

DLV11-F

DLV11-F OFF LINE TEST
MD-11-DVDVC-A

EP-DVDVC-A-DL-A

COPYRIGHT © 1977

FICHE 1 OF 1

OCT 1977

digital

MADE IN USA

The microfiche card contains 100 frames of data. The first 99 frames are organized into a 10x10 grid. Each frame contains a different set of data, including text, diagrams, and tables. The 100th frame is blank.

Frame	Frame	Frame	Frame	Frame	Frame	Frame	Frame	Frame	Frame
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

B01

EOF102F0R380411
DVDVCH.P11

08-AUG-77 09:16

0000080811-DVDVCH020 MACY11 27(108871008)AUG-77 09128DR29 7VBASE0

00010000

770920
SEQ 0001

.REM 2

IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DVDVC-A-D

PRODUCT NAME: DLV11-F OFFLINE TEST

PRODUCT DATE: AUGUST, 1977

AUTHOR: ODES CHOATE

MAINTAINER: DIAGNOSTIC ENGINEERING GROUP

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS MANUAL.

THE SOFTWARE DESCRIBED IN THIS DOCUMENT IS FURNISHED TO THE PURCHASER UNDER A LICENSE FOR USE ON A SINGLE COMPUTER SYSTEM AND CAN BE COPIED (WITH INCLUSION OF DIGITALS COPYRIGHT NOTICE) ONLY FOR USE IN SUCH SYSTEM, EXCEPT AS MAY OTHERWISE BE PROVIDED IN WRITING BY DIGITAL.

DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL.

COPYRIGHT (C) 1977 DIGITAL EQUIPMENT CORPORATION

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
72TABLE OF CONTENTS

1.0	GENERAL PROGRAM INFORMATION.
1.1	PROGRAM PURPOSE (ABSTRACT).
1.2	SYSTEM REQUIREMENTS.
1.3	RELATED DOCUMENTS AND STANDARDS.
1.4	DIAGNOSTIC HIERARCHY PREREQUISITES.
1.5	ASSUMPTIONS.
2.0	OPERATING INSTRUCTIONS.
2.1	LOADING AND STARTING PROCEDURES.
2.2	SPECIAL ENVIRONMENTS.
2.3	OPERATIONAL SWITCH SETTINGS
2.4	PROGRAM OPTIONS.
2.5	EXECUTION TIMES.
3.0	ERROR INFORMATION.
3.1	ERROR REPORTING PROCEDURE.
3.2	ERROR HALTS.
4.0	PERFORMANCE AND PROGRESS REPORTS.
4.1	PERFORMANCE REPORTS.
5.0	DEVICE INFORMATION TABLES.
6.0	SUMMARY OF TESTS AND SPECIAL SUBROUTINES

1.0 GENERAL PROGRAM INFORMATION.

1.1 PROGRAM PURPOSE (ABSTRACT).

THIS DIAGNOSTIC IS A LOGIC TEST TO VERIFY THE OPERATION OF THE DLV11-F SERIAL LINE INTERFACE. THE USER CAN SELECTIVELY ENABLE AND DISABLE TESTING OF THE OPTIONS BY ALTERING THE CONTENTS OF 'SUSER'. THE DIAGNOSTIC IS DESIGNED TO TEST AND DETECT FAULTS TO THE LOGIC LEVEL (NOT TO THE CHIP LEVEL). THIS TEST OPERATES ON UP TO SIXTEEN(16) IDENTICALLY CONFIGURED DLV11-F SERIAL LINE INTERFACES. THE DEFAULT ADDRESSES ARE:

177560 -CONSOLE INTERFACE DEVICE ADDRESS
175610 -FIRST SERIAL LINE ADDRESS OF 15 CONSECUTIVE
SERIAL LINE DEVICES.

60 - VECTOR FOR CONSOLE INTERFACE.
300 - VECTOR FOR FIRST OF 15 DEVICES.

THIS PROGRAM IS DESIGNED TO RUN ON ANY PDP-11 WITH 4K OF MEMORY AND A DLV11-F (LSI-BUS) MODULE. IT CAN RUN UNDER XXDP, APT, AND ACT MONITORS, AND ON PROCESSORS WITH NO HARDWARE SWITCH REGISTER. POWER FAIL IS SUPPORTED.

1.2 SYSTEM REQUIREMENTS.

HARDWARE REQUIREMENTS:

ANY PDP-11 FAMILY PROCESSOR
4K MEMORY - MINIMUM
A SPECIAL WRAP CONNECTOR OR EQUIVALENT (OPTIONAL)

SOFTWARE REQUIREMENTS:

THIS DIAGNOSTIC IS DESIGNED TO RUN IN ANY OF THE
FOLLOWING WAYS:

S.P.C. ALONE
WITH APT MONITOR
WITH ACT MONITOR
WITH XXDP MONITOR (CHAINABLE)

1.3 RELATED DOCUMENTS AND STANDARDS.

DIAGNOSTIC ENGINEERING STANDARDS AND CONVENTIONS
APT
ACT
SYSMAC

175-003-009-02
MD-11-DZZMA
AUTOCAT-11-QZAU8
MD-11-DZQAC

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES.

