

RP02/03

DEVICE ROUTINE (MPG)
MD-11-DTR3A-A

EP-DTR3A-A DL-A
COPYRIGHT © 1976
FICHE 1 OF 1

NOV 1976
digital
MADE IN U.S.A

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16
17	18	19	20
21	22	23	24
25	26	27	28
29	30	31	32
33	34	35	36
37	38	39	40
41	42	43	44
45	46	47	48
49	50	51	52
53	54	55	56
57	58	59	60
61	62	63	64
65	66	67	68
69	70	71	72
73	74	75	76
77	78	79	80
81	82	83	84
85	86	87	88
89	90	91	92
93	94	95	96
97	98	99	100

IDENTIFICATION

DEC ASSUMES NO RESPONSIBILITY FOR USE OR RELIABILITY
OF ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY
DEC.

%

CO1

MAINDEC-11-DTR3A-A RPD2/RPD3 DEVICE ROUTINE FOR MPG MACY11 27(732) 24-SEP-76 14:11 PAGE 2
DTR3AA.P11 REVISION HISTORY

SEQ 0073

4555E
47

.SBTTL REVISION HISTORY

; JUL 76 DTR3A-A INITIAL RELEASE

.SBTTL STANDARD DEVICE ROUTINE TABLE

.TITLE MAINDEC-11-DTR3A-A RPO2/RPO3 DEVICE ROUTINE FOR MPG

;REVISION 'A'

;FILENAME OF "TR3A00.MPG" ON MPG/XXDP MEDIA

;MACY11: DTR3A',DTR3A'/CRF:SYM/DOC=DTR3A'.P11

;LNKX11: DTR3A'.MPG/B:0=DTR3A'/E

;PAPER TAPE: PUNCH DTR3A'.MPG/FILE:ELEV

000000'

.CSECT RP11
.DSABL GBL;THE FOLLOWING TABLE IS IN THE STANDARDIZED FORMAT REQUIRED
;TO INTERFACE WITH MPG.

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

LOCZ: .WORD DVREND-
 DFLGMD: .WORD 0
 WAITMD= 100000
 WAITTN= 4
 DETERM= 2
 IOERR= 1
 CYL: .WORD 0
 HEAD: .WORD 0
 SECT: .WORD 0
 RTRY: .WORD 3
 SIZE: .WORD 0
 ERR1: .WORD 0
 DREGAD: .WORD 176710
 IVCTAD: .WORD 254
 PSMD: .WORD 240
 CI0BSY: .WORD 0
 CUPGER: .WORD 0
 ULIST: .WORD 0
 CLIST: .WORD 0
 BINASC: .WORD 0
 BTASLZ: .WORD 0
 DECASC: .WORD 0
 CSYSFW: .WORD 0
 SETVEC: .WORD 0
 CLRVEC: .WORD 0
 TSTVEC: .WORD 0
 RTNINT: .WORD 0

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

EO1

MA:NDDEC-11-DTR3A-A RPD2/RPD3 DEVICE ROUTINE FOR MPG
DTR3AA.P11 STANDARD DEVICE ROUTINE TABLE

MACY11 27(732) 24-SEP-76 14:11 PAGE 3-1

SEQ 0075

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

.SBTTL COMPILER TABLES & CONSTANT AREAS

114					
115					
116					
117	000116	050122	051504	DVREGS:	.ASCII /RPOS/
118	000122	000000			.WORD 0
119	000124	050122	051105		.ASCII /RPER/
120	000130	000000			.WORD 2
121	000132	050122	051503		.ASCII /RPCS/
122	000136	000000			.WORD 4
123	000140	050122	041527		.ASCII /RPMC/
124	000144	000000			.WORD 6
125	000146	050122	040502		.ASCII /RPBA/
126	000152	000010			.WORD 10
127	000154	050122	040503		.ASCII /RPCA/
128	000160	000012			.WORD 12
129	000162	050122	040504		.ASCII /RPDA/
130	000166	000014			.WORD 14
131	000170	050122	030515		.ASCII /RPM1/
132	000174	000016			.WORD 16
133	000176	050122	031115		.ASCII /RPM2/
134	000202	000020			.WORD 20
135	000204	050122	031515		.ASCII /RPM3/
136	000210	000022			.WORD 22
137	000212	052523	040503		.ASCII /SUCA/
138	000216	000024			.WORD 24
139	000220	044523	047514		.ASCII /SILO/
140	000224	000026			.WORD 26
141		000226		DVREGS=	.
142					
143					
144	000226	120	201	DVCHDS:	.BYTE 120,201
145	000230	002232			.WORD READ-
146	000232	130	201		.BYTE 130,201
147	000234	002256			.WORD WRITE-
148	000236	376	000		.BYTE 376,0
149	000240	002072			.WORD NOWAIT-
150	000242	375	000		.BYTE 375,0
151	000244	002046			.WORD WAIT-
152	000246	374	000		.BYTE 374,0
153	000250	001166			.WORD REPORT-
154	000252	373	000		.BYTE 373,0
155	000254	001162			.WORD REPORT-
156	000256	372	000		.BYTE 372,0
157	000260	001632			.WORD STEPUP-
158	000262	371	000		.BYTE 371,0
159	000264	001760			.WORD STEPON-
160	000266	370	201		.BYTE 370,201
161	000270	002252			.WORD RDNOSK-
162	000272	367	201		.BYTE 367,201
163	000274	002260			.WORD WRNOSK-
164	000276	366	201		.BYTE 366,201
165	000300	002266			.WORD WRCK-
166	000302	365	000		.BYTE 365,0
167	000304	002312			.WORD SEEK-
168	000306	364	000		.BYTE 364,0
169	000310	002330			.WORD HOMESK-

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

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

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

;TABLE TERMINATOR

;PACK TABLE EXTENSION

;MODEL VECTOR TABLE EXTEN.

000604	000373	001140	.WORD	373,MSFMT1-LOC2
000610	000372	001164	.WORD	372,MSFMT5-LOC2
000614	000371	001164	.WORD	371,MSFMT5-LOC2
000620	000370	001146	.WORD	370,MSFMT3-LOC2
000624	000367	001155	.WORD	367,MSFMT4-LOC2
000630	000366	001141	.WORD	366,MSFMT2-LOC2
000634	000365	001140	.WORD	365,MSFMT1-LOC2
000640	000364	001140	.WORD	364,MSFMT1-LOC2
000644	000363	001140	.WORD	363,MSFMT1-LOC2
000650	000362	001140	.WORD	362,MSFMT1-LOC2
000654	000361	001140	.WORD	361,MSFMT1-LOC2
000660	000360	001140	.WORD	360,MSFMT1-LOC2
000664	000357	001140	.WORD	357,MSFMT1-LOC2
000670	000356	001140	.WORD	356,MSFMT1-LOC2
000674	000355	001140	.WORD	355,MSFMT1-LOC2

.....
 DVCPT: COMPILER TABLE EXTENSION

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

```

279 001100' 003 355 .BYTE 3,355 ;MODE 10
280 001102' 004537 000012 .WORD 4537,10.
281
282
283
284

```

```

285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330

```

DEVICE INTERFACE WORD SYMBOL TABLE

```

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

```

```

309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330

```

MODEL STATEMENT TABLE EXTENSION

```

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

```

```

309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330

```

;DEVICE ROUTINE CONSTANTS & EQUATES

```

MSKPST= .
ISTAT= . ;STORAGE FOR DEV REG'S AT INT

```

```

.WORD 0,0,0,0,0,0
.WORD 0,0,0,0,0

```

```

CSTAT: .BLKW 11. ;DEV REG CURRENT VALUES STORAGE

```

```

OBJADR: .WORD 0 ;CURR ADR IN USER PROG
CNTADR: .WORD 0 ;ADR OF BYTE COUNT TOTALS
CURFLG: .WORD 0 ;FLAG WORD OF CURR CMND
CURCMD: .WORD 0 ;CURR CMND CODE
CURADR: .WORD 0 ;CURR BUS ADDRESS
CURCNT: .WORD 0 ;NEG WORD CNT FOR CURR CMND
FINCNT: .WORD 0 ;FINAL WORD CNT (RPWC)
CURRTY: .WORD 0 ;CURR RETRY COUNT
RTRYIP: .WORD 0 ;RETRY IN PROGRESS FLAG

```

```

COUNTS:
BYRD: .WORD 0 ;BYTES READ COUNT
BYWR: .WORD 0 ;BYTES WRITTEN COUNT

```

331	001274'	000000			
332	001276'	000000	BYCK:	.WORD	0
333	001500'	000000			
334	001302'	000000	RDCNT:	.WORD	0
335	001304'	000000	WRCNT:	.WORD	0
336	001306'	000000	CKCNT:	.WORD	0
337	001310'	000000	SKCNT:	.WORD	0
338	001312'	000000	MISCNT:	.WORD	0
339	001314'	000000	ERRCNT:	.WORD	0
340	001316'	000000	DATAER:	.WORD	0
341	001320'	000000	TIECNT:	.WORD	0
342	001322'	000000	CSECNT:	.WORD	0
343	001324'	000000	WPECNT:	.WORD	0
344	001326'	000000	LPECNT:	.WORD	0
345	001330'	000000	WCECNT:	.WORD	0
346	001332'	000000	RETRY:	.WORD	0
347	001334'	000000	INTCNT:	.WORD	0
348					
349	001336'		HSKPEN=	.	
350					
351	001336'	000000	RPCSV:	.WORD	0
352		010000	MODE=	10000	
353		004000	HEADER=	4000	
354					
355					
356					
357					
358		100000	ERR=	100000	
359		040000	HE=	40000	
360					
361		000013	REGNUM=	11.	
362		000024	CNTNUM=	HSKPEN-COUNTS/2	
363					
364		000001	MMVER=	1	
365					
366		000000	XXXX=	0	
367					
368		177774	RPDS=	-4	
369		177776	RPER=	-2	
370		000000	RPCS=	0	
371		000002	RPWC=	+2	
372		000004	RPBA=	+4	
373		000006	RPCA=	+6	
374		000010	RPDA=	+10	
375		000012	RPM1=	+12	
376		000014	RPM2=	+14	
377		000016	RPM3=	+16	
378		000020	SUCA=	+20	
379		000022	SILO=	+22	
380					
381					
382	001340'		PATCH:	.REPT	10.
383				.WORD	0
384				.ENDR	

;BYTES CHECKED COUNT

;READ CMD COUNT

;WRITE CMD COUNT

;CHECK CMD COUNT

;SEEK CMD COUNT

;MISC. CMD COUNT

;DEVICE ERRORS COUNT

;DATA ERRORS COUNT

;TIMING ERRORS (DLT)

;CHECKSUM ERRORS

;WORD PARITY ERRORS

;LONGITUDINAL PARITY ERRORS

;WRITE CHECK ERRORS

;# OF RETRIES ON I/O CMDS

;INTERRUPTS COUNT

;BASE VALUE FOR RPCS REG

; MODE SELECT BIT - 1 = PDP-10/15

; HEADER BIT - 1 = HEADER OPERATION

DEVICE BIT EQUATES

;# OF DEVICE REGISTERS

;# OF STATISTICAL COUNT WORDS

;SYS FLGWD BIT DEFINITIONS

;VALUE TO BE TAILORED BY DEV ROUT

;EQUATES FOR DEVICE REGISTER NAMES

;RELATIVE TO R4'S CONTENTS

;PATCH AREA

K01

MAINDEC-11-DTR3A-A RP02/RP03 DEVICE ROUTINE FOR MPG MACY11 27(732) 24-SEP-76 14:11 PAGE 5
 DTR3AA.P11 RP02/RP03 SUPPORT ROUTINES ENTERED FROM MPG

