER= /
2345 012770' 054130 054130 054130 CNTFER: .ASCII /XXXXXX HCE= /
2346 013004' 054130 054130 054130 CNTHCE: .ASCII /XXXXXX DCK= /
2347 013020' 054130 054130 054130 CNTDCK: .ASCII /XXXXXX/<015><012><011><011>/WCE= /
2348 013037' 130 054130 054130 CNTWCE: .ASCII /XXXXXX/<015><012><011>/TOTAL RETRY: /
2349 013066' 054130 054130 054130 CNTRTY: .ASCII /XXXXXX/<015><012><011>/INTERRUPTS: /
2350 013114' 054130 054130 054130 CNTINT: .ASCII /XXXXXX/
2351 013122' 013122' CNTSEN=
2352 013122' 054503 036514 040 INFOMG: .ASCII /CYL= /
2353 013127' 130 054130 054130 IFCYL: .ASCII /XXXXXX HEAD= /
2354 013144' 054130 054130 054130 IFHEAD: .ASCII /XXXXXX SECT= /
2355 013161' 130 054130 054130 IFSECT: .ASCII /XXXXXX/
2356 013170' 013170' .EVEN
2357 013170' 052123 047115 020124 STMNMG: .ASCII /STMT # /
2358 013200' 054130 054130 054130 STMNUM: .ASCII /XXXXXX/
2359 013206' 051105 047522 020122 DKEMSG: .ASCII /ERROR BITS://<015><012><011>
2360 013224' 000100 CODFLD: .BLKB 64.
2361 013324' 042502 047506 042522 BEFIO: .ASCII 'BEFORE ISSUING I/O CMND:'
2362 013354' 043101 042524 020122 AFTIO: .ASCII 'AFTER ISSUING I/O CMND:'
2363 013403' 124 047457 047440 CRT0: .ASCIZ 'T/O ON CRESET'
2364 013421' 124 046511 047505 IOT0: .ASCIZ 'TIMEOUT ON I/O'

```

2365	013440'	027524	020117	047117	ACQTO: .ASCIZ	'T/O ON DISK ACQUIRE'
2366	013464'	047516	026516	054105	NONEXD: .ASCIZ	'NON-EXISTENT DRIVE'
2367	013507'	125	051516	043101	INITUS: .ASCII	'L -AFE'
2368	013516'	051105	047522	020122	INITDE: .ASCIZ	'E:JR ON INITIATION'
2369	013542'	044504	045523	044440	OFFLIN: .ASCIZ	'DISK IS OFF-LINE'
2370	013563'	104	051120	047040	DPNSET: .ASCIZ	'DPR NOT SET'
2371	013577'	116	047117	044455	NOITER: .ASCII	'NON-INT '
2372	013607'	111	047457	052040	IOTERM: .ASCIZ	'I/O TERMINATION ERROR'
2373	013635'	111	052116	053440	NOATA: .ASCIZ	'INT WITHOUT ATA'
2374	013655'	125	042516	050130	UXPATA: .ASCIZ	'UNEXP ATA COND'
2375	013674'	054105	040510	051525	RTYEXH: .ASCIZ	'EXHAUSTED RETRIES'
2376	013716'	047111	020126	041505	INVPAT: .ASCIZ	'INV ECC BIT PATTERN'
2377	013742'	047111	020126	041505	INVPOS: .ASCIZ	'INV ECC BIT POSITION'
2378	013767'	111	053116	052440	INVDVN: .ASCIZ	'/INV UNIT #/'
2379					.EVEN	
2380					.LIST	BEX
2381						
2382	014002'				DVREND= .	