NO SPECIAL DIAGNOSTICS ARE REQUIRED TO RUN BEFORE THIS, BUT THE PROCESSOR, MEMORY, AND BUS ARE ASSUMED TO BE FULLY OPERATIONAL.

1.5 ASSUMPTIONS.

THIS DIAGNOSTIC ASSUMES THAT THE OPERATOR HAS INITIALIZED LOCATION 'SUSWR' AND 'SDEVM' TO THE PROPER VALUES.

2.0 OPERATING INSTRUCTIONS.

2.1 LOADING AND STARTING PROCEDURES.

USE STANDARD PROCEDURE FOR PDP-11 ABSOLUTE BINARY FORMATTED MEDIA.

THIS DIAGNOSTIC HAS ONLY ONE (1) STARTING ADDRESS. 200 FOR START AND RESTART.

THE USER CAN SELECT A SPECIFIC TEST TO BE EXECUTED BY SETTING SWITCH 8 IN THE SWITCH REGISTER AND THE TEST NUMBER (IN OCTAL) IN THE LOWER BYTE. (NOTE: ALL TESTS PREVIOUS TO THE SELECTED ONE ARE EXECUTED IN QUICK VERIFY MODE.)

2.2 SPECIAL ENVIRONMENTS.

THIS DIAGNOSTIC FOLLOWS THE STANDARD PROCEDURE FOR RUNNING UNDER APT ACT, XXDP MONITORS, AS DESCRIBED IN THEIR RESPECTIVE PROCEDURES MANUAL AND SYSMAC PACKAGE.

2.3 OPERATIONAL SWITCH SETTINGS

IF THE DIAGNOSTIC IS RUN ON A CPU WITHOUT A SWITCH REGISTER THEN A SOFTWARE SWITCH REGISTER IS USED WHICH ALLOWS THE USER THE SAME SWITCH OPTIONS AS THE HARDWARE SWITCH REGISTER. IF THE HARDWARE SWITCH REGISTER DOES NOT EXIST OR IF ONE DOES AND IT CONTAINS ALL ONES (177777) THEN THE SOFTWARE SWITCH REGISTER (LOC. 176) IS USED.

CONTROL:

THIS PROGRAM ALSO SUPPORTS THE DYNAMIC LOADING OF THE SOFTWARE SWITCH REGISTER (LOC. 176) FROM THE TTY. THIS CAN BE ACCOMPLISHED BY DOING THE FOLLOWING:

- 1) TYPE CONTROL G (1G); THIS WILL ALLOW THE TTY TO ENTER DATA INTO LOC. 176 AT SELECTED POINTS WITHIN THE

127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181

PROGRAM.

- 2) THE MACHINE WILL THEN TYPE: 'SWR=XXXXXX NEW=' (XXXXXX IS THE OCTAL CONTENTS OF THE SOFTWARE SWITCH REGISTER.)
- 3) AFTER THE 'NEW=' HAS BEEN TYPED THEN THE OPERATOR CAN DO ONE OF THE FOLLOWING AT THE TTY:
 - A) TYPE A NUMBER TO BE LOADED INTO LOC. 176 FOLLOWED BY A <CR>. (ONLY NUMBERS BETWEEN 0-7 WILL BE ACCEPTED). LEADING ZEROS NEED NOT BE TYPED, AND IF MORE THAN 6 DIGITS ARE TYPED THE LAST 6 WILL BE USED. IF A <CR> IS THE FIRST KEY DEPRESSED THE SOFTWARE SWITCH REGISTER CONTENTS WILL NOT BE CHANGED.
 - B) A RUBOUT WILL DELETE THE LAST INPUT VALUE AND WILL DELIMIT ALL DELETED CHARACTERS BETWEEN BACK SLASHES.
 - C) IF A CONTROL U <IU> IS DEPRESSED THEN THE PROGRAM WILL SEND YOU BACK TO STEP 2.
 - D) IF THE INPUT CHARACTER IS NOT ONE OF THE CHARACTERS MENTIONED ABOVE THEN A QUESTION MARK (?) WILL BE TYPED AND WILL WAIT FOR THE OPERATOR TO ENTER THE "SWREG" DATA AGAIN USING VALID CHARACTERS.

DYNAMIC SWITCH REGISTER

BIT 15 - HALT ON ERROR
 14 - LOOP ON TEST
 13 - INHIBIT ERROR TIMEOUTS
 12 - (UNUSED)
 11 - INHIBIT ITERATIONS
 10 - BELL ON ERROR
 9 - LOOP ON ERROR
 8 - LOOP ON TEST IN SWR<7:0>
 7:0 - TEST NUMBER TO LOOP ON (USED WITH BIT 8)

2.4 PROGRAM OPTIONS.

THIS PROGRAM WILL SUPPORT TESTING OF MULTIPLE DLV11-F'S. IT REQUIRES THE ADDRESS OF THE FIRST RCSR (STORED AT '\$BASE') AND ITS INTERRUPT VECTOR (STORED AT '\$VECT1'); AND WILL BE ABLE TO ADDRESS ANY DLV11-F STARTING AT THE SPECIFIED BASE ADDRESS UP TO 16 CONSECUTIVE DEVICES.