SEQ 0081

.SBTTL RP02/RP03 SUPPORT ROUTINES ENTERED FROM MPG

;DEVICE ROUTINE HOUSEKEEPING

JSR RS,HSKEEP
 .WORD 0 OR 1
 R2 = PROG'S OPSW
 DESTROYS R0,R1

S/R CALL
 0 = DO HSKP PER OPSW
 1 = UNCOND. DO HSKP

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

#3 RTRY
 RPCSV
 PC,R0
 #HSKPST-.R0
 #HSKPEN-HSKPST/2,R1
 (R5)+
 #HSKPEP,R2
 10\$
 #CNTNUM,R1
 (R0)+
 R1
 10\$
 R5

;INIT # OF RETRY ATTEMPTS
 ;INITIALIZE RPCS VALUE
 ;SET UP FIRST WD ADR
 ;SET UP # OF WORDS
 ;UNCONDITIONALLY DO HSKP?
 ;N,Y-10\$
 ;OPSW SPECIFY DON'T HSKP COUNTS?
 ;Y,N-10\$
 ;REMOVE THEM FROM LOOP COUNT
 ;HSKP ALL NECESSARY AREAS

;EXIT IN-LINE

;RP02/RP03 REPORT ROUTINE

JSR RS,REPORT
 .WORD FLAGWORD
 :
 :
 :

S/R CALL
 FLAGWORD
 BIT 15 = CHND MODE CALL
 BIT 9 = PROG STMT CALL
 BIT 1 = DO STATUS REPORT
 BIT 0 = DO COUNTS REPORT

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

RO,SAVREG
 PC,SUPTAD
 (R5),R4
 #2,R4
 5\$
 RS,STSTAT
 CSTAT-
 #177776,R4
 15\$
 PC,R0
 #COUNTS-.R0
 #CNTNUM,R1
 (R0)+
 15\$
 R1
 10\$
 DVREX
 PC,DEVID
 #2,R4

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

L01

MAINDEC-11-OTR3A-A RP02/RP03 DEVICE ROUTINE FOR MPG MACY11 27(732) 24-SEP-76 14:11 PAGE 5-1
 OTR3AA.P11 RP02/RP03 SUPPORT ROUTINES ENTERED FROM MPG

SE0 0082

```

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

```

MO1

MAINDEC-11-CR3A-A RP02/RP03 DEVICE ROUTINE FOR MPG MACY11 27(732) 24-SEP-76 14:11 PAGE 5-2
 DTR3A6.P11 RP02/RP03 SUPPORT ROUTINES ENTERED FROM MPG

SEQ 0083

```

498 001746' 004607 .WORD CMDCSK-RPTBAS
499 001750' 004624 .WORD CMDCMS-RPTBAS
500 001752' 004652 .WORD CNTERR-RPTBAS
501 001754' 004667 .WORD CNTDER-RPTBAS
502 001756' 004717 .WORD CNTTIE-RPTBAS
503 001760' 004734 .WORD CNTCSE-RPTBAS
504 001762' 004750 .WORD CNTWPE-RPTBAS
505 001764' 004767 .WORD CNTLPE-RPTBAS
506 001766' 005003 .WORD CNTWCE-RPTBAS
507 001770' 005032 .WORD CNTRTY-RPTBAS
508 001772' 005060 .WORD CNTINT-RPTBAS

```

;TIMEOUT ERROR ROUTINE

;JSR RS,TOUTER

S/R CALL

```

517 001774' 004067 003346 TOUTER: JSR R0,SAVREG ;SAVE ALL REGISTERS
518 002000' 004767 003374 JSR PC,SUPTAD ;SET UP RPCS1 & PROG TBL ADR'S
519 002004' 004567 003414 JSR R5,STAT ;STORE CURRENT STATUS
520 002010' 177204 .WORD CSTAT-
521 002012' 004567 002332 JSR R5,TVECT ;CK IF I HAVE VECTOR CONTROL
522 002016' 000404 BR 10$ ;BR IF I DON'T
523 002020' 042714 020100 BIC #20100,(R4) ;RESET BOTH INT ENABLES
524 002024' 004767 002274 JSR PC,RINTV ;RESET THE INTERRUPT VECTOR
525 002030' 042713 000010 10$: BIC #20100,(R3) ;RESET WAITING FOR I/O FLG
526 002034' 004567 002346 JSR R5,ERRCSI ;ISSUE I/O TIMEOUT ERROR MSG
527 002040' 002441 .WORD IOTO-ERMBAS
528 002042' 004067 003314 JSR R0,RESREG ;RESTORE REGISTERS
529 002046' 012605 MOV (SP)+,R5 ;REMOVE RETURN ADR
530 002050' 000177 175774 JMP @CUPGER ;GO TO ERROR EXIT

```

;KILL USER PROGRAM ROUTINE

;JSR RS,KILL

S/R CALL

;R3 = PROG TBL ADR

;DESTROYS R0,R1

```

541 002054' 016701 175744 KILL: MOV DREGAD,R1 ;GET DEV REG ADR
542 002060' 062701 000004 ADD #4,R1 ;POINT AT RPCS REG
543 002064' 004567 002260 JSR R5,TVECT ;DO I HAVE VECTOR CONTROL?
544 002070' 000407 BR KILLEX ;BR IF I DON'T
545 002072' 032711 020100 BIT #20100,(R1) ;ARE INT ENABLES SET?
546 002076' 001402 BEQ 10$ ;Y,N-10$
547 002100' 042711 020100 BIC #20100,(R1) ;RESET INT ENABLES
548 002104' 004767 002214 10$: JSR PC,RINTV ;RESET INT VECTOR INFO
549 002110' 000205 KILLEX: RTS RS ;EXIT IN-LINE

```

.SBTTL RP02/RP03 NON-I/O & NON-INTERRUPT FUNCTION ROUTINES

;"STEPUP" FUNCTION ROUTINE

```

551                                     ;JSR    R5,STEPUP      FUNCTION CALL
552                                     ;.WORD  NBR          INCREMENT FACTOR
553
554                                     ;"STEPUP" FUNCTION ROUTINE
555
556                                     ;JSR    PC,STPCOM      ;DO COMMON SETUP
557                                     ;.WORD  NBR          ;ADD INCR VALUE TO SECT #
558                                     ;DO COMMON SETUP
559 002112' 004767 000060 STEPUP: JSR    PC,STPCOM      ;DO COMMON SETUP
560 002116' 062502          ADD    (R5)+,R2      ;ADD INCR VALUE TO SECT #
561 002120' 020203 10$: CMP    R2,R3          ;SECT # IN RANGE?
562 002122' 103403          BLO    15$          ;N,Y-15$
563 002124' 160302          SUB    R3,R2          ;ADJ SECT # DOWNWARDS
564 002126' 001201          INC    R1          ;ADD 1 TO HEAD #
565 002130' 000773          BR     10$          ;GO CK NEW SECT # VALUE
566 002132' 020127 000024 15$: CMP    R1,#20.    ;HEAD # IN RANGE?
567 002136' 103404          BLO    20$          ;N,Y-20$
568 002140' 162701 000024 SUB    #20.,R1        ;ADJ HEAD # DOWNWARDS
569 002144' 005200          INC    R0          ;ADD 1 TO CYL #
570 002146' 000771          BR     15$          ;GO CHECK NEW HEAD #
571 002150' 020004 20$: CMP    R0,R4          ;CYL # IN RANGE?
572 002152' 103402          BLO    STEPEX        ;N,Y-STEPEX
573 002154' 160400          SUB    R4,R0          ;ADJ CYL # DOWNWARDS
574 002156' 000774          BR     20$          ;GO CK IT NOW
575 002160' 010067 175620 STEPEX: MOV    R0,CYL    ;STORE NEW CYL,HEAD,SECT VALUES
576 002164' 010167 175616          MOV    R1,HEAD
577 002170' 010267 175614          MOV    R2,SECT
578 002174' 000205          RTS     R5
579                                     ;EXIT TO USER PROG
580
581 002176' 016700 175602 STPCOM: MOV    CYL,R0      ;GET CYL, HEAD, & SECT VALUES
582 002202' 016701 175600          MOV    HEAD,R1    ;IN REGISTERS
583 002206' 016702 175576          MOV    SECT,R2
584 002212' 004767 003162          JSR    PC,SUPTAD    ;GET PROG TBL ADR IN R3
585 002216' 012704 000313          MOV    #203.,R4    ;PRESET # OF CYL'S TO RP02
586 002222' 032763 000020 000032 BIT    #20,PMDLCD(R3) ;THIS AN RP03?
587 002230' 001402          BEQ    STPC10          ;Y,N-STPC10
588 002232' 012704 000626          MOV    #406.,R4    ;SET UP # OF RP03 CYL'S
589 002236' 012703 000012 STPC10: MOV    #10.,R3    ;SET UP # OF SECTORS
590 002242' 000207          RTS     PC              ;EXIT IN-LINE
591
592
593                                     ;"STEPDN" FUNCTION ROUTINE
594
595                                     ;JSR    R5,STEPDN      FUNCTION CALL
596                                     ;.WORD  NBR          DECREMENT FACTOR
597
598 002244' 004767 177726 STEPDN: JSR    PC,STPCOM    ;DO COMMON SETUP
599 002250' 162502          SUB    (R5)+,R2      ;SUB DECR FACTOR FROM SECT #
600 002252' 020203 40$: CMP    R2,R3          ;SECT # IN RANGE?
601 002254' 103403          BLO    45$          ;N,Y-45$
602 002256' 060302          ADD    R3,R2          ;ADJ SECT # UPWARDS
603 002260' 005301          DEC    R1          ;DECR HEAD # BY 1
604 002262' 000773          BR     40$          ;GO CK NEW SECT # VALUE
605 002264' 020127 000024 45$: CMP    R1,#20.    ;HEAD # IN RANGE?
606 002270' 103404          BLO    50$          ;N,Y-50$

```

```

607 002272' 062701 000024      ADD      #20.,R1      :ADJ IT UPWARDS
608 0 76' 007300      DEC      R0      :DECR CYL # BY 1
609 0 00' 00771      BR      45$      :GO CHECK NEW HEAD #
610 002302' 020004      SOS:      CMP      R0,R4      :CYL # IN RANGE?
611 002304' 103725      BLO      STEPEX      :N.Y-STEPEX
612 002306' 060400      ADD      R4,R0      :ADJ IT UPWARDS
613 002310' 000774      BR      50$      :GO CK IT NOW
614
615
616      ;"WAIT" FUNCTION ROUTINE
617
618      ;JSR      RS,WAIT      FUNCTION CALL
619
620 002312' 042767 100000 175462 WAIT: BIC      #WAITMD,DFLGMD      :RESET THE "NOWAIT" FLAG
621 002320' 004767 001552      JSR      PC,CKDBSY      :WAIT IF BUSY & DO TERMINATION
622 002324' 004767 001774      JSR      PC,RINTV      :RESET THE INTERRUPT VECTOR
623 002330' 000205      RTS      RS      :EXIT IN-LINE
624
625
626      ;"NOWAIT" FUNCTION ROUTINE
627
628      ;JSR      RS,NOWAIT      FUNCTION CALL
629
630 002332' 052767 100000 175442 NOWAIT: BIS      #WAITMD,DFLGMD      :SET THE "NOWAIT" FLAG
631 002340' 000205      RTS      RS      :EXIT IN-LINE
632
633
634      ;"H0RON" FUNCTION ROUTINE
635
636      ;JSR      RS,H0RON      FUNCTION CALL
637
638 002342' 052767 014000 176766 H0RON: BIS      #HEADER+MODE,RPCSV      :SET THE HEADER & MODE BITS
639 002350' 000205      RTS      RS      :EXIT IN-LINE
640
641
642      ;"H0ROFF" FUNCTION ROUTINE
643
644      ;JSR      RS,H0ROFF      FUNCTION CALL
645
646 002352' 042767 014000 176756 H0ROFF: BIC      #HEADER+MODE,RPCSV      :RESET THE HEADER & MODE BITS
647 002360' 000205      RTS      RS      :EXIT IN-LINE
648
649
650      ;"MODE11" FUNCTION ROUTINE
651
652      ;JSR      RS,MODE11      FUNCTION CALL
653
654 002362' 042767 010000 176746 MODE11: BIC      #MODE,RPCSV      :RESET THE MODE BIT
655 002370' 000205      RTS      RS      :EXIT IN-LINE