```

2384          .SBTTL FORMATS FOR PROGRAM & DEVICE ROUTINE TABLES
2385          ; PROGRAM TABLE FORMAT
2386
2387          PTLGTH= 162. ;PROGRAM TABLE LENGTH - NON MEM MGMT VERSION OF MPG
2388          ;(PTLGTH= 212. ;PROGRAM TABLE LENGTH - MEM MGMT VERSION OF MPG)
2389
2390          PFLGWD= +0. ;PROGRAM FLAG WORD - 1 WORD
2391
2392          000000
2393
2394          000002          URSTOP= 2          ; 1 = USER HAS STOPPED THIS PROGRAM
2395          000004          ERSTOP= 4          ; 1 = AN ERROR HAS STOPPED THIS PROGRAM
2396          000010          WT4IOT= 10         ; 1 = WAITING FOR I/O TERMINATION
2397          000020          CTPRIO= 20         ; 1 = CONSOLE OR PRINTER I/O IN PROGRESS
2398          000040          SETDED= 40         ; 1 = THIS PROG SET THE PRT DEV DEDICATED FLAG
2399          000100          OCPRES= 100        ; 1 = OBJ CODE IS PRESENT
2400          000200          USEUBM= 200       ; 1 = THIS PROG USES THE UNIBUS MAP (MEM MGMT ONLY)
2401          100000          ACTIVE= 100000    ; 1 = PROGRAM IS ACTIVE (SPECIFIED FOR EXECUTION)
2402
2403          000002          POPSW= +2. ;PROGRAM'S OPERATION SWITCHES - 1 WORD
2404
2405          100000          STONER= 100000     ; 1 = STOP PROG EXECUTION UPON ERROR
2406          040000          CYCPRG= 40000     ; 1 = CYCLE PROGRAM (ON CURRENT DEVICE)
2407          020000          PRONER= 20000     ; 1 = DO NOT PRINT ON ERROR
2408          010000          BIT12= 10000     ; 0 = NOT USED
2409          004000          BIT11= 4000      ; 0 = NOT USED
2410          002000          CYCDVL= 2000     ; 1 = CYCLE THE DEVICE LIST
2411          001000          GTNXTD= 1000     ; 1 = CYCLE ON SAME DEVICE UPON ERROR
2412          000400          DOERCK= 400      ; 1 = DON'T DO ERROR CHECKING
2413          000200          SPOPER= 200     ; 1 = DEVICE SPECIAL OPERATION
2414          000100          BIT6= 100       ; 0 = NOT USED
2415          000040          DOIOT= 40        ; 1 = DO NOT PERFORM I/O TIMEOUT
2416          000020          AUTORP= 20       ; 1 = DO NOT AUTOMATICALLY DISPLAY COUNTS
2417          000010          AURPEP= 10      ; 1 = AUTO DISPLAY COUNTS AT END OF FINAL PASS ONLY
2418          000004          HSKPEP= 4        ; 1 = HOUSEKEEP COUNTS ONLY AT RUN COMMAND
2419          000002          PFBBOV= 2        ; 1 = PRINT FIRST BAD BYTE ONLY ON VERIFY
2420          000001          NOCOMP= 1        ; 1 = DO NOT PRINT PROG COMPLETED MSG
2421
2422          000004          PFWADR= +4. ;*;PROGRAM FLAGWORD ADDRESS - 1 WORD
2423
2424          000006          PASCIN= +6. ;PROGRAM'S NUMBER IN ASCII - 1 WORD
2425
2426          000010          PNAME= +8. ;PROGRAM'S NAME IN ASCII - 6 BYTES
2427
2428          000016          PRDIOA= +14. ;ADDRESS OF READ I/O AREA - 1 WORD
2429
2430          000020          PWRIOA= +16. ;ADDRESS OF WRITE I/O AREA - 1 WORD