EXAMPLES: \$BASE: 175610
\$VECT1: 300

THE PROGRAM WILL BE ABLE TO TEST ANY DLV11-F WITHIN THE ADDRESS RANGE 175610 --> 176000

\$BASE AND \$VECT1 DEFAULT TO 175610 AND 300 RESPECTIVELY.

 182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236

G01

MAINDEC-11-DVDC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 7
 DVDVCA.P11 08-AUG-77 09:16

SEQ 0006

237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279

THE PROGRAM ASSOCIATES UNIT NUMBERS AS FOLLOWS: (NUMBERS IN PARENTHESES ARE OCTAL)

UNIT#0 -- BASE ADDRESS STORED AT '\$BASE'
 ASSOCIATED BASE VECTOR STORED AT '\$VECT1'
 UNIT#1 -- BASE ADDRESS + (10)
 BASE VECTOR + (10)

UP TO

UNIT#14 -- BASE ADDRESS + (160)
 BASE VECTOR + (160)

LOCATION '\$DEVN' IS USED AS A BIT MAP TO INDICATE WHICH UNIT NUMBERS ARE PRESENT AND WILL BE TESTED.

BIT 15	BIT 14	-	-	-	BIT 1	BIT 0
CON-	UNIT				UNIT	UNIT
SOLE	14				01	00

A BIT MAP CAN BE ENTERED AT '\$DEVN' PRIOR TO STARTING THE PROGRAM.

EXAMPLE:

\$BASE: 175610
 \$VECT1: 300
 \$DEVN: 100013

THE PROGRAM WILL TEST-

UNIT#0	175610	300
UNIT#1	175620	310
UNIT#3	175640	330
CONSOLE	177560	60

OPTIONS

LOCATION \$USWR CONTAINS ALL THE USER SELECTABLE OPTIONS. THE VALUES IN THIS WORD MUST CONFORM TO THE ACTUAL BOARD CONFIGURATION.

H01

THE DEFAULT VALUE OF SUSWR IS AS FOLLOWS:

BIT POSITION	DEFINITION	DEFAULT VALUE
0-3	# OF DATA BITS	10(8) = 8
4	PARITY ENABLED	0 = NO
5	EVEN ODD PARITY	0 = ODD
6	COMMON SPEED	1 = YES
7	PROGRAMMABLE BAUD RATE	0 = NO
8-11	BAUD RATE OFFSET (SEE FOLLOWING NOTE)	02(8) = 110 BAUD
12	BREAK GENERATION ENABLED	1 = YES
13	WRAP CONNECTOR INSTALLED	0 = NO
14	MAINT JUMPER	0 = NO
15	ERROR BITS ENABLED	0 = NO

NOTE ON BITS (7:11)

WHEN THE PROGRAMABLE BAUD RATE OPTION IS ENABLED THE PROGRAMABLE BAUD RATE TEST WILL EXIT WITH THE BAUD RATE SET TO THE SELECTED VALUE. TO CHANGE THE DEFAULT VALUE OF 110 BAUD REPLACE BITS (11:8) WITH THE OFFSET INDICATED IN THE TABLE AT THE END OF THE PBR TEST.(TEST #16)

NOTE ON BIT 14

THIS SWITCH WHEN ON WILL ALLOW THE DIAGNOSTIC TO TEST IN MAINTAINCE MODE. IT IS ASSUMED THAT THE MAINTAINCE JUMPER IS INSTALLED ON ALL OF THE DLV11-F MODULES WHEN THIS BIT IS SET.

2.5 EXECUTION TIMES.

EXECUTION TIMES ARE FOR AN LSI-11 PROCESSOR WITH ALL OPTIONS ENABLED ON THE DLV11-F (EXCEPT FOR PROGRAMMABLE BAUD RATE), AT 110 BAUD.

FIRST PASS- 2 MINUTES
 ADDITIONAL PASSES 2 MINUTES
 ADDITIONAL DEVICES 2 MINUTES

THE TEST TIME IS BAUD RATE DEPENDANT; HIGHER BAUD GIVES SHORTER PASS TIMES.

280
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

3.0 ERROR INFORMATION.

3.1 ERROR REPORTING PROCEDURE.

SINCE THIS DIAGNOSTIC WAS DESIGNED TO FIT IN 4-K OF MEMORY THE
ERROR TYPEOUT IS VERY BRIEF. THE FORMAT OF THE ERROR TYPEOUT
IS AS FOLLOWS:

TEST#+++++,ERROR#+++++,PC=+++++,ADDRESS=+++++,VECTOR=+++++

WHERE ALL VALUES TYPED ARE OCTAL.
THE ADDRESS AND VECTOR REFER TO THE FAILING SLU'S.
FOR FURTHER INFORMATION THE LISTING MUST BE CONSULTED.
BITS 15,13,10 AND 9 OF THE SWITCH REGISTER CONTROL THE
SEQUENCE OF EVENTS AFTER AN ERROR IS CAUGHT.

BIT 15 - CAUSES THE PROGRAM TO HALT IN THE ERROR
ROUTINE. CONTINUING THE PROGRAM CAUSES IT TO
PROCEED.

BIT 13 - DISABLES THE PRINTING OF THE ERROR MESSAGE.