```

```

657                                     ;"MODE10" FUNCTION ROUTINE
658
659                                     ;JSR    RS,MODE10          FUNCTION CALL
660
661     002372' 052767 010000 176736 MODE10^ BIS      #MODE,RPCSV      ;SET THE MODE BIT
662     002400' 000205                                RTS        RS       ;EXIT IN-LINE
663
664                                     ;"IDLE" & "CRESET" FUNCTION ROUTINES
665
666                                     ;JSR    RS,IDLE           FUNCTION CALL
667                                     ;JSR    RS,CRESET         FUNCTION CALL
668
669
670     002402'                               IDLE:
671     002402' 004767 001470 CRESET: JSR    PC,CKOBSY      ;GO CK IF DEV IS BUSY
672     002406' 004767 176700             INC    MISCNT      ;ADD 1 TO MISC. CNTD CNT
673     002412' 005467 175404             CLR    ERRI        ;RESET THE ERROR INDICATOR
674     002416' 012714 000001             MOV    #1,(R4)      ;ISSUE THE IDLE FUNCTION CODE
675     002422' 012700 000012             MOV    #10.,RO      ;SET UP DELAY COUNT
676     002430' 005300 SS:                DEC    RO           ;DELAY FOR A FEW MICROSECONDS
677     002430' 001376             BNE    SS
678     002432' 012714 000001             MOV    #1,(R4)      ;ISSUE THE IDLE FUNCTION CODE
679     002436' 105714 10S:               TSTB   (R4)         ;READY SET YET?
680     002440' 100407             BMI    ZOS                ;N,Y-ZOS
681     002442' 005300             DEC    RO                 ;TIMED OUT?
682     002444' 100774             BMI    IOS                ;Y,N-IOS
683     002446' 004567 001726             JSR    RS,ERRCS      ;GO ISSUE CRESET TIMEOUT ERROR
684     002452' 002416             .WORD    CRTO-ERMB.C
685     002454' 000177 175370             JMP     @CUPGER
686     002460' 000205 ZOS:              RTS    RS            ;GO TO ERR RET POINT IN MPG

```

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

;"READ" FUNCTION ROUTINE

```

:JSR    RS, READ
:..WORD  ADR
:..WORD  ADR
:..WORD  CNT
:..WORD  DEV

```

```

FUNCTION CALL
DATA ADDRESS (BITS 16 - 17)
DATA ADDRESS (BITS 0 - 15)
BYTE COUNT
(NOT USED)

```

```

688
689
690
691
692
693
694
695
696
697
698
699 002462' 012702 000105
700 002466' 012701 000275
701 002472' 004767 001400
702 002476' 005267 176600
703 002502' 010700
704 002504' 062700 176564
705 002510' 000456
706
707
708
709
710
711
712
713
714
715
716 002512' 012702 000103
717 002516' 012701 000275
718 002522' 004767 001350
719 002526' 005267 176552
720 002532' 010700
721 002534' 062700 176540
722 002540' 000442
723
724
725
726
727
728
729
730
731
732 002542' 012702 000117
733 002546' 012701 000256
734 002552' 000747

```

```

READ:  MOV    #105, R2
        MOV    #275, R1
RDCOM: JSR     PC, CKDBSY
        INC     RDCNT
        MOV     PC, R0
        ADD     #BYRD+2-., R0
        BR      CHDCOM

```

```

:SET UP READ FUNCT CODE
:SET UP CHND FLAG WORD
:GO CK IF DEV IS BUSY
:ADD 1 TO READ CHND CNT
:SET UP ADR OF BYTES READ CNT
:GO TO CHND COMMON PROCESSING

```

;"WRITE" FUNCTION ROUTINE

```

:JSR    RS, WRITE
:..WORD  ADR
:..WORD  ADR
:..WORD  CNT
:..WORD  DEV

```

```

FUNCTION CALL
DATA ADDRESS (BITS 16 - 17)
DATA ADDRESS (BITS 0 - 15)
BYTE COUNT
(NOT USED)

```

```

WRITE:  MOV    #103, R2
        MOV    #275, R1
WRCOM: JSR     PC, CKDBSY
        INC     WRCNT
        MOV     PC, R0
        ADD     #BYWR+2-., R0
        BR      CHDCOM

```

```

:SET UP WRITE FUNCT CODE
:SET UP CHND FLAG WORD
:GO CK IF DEV IS BUSY
:ADD 1 TO WRITE CHND CNT
:SET UP ADR OF BYTES WRITTEN CNT
:GO TO CHND COMMON PROCESSING

```

;"RDNOSK" FUNCTION ROUTINE

```

:JSR    RS, RDNOSK
:..WORD  ADR
:..WORD  ADR
:..WORD  CNT

```

```

FUNCTION CALL
DATA ADDRESS (BITS 16 - 17)
DATA ADDRESS (BITS 0 - 15)
BYTE COUNT

```

```

RDNOSK: MOV    #117, R2
        MOV    #256, R1
        BR     RDCOM

```

```

:SET UP READ (NO SEEK) FUNCT CODE
:SET UP CHND FLAG WORD
:GO TO COMMON READ PROCESSING

```

```

736                                     ;"WRNOSK" FUNCTION ROUTINE
737
738                                     ;JSR      RS,WRNOSK      FUNCTION CALL
739                                     ;.WORD    ADR            DATA ADDRESS (BITS 16 - 17)
740                                     ;.WORD    ADR            DATA ADDRESS (BITS 0 - 15)
741                                     ;.WORD    CNT            BYTE COUNT
742
743 002554' 012702 000113      WRNOSK: MOV      #113,R2      ;SET UP WRITE (NO SEEK) FUNCT CODE
744 002560' 012701 000256      MOV      #256,R1      ;SET UP CHND FLAG WORD
745 002564' 000756      BR          WRNCOM      ;GO TO WRITE COMMON PROCESSING
746
747                                     ;"WRCK" FUNCTION ROUTINE
748
749                                     ;JSR      RS,WRCK      FUNCTION CALL
750                                     ;.WORD    ADR            DATA ADDRESS (BITS 16 - 17)
751                                     ;.WORD    ADR            DATA ADDRESS (BITS 0 - 15)
752                                     ;.WORD    CNT            BYTE COUNT
753
754 002566' 012702 000107      WRCK:  MOV      #107,R2      ;SET UP WRITE CHECK FUNCT CODE
755 002572' 012701 000276      MOV      #276,R1      ;SET UP CHND FLAG WORD
756 002576' 004767 001274      JSR      PC,CKDBSY      ;GO CK IF DEV IS BUSY
757 002582' 005267 176500      INC      CKCNT      ;ADD 1 TO CHECK CHND COUNT
758 002586' 010700      MOV      PC,R0      ;SET UP ADR OF BYTES
759 002590' 062700 176470      ADD      #BYCK+2--,R0      ;CHECKED COUNT
760 002594' 000414      BR          CHDCOM      ;GO TO CHND COM PROCESSING
761
762                                     ;"SEEK" FUNCTION ROUTINE
763
764                                     ;JSR      RS,SEEK      FUNCTION CALL
765
766 002616' 012702 000111      SEEK:  MOV      #111,R2      ;SET UP SEEK FUNCT CODE
767 002622' 012701 000160      SKCOM: MOV      #160,R1      ;SET UP CHND FLAG WORD
768 002626' 004767 001244      JSR      PC,CKDBSY      ;GO CK IF DEV IS BUSY
769 002632' 005267 176452      INC      SKCNT      ;ADD 1 TO SEEK CHND COUNT
770 002636' 000403      BR          CHDCOM      ;GO TO CHND COMMON PROCESSING
771
772                                     ;"HOMESK" & "RECAL" FUNCTION ROUTINES
773
774                                     ;JSR      RS,HOMESK      FUNCTION CALL
775                                     ;JSR      RS,RECAL      FUNCTION CALL
776
777 002640' 012702 000115      HOMESK: MOV      #115,R2      ;SET UP HOME SEEK FUNCT CODE
778 002644' 000766      RECAL:  BR          SKCOM      ;GO TO COMMON SEEK PROCESSING
779
780
781
782

```

; INTERRUPT TYPE I/O FUNCTION COMMON PROCESSING ROUTINE

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

; CMND FLAGWORD FORMAT:

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

784					
785					
786					
787					
788					
789					
790					
791					
792					
793					
794					
795					
796					
797					
798					
799					
800					
801					
802					
803					
804					
805	002646	010067	176372	CMDCOM: MOV	R0, CNTADR
806	002652	010167	176370	MOV	R1, CURFLG
807	002656	032701	000003	BIT	#3, R1
808	002662	001416		BEQ	10\$
809	002664	012567	176362	MOV	(R5)+, CURADR
810	002670	012567	176360	MOV	(R5)+, CURADR+2
811	002674	012500		MOV	(R5)+, R0
812	002676	000241		CLC	
813	002700	000000		ROR	R0
814	002702	005400		NEG	R0
815	002704	010067	176346	MOV	R0, CURCNT
816	002710	032701	003001	BIT	#1, R1
817	002714	001401		PEQ	10\$
818	002716	005725		TST	(R5)+
819	002720	116300	000035	10\$: MOV	PCURDV(R3), R0
820	002724	020027	000007	CMF	R0, #7
821	002730	101411		BLOS	20\$
822	002732	004567	001450	JSR	R5, ERRCS1
823	002736	002634		WORD	INVDVN-ERMBAS
824	002740	005267	176352	INC	DATAER
825	002744	005367	176344	DEC	ERRCNT
826	002750	000177	175074	15\$: JMP	2CUPGER
827	002754	000300		20\$: SWAB	R0
828	002756	050002		BIS	R0, R2
829	002760	056702	176352	BIS	RPCSV, R2
830	002764	010267	176260	MOV	R2, CURCMD
831	002770	000300		SWAB	R0
832	002772	110064	000001	MOV	R0, 1(R4)
833	002776	032764	040000 177774	BIT	#40000, RPDS(R4)
834	003004	001004		BNE	30\$
835	003006	004567	001366	JSR	R5, ERRCS
836	003012	002601		WORD	UNOFFL-ERMBAS
837	003014	000755		BR	15\$
838	003016	005764	177774	30\$: TST	RPDS(R4)
839	003022	100404		BMI	40\$