2431
2432          000022          PSRCST= +18. ;SOURCE STATEMENTS START ADDRESS - 1 WORD
2433
2434          000024          POBJST= +20. ;OBJECT CODE START ADDRESS - 1 WORD
2435
2436          000026          PLNGTH= +22. ;PROG AREA LENGTH (OBJ END MINUS PROG TBL START) - 1 WORD
2437
2438          000030          PTOCNT= +24. ;I/O TIMEOUT COUNT - 1 WORD
2439
    
```

2440	000032	PMDLCD= +26.	;DEV ROUT MODEL # CODE - 1 WORD
2441			
2442	000034	PDPNTR= +28.	;CURRENT DEVICE NUMBER POINTER - 1 BYTE
2443			
2444	000035	PCURDV= +29.	;CURRENT DEVICE # - 1 BYTE
2445			
2446	000036	PDNUMS= +30.	;DEVICE NUMBERS - 16 BYTES
2447			
2448	000056	PTEM0= +46.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2449			
2450	000060	PTEM1= +48.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2451			
2452	000062	PTEM2= +50.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2453			
2454	000064	PTEM3= +52.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2455			
2456	000066	PTEM4= +54.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2457			
2458	000070	PTEM5= +56.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2459			
2460	000072	PTEM6= +58.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2461			
2462	000074	PTEM7= +60.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2463			
2464	000076	PTEM8= +62.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2465			
2466	000100	PTEM9= +64.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2467			
2468	000102	PTEM10= +66.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2469			
2470	000104	PTEM11= +68.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2471			
2472	000106	PTEM12= +70.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2473			
2474	000110	PTEM13= +72.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2475			
2476	000112	PTEM14= +74.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2477			
2478	000114	PTEM15= +76.	;USER PROGRAM TEMPORARY STORAGE - 1 WORD
2479			
2480	000116	PNBR= +78.	;NUMBER OF BYTES TO TRANSFER ON MOVE (NBR) - 1 WORD
2481			
2482	000120	PSRC= +80.	;DATA SOURCE ADDRESS ON MOVE (SRC) - 1 WORD
2483			
2484	000122	PDST= +82.	;DATA DESTINATION ADDRESS ON MOVE (DST) - 1 WORD
2485			
2486	000124	PSTKCT= +84.	;# OF WORDS (X 2) SAVED OFF STACK - 1 WORD
2487			
2488	000126	PSTKSV= +86.	;STACK WORDS STORAGE AREA - 30 WORDS
2489			
2490	000222	PSVREG= +146.	;USER'S R0 THRU R5 REGISTERS STORAGE AREA - 6 WORDS
2491			
2492	000236	PUSRPC= +158.	;USER'S CURRENT PROGRAM COUNTER - 1 WORD
2493			



2571  
2572  
2573  
2574  
2575  
2576  
2577  
2578  
2579  
2580  
2581  
2582  
2583  
2584  
2585  
2586  
2587  
2588  
2589  
2590  
2591  
2592  
2593  
2594  
2595  
2596  
2597  
2598  
2599  
2600  
2601  
2602  
2603  
2604  
2605  
2606  
2607  
2608  