BIT 10 - CAUSES THE BELL TO RING.

BIT 9 - CAUSES THE DIAGNOSTIC TO LOOP FROM BEGINNING
OF TEST TO ERROR.

THE ERROR ROUTINE SUPPORTS THE CONTROL G FUNCTION.

3.2 ERROR HALTS.

THE ONLY HALT IN THIS DIAGNOSTIC IS IN THE ERROR ROUTINE, AND
IS EXECUTED ONLY IF BIT 15 OF THE SWITCH REGISTER IS A ONE
WHEN AN ERROR OCCURS.

4.0 PERFORMANCE AND PROGRESS REPORTS.

4.1 PERFORMANCE REPORTS.

AS EACH DEVICE COMPLETES ONE PASS OF THE DIAGNOSTIC THE
FOLLOWING WILL BE TYPED:

CSR:+++++,VECTOR:+++++,ERRORS:+++++

WHERE. 'CSR:+++++' IS THE DEVICE CSR UNDER TEST
'VECTOR:++' IS THE ASSOCIATED VECTOR
AND 'ERRORS:++' IS THE TOTAL NUMBER OF ERRORS ON THIS DEVICE
ON THIS PASS.

J01

SEQ 0009

NOTE

THIS IS TYPED AFTER THE DEVICE HAS COMPLETED ITS PASS.

AFTER ALL DEVICES HAVE BEEN EXERCISED AN END PASS STATEMENT IS
TYPED:

5.0 "ENDPASS*****."
DEVICE INFORMATION TABLES.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RCSR:					RCVR:				RCVR:	RCVR:						RDR
					ACT				DONE	IE						ENB
RBUF:	ERRO	OR	FR	P												
	R	ERR	ERR	ERR												
TCSR:	PROGRAMMABLE BAUD				PBR				XMIT	XMIT				MAIN		BREA
	RATE SELECT				ENAB				RDY	IE				T		K
TBUF:																

NOTE

BLANK BOXES INDICATE UNUSED AND RESERVED
BIT POSITIONS. SEE THE LISTING FOR AN
EXPLANATION OF THE BITS.

6.0 SUMMARY OF TESTS AND SPECIAL SUBROUTINES.

TEST 1 ADDRESSABILITY

THIS TEST VERIFIES THAT THE ADDRESS AS PLACED IN THE
HARDWARE P-TABLE TO BE CORRECT AND THE DLV11-F
RESPONDS TO THAT ADDRESS SPACE.

K01

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 11
DVDVCA.P11 08-AUG-77 09:16

SEQ 0010

433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483

THE FOLLOWING 8 TESTS TEST ALL 'READ WRITE' BITS

TEST 2 BREAK - TCSR0 SET, CLEAR, RESET
---- -

TEST 3 MAINT - TCSR2 SET, CLEAR, RESET
---- -

TEST 4 XMITIE - TCSR6 SET, CLEAR, RESET
---- -

TEST 5 RCVRIE - RCSR6 SET, CLEAR, RESET
---- -

THE FOLLOWING 4 TESTS VERIFY THAT RESET (INIT) INITIALIZES
READ ONLY BITS.

TEST 6 RCVRDONE - RCSR 7 - IS CLEARED BY INIT
---- --

TEST 7 RCVRACT - RCSR 11 - 15 CLEARED BY INIT
---- -

TEST 10 XMITRDY - TCSR 7 - IS SET BY INIT
---- --

TEST 11 XMIT RDY - TCSR 7 - CLEARS WHEN TBUF IS LOADED
---- --
WITH A CHARACTER AND THAT IT SETS WITHIN A
REASONABLE AMOUNT OF TIME.

TEST 12 OUTPUTTING A CHAR FROM TBUF (WITH MAINT SET)
---- --
RESULTS IN RCVRDONE SETTING WITHIN A
REASONABLE AMOUNT OF TIME AND THAT RESET
CLEARS THE BIT.

TEST 13 RCVRDONE IS CLEARED BY READING RBUF
---- --

L01

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 12
DVDVCA.P11 08-AUG-77 09:16

SEQ 0011

484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536

TEST 14 RCVRACT - RCSR 11 - SETS WHEN A START BIT IS
----- --
RECEIVED AND CLEARS WHEN RCVRDONE - RCSR 7 -
SETS

TEST 15 OVERRUN BIT - RBUF 14
----- --

TEST 16 PROGRAMMABLE BAUD RATE TEST TEST AT ALL SPEEDS
----- --
AVAILABLE A COMPARISON WILL BE MADE TO SEE IF
NEW TIME IS LESS THAN PREVIOUS.

TEST 17 TRANSMITTER INTERRUPT LOGIC TEST
----- --
LOGICALLY THIS IS 4 SEPARATE TESTS
A) DOES TRANSMITTER INTERRUPT LOGIC WORK
B) AT PRIORITY OF 0
C) AND ONLY ONCE
D) BUT NOT WITH INTERRUPT ENABLE CLEAR

TEST 20 RECEIVER INTERRUPT LOGIC TEST THIS TEST COVERS ALL
----- --
OF THE RECEIVER SIDE OF THE INTERRUPT LOGIC IN
CHARACTER MODE.