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

```

840 003024' 004567 001350      JSR      RS,ERRCS      ;ISSUE UNIT NOT RDY ERROR MSG
841 003030' 002517      .WORD    UNRDY-ERMBAS
842 003032' 000746      BR        15$
843 003034' 032764 001000 177774 40$: BIT      01000,RPDS(R4) ;GO TO ERROR EXIT
844 003042' 001404      BEQ       50$      ;IS THE UNIT UNSAFE?
845 003044' 004567 001330      JSR      RS,ERRCS      ;Y,N-50$
846 003050' 002460      .WORD    INITUS-ERMBAS      ;ISSUE UNSAFE ON INIT ERROR MSG
847 003052' 000736      BR        15$
848 003054' 006300      50$:  ASL      R0      ;GO TO ERROR EXIT
849 003056' 060700      ADD      PC,R0      ;USING THE UNIT #, INDEX INTO
850 003060' 062700 000136      ADD      @ATATBL-,R0 ;THE ATTN TBL FOR BITS FOR
851 003064' 112067 000146      MOVB     (R0)+,MYATA ;THIS UNIT #
852 003070' 111067 000144      MOVB     (R0),OTHATA ;STORE UNIT'S ATTN BIT
853 003074' 016400 177774      MOV      RPDS(R4),R0 ;STORE OTHER ATTN BITS
854 003100' 136700 000132      BITB     MYATA,R0 ;GET THE ATTN BITS
855 003104' 001403      BEQ       55$      ;IS MY ATTN ALREADY SET?
856 003106' 116764 000124 177774 55$: MOVB     MYATA,RPDS(R4) ;Y,N-55$
857 003114' 004767 000122      JSR      PC,SUIORG ;KNOCK IT DOWN
858 003120' 016767 174666 176134      MOV      RTRY,CURRTY ;GO SET UP REGS FOR I/O
859 003126' 000767 176132      CLR      RTRYIP ;INITIALIZE RETRY COUNT
860 003132' 012767 002504 001132      MOV      @IOTERM-ERMBAS,INTEAD ;CLEAR RETRY IN PROGRESS FLAG
861 003140' 042767 000005 174634      BIC      @IOERR+MYATTN,DFLGMD ;INIT TERMINATION ERROR MSG
862 003146' 002767 000002 174626      BIS      @IOTERM,DFLGMD ;HACK INT FLAGS
863 003154' 002713 000010      BIS      @WT4IOT,(R3) ;SET THE "PROCESS TERMINATION" FLAG
864 003160' 116764 176065 000001      MOVB     CURCMD+1,1(R4) ;SET WAITING FOR I/O TERM FLAG
865 003166' 110214      MOVB     R2,(R4) ;LOAD HIGH BYTE OF RPCS REG
866 003170' 005767 174606      TST      DFLGMD ;ISSUE SPECIFIED CMD
867 003174' 100203      BPL      60$      ;"NOWAIT" BIT SET?
868 003176' 042713 000010      BIC      @WT4IOT,(R3) ;Y,N-60$
869 003202' 000404      BR        70$      ;RESET WAITING FOR I/O TERM
870 003204' 004577 174636 60$: JSR      RS,@CIOBSY ;GO TO EXIT
871 003210' 004767 000762 70$: JSR      PC,PROCTH ;WAIT FOR I/O TO COMPLETE
872 003214' 000205      RTS       RS      ;GO PROCESS TERMINATION
873                                     ;EXIT IN-LINE TO USER PROG
874
875 003216' 001 376      ATATBL: .BYTE    001,376
876 003220' 002 375      .BYTE    002,375
877 003222' 004 373      .BYTE    004,373
878 003224' 010 367      .BYTE    010,367
879 003226' 020 357      .BYTE    020,357
880 003230' 040 337      .BYTE    040,337
881 003232' 100 277      .BYTE    100,277
882 003234' 200 177      .BYTE    200,177
883
884 003236' 000000      MYATA:  .WORD    0
885 003240' 000000      OTHATA:  .WORD    0

```

```

887                                     ;SET UP DEVICE REGS FOR I/O
888
889                                     ;JSR    PC,SUIORG      S/R CALL
890
891                                     ;R4 = RPCS ADR
892                                     ;R3 = PROG TBL ADR
893                                     ;R2 = CMND CODE
894                                     ;R1 = CMND FLAGWORD
895
896                                     ;DESTROYS R0
897
898 003242' 032701 000060      SUIORG: BIT    #60,R1      ;NEED TO SET UP CYL/HEAD/SECT?
899 003246' 001414      BEQ    10$      ;Y,N-10$
900 003250' 116764 174534 000010      MOVB   SECT,RPDA(R4) ;STORE SECT #
901 003256' 032701 000020      BIT    #20,R1      ;NEED CYL & HEAD #?
902 003262' 001406      BEQ    10$      ;Y,N-10$
903 003264' 116764 174516 000011      MOVB   HEAD,RPDA+1(R4) ;LOAD HEAD #
904 003272' 016764 174506 000006      MOV    CYL,RPDA(R4)  ;LOAD CYL #
905 003300' 032701 000004      10$: BIT    #4,R1      ;DATA XFER CMND?
906 003304' 001417      BEQ    20$      ;Y,N-20$
907 003306' 016700 175740      MOV    CURADR,R0      ;GET HIGH BITS OF ADR
908 003312' 042700 177774      BIC    #177774,R0      ;RESET BITS ABOVE A17
909 003316' 006300      ASL    R0      ;ALIGN BITS A16 & A17
910 003320' 006300      ASL    R0      ;TO THEIR POSITIONS IN RPCS
911 003322' 006300      ASL    R0
912 003324' 006300      ASL    R0
913 003326' 050002      BIS    R0,R2      ;SET THEM INTO CMND CODE WORD
914 003330' 016764 175720 000004      MOV    CURADR+2,RPDA(R4) ;LOAD BITS 0-15 OF ADR
915 003336' 016764 175714 000002      MOV    CURCNT,RPWC(R4) ;LOAD WORD COUNT
916 003344' 012763 072460 000030      20$: MOV    #30000.,PTOCNT(R3) ;INITIALIZE 10 SEC TIMEOUT CNT
917 003352' 000207      RTS    PC      ;EXIT IN-LINE

```

.SBTTL RP02/RP03 INTERRUPT SERVICE ROUTINE

```

919
920
921
922 003354' 004067 001766      RPINT: JSR    R0, SAVREG      ;SAVE ALL REGISTERS
923 003360' 004567 002040      JSR    RS, STSTAT      ;GO STORE ALL DEV REG'S
924 003364' 175602              WORD    ISTAT-
925 003366' 005267 175742      INC     INT CNT
926 003372' 004767 002002      JSR    PC, SUPTAD      ;ADD 1 TO INTERRUPT CNT
927 003376' 016701 175644      MOV     CURFLG, R1      ;SET UP PROG TBL & RPCS1 ADR'S
928 003402' 032767 000004 174372 BIT     MYATTN, DFLGWD      ;GET THIS CMND'S FLGWD
929 003410' 001032              BNE     ATNINT          ;WAITING FOR ATTN INT?
930 003412' 042714 000100      BIC     #100, (R4)        ;N, Y-ATNINT
931 003416' 016467 000002 175634 MOV     RPWC(R4), FINCNT      ;RESET INTERRUPT ENABLE
932 003424' 032714 140000      BIT     #140000, (R4)      ;STORE FINAL WORD COUNT
933 003430' 001055              BNE     ERRFND          ;'ERR' OR 'HE' SET?
934 003432' 032701 000100      BIT     #100, R1          ;N, Y-ERRFND
935 003436' 001411              BEQ     CLRWTF          ;CMND RESULT IN TWO INTERRUPTS?
936 003440' 052767 000004 174334 BIS     MYATTN, DFLGWD      ;Y, N-CLRWTF
937 003446' 052714 020000      BIS     #20000, (R4)       ;SET 'WAITING FOR ATTN' FLAG
938 003452' 000405              BR      INTEX           ;SET ATTN INTERRUPT ENABLE
939 003454' 052767 000001 174320 SETERR: BIS     #IOERR, DFLGWD      ;GO TO INTERRUPT EXIT
940 003462' 042713 000010      CLRWTF: BIC     MYATTN, (R3)      ;SET THE TERMINATION I/O ERR FLAG
941 003466' 004067 001670      INTEX: JSR    R0, RESREG      ;RESET WAITING FOR I/O TERM
942 003472' 000177 174376      JMP     RTNINT          ;RESTORE ALL REGISTERS
943
944 003476' 042714 020000      ATNINT: BIC     #20000, (R4)       ;EXIT FROM INTERRUPT
945 003502' 032714 140000      BIT     #140000, (R4)       ;RESET ATTN INT ENABLE
946 003506' 001362              BNE     SETERR          ;'ERR' OR 'HE' SET?
947 003510' 136764 177522 177774 BITB     MYATA, RPDS(R4)      ;N, Y-SETERR
948 003516' 001404              BEQ     10$             ;IS MY ATTN BIT SET?
949 003520' 116764 177512 177774 MOVB     MYATA, RPDS(R4)      ;Y, N-10$
950 003526' 000755              BR      CLRWTF          ;RESET MY ATTN BIT
951 003530' 136764 177504 177774 10$: BITB     OTHATA, RPDS(R4)      ;GO TO INT EXIT
952 003536' 001406              BEQ     20$             ;IS ANOTHER ATTN BIT SET?
953 003540' 116764 177474 177774 MOVB     OTHATA, RPDS(R4)      ;Y, N-20$
954 003546' 052714 020000      BIS     #20000, (R4)       ;RESET OTHER ATTN BITS
955 003552' 000745              BR      INTEX           ;SET ATTN INT ENABLE AGAIN
956 003554' 012767 002532 000510 20$: MOV     MYATA-ERMBAS, INTEAD      ;GO TO INT EXIT
957 003562' 000734              BR      SETERR          ;SET UP 'ATTN NOT SET' ERR MSG ADR
958
959 003564' 032764 177407 177776 ERRFND: BIT     #177407, RPER(R4)      ;GO SET ERROR FLAG & EXIT
960 003572' 001330              BNE     SETERR          ;ANY HARD ERROR BITS IN RPER?
961 003574' 016400 177774      MOV     RPDS(R4), R0      ;N, Y-SETERR
962 003600' 032700 015000      BIT     #15000, R0          ;GET THE RPDS REG
963 003604' 001323              BNE     SETERR          ;ANY HARD ERROR BITS IN RPDS?
964 003606' 005100              COM     R0              ;N, Y-SETERR
965 003610' 032700 140000      BIT     #140000, R0          ;FLIP RPDS BITS
966 003614' 001317              BNE     SETERR          ;STILL ON-LINE & READY?
967 003616' 016400 177776      MOV     RPER(R4), R0      ;Y, N-SETERR
968 003622' 032700 000370      BIT     #370, R0          ;GET THE RPER REG
969 003626' 001712              BEQ     SETERR          ;RETRYABLE ERROR?
970 003630' 032701 000200      BIT     #200, R1          ;Y, N-SETERR
971 003634' 001707              BEQ     SETERR          ;DO RETRIES ON THIS CMND?
972
973 003636' 005767 175422      CKRTRY: TST     RTRYIP          ;Y, N-SETERR
974 003642' 001070              BNE     55$             ;ALREADY DONE RETRIES ON THIS CMND?