2609  
2610  
2611  
2612  
2613  
2614  
2615  
2616  
2617  
2618  
2619  
2620  
2621  
2622  
2623  
2624  
2625  
2626  
2627  
2628  
2629  
2630  
2631  
2632  
2633  
2634  
2635  
2636  
2637  
2638  
2639  
2640  
2641  
2642  
2643  
2644  
2645  
2646  
2647  
2648  
2649  
2650  
2651  
2652  
2653  
2654  
2655  
2656  
2657  
2658  
2659  
2660  
2661  
2662  
2663  
2664  
2665  
2666  
2667  
2668  
2669  
2670  
2671  
2672  
2673  
2674  
2675  
2676

; DEVICE ROUTINE TABLE

000116	DRTLTH= 78.	;DEVICE ROUTINE TABLE LENGTH
	:	
000000	DEVRSZ= +0.	;DEVICE ROUTINE SIZE IN BYTES - 1 WORD
000002	DEVFWD= +2.	;DEVICE ROUTINE FLAGWORD - 1 WORD
000004	DEVIW1= +4.	;DEVICE INTERFACE WORD # 1 - 1 WORD
000006	DEVIW2= +6.	;DEVICE INTERFACE WORD # 2 - 1 WORD
000010	DEVIW3= +8.	;DEVICE INTERFACE WORD # 3 - 1 WORD
000012	DEVIW4= +10.	;DEVICE INTERFACE WORD # 4 - 1 WORD
000014	DEVIW5= +12.	;DEVICE INTERFACE WORD # 5 - 1 WORD
000016	DEVIW6= +14.	;DEVICE INTERFACE WORD # 6 - 1 WORD
000020	DEVIW7= +16.	;DEVICE INTERFACE WORD # 7 - 1 WORD (SIZE)
000022	DEVIW8= +18.	;DEVICE INTERFACE WORD # 8 - 1 WORD (ERR)
000024	DEVDR= +20.	;DEVICE REGISTERS ADDRESS - 1 WORD
000026	DEVIVA= +22.	;DEVICE INTERRUPT VECTOR ADDRESS - 1 WORD
000030	DEVRRS= +24.	;DEVICE READ PROCESSOR STATUS WORD (BUS REQ) - 1 WORD
000032	DEVWPS= +26.	;DEVICE WRITE PROC STATUS WORD (BUS REQ) - 1 WORD
000034	DHKNAD= +28.	;DEVICE ROUT HOUSEKEEPING ROUT REL ENTRY ADR - 1 WORD
000036	DERPAD= +30.	;DEVICE ROUT REPORT ROUT REL ENTRY ADR - 1 WORD
000040	DKILAD= +32.	;DEVICE ROUT KILL ROUTINE REL ENTRY ADR - 1 WORD
000042	DECTAD= +34.	;DEVICE ROUT ERROR COUNTER REL ADR - 1 WORD
000044	DTOEAD= +36.	;DEVICE ROUT TIMEOUT ERR ROUT REL ENTRY ADR - 1 WORD
000046	DEVI0B= +38.	;DEVICE I/O BUSY BRANCH ADDRESS (CIOBSY) - 1 WORD
000050	DEVDER= +40.	;DEVICE ERROR BRANCH ADDRESS (CUPGER) - 1 WORD
000052	DVUPRT= +42.	;USER MODE PRINT BRANCH ADDRESS (ULIST) - 1 WORD
000054	DVCPRT= +44.	;CMND MODE PRINT BRANCH ADDRESS (CLIST) - 1 WORD
000056	DEVBT= +46.	;CONVERT BINARY TO ASCII BR ADR (BINASC) - 1 WORD
000060	DVBTD= +48.	;CONVERT BINARY TO DECIMAL ASCII BR ADR (BTASLZ) - 1 WORD

2577	000062	DVPDTA= +50.	;CONVEK; PACKED DECIMAL TO ASCII BR ADR (DECASC) - 1 WORD
2578			
2579	000064	DVSFWD= +52.	;MPG SYSTEM FLAGWORD ADDRESS (CSYSFW) - 1 WORD
2580			
2581	000066	DVSVEC= +54.	;SET INTERRUPT VECTOR BR ADR (SETVEC) - 1 WORD
2582			
2583	000070	DVCVEC= +56.	;CLEAR INTERRUPT VECTOR BR ADR (CLRVEC) - 1 WORD
2584			
2585	000072	DVTVEC= +58.	;TEST INTERRUPT VECTOR BR ADR (TSTVEC) - 1 WORD
2586			
2587	000074	DVRINT= +60.	;RETURN FROM INTERRUPT BR ADR (RTNINT) - 1 WORD
2588			
2589	000076	DVGETB= +62.	;GET DATA BYTE BR ADR (GETBYT) - 1 WORD
2590			
2591	000100	DVPUTB= +64.	;PUT DATA BYTE BR ADR (PUTBYT) - 1 WORD
2592			