TEST 21 TEST ACTUAL DATA TRANSFERED NON-INTERRUPT
----- --
MAINTENANCE BIT SET

TEST 22 TEST DATA THROUGH WRAP
----- --

TEST 23 FULL DATA TRANSFER WITH INTERRUPTS AND MAINTENANCE
----- --
MODE.

TEST 24 TEST BREAK GENERATION LOGIC TRANSMIT KNOWN CHAR
----- --
WITH BREAK SET AND COMPARE RECEIVED WITH 0.

TEST 25 NOT A TEST - SEND BACK TO LOOP
----- --

MO1

SEQ 0012

NOTE

FOR ALL OF THE FOLLOWING ROUTINES THE USE
OF (R5) IS PART OF THE LINKAGE MECHANISM
BETWEEN THE CALLER AND THE CALLED.

ROUTINE:TIMER

THIS ROUTINE IS USED TO TEST THE STATUS OF
ANY BIT IN ANY REGISTER.

INPUTS:

HOWLONG THE MAXIMUM AMOUNT OF TIME TO
SPEND IN THIS ROUTINE.
WHICHBIT A MASK WITH THE BIT(S) SET THAT
ARE TO BE CHECKED
REG A POINTER TO THE REGISTER TO BE
CHECKED
SETCLR THE DESIRED RESULTS -- EITHER SET
OR CLEAR

OUTPUT:

THE 'C' BIT IS SET TO INDICATE AN ERROR BUT IT
IS TESTED BY THE IF.ERROR STATEMENT.

ROUTINE:DATLNG

THIS ROUTINE SETS UP A MASK FOR DATA, WITH -

INPUT:

NOTHING IS PASSED TO THIS ROUTINE BUT GLOBAL
INFORMATION IS ASSUMED TO EXIST:
\$USWR-- THE WORD FOR SOFTWARE PARAMETERS
DATA-- A MASK FOR THE LOCATION OF THE OCTAL
NUMBER OF DATA BITS

OUTPUT----

MASK-- A MASK OF BINARY ZEROS RIGHT-JUSTIFIED
THE NUMBER OF WHICH IS DEFINED IN \$USWR WORD.

ROUTINE:WAIT

THIS ROUTINE IS USED TO DELAY EXECUTION OF THE
MAIN PROGRAM FOR A SPECIFIED AMOUNT OF TIME.
THIS IS ACCOMPLISHED BY INCREMENTING A
REGISTER UP TO A LIMIT. THE INNER LOOP IS SET
TO APPROXIMATE 1 MICRO SEC.

SERVICE ROUTINE: INTSRV

THIS GLOBAL ROUTINE DOES NOTHING BUT INCREMENT

537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590

NO1

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 14
DVDVCA.P11 08-AUG-77 09:16

SEQ 0013

591
592
593
594
595
596
597
598
599
600
601
602
603
604

'INTFLAG' EACH TIME IT IS CALLED. IT ASSUMES
THAT THE MAIN CALLING ROUTINE WILL KNOW WHAT
TO LOOK FOR.

ROUTINE:CYCLE

THIS ROUTINE CAUSES ADRS TO POINT TO THE
ADDRESS OF DLV11-F UNDER TEST. ADRS +2 TO
POINT TO THE VECTOR OF THE DLV11-F UNDER TEST.
IT KEEPS TRACK OF THE CURRENT DEVICE AND BIT
MASKS.

```

605      2
606      .TITLE MAINDEC-11-DVDVC-A
607      .#COPYRIGHT (C) 1977
608      .#DIGITAL EQUIPMENT CORP.
609      .#MAYNARD, MASS. 01754
610      .#
611      .#PROGRAM BY ODES CHOATE
612      .#
613      .#THIS PROGRAM WAS ASSEMBLED USING THE PDP-11 MAINDEC SYSMAC
614      .#PACKAGE (MAINDEC-11-DZQAC-C3), JAN 19, 1977.
615      .#
616      .SBTTL OPERATIONAL SWITCH SETTINGS
617      .#
618      .#          SWITCH          USE
619      .#          -----
620      .#          15          HALT ON ERROR
621      .#          14          LOOP ON TEST
622      .#          13          INHIBIT ERROR TYPEOUTS
623      .#          11          INHIBIT ITERATIONS
624      .#          10          BELL ON ERROR
625      .#          9          LOOP ON ERROR
626      .#          8          LOOP ON TEST IN SWR<7:0>
627
628      .SBTTL BASIC DEFINITIONS
629
630      .#INITIAL ADDRESS OF THE STACK POINTER *** 1100 ***
631      001100      STACK= 1100
632      .EQUIV EMT,ERROR      ;;BASIC DEFINITION OF ERROR CALL
633      .EQUIV IOT,SCOPE      ;;BASIC DEFINITION OF SCOPE CALL
634
635      .#MISCELLANEOUS DEFINITIONS
636      000011      HT= 11      ;;CODE FOR HORIZONTAL TAB
637      000012      LF= 12      ;;CODE FOR LINE FEED
638      000015      CR= 15      ;;CODE FOR CARRIAGE RETURN
639      000200      CRLF= 200    ;;CODE FOR CARRIAGE RETURN-LINE FEED
640      177776      PS= 177776  ;;PROCESSOR STATUS WORD
641      .EQUIV PS,PSW
642      177774      STKLMT= 177774 ;;STACK LIMIT REGISTER
643      177772      PIRQ= 177772 ;;PROGRAM INTERRUPT REQUEST REGISTER
644      177570      DSWR= 177570 ;;HARDWARE SWITCH REGISTER
645      177570      DDISP= 177570 ;;HARDWARE DISPLAY REGISTER
646
647      .#GENERAL PURPOSE REGISTER DEFINITIONS
648      000000      R0= %0      ;;GENERAL REGISTER
649      000001      R1= %1      ;;GENERAL REGISTER
650      000002      R2= %2      ;;GENERAL REGISTER
651      000003      R3= %3      ;;GENERAL REGISTER
652      000004      R4= %4      ;;GENERAL REGISTER
653      000005      R5= %5      ;;GENERAL REGISTER
654      000006      R6= %6      ;;GENERAL REGISTER
655      000007      R7= %7      ;;GENERAL REGISTER
656      000006      SP= %6      ;;STACK POINTER
657      000007      PC= %7      ;;PROGRAM COUNTER
658
659      .#PRIORITY LEVEL DEFINITIONS
660      000000      PRO= 0      ;;PRIORITY LEVEL 0