```

975	003644'	005767	175412		TST	CURRTY	:ARE RETRIES SPECIFIED?
976	003650'	001701			BEQ	SETERR	:Y,N-SETERR
977	003652'	032700	000020		BIT	#20,R0	:TIMING ERROR?
978	003656'	001411			BEQ	42\$	:Y,N-42\$
979	003660'	005267	175434		INC	TIECNT	:ADD 1 TO TIMING ERROR COUNT
980	003664'	012767	046524	003434	MOV	#TH,RTYID	:SET RETRY I.D. TO "TME"
981	003672'	012767	042505	003430	MOV	#EE,RTYID+2	
982	003700'	000451			BR	55\$	:GO CK RETRY COUNT
983	003702'	032700	000040	42\$:	BIT	#40,R0	:CHECKSUM ERROR?
984	003706'	001411			BEQ	44\$	:Y,N-44\$
985	003710'	005267	175406		INC	CSECNT	:ADD 1 TO CHECKSUM ERROR COUNT
986	003714'	012767	051503	003404	MOV	#CS,RTYID	:SET RETRY I.D. TO "CSME"
987	003722'	012767	042515	003400	MOV	#ME,RTYID+2	
988	003730'	000435			BR	55\$	:GO CK RETRY COUNT
989	003732'	032700	000100	44\$:	BIT	#100,R0	:WORD PARITY ERROR?
990	003736'	001406			BEQ	46\$	:Y,N-46\$
991	003740'	005267	175360		INC	WPECNT	:ADD 1 TO WORD PARITY ERROR COUNT
992	003744'	012767	050127	003354	MOV	#WP,RTYID	:SET RETRY I.D. TO "WPE"
993	003752'	000421			BR	50\$	:GO CK RETRY COUNT
994	003754'	032700	000200	46\$:	BIT	#200,R0	:LONGITUDINAL PARITY ERROR?
995	003760'	001406			BEQ	48\$	:Y,N-48\$
996	003762'	005267	175340		INC	LPECNT	:ADD 1 TO LONGITUDINAL PARITY COUNT
997	003766'	012767	050114	003332	MOV	#LP,RTYID	:SET RETRY I.D. TO "LPE"
998	003774'	000410			BR	50\$	:GO CK RETRY COUNT
999	003776'	032700	000010	48\$:	BIT	#10,R0	:WRITE CHECK ERROR?
1000	004002'	001410			BEQ	55\$	:Y,N-55\$
1001	004004'	005267	175320		INC	WCECNT	:ADD 1 TO WRITE CHECK ERROR COUNT
1002	004010'	012767	041527	003310	MOV	#WC,RTYID	:SET RETRY I.D. TO "WCE"
1003	004016'	012767	020105	003304	MOV	#E,RTYID+2	
1004	004024'	005367	175232	50\$:	DEC	CURRTY	:DECREMENT RETRY COUNT
1005	004030'	100004		55\$:	BPL	60\$	:CNT EXHAUSTED? (Y,N-60\$)
1006	004032'	012767	002547	000232	MOV	#RTYEXH-ERMBAS,INTEND	:SET UP EXHAUSTED RETRIES ERR MSG ADDR
1007	004040'	000605			BR	SETERR	:GO TO ERR R EXIT
1008	004042'	005267	175264	60\$:	INC	RETRY\$	:ADD 1 TO RETRY TOTAL CNT
1009	004046'	005267	175212		INC	RTRYIP	:SET RETRY IN PROGRESS FLAG
1010	004052'	016702	175172		MOV	CURCMD,R2	:GET CURR CMD IN R2
1011	004056'	004767	177160		JSR	PC,SUIORG	:SET UP DEV REGS
1012	004062'	116764	175163	000001	MOVB	CURCMD+1,1(R4)	:LOAD HIGH BYTE OF RPCS
1013	004070'	110214			MOVB	R2,(R4)	:RE-ISSUE THE ORIG CMD
1014	004072'	000167	177370		JMP	INTX	:GO TO INT EXIT

```

1016 .SBTTL SUBROUTINES FOR RP02/RP03 FUNCTION ROUTINES
1017
1018 ;CHECK IF DEVICE IS BUSY AND WAIT IF IT IS
1019
1020 ;JSR PC,CKDBSY S/R CALL
1021
1022 ;DESTROYS R0,R3,R4
1023
1024 ;ON EXIT: R3 = PROG TBL ADR
1025 ; R4 = RPCS1 ADR
1026
1027
1028 004076' 004767 001276 CKDBSY: JSR PC,SUPTAD ;SET UP PROG TBL & RPCS1 ADR'S
1029 004102' 032714 020100 10S: BIT #20100,(R4) ;EITHER INT ENABLE ON?
1030 004106' 001403 173732 20S: BEQ 20S ;Y,N-20S
1031 004110' 004577 173732 JSR RS,20C10BSY ;RELEASE CONTROL
1032 004114' 000772 173656 20S: BR 10S ;GO CK AGAIN
1033 004116' 032767 000002 173656 20S: BIT #00TERM,DFLGWD ;HAVE TO PROCESS PREV TERMINATION?
1034 004124' 001403 000044 30S: BEQ 30S ;Y,N-30S
1035 004126' 004767 000044 JSR PC,PROCTM ;GO PROCESS TERMINATION
1036 004132' 000763 173666 000012 30S: BR 10S ;GO CK INT ENABLE AGAIN
1037 004134' 016767 173662 000006 30S: MOV IVCTAD,40S ;STORE INT VECTOR ADR
1038 004142' 016767 173712 40S: MOV PSWD,45S ;STORE PROC STATUS WORD
1039 004150' 004577 173712 JSR RS,2SETVEC ;GO SET UP INTERRUPT VECTOR
1040 004154' 000000 40S: .WORD XXXX ;INT VECTOR ADR
1041 004156' 000000 45S: .WORD XXXX ;PSW
1042 004160' 177174 .WORD RPINT- ;REL INT ROUT ADR
1043 004162' 010567 175054 STMADR: MOV RS,0BJADR ;SAVE CURR USER STMT ADR
1044 004166' 162767 000004 175046 SUB #4,0BJADR
1045 004174' 000207 RTS PC ;EXIT IN-LINE
1046
1047 ;PROCESS TERMINATION OF PREVIOUS I/O FUNCTION
1048
1049 ;JSR PC,PROCTM S/R CALL
1050
1051 ;R3 = PROG TABLE ADR
1052
1053 ;DESTROYS R0
1054
1055
1056 004176' 010146 PROCTM: MOV R1,-(SP) ;SAVE R1 & R2
1057 004200' 010246 MOV R2,-(SP)
1058 004202' 042767 000002 173572 BIC #00TERM,DFLGWD ;RESET PROCESS TERMINATION FLAG
1059 004210' 032767 000010 175030 BIT #10,CURFLG ;INCR BYTE COUNT?
1060 004216' 001417 175032 20S: BEQ 20S ;Y,N-20S
1061 004220' 016700 175032 MOV CURCNT,R0 ;GET INITIAL WORD CNT
1062 004224' 005400 NEG R0 ;MAKE IT POSITIVE AGAIN
1063 004226' 016701 175026 MOV FINCNT,R1 ;GET FINAL WORD CNT
1064 004232' 100001 BPL 10S ;IS IT NEGATIVE? (Y,N-10S)
1065 004234' 005401 NEG R1 ;MAKE IT POSITIVE
1066 004236' 160100 10S: SUB R1,R0 ;SUB REMAINING CNT FROM INITIAL CNT
1067 004240' 006300 ASL R0 ;MAKE IT A BYTE CNT
1068 004242' 010067 173552 MOV R0,SIZE ;STORE # OF BYTES ACTUALLY XFERRED
1069 004246' 016701 174772 MOV CNTADR,R1 ;GET ADR OF BYTE CNT TOTALS
1070 004252' 060011 ADD R0,(R1) ;ADD IN THIS CNT
1071 004254' 005541 ADC -(R1) ;UPDATE MOST SIGNF WORD OF CNT

```

```

1072 004256' 032767 000001 173516 20$: BIT      #IOERR,DFLGND      ;WAS THERE AN ERROR?
1073 004264' 001412      BEQ      PROCEX      ;Y,N-PROCEX
1074 004266' 004557 000132      JSR      RS,EPRIS      ;GO ISSUE I/O TERMINATION
1075 004272' 002504      INTEND: .WORD    ITERM-ERMBAS      ;ERROR MSG
1076 004274' 004767 000024      JSR      PC,RINTV      ;RESET THE INT VECTOR
1077 004300' 012602      MOV      (SP)+,R2      ;RESTORE R1 & R2
1078 004302' 012601      MOV      (SP)+,R1
1079 004304' 004577 173540      JSR      RS,2CUPGER      ;GO TO MPG ERR RETN POINT
1080 004310' 000207      RTS      PC      ;RETURN IN-LINE
1081 004312' 004767 000006      PROCEX: JSR      PC,RINTV      ;GO RESET INT VECTOR
1082 004316' 012602      MOV      (SP)+,R2      ;RESTORE R1 & R2
1083 004320' 012601      MOV      (SP)+,R1
1084 004322' 000207      RTS      PC      ;EXIT IN-LINE
1085
1086
1087      ;RESET INTERRUPT VECTOR S/R
1088
1089      ;JSR      PC,RINTV      S/R CALL
1090
1091      ;R3 = PROG TBL ADR
1092
1093      ;DESTROYS R0
1094
1095 004324' 004567 000020      RINTV: JSR      RS,TVECT      ;GO CK IF I HAVE VECTOR CONTROL
1096 004330' 000406      BR      RINTEX      ;BR IF I DON'T
1097 004332' 016767 173470 000004      MOV      IVCTAD,10$      ;GET CURR INT VECT ADR
1098 004340' 004577 173524      JSR      RS,2CLAVEC      ;GO HAVE MPG CLEAR IT
1099 004344' 000600      10$: .WORD    XXXX
1100 004346' 000207      RINTEX: RTS      PC      ;EXIT IN-LINE
1101
1102
1103      ;TEST INTERRUPT VECTOR S/R
1104
1105      ;JSR      RS,TVECT      S/R CALL
1106      ;BR      LABEL      EXECUTED IF NOT SAME
1107
1108      ;R3 = PROG TBL ADR
1109
1110      ;DESTROYS R0
1111
1112 004350' 016767 173452 000010      TVECT: MOV      IVCTAD,20$      ;GET CURR INT VECT ADR
1113 004356' 016346 000004      MOV      PFWADR(R3),-(SP)      ;STORE FLGND ADR TO IDENTIFY ME
1114 004362' 004577 173504      JSR      RS,2TSTVEC      ;DO I HAVE VECTOR CONTROL?
1115 004366' 000000      20$: .WORD    XXXX      ;MPG WILL TELL ME SINCE I CAN'T
1116 004370' 176764      .WORD    RPINT-      ;GET AT LOWER MEM IF MEM MGMT
1117 004372' 000401      BR      TVECTX      ;BR IF I DON'T HAVE CNTRL
1118 004374' 005725      TST      (RS)+      ;BYPASS BR INST IN S/R CALL
1119 004376' 000205      TVECTX: RTS      RS      ;EXIT IN-LINE