2593	000102	DEVSTP= +66.	;DEVICE ROUT REL SYMBOL TABLE POINTER - 1 WORD
2594			
2595	000104	DEVETP= +68.	;DEVICE ROUT REL ENTRY TABLE POINTER - 1 WORD
2596			
2597	000106	DVPTEP= +70.	;PACK TABLE EXTEN. REL POINTER - 1 WORD
2598			
2599	000110	DVVTEP= +72.	;VECTOR TABLE EXTEN. REL POINTER - 1 WORD
2600			
2601	000112	DVCTEP= +74.	;COMPILER TBL EXTEN. REL POINTER - 1 WORD
2602			
2603	000114	DVIWSP= +76.	;DEVICE INTERFACE WORD SYMBOL TBL REL POINTER - 1 WORD
2604			
2605	000116	DRTEND= +78.	;END OF DEVICE ROUTINE TABLE
2606			
2607			
2608	000001	.END	

DTRPA8.P11 SYMBOL TABLE

ACQERD	005114R	002	CNTCEC	012655R	002	DEVIPR	011640R	002	DVREND=	014002R	002	INITDE	013516R	002
ACQERC	005116R	002	CNTDCK	013020R	002	DEVIVA=	000026	002	DVREX	003006R	002	INITUS	013507R	002
ACQERR	005104R	002	CNTDER	012572R	002	DEVIMI=	000004	002	DVRGMC	012340R	002	INTCNT	002340R	002
ACQGDK	005126R	002	CNTDLT	012720R	002	DEVIM2=	000006	002	DVRINT=	000074	002	INTEAD	007576R	002
ACQMSK	004412R	002	CNTDTE	012734R	002	DEVIM3=	000010	002	DVSFMD=	000064	002	INTEX	006232R	002
ACQIAD	004656R	002	CNTERR	012634R	002	DEVIM4=	000012	002	DVSVEC=	000066	002	INVDVN	013767R	002
ACQRTY	004414R	002	CNTFER	012770R	002	DEVIM5=	000014	002	DVTVEC=	000072	002	INVPAT	013716R	002
ACQTO	013440R	002	CNTHCE	013004R	002	DEVIM6=	000016	002	DVUPRT=	000052	002	INVPOS	013742R	002
ACTIVE=	100000	002	CNTHCR	012751R	002	DEVIM7=	000020	002	DVVTPE=	000110	002	IOERR =	000001	002
AFTIO	013354R	002	CNTINT	013114R	002	DEVIM8=	000022	002	EBSBAS	010136R	002	IOTERM	013607R	002
ANYIOI=	000200	002	CNTNUM=	000027	002	DEVIPS=	000030	002	EBSTAT	010140R	002	IOTO	013421R	002
RPORT	003666R	002	CNTRTY	013066R	002	DEVRSZ=	000000	002	ECI =	004000	002	ISTAT =	002134R	002
ATA =	100000	002	CNTSEN=	013122R	002	DEVSTP=	000102	002	ECIOFF	003736R	002	IVCTAD	000026R	002
ATATBL	005136R	002	CNTSMG	012410R	002	DEVVPS=	000032	002	ECION	003726R	002	JINTEX	006650R	002
ATIMSG	012243R	002	CNTHCE	013037R	002	DFLGMD	000022R	002	ERCOTB	010472R	002	JSETER	006670R	002
AURPEP=	000010	002	COOFLD	013224R	002	DHKPAD=	000034	002	ERMBAS	010104R	002	KILL	003370R	002
AUTORP=	000020	002	CORFLG=	002000	002	DISCNT	002706R	002	ERR =	040000	002	KILLEX	003420R	002
BAI =	000010	002	COROFF	004016R	002	DISPST	011652R	002	ERRADR	002342R	002	KPAR4 =	172350	002
BAIOFF	003776R	002	CORON	004006R	002	DKEMSG	013206R	002	ERRCNT	002312R	002	KPOR4 =	172310	002
BAION	003766R	002	COUNTS	002264R	002	DKILAD=	000040	002	ERRCOM	007744R	002	LOCZ	000000R	002
BCMCK	012472R	002	CPU70 =	000010	002	DLTCNT	002320R	002	ERRCS	007704R	002	MCPE =	020000	002
BCMRD	012424R	002	CRESET	004066R	002	DOCORR	007206R	002	ERRCS1	007712R	002	MISCNT	002310R	002
BCMR	012446R	002	CRLF	012462R	002	DOERCK=	000400	002	ERREX	010470R	002	MYVER =	000001	002
BEFIO	013324R	002	CRT0	013403R	002	DOIOI =	000040	002	ERRFND	006344R	002	NOL =	010000	002