```

661	000040	PR1=	40	::PRIORITY LEVEL 1
662	000100	PR2=	100	::PRIORITY LEVEL 2
663	000140	PR3=	140	::PRIORITY LEVEL 3
664	000200	PR4=	200	::PRIORITY LEVEL 4
665	000240	PR5=	240	::PRIORITY LEVEL 5
666	000300	PR6=	300	::PRIORITY LEVEL 6
667	000340	PR7=	340	::PRIORITY LEVEL 7

:*"SWITCH REGISTER" SWITCH DEFINITIONS

670	100000	SW15=	100000
671	040000	SW14=	40000
672	020000	SW13=	20000
673	010000	SW12=	10000
674	004000	SW11=	4000
675	002000	SW10=	2000
676	001000	SW09=	1000
677	000400	SW08=	400
678	000200	SW07=	200
679	000100	SW06=	100
680	000040	SW05=	40
681	000020	SW04=	20
682	000010	SW03=	10
683	000004	SW02=	4
684	000002	SW01=	2
685	000001	SW00=	1
686		.EQUIV	SW09, SW9
687		.EQUIV	SW08, SW8
688		.EQUIV	SW07, SW7
689		.EQUIV	SW06, SW6
690		.EQUIV	SW05, SW5
691		.EQUIV	SW04, SW4
692		.EQUIV	SW03, SW3
693		.EQUIV	SW02, SW2
694		.EQUIV	SW01, SW1
695		.EQUIV	SW00, SW0

:*DATA BIT DEFINITIONS (BIT00 TO BIT15)

698	100000	BIT15=	100000
699	040000	BIT14=	40000
700	020000	BIT13=	20000
701	010000	BIT12=	10000
702	004000	BIT11=	4000
703	002000	BIT10=	2000
704	001000	BIT09=	1000
705	000400	BIT08=	400
706	000200	BIT07=	200
707	000100	BIT06=	100
708	000040	BIT05=	40
709	000020	BIT04=	20
710	000010	BIT03=	10
711	000004	BIT02=	4
712	000002	BIT01=	2
713	000001	BIT00=	1
714		.EQUIV	BIT09, BIT9
715		.EQUIV	BIT08, BIT8
716		.EQUIV	BIT07, BIT7

```

717      .EQUIV BIT06,BIT6
718      .EQUIV BIT05,BIT5
719      .EQUIV BIT04,BIT4
720      .EQUIV BIT03,BIT3
721      .EQUIV BIT02,BIT2
722      .EQUIV BIT01,BIT1
723      .EQUIV BIT00,BIT0
724
725      ;#BASIC "CPU" TRAP VECTOR ADDRESSES
726      ERRVEC= 4      ;: TIME OUT AND OTHER ERRORS
727      RESVEC= 10     ;: RESERVED AND ILLEGAL INSTRUCTIONS
728      TBITVEC= 14    ;: "T" BIT
729      TRTVEC= 14     ;: TRACE TRAP
730      BPTVEC= 14     ;: BREAKPOINT TRAP (BPT)
731      IOTVEC= 20     ;: INPUT/OUTPUT TRAP (IOT) **SCOPE**
732      PMRVEC= 24     ;: POWER FAIL
733      EMTVEC= 30     ;: EMULATOR TRAP (EMT) **ERROR**
734      TRAPVEC= 34    ;: "TRAP" TRAP
735      TKVEC= 60      ;: TTY KEYBOARD VECTOR
736      TPVEC= 64      ;: TTY PRINTER VECTOR
737      PIRQVEC= 240   ;: PROGRAM INTERRUPT REQUEST VECTOR
738
739      ILLMEM= 4
740      ADRS= R1
741      GOOD= R2
742      BAD= R3
743      REGISTER= R1
744      BIT= R2
745      FUNCT= R3
746      LEAD= R2
747      FOLLOW= R4
748      DLADDR= 175610
749
750      ; THE FOLLOWING DEFINITIONS APPLY TO THE GLOBAL SUBS
751      SET= -1
752      CLR= 0
753
754      ;*****
755      ; RCSR REGISTER BIT NAMES
756      ;*****
757      : UNUSED BIT15
758      : UNUSED BIT14
759      : UNUSED BIT13
760      : UNUSED BIT12
761      RCVRACT= BIT11 ; RECEIVER ACTIVE INDICATOR
762      : UNUSED BIT10
763      : UNUSED BIT09
764      : UNUSED BIT08
765      RCVRDONE= BIT07 ; RECEIVER DONE
766      RCVRIE= BIT06 ; RECEIVER INTERRUPT ENABLE
767      : UNUSED BIT05
768      : UNUSED BIT04
769      : UNUSED BIT03
770      : UNUSED BIT02
771      : UNUSED BIT01
772      RDRRUN= BIT00 ; READER RUN