```

```

1121                                     ;ERROR INFORMATION DISPLAY S/R
1122
1123                                     ;JSR    RS,ERRCS          S/R CALL FOR CURR STATUS
1124                                     ;ERRCS1          S/R CALL FOR CURR STATUS W/O STORING
1125                                     ;ERRIS          S/R CALL FOR INT STATUS
1126                                     ;.WORD MSGADR-ERMBAS REL ADR OF ERROR MSG
1127
1128                                     ;R3 = PROG TABLE ADR
1129                                     ;DESTROYS R0,R1,R2
1130
1131 004400' 004567 001020 ERRCS: JSR    RS,STSTAT          ;STORE CURR STATUS
1132 004404' 174610          .WORD CSTAT-                ;
1133 004406' 012767 174246 000332 ERRCS1: MOV    #CSTAT-ERSTAD,ERSTAD ;STORE ADR OF CURR STATUS
1134 004414' 012767 174436 000144          MOV    #CSTAT+6-EBSBAS,EBSTAT ;
1135 004422' 004406          BR      ERRCOM                ;GO TO COMMON POINT
1136 004424' 012767 174220 000314 ERRIS: MOV    #ISTAT-ERSTAD,ERSTAD ;STORE ADR OF LAST INT STATUS
1137 004432' 012767 174410 000126          MOV    #ISTAT+6-EBSBAS,EBSTAT ;
1138 004440' 012567 000066          ERRCOM: MOV    (RS)+,ERMBAS        ;STORE MSG ADR
1139 004444' 005267 174644          INC    ERRCNT          ;ADD 1 TO ERROR CNT
1140 004450' 012767 000001 173344          MOV    #1,ERRPRI          ;SET THE ERROR INDICATOR
1141 004456' 032763 020400 000002          BIT    #DOERCK+PRONER,POPSW(R3) ;ERROR CHECKING OR PRINTING INHIBITED?
1142 004464' 001163          BNE    ERREX                ;Y N-ERREX
1143 004466' 010446          MOV    R4,-(SP)              ;SAVE R4 & RS
1144 004470' 010546          MOV    RS,-(SP)
1145 004472' 005304          CLR    R4                    ;SET USER MODE PRINT FLAG
1146 004474' 004767 000750          JSR    PC,DEVID        ;DISPLAY DEVICE I.D.
1147 004500' 010700          MOV    PC,R0                ;GET START ADR OF ERROR MSG
1148 004502' 062700 000030          ADD    #ERMBAS-.,R0
1149 004506' 061000          ADD    (R0),R0
1150 004510' 012701 177777          MOV    #-1,R1
1151 004514' 005201          10$: INC    R1                ;INITIALIZE MSG LENGTH
1152 004516' 105720          TSTB   (R0)+                ;ADD 1 TO MSG LENGTH
1153 004520' 001375          BNE    10$                    ;MSG TERMINATOR?
1154 004522' 010167 000006          MOV    R1,ERMBAS+2    ;Y N-10$
1155 004526' 004567 001266          JSR    RS,PRINT        ;STORE MSG LENGTH
1156 004532' 000000          ERMBAS: .WORD XXXX            ;PRINT ERROR MSG SPECIFIED
1157 004534' 000000          .WORD XXXX
1158 004536' 026727 177770 002634          CMP    ERMBAS,#INVDVN-ERMBAS ;INVALID UNIT # MSG OR HIGHER?
1159 004544' 103103          BHIS   ERRSNM                ;N,Y-ERRSNM
1160 004546' 010701          MOV    PC,R1                ;GET ADR OF CODE AREA IN ERR MSG
1161 004550' 062701 002300          ADD    #C00FLD-.,R1
1162 004554' 010700          MOV    PC,R0                ;SET UP ADR OF ERROR CODE TBL
1163 004556' 062700 000260          ADD    #ERCOTB-.,R0
1164 004562' 010702          MOV    PC,R2                ;SET UP ADR OF STORED DEV REG'S
1165 004564' 062702          EBSBAS: ADD    (PC)+,R2
1166 004566' 174436          EBSTAT: .WORD CSTAT+6-EBSBAS ;
1167 004570' 012767 000015 000142          MOV    #13.,70$          ;INITIALIZE MSG LENGTH
1168 004576' 012746 000100          MOV    #64,-(SP)        ;INITIALIZE CODE FIELD CNT
1169 004602' 014205          15$: MOV    -(R2),R5          ;GET NEXT DEV REG WORD
1170 004604' 000305          17$: SWAB   R5                ;GET DESIRED BYTE IN LOW BYTE
1171 004606' 112004          20$: MOVB   (R0)+,R4          ;GET FLAG & LENGTH BYTE
1172 004610' 001444          BEQ     60$                    ;END OF CODE TBL? (N,Y-60$)
1173 004612' 122704 000377          CMPB   #377,R4        ;GO TO NXT DEV REG WORD?
1174 004616' 001771          BEQ     15$                    ;N,Y-15$
1175 004620' 122704 000376          CMPB   #376,R4        ;GO TO NXT BYTE IN DEV REG WORD?
1176 004624' 001767          BEQ     17$                    ;N,Y-17$

```

Line	Address	Hex	Label	Op	Op2	Op3	Op4	Op5	Op6	Op7	Op8	Op9	Op10	Op11	Op12	Op13	Op14	Op15	Op16	Op17	Op18	Op19	Op20	Op21	Op22	Op23	Op24	Op25	Op26	Op27	Op28	Op29	Op30	Op31	Op32	Op33	Op34	Op35	Op36	Op37	Op38	Op39	Op40	Op41	Op42	Op43	Op44	Op45	Op46	Op47	Op48	Op49	Op50	Op51	Op52	Op53	Op54	Op55	Op56	Op57	Op58	Op59	Op60	Op61	Op62	Op63	Op64	Op65	Op66	Op67	Op68	Op69	Op70	Op71	Op72	Op73	Op74	Op75	Op76	Op77	Op78	Op79	Op80	Op81	Op82	Op83	Op84	Op85	Op86	Op87	Op88	Op89	Op90	Op91	Op92	Op93	Op94	Op95	Op96	Op97	Op98	Op99	Op100	Op101	Op102	Op103	Op104	Op105	Op106	Op107	Op108	Op109	Op110	Op111	Op112	Op113	Op114	Op115	Op116	Op117	Op118	Op119	Op120	Op121	Op122	Op123	Op124	Op125	Op126	Op127	Op128	Op129	Op130	Op131	Op132	Op133	Op134	Op135	Op136	Op137	Op138	Op139	Op140	Op141	Op142	Op143	Op144	Op145	Op146	Op147	Op148	Op149	Op150	Op151	Op152	Op153	Op154	Op155	Op156	Op157	Op158	Op159	Op160	Op161	Op162	Op163	Op164	Op165	Op166	Op167	Op168	Op169	Op170	Op171	Op172	Op173	Op174	Op175	Op176	Op177	Op178	Op179	Op180	Op181	Op182	Op183	Op184	Op185	Op186	Op187	Op188	Op189	Op190	Op191	Op192	Op193	Op194	Op195	Op196	Op197	Op198	Op199	Op200	Op201	Op202	Op203	Op204	Op205	Op206	Op207	Op208	Op209	Op210	Op211	Op212	Op213	Op214	Op215	Op216	Op217	Op218	Op219	Op220	Op221	Op222	Op223	Op224	Op225	Op226	Op227	Op228	Op229	Op230	Op231	Op232	Op233	Op234	Op235	Op236	Op237	Op238	Op239	Op240	Op241	Op242	Op243	Op244	Op245	Op246	Op247	Op248	Op249	Op250	Op251	Op252	Op253	Op254	Op255	Op256	Op257	Op258	Op259	Op260	Op261	Op262	Op263	Op264	Op265	Op266	Op267	Op268	Op269	Op270	Op271	Op272	Op273	Op274	Op275	Op276	Op277	Op278	Op279	Op280	Op281	Op282	Op283	Op284	Op285	Op286	Op287	Op288	Op289	Op290	Op291	Op292	Op293	Op294	Op295	Op296	Op297	Op298	Op299	Op300	Op301	Op302	Op303	Op304	Op305	Op306	Op307	Op308	Op309	Op310	Op311	Op312	Op313	Op314	Op315	Op316	Op317	Op318	Op319	Op320	Op321	Op322	Op323	Op324	Op325	Op326	Op327	Op328	Op329	Op330	Op331	Op332	Op333	Op334	Op335	Op336	Op337	Op338	Op339	Op340	Op341	Op342	Op343	Op344	Op345	Op346	Op347	Op348	Op349	Op350	Op351	Op352	Op353	Op354	Op355	Op356	Op357	Op358	Op359	Op360	Op361	Op362	Op363	Op364	Op365	Op366	Op367	Op368	Op369	Op370	Op371	Op372	Op373	Op374	Op375	Op376	Op377	Op378	Op379	Op380	Op381	Op382	Op383	Op384	Op385	Op386	Op387	Op388	Op389	Op390	Op391	Op392	Op393	Op394	Op395	Op396	Op397	Op398	Op399	Op400	Op401	Op402	Op403	Op404	Op405	Op406	Op407	Op408	Op409	Op410	Op411	Op412	Op413	Op414	Op415	Op416	Op41
------	---------	-----	-------	----	-----	-----	-----	-----	-----	-----	-----	-----	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	------

;ERROR MESSAGE CODE TABLE

;377 = GO TO NEXT DEVICE REGISTER WORD
 ;376 = GO TO NEXT DEVICE REGISTER BYTE
 ;BYTE 0 CONTAINS FLAG BIT & I.D. NAME LENGTH
 ;BITS 0-2 = LENGTH OF BIT MASK + I.D. NAME
 ;BIT 7 = BIT = 0 IS AN ERROR CONDITION
 ;BYTE 1 IS THE BIT MASK
 ;BYTES 2 THRU 9 ARE THE BIT'S ASCII I.D.

1229							
1230							
1231							
1232							
1233							
1234							
1235							
1236							
1237							
1238							
1239							
1240							
1241							
1242							
1243	005036	100004	051105	122	ERCOTB:	.ASCII	<004><200>/ERR/
1244	005043	003	044100	105		.ASCII	<003><100>/HE/
1245	005047	377				.BYTE	377
1246	005050	100004	050127	126		.ASCII	<004><200>/MPV/
1247	005055	004	043100	053125		.ASCII	<004><100>/FUV/
1248	005057	020074	054116	103		.ASCII	<004><040>/NXC/
1249	005074	004004	047020	052130		.ASCII	<004><020>/NXT/
1250	005101	006	054116	123		.ASCII	<004><010>/NXS/
1251	005106	042507	050004	047522		.ASCII	<006><004>/PROGE/
1252	005110	001005	046506	042524		.ASCII	<005><002>/FMTE/
1253	005116	000406	047515	042504		.ASCII	<006><001>/MODEE/
1254	005124	105					
1255	005125	376				.BYTE	376
1256	005128	100004	050114	105		.ASCII	<004><200>/LPE/
1257	005133	004	053500	042520		.ASCII	<004><100>/MPE/
1258	005140	020005	051503	042515		.ASCII	<005><040>/CSME/
1259	005146	010006	044524	042515		.ASCII	<006><020>/TIMEE/
1260	005154	105					
1261	005155	004	053410	042503		.ASCII	<004><010>/WCE/
1262	005162	002005	054116	042515		.ASCII	<005><004>/NDME/
1263	005170	001005	047505	042520		.ASCII	<005><002>/EOPE/
1264	005176	000407	051504	042513		.ASCII	<007><001>/DSKERR/
1265	005204	051122					
1266	005206	377				.BYTE	377
1267	005207	206	051600	051125		.ASCII	<206><200>/SURDY/
1268	005214	054504					
1269	005216	040205	052523	046117		.ASCII	<205><100>/SUOL/
1270	005224	010004	047110	106		.ASCII	<004><020>/HNF/
1271	005231	005	051410	051525		.ASCII	<005><010>/SUSI/
1272	005236	111					
1273	005237	005	051402	043125		.ASCII	<005><002>/SUFU/
1274	005244	125					
1275	005245	005	051401	053525		.ASCII	<005><001>/SUMP/
1276	005252	120					
1277	005253	376				.BYTE	376
1278	005254	100006	052101	047124		.ASCII	<006><200>/ATTN7/
1279	005262	067					
1280	005263	006	040500	052124		.ASCII	<006><100>/ATTN6/
1281	005270	033116					
1282	005272	020006	052101	047124		.ASCII	<006><040>/ATTN5/
1283	005300	065					