BINASC	000056R	002	CSTAT	002210R	002	DOTERM=	000002	002	ERRI	000022R	002	MSFHT1	002106R	002
BIT11 =	004000	002	CSYSFW	000064R	002	DPNSET	013563R	002	ERRIS	007730R	002	MSFHT2	002107R	002
BIT12 =	010000	002	CTPRIO=	000020	002	DPR =	000400	002	ERRSNM	010410R	002	MSFHT3	002114R	002
BIT6 =	000100	002	CUPGER	000050R	002	DREGAD	000024R	002	ERSTAD	010402R	002	MSFHT4	002123R	002
BPORT	003676R	002	CURADR	002352R	002	DRESET	004136R	002	ERSTOP=	000004	002	MSFHT5	002132R	002
BTASLZ	000060R	002	CURCMD	002350R	002	DRTEND=	000116	002	EVEN	003716R	002	MYATA	005156R	002
BYCK	002274R	002	CURCNT	002356R	002	DRTLTH=	000116	002	FERCNT	002326R	002	NED =	010000	002
BYRD	002264R	002	CURFLG	002346R	002	DTECNT	002322R	002	FINCNT	002322R	002	NOATA	013635R	002
BYMR	002270R	002	CURMSG	012257R	002	DTOEAD=	000044	002	FMT20	004000R	002	NOCOMP=	000001	002
CECCER	002314R	002	CURPBC	002360R	002	DVA =	004000	002	FMT22	004046R	002	NOICOM	004174R	002
CI0BSY	000046R	002	CURPSW	002370R	002	DVBTDA=	000060	002	FMT28=	010000	002	NOITER	013577R	002
CKCNT	002304R	002	CURRTY	002364R	002	DVCNDS	000322R	002	GETBYI	000076R	002	NONEXD	013464R	002
CKCOM	005272R	002	CYCDVL=	000000	002	DVCPTI=	000054	002	GOTDSK	004662R	002	NO:AIT	003656R	002
CKCORR	006674R	002	CYCPRG=	040000	002	DVCPTI	001454R	002	GTNXTD=	001000	002	OCFRES=	000100	002
CK0BSY	007334R	002	CYL	003004R	002	DVCTEP=	000112	002	HARDE	006654R	002	OOD	003706R	002
CKRTY	006424R	002	DATAER	002316R	002	DVCVEC=	000070	002	HCECNT	002330R	002	OFFLIN	013542R	002
CKSC	006242R	002	DCKCNT	002332R	002	DVGETB=	000076	002	HCI =	002000	002	OFFSET	005360R	002
CKUNLD	003212R	002	DEASC	000062R	002	DVINSP=	000114	002	HCI0FF	003756R	002	OTHATA	005160R	002
CLIST	000054R	002	DECTAD=	000042	002	DVINST	002054R	002	HCI0N	003746R	002	PAKACK	004152R	002
CLRVEC	000070R	002	DERPAD=	000036	002	DVMYTE	001230R	002	HCRCNT	002324R	002	PARITY=	000020	002
CLRWF	006210R	002	DEVBTA=	000056	002	DVPDTA=	000062	002	HEAD	000006R	002	PASCIN=	000006	002
CHDCCK	012553R	002	DEVDER=	000050	002	DVPKTE	000560R	002	HSKEEP	002450R	002	PATCH	002400R	002
CHDCMS	012606R	002	DEVORA=	000024	002	DVPTEP=	000106	002	HSKPEN=	002372R	002	PC =	000007	002
CHDCOM	005416R	002	DEVETP=	000104	002	DVPUTB=	000100	002	HSKPEP=	000004	002	PCURDV=	000035	002
CHDCRD	012525R	002	DEVFMD=	000002	002	DVRT01	012346R	002	HSKPST=	002134R	002	PONUMS=	000036	002
CHDCSK	012571R	002	DEVID	011522R	002	DVRT02	012364R	002	IFCYL	013127R	002	PDPNTR=	000034	002
CHDCWR	012540R	002	DEVIIV	011620R	002	DVRT03	012402R	002	IFHEAD	013144R	002	PDRCON=	077406	002
CHDISU=	000100	002	DEVIML	011646R	002	DVREGE=	000322R	002	IFSECT	013161R	002	POST =	000122	002
CNTADR	002344R	002	DEVI08=	000046	002	DVREGS	000116R	002	INFONG	013122R	002	PFBBOV=	000002	002

PFLGND=	000000		PTEM2 =	000062		RPAS =	000016		R1 =	%000001		TOUTER	003074R	002
PFWADR=	000004		PTEM3 =	000064		RPBA =	000004		R2 =	%000002		TRE =	040000	