```


F02

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 19
 DVDVCA.P11 08-AUG-77 09:16 BASIC DEFINITIONS

SEQ 0018

```

829      000040      TDATA5=      BIT05      : \
830      000020      TDATA4=      BIT04      : \ TRANSMITTER DATA BUFFER
831      000010      TDATA3=      BIT03      : \
832      000004      TDATA2=      BIT02      : \
833      000002      TDATA1=      BIT01      : \
834      000001      TDATA0=      BIT00      : /
835
836
837      ;*****
838      ; FLAG BITS TO BE USE OR CLEARED IN SUSMR.
839
840      000017      DATA =      17
841      000020      PARITY =      20
842      000040      EVENOOD =      40
843      000100      COMSPD =      100
844      000200      PBR =      200
845
846      ; BAUDE MUST BE ON THE UPPER
847      ; BYTE BOUNDARY OF SUSMR.--4 BITS
848      007400      BAUD =      7400
849      010000      BRK =      10000
850      020000      WRAP =      20000
851      040000      MAINTJUMP =      40000
852      100000      ERRBITS =      100000
853      ;*****
854      .SBTTL TRAP CATCHER
855
856      000000      .=0
857      ;*ALL UNUSED LOCATIONS FROM 4 - 776 CONTAIN A ".+2,HALT"
858      ;*SEQUENCE TO CATCH ILLEGAL TRAPS AND INTERRUPTS
859      ;*LOCATION 0 CONTAINS 0 TO CATCH IMPROPERLY LOADED VECTORS
860      000174      .=174
861      000174      000000      DISPREG: .WORD 0      ;;SOFTWARE DISPLAY REGISTER
862      000176      000000      SWREG: .WORD 0      ;;SOFTWARE SWITCH REGISTER
863
864      000200      000137      001336      .SBTTL STARTING ADDRESS(ES)
      JMP 2*START ;;JUMP TO STARTING ADDRESS OF PROGRAM

```

G02

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 20
 DVDVCA.P11 08-AUG-77 09:16 ACT11 HOOKS

SEQ 0019

```

865      .SBTTL ACT11 HOOKS
866
867      ;;*****
868      ;HOOKS REQUIRED BY ACT11
869      $SVPC=.          ;SAVE PC
870      .=46
871      SENDAD          ;;1)SET LOC.46 TO ADDRESS OF SENDAD IN .SEOP
872      .=52
873      .WORD 0          ;;2)SET LOC.52 TO ZERO
874      .=$SVPC          ;; RESTORE PC
875      .=1000
876      .SBTTL APT PARAMETER BLOCK
877
878      ;;*****
879      ;SET LOCATIONS 24 AND 44 AS REQUIRED FOR APT
880      ;;*****
881      .SX=.          ;SAVE CURRENT LOCATION
882      .=24          ;SET POWER FAIL TO POINT TO START OF PROGRAM
883      200          ;FOR APT START UP
884      .=44          ;POINT TO APT INDIRECT ADDRESS PNTR.
885      $APTHDR       ;POINT TO APT HEADER BLOCK
886      .=$X          ;RESET LOCATION COUNTER
887      ;;*****
888      ;SETUP APT PARAMETER BLOCK AS DEFINED IN THE APT-PDP11 DIAGNOSTIC
889      ;INTERFACE SPEC.
890
891      $APTHD:
892      $HIBTS: .WORD 0          ;TWO HIGH BITS OF 18 BIT MAILBOX ADDR.
893      $MBAOR: .WORD $MAIL      ;ADDRESS OF APT MAILBOX (BITS 0-15)
894      $TSTM:  .WORD 5          ;RUN TIM OF LONGEST TEST
895      $PASTM:  .WORD 45.       ;RUN TIME IN SECS. OF 1ST PASS ON 1 UNIT (QUICK VERIFY)
896      $UNITM:  .WORD 30.       ;ADDITIONAL RUN TIME (SECS) OF A PASS FOR EACH ADDITIONAL UNIT
897      .WORD $ETEND-$MAIL/2 ;;LENGTH MAILBOX-ETABLE(WORDS)