;RPCS - BYTE 1

;RPER - BYTE 1

;RPER - BYTE 0

;RPDS - BYTE 1

;RPDS - BYTE 0

1274	005301'	006	040420	052124	.ASCII	<006><020>/ATTN4/	
	005306'	032116					
1275	005310'	004006	052101	047124	.ASCII	<006><010>/ATTN3/	
	005316'	063					
1276	005317'	006	040404	052124	.ASCII	<006><004>/ATTN2/	
	005324'	031116					
1277	005326'	001006	052101	047124	.ASCII	<006><002>/ATTN1/	
	005334'	061					
1278	005335'	006	040401	052124	.ASCII	<006><001>/ATTN0/	
	005342'	030116					
1279	005344'	000			.BYTE	0	;TABLE TERMINATOR
1280	005346'				.EVEN		

```

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

```

E03

MAINDEC-11-DTR3A-A RPO2/RPO3 DEVICE ROUTINE FOR MPG MACY11 27(732) 24-SEP-76 14:11 PAGE 10-1
DTR3AA.P11 SUBROUTINES FOR RPO2/RPO3 DEVICE ROUTINE

SEQ 0101

```

1338
1339
1340
1341
1342
1343
1344
1345
1346
1347 005450 012700 031060 DEVID: MOV #02,R0 ;INITIALIZE TO RPO2
1348 005454 032763 000020 000032 BIT #20,PHOLD(R3) ;IS THIS AN RPO2?
1349 005462 001402 BEQ 105 ;N.Y-105
1350 005464 012700 031460 MOV #03,R0 ;SET UP FOR AN RPO3
1351 005470 010067 000516 105: MOV R0,UNITMG+6 ;TAILOR DEV I.D. MSG
1352 005474 012767 000026 000056 MOV #22,DEVIML ;INITIALIZE TO NORMAL MSG LGTH
1353 005502 116300 000035 MOVB PCURDV(R3),R0 ;GET CURR UNIT #
1354 005506 020027 000007 CMP R0,#7 ;VALID UNIT #?
1355 005512 101007 BHI DEVIIV ;Y N-DEVIIV
1356 005514 004577 172340 JSR RS,28TASLZ ;CONVERT # TO DECIMAL ASCII
1357 005520 000510 .WORD UNASCI- ;
1358 005522 016767 000506 000510 MOV UNASCI+4,UNASCI ;MOVE ASCII # TO 1ST TWO DIGITS
1359 005524 000410 BR DEVIIV ;GO ISSUE MSG
1360 005540 042700 177400 000032 000020 DEVIIV: MOV #26,DEVIML ;SET ERR COND MSG LGTH
1361 005544 004577 172306 BIC #177400,R0 ;RESOLV HIGH BYTE
1362 005546 000460 .WORD UNASCI- ;CONVERT BINARY # TO ASCII
1363 005548 004567 000242 DEVIIV: JSR RS,PRINT ;GO ISSUE UNIT # MSG
1364 005550 000426 .WORD UNITMG- ;
1365 005552 000026 DEVIIV: .WORD 22 ;
1366 005554 000207 RTS PC ;EXIT IN-LINE
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376 005564 010346 DISPST: MOV R3,-(SP) ;SAVE R3
1377 005566 010503 MOV R3,R3 ;GET REL DATA ADR
1378 005570 062503 ADD (R3)+,R3 ;MAKE IT ABS
1379 005572 010546 MOV R5,-(SP) ;SAVE R5
1380 005574 010705 MOV PC,R5 ;SET UP ADR OF REG NAMES IN ASCII
1381 005576 062705 172320 ADD #0VREGS-,R5 ;
1382 005578 012746 000013 105: MOV #11,-(SP) ;SET UP # OF REGISTERS TO DISPLAY
1383 005580 012700 000003 MOV #3,R0 ;SET UP 3 REG LOOP CNT
1384 005582 010701 000422 MOV PC,R1 ;POINT AT REG NAME IN MSG
1385 005584 062701 155: MOV #0VRGMC-,R1 ;
1386 005586 012521 MOV (R5)+,(R1)+ ;MOVE REG NAME TO MSG
1387 005588 012521 MOV (R5)+,(R1)+ ;
1388 005590 005725 TST (R5)+ ;POINT TO NEXT NAME
1389 005592 062701 000012 ADD #10-,R1 ;POINT TO NEXT FIELD IN MSG
1390 005594 005300 DEC R0 ;DONE 3 REGS?
1391 005596 001371 BNE 155 ;Y N-155
1392 005598 012300 MOV (R3)+,R0 ;CONVERT OCTAL REGISTER CONTENTS
1393 005600 004577 172212 JSR RS,28NASC ;FOR 3 REGISTERS TO ASCII

```

```

1394 005644' 000400      .WORD  DVRT01-      ;AND PLACE IN THE MSG
1395 005646' 012300      MOV      (R3)+,R0
1396 005648' 004577 172202 JSR      RS,281NASC
1397 005650' 000406      .WORD  DVRT02-
1398 005652' 012300      MOV      (R3)+,R0
1399 005654' 004577 172172 JSR      RS,281NASC
1400 005656' 000414      .WORD  DVRT03-
1401 005658' 012767 000050 000034 MOV      #40,30$      ;INITIALIZE MSG LENGTH TO 3 REGS
1402 005660' 162716 000003 SUB      #3,(SP)      ;DECR REGISTER CNT
1403 005700' 100005 BPL      25$      ;(< 3 REGS) (Y,N-25$)
1404 005702' 162767 000016 000020 20$: SUB      #14,30$      ;SHORTEN MSG LENGTH BY 1 REG
1405 005710' 005216 INC      (SP)      ;INCR REG REG CNT
1406 005712' 100773 BMT      20$      ;CNT BACK TO 0? (Y,N-20$)
1407 005714' 010346 25$: MOV      R3,-(SP)      ;SAVE REG DATA PNTR
1408 005716' 016603 000006 MOV      6(SP),R3      ;RESTORE PROG TBL ADR
1409 005722' 004567 000072 JSR      RS,PRINT      ;GO PRINT THE MSG
1410 005726' 000310      .WORD  DVRGMG-
1411 005730' 000050 30$: .WORD  40
1412 005732' 012603 MOV      (SP)+,R3      ;RESTORE REG DATA PNTR
1413 005734' 005716 TST      (SP)      ;MORE REGS TO GO?
1414 005736' 001323 BNE      10$      ;N,Y-10$
1415 005740' 005726 TST      (SP)+      ;REMOVE CNT FROM STACK
1416 005742' 012605 MOV      (SP)+,RS      ;RESTORE RS & R3
1417 005744' 012603 MOV      (SP)+,R3
1418 005746' 000205 RTS      RS      ;EXIT IN-LINE
1419
1420
1421      ;DISPLAY CYL/HEAD/SECT WORDS' VALUES
1422
1423      ;JSR      PC,PRTIWD      S/R CALL
1424      ;DESTROYS R0,R1,R2
1425
1426 005750' 016700 172030 PRTIWD: MOV      CYL,R0      ;GET CYL VALUE
1427 005754' 004577 172076 JSR      RS,281NASC      ;CONVERT ITS VALUE TO ASCII
1428 005760' 000773      .WORD  IFCYL-
1429 005762' 016700 172020 MOV      HEAD,R0      ;GET & CONVERT HEAD VALUE
1430 005766' 004577 172064 JSR      RS,281NASC
1431 005772' 000776      .WORD  IFHEAD-
1432 005774' 016700 172010 MOV      SECT,R0      ;GET & CONVERT SECT VALUE
1433 006000' 004577 172052 JSR      RS,281NASC
1434 006004' 001001      .WORD  IFSECT-
1435 006006' 004567 000006 JSR      RS,PRINT      ;PRINT MSG WITH THEIR VALUES
1436 006012' 000734      .WORD  INFOG-
1437 006014' 000045      .WORD  37
1438 006016' 000207 RTS      PC      ;EXIT IN-LINE

```

```

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

```

```

1480 .SBTTL RPO2/RPO3 MESSAGE STORAGE AREA
1481
1482
1483 .NLIST BEX
1484
1485 .EVEN
1486 006134' 021520 PNMMSG: .ASCII /P# /
1487 006136' 054130 011 PROGNM: .ASCII /XX/<011>
1488 006141' 101 020124 040514 ATIMSG: .ASCII /AT LAST INT:/
1489 006155' 103 051125 042522 CUPMSG: .ASCII /CURRENTLY:/
1490 006167' 105 042116 047440 RENDMG: .ASCII /END OF REPORT/
1491
1492 006204' 025052 025052 050122 UNITMG: .ASCII /###RXX DISK UNIT: /
1493 006230' 054130 054130 054130 UNASCI: .ASCII /XXXXXX/
1494
1495 006236' 054130 054130 020075 DVRGMG: .ASCII /XXXX= /
1496 006240' 054130 054130 054130 DVJT1: .ASCII /XXXXXX XXXX= /
1497 006242' 054130 054130 054130 DVJT2: .ASCII /XXXXXX XXXX= /
1498 006244' 054130 054130 054130 DVJT3: .ASCII /XXXXXX/
1499 006246' 054130 054130 054130 CNTSMG: .ASCII /BYTES: RD= /
1500 006248' 054130 054130 054130 BCHKD: .ASCII /XXXXXXXXXXXXXXXXX WR= /
1501 006250' 054130 054130 054130 BCHKR: .ASCII /XXXXXXXXXXXXXXXXX/
1502 006252' 054130 054130 054130 CRLF: .ASCII (015)<012>
1503 006254' 054130 054130 054130 BCHK: .ASCII (011)<011>/CK= /
1504 006256' 054130 054130 054130 BCHKD: .ASCII /XXXXXXXXXXXXXXXXX/(015)<012><011>/CHND: RD= /
1505 006258' 054130 054130 054130 CJCRO: .ASCII /XXXXXX WR= /
1506 006260' 054130 054130 054130 CJCUR: .ASCII /XXXXXX CK= /
1507 006262' 054130 054130 054130 CJCCK: .ASCII /XXXXXX/(015)<012><011><011>/SK= /
1508 006264' 054130 054130 054130 CJCCK: .ASCII /XXXXXX MISC= /
1509 006266' 054130 054130 054130 CJCCK: .ASCII /XXXXXX/(015)<012><011>/ERRORS: DEV= /
1510 006268' 054130 054130 054130 CJCCK: .ASCII /XXXXXX DATA= /
1511 006270' 054130 054130 054130 CNTLDR: .ASCII /XXXXXX/(015)<012><011>/RETRY: TIME= /
1512 006272' 054130 054130 054130 CNTTIE: .ASCII /XXXXXX CSME= /
1513 006274' 054130 054130 054130 CNTCSE: .ASCII /XXXXXX MPE= /
1514 006276' 054130 054130 054130 CNTWPE: .ASCII /XXXXXX/(015)<012><011><011>/LPE= /
1515 006278' 054130 054130 054130 CNTLPE: .ASCII /XXXXXX MCE= /
1516 006280' 054130 054130 054130 CNTWCE: .ASCII /XXXXXX/(015)<012><011>/TOTAL RETRY: /
1517 006282' 054130 054130 054130 CNTTRY: .ASCII /XXXXXX/(015)<012><011>/INTERRUPTS: /
1518 006284' 054130 054130 054130 CNTINT: .ASCII /XXXXXX/
1519 006286' 054130 054130 054130 CNTSEN: .ASCII /CYL= /
1520 006288' 054130 054130 054130 INFOG: .ASCII /XXXXXX HEAD= /
1521 006290' 054130 054130 054130 IFCYL: .ASCII /XXXXXX SECT= /
1522 006292' 054130 054130 054130 IFHEAD: .ASCII /XXXXXX/
1523 006294' 054130 054130 054130 IFSECT: .ASCII /XXXXXX/
1524
1525 007014' 052123 047115 020124 STNMG: .ASCII /STMT # /
1526 007024' 054130 054130 054130 STNMN: .ASCII /XXXXXX/
1527 007032' 051105 047522 020122 EFBMSG: .ASCII /ERROR BITS:/(015)<012><011>
1528 007040' 000100 000100 000100 CODFLD: .BLKB 64.
1529 007150' 027524 020117 047117 CRT0: .ASCII 'T/O ON IDLE/RESET'
1530 007173' 124 046511 047505 IOT0: .ASCII 'TIMEOUT ON I/O'
1531 007212' 047125 040523 042506 INITUS: .ASCII 'UNSAFE ON INITIATION'
1532 007236' 027511 020117 042524 IOTERM: .ASCII 'I/O TERMINATION ERROR'
1533 007264' 052101 047124 047040 NOATA: .ASCII 'ATTN NOT SET'
1534
1535 007301' 105 044130 052501 RTYEXH: .ASCII 'EXHAUSTED RETRIES ON '