PLNGTH=	000026		PTEM4 =	000066		RPBAE =	000050		R3 =	%000003		TRMINT	006156R	002
PMOLCD=	000032		PTEM5 =	000070		RPCC =	000036		R4 =	%000004		TSTVEC	000072R	002
PNAME =	000010		PTEM6 =	000072		RPCS1 =	000000		RS =	%000005		TVECT	007654R	002
PNR =	000116		PTEM7 =	000074		RPCS1V	002372R	002	SAVREG	011340R	002	TVECTX	007702R	002
PNMSG =	012236R	002	PTEM8 =	000076		RPCS2 =	000010		SEARCH	005346R	002	ULIST	000052R	002
POBJST=	000024		PTEM9 =	000100		RPCS2V	002374R	002	SECT	000010R	002	UNASCI	012332R	002
POPSM =	000002		PTEND =	000242		RPCS3 =	000052		SEEK	005324R	002	UNITMG	012306R	002
PORT =	002000		PTLGTH=	000242		RPOA =	000006		SETDED=	000040		UNLDIP=	000004	
PROIOR=	000016		PTCNT =	000030		RPOB =	000022		SETERR	006202R	002	UNLOAD	004144R	002
PRINT =	012122R	002	PTSIZE=	000240		RPOC =	000034		SETVEC	000066R	002	UNS =	040000	
PROCEX	007616R	002	PUSRPC=	000236		RPOS =	000012		SIZE	00002UR	002	URSTOP=	000002	
PROCTM	007502R	002	PUTBYT	000100R	002	RPO1 =	000026		SKCNT	002306R	002	USEURM=	000200	
PROGNM	012240R	002	PWRIOR=	000020		RPEC1 =	000044		SKCOM	005334R	002	USMTPS=	000002	
PROMER=	020000		P4CONS=	100000		RPEC2 =	000046		SP =	%000006		UXPATA	013655R	002
PRTX =	012234R	002	RDCNT =	002300R	002	RPER1 =	000014		SOPER=	000200		VV =	000100	
PRTIND	012052R	002	RDCOM	005172R	002	RPER2 =	000040		STEPON	003570R	002	VVFLC =	001000	
PS =	177776		RHD =	005242R	002	RPER3 =	000042		STEPEX	003470R	002	VVGF =	004036R	002
PSRC =	000120		RDPSET	004160R	002	RPLA =	000020		STEPUP	003422R	002	VVGN	004026R	002
PSRCST=	000022		READ =	005162R	002	RPMR =	000024		STADR	007466R	002	WAIT	003636R	002
PSTKCT=	000124		RECAL	005404R	002	RPOF =	000032		STANMG	013170R	002	WAITMD=	100000	
PSTKSV=	000126		REGNUM=	000026		RPOFV	002376R	002	STANUM	013200R	002	WCECNT	002334R	002
PSVREG=	000222		REL =	004166R	002	RPSN =	000030		STONER=	100000		WRCK	005266R	002
PSWD =	000030R	002	RENDMG	012271R	002	RPTBAS	002752R	002	STPCEX	003566R	002	WRCKHD	005316R	002
PTEM0 =	000056		REPORT	002530R	002	RPTEND	002776R	002	STPCOM	003506R	002	WRCNT	002302R	002
PTEM1 =	000060		REPTBL	003016R	002	RPTLP	002734R	002	STPC10	003546R	002	WRCOM	005222R	002
PTEM10=	000102		RESREG	011354R	002	RPWC =	000002		STSTAT	011412R	002	WRHD	005254R	002
PTEM11=	000104		RETCTR	005372R	002	RTNINT	000074R	002	SUIORG	005656R	002	WRITE	005212R	002
PTEM12=	000106		RETRYS	002336R	002	RTRY	000012R	002	SUPTAD	011372R	002	WT410T=	000010	
PTEM13=	000110		RHPINT	005766R	002	RTRYIP	002366R	002	SWIER=	000020		XXXX =	000000	
PTEM14=	000112		RINTEX	007652R	002	RTYEXH	013674R	002	SWOVER=	000010		. =	014002R	002

. ABS. 000000 000  
 000000 001  
 RJP11 014002 002

ERRORS DETECTED: 0  
 DEFAULT GLOBALS GENERATED: 0  
 \*, DTRPAB/NL:TOC/DOC=DTRPAB.P11  
 RUN-TIME: 8 17 1 SECONDS  
 RUN-TIME RATIO: 108/27=3.9  
 CORE USED: 6K (11 PAGES)  
 DOCUMENT PAGES: 57