```

```

898      .SBTTL COMMON TAGS
899
900      ;;*****
901      ;;THIS TABLE CONTAINS VARIOUS COMMON STORAGE LOCATIONS
902      ;;USED IN THE PROGRAM.
903
904      001100      .=1100
905      001100      SCMTAG:      ;;START OF COMMON TAGS
906      001100      000000      .WORD      0      ;;CONTAINS THE TEST NUMBER
907      001102      000      .BYTE      0      ;;CONTAINS ERROR FLAG
908      001103      000      .BYTE      0      ;;CONTAINS SUBTEST ITERATION COUNT
909      001104      000000      .WORD      0      ;;CONTAINS SCOPE LOOP ADDRESS
910      001106      000000      .WORD      0      ;;CONTAINS SCOPE RETURN FOR ERRORS
911      001110      000000      .WORD      0      ;;CONTAINS TOTAL ERRORS DETECTED
912      001112      000000      .WORD      0      ;;CONTAINS ITEM CONTROL BYTE
913      001114      000      .BYTE      0      ;;CONTAINS MAX. ERRORS PER TEST
914      001115      001      .BYTE      1      ;;CONTAINS PC OF LAST ERROR INSTRUCTION
915      001115      000000      .WORD      0      ;;CONTAINS ADDRESS OF 'GOOD' DATA
916      001120      000000      .WORD      0      ;;CONTAINS ADDRESS OF 'BAD' DATA
917      001122      000000      .WORD      0      ;;CONTAINS 'GOOD' DATA
918      001124      000000      .WORD      0      ;;CONTAINS 'BAD' DATA
919      001126      000000      .WORD      0      ;;RESERVED--NOT TO BE USED
920      001130      000000      .WORD      0
921      001132      000000      .WORD      0
922      001134      000      .BYTE      0      ;;AUTOMATIC MODE INDICATOR
923      001135      000      .BYTE      0      ;;INTERRUPT MODE INDICATOR
924      001136      000000      .WORD      0
925      001140      177570      .WORD      DSWR      ;;ADDRESS OF SWITCH REGISTER
926      001142      177570      .WORD      DDISP      ;;ADDRESS OF DISPLAY REGISTER
927      001144      177560      .WORD      177560      ;;TTY KBD STATUS
928      001146      177562      .WORD      177562      ;;TTY KBD BUFFER
929      001150      177564      .WORD      177564      ;;TTY PRINTER STATUS REG. ADDRESS
930      001152      177566      .WORD      177566      ;;TTY PRINTER BUFFER REG. ADDRESS
931      001154      000      .BYTE      0      ;;CONTAINS NULL CHARACTER FOR FILLS
932      001155      002      .BYTE      2      ;;CONTAINS # OF FILLER CHARACTERS REQUIRED
933      001156      012      .BYTE      12      ;;INSERT FILL CHARS. AFTER A "LINE FEED"
934      001157      000      .BYTE      0      ;;"TERMINAL AVAILABLE" FLAG (BIT<07>=0=YES)
935      001160      000000      .WORD      0      ;;MAX. NUMBER OF ITERATIONS
936      001162      000000      .WORD      0      ;;ESCAPE ON ERROR ADDRESS
937      001164      177607      .ASCIZ      <207><377><377>      ;;CODE FOR BELL
938      001170      077      .ASCIZ      /?/      ;;QUESTION MARK
939      001171      015      .ASCIZ      <15>      ;;CARRIAGE RETURN
940      001172      000012      .ASCIZ      <12>      ;;LINE FEED
941      ;;*****
942      .SBTTL APT MAILBOX-ETABLE
943
944      ;;*****
945      .EVEN
946      001174      000000      .WORD      MSGTY      ;;APT MAILBOX
947      001174      000000      .WORD      AFATAL      ;;MESSAGE TYPE CODE
948      001176      000000      .WORD      ATESTN      ;;FATAL ERROR NUMBER
949      001200      000000      .WORD      APASS      ;;TEST NUMBER
950      001202      000000      .WORD      ADEVCT      ;;PASS COUNT
951      001204      000000      .WORD      AUNIT      ;;DEVICE COUNT
952      001206      000000      .WORD      AMSGAD      ;;I/O UNIT NUMBER
953      001210      000000      .WORD      AMSGAD      ;;MESSAGE ADDRESS

```

000377

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 22
 DVDVCA.P11 08-AUG-77 09:16 APT MAILBOX-ETABLE

SEQ 0021

954	001212	000000	\$MSGLG: .WORD	AMSGLG	:: MESSAGE LENGTH
955	001214		\$ETABLE:		:: APT ENVIRONMENT TABLE
956	001214	000	\$ENV: .BYTE	AENV	:: ENVIRONMENT BYTE
957	001215	000	\$ENVH: .BYTE	AENVH	:: ENVIRONMENT MODE BITS
958	001216	000000	\$SWREG: .WORD	ASWREG	:: APT SWITCH REGISTER
959	001220	011110	\$USWR: .WORD	AUSWR	:: USER SWITCHES
960	001222	000000	\$CPUOP: .WORD	ACPUOP	:: CPU TYPE, OPTIONS
961			::		BITS 15-11=CPU TYPE
962			::		11/04=01, 11/05=02, 11/20=03, 11/40=04, 11/45=05
963			::		11/70=06, PDQ