```

MAINDEC-11-DTR3A-A RPO2/RPO3 DEVICE ROUTINE FOR MPG
DTR3A.F11 RPO2/RPO3 MESSAGE STORAGE AREA

MACY11 27(732) 24-SEP-76 14:11 PAGE 11-1

SEQ 0105

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

```

1545          .SBTTL FORMATS FOR PROGRAM & DEVICE ROUTINE TABLES
1546
1547          ; PROGRAM TABLE FORMAT
1548
1549          000242      PTLGTH= 162.      ;PROGRAM TABLE LENGTH - NON MEM MGMT VERSION OF MPG
1550
1551          ;(PTLGTH= 212. ;PROGRAM TABLE LENGTH - MEM MGMT VERSION OF MPG)
1552
1553          000000      PFLGWD= +0.      ;PROGRAM FLAG WORD - 1 WORD
1554
1555          000002      URSTOP= 2          ; 1 = USER HAS STOPPED THIS PROGRAM
1556          000004      ERSTOP= 4          ; 1 = AN ERROR HAS STOPPED THIS PROGRAM
1557          000010      WTIOT= 10          ; 1 = WAITING FOR I/O TERMINATION
1558          000020      CTPRIO= 20          ; 1 = CONSOLE OR PRINTER I/O IN PROGRESS
1559          000040      SETDED= 40          ; 1 = THIS PROG SET THE PRT DEV DEDICATED FLAG
1560          000100      OCPRES= 100          ; 1 = OBJ CODE IS PRESENT
1561          000200      USEUBM= 200          ; 1 = THIS PROG USES THE UNIBUS MAP (MEM MGMT ONLY)
1562          100000      ACTIVE= 100000      ; 1 = PROGRAM IS ACTIVE (SPECIFIED FOR EXECUTION)
1563
1564          000002      POPSW= +2.      ;PROGRAM'S OPERATION SWITCHES - 1 WORD
1565
1566          100000      STONER= 100000      ; 1 = STOP PROG EXECUTION UPON ERROR
1567          040000      CYCPRG= 40000      ; 1 = CYCLE PROGRAM (ON CURRENT DEVICE)
1568          020000      PRONER= 20000      ; 1 = DO NOT PRINT ON ERROR
1569          010000      BIT12= 10000      ; 0 = NOT USED
1570          004000      BIT11= 4000        ; 0 = NOT USED
1571          002000      CYCDVL= 2000        ; 1 = CYCLE THE DEVICE LIST
1572          001000      GTNXTD= 1000        ; 1 = CYCLE ON SAME DEVICE UPON ERROR
1573          000400      DOERCK= 400          ; 1 = DON'T DO ERROR CHECKING
1574          000200      SPOPER= 200          ; 1 = DEVICE SPECIAL OPERATION
1575          000100      BIT6= 100           ; 0 = NOT USED
1576          000040      DOIOT= 40           ; 1 = DO NOT PERFORM I/O TIMEOUT
1577          000020      AUTORP= 20           ; 1 = DO NOT AUTOMATICALLY DISPLAY COUNTS
1578          000010      AURPEP= 10           ; 1 = AUTO DISPLAY COUNTS AT END OF FINAL PASS ONLY
1579          000004      HSKPEP= 4            ; 1 = HOUSEKEEP COUNTS ONLY AT RUN COMMAND
1580          000002      PFBBOV= 2           ; 1 = PRINT FIRST BAD BYTE ONLY ON VERIFY
1581          000001      NOCOMP= 1           ; 1 = DO NOT PRINT PROG COMPLETED MSG
1582
1583          000004      PFHADR= +4.      ;*;PROGRAM FLAGWORD ADDRESS - 1 WORD
1584
1585          000006      PASCIN= +6.      ;PROGRAM'S NUMBER IN ASCII - 1 WORD
1586
1587          000010      PNAME= +8.      ;PROGRAM'S NAME IN ASCII - 6 BYTES
1588
1589          000016      PRDIOA= +14.      ;ADDRESS OF READ I/O AREA - 1 WORD
1590
1591          000020      PWRIOA= +16.      ;ADDRESS OF WRITE I/O AREA - 1 WORD
1592
1593          000022      PSRCST= +18.      ;SOURCE STATEMENTS START ADDRESS - 1 WORD
1594
1595          000024      POBJST= +20.      ;OBJECT CODE START ADDRESS - 1 WORD
1596
1597          000026      PLNGTH= +22.      ;PROG AREA LENGTH (OBJ END MINUS PROG TBL START) - 1 WORD
1598
1599          000030      PTOCNT= +24.      ;I/O TIMEOUT COUNT - 1 WORD
1600

```

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

L03

MAINDEC-11-DTR3A-A RPO2/RPO3 DEVICE ROUTINE FOR MPG MACY11 27(732) 24-SEP-76 14:11 PAGE 12-2
DTR3AA.P11 FORMATS FOR PROGRAM & DEVICE ROUTINE TABLES

SEQ 0108

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

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

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

;END OF MEM MGMT ONLY ENTRIES

000240

PTSIZE= +160. ;PROGRAM TABLE SIZE IN BYTES - 1 WORD - NON MEM MGMT
;(PTSIZE= +210. ;PROGRAM TABLE SIZE IN BYTES - 1 WORD - MEM MGMT VERSION)
PTEND= +162. ;END OF PROGRAM TABLE - NON MEM MGMT VERSION
;(PTEND= +212. ;END OF PROGRAM TABLE - MEM MGMT VERSION)

000242

```

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

```

N03

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

SEQ 0110

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

ACTIVE=	100000		CURADR	001252R	002	DVCVEC=	000070		IDLE	002402R	002	PROIOR=	000016	
ATATBL	003216R	002	CURCMD	001250R	002	DVGETB=	000076		IFCYL	006753R	002	PRINT	006020R	002
ATTMSG	006141R	002	CURCNT	001255R	002	DVWSP=	000114		IFHEAD	006770R	002	PROCEX	004312R	002
ATTNPT	003476R	002	CURFLG	001246R	002	DVWST	001106R	002	IFSECT	007005R	002	PROCTH	004176R	002
AURPEP=	000010		CURMSG	006155R	002	DVMVTE	000570R	002	INFOMG	006746R	002	PROGNH	006136R	002
AUTORP=	000020		CURRTY	001262R	002	DVPOTA=	000062		INITUS	007212R	002	PRONER=	020000	
BCMCK	006370R	002	CYCOVL=	002000		DVPKTE	000350R	002	INTCNT	001334R	002	PRTEX	006132R	002
BCMRD	006322R	002	CYCPRG=	040000		DVPTEP=	000106		INTEAD	004272R	002	PRTIND	005750R	002
BCMR	006344R	002	CYL	000004R	002	DVPUTB=	000100		INTEX	003466R	002	PSRC	000120	
BINASC	000056R	002	DATADR	001316R	002	DVROT1	006244R	002	INMOVN	007366R	002	PSRCST=	000022	
BIT11 =	004000		DECASC	000062R	002	DVROT2	006262R	002	IOERR =	000001		PSTKCT=	000124	
BIT12 =	010000		DECTAD=	000042		DVROT3	006300R	002	IOTERM	007236R	002	PSTKSV=	000126	
BITE =	000100		DERPAD=	000036		DVREG=	000226R	002	IOTO	007173R	002	PSVREG=	000222	
BTASLZ	000060R	002	DEVSTA=	000056		DVREGS	000116R	002	ISTAT =	001166R	002	PSMO	000030R	002
BYCK	001276R	002	DEVDER=	000050		DVREND=	007402R	002	IVCTAD	000026R	002	PTEHD =	000056	
BYRD	001266R	002	DEVDR=	000024		DVREX	001714R	002	KILL	002054R	002	PTEH1 =	000060	
BYMR	001272R	002	DEVETP=	000104		DVRGMC	006236R	002	KILLEX	002110R	002	PTEH10=	000102	
CI08SY	000046R	002	DEVFND=	000002		DVRINT=	000074		LOC2	000000R	002	PTEH11=	000104	
CKCNT	001306R	002	DEVID	005450R	002	DVSFND=	000064		LPECNT	001326R	002	PTEH12=	000106	
CK03SY	004076R	002	DEVIV	005532R	002	DVSVEC=	000066		MISCNT	001312R	002	PTEH13=	000110	
CKRTRY	003636R	002	DEVINL	005560R	002	DVTVEC=	000072		MMVER =	000001		PTEH14=	000112	
CLIST	000054R	002	DEVIOB=	000046		DVUPRT=	000052		MODE =	010000		PTEH15=	000114	
CLRVEC	000070R	002	DEVIPR	005552R	002	DVUTEP=	000110		MODE10	002372R	002	PTEH2 =	000062	
CLP-IF	003462R	002	DEVIVA=	000026		EBSRBS	004564R	002	MODE11	002362R	002	PTEH3 =	000064	
CHDOCK	006451R	002	DEVINI=	000004		EBSTAT	004566R	002	MSFMT1	001140R	002	PTEH4 =	000066	
CHDOHS	006504R	002	DEVIN2=	000006		EBMSG	007032R	002	MSFMT2	001141R	002	PTEH5 =	000070	
CHDCOM	002646R	002	DEVIN3=	000010		ERCOTB	005036R	002	MSFMT3	001146R	002	PTEH6 =	000072	
CHDCRD	006423R	002	DEVIN4=	000012		ERNBS	004532R	002	MSFMT4	001155R	002	PTEH7 =	000074	
CHDCSK	006467R	002	DEVIN5=	000014		ERR	100000		MSFMT5	001164R	002	PTEH8 =	000076	
CHDCWR	006436R	002	DEVIN6=	000016		ERRCNT	001314R	002	MYATA	003236R	002	PTEH9 =	000100	
CNTADR	0													

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

ERRORS DETECTED: 0
DEFAULT GLOBALS GENERATED: 0

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

DOCUMENT PAGES: 41

Speaker routine & Records. 20 mag. 128 disk reader, 2 disk writer, 41 cases

[illegible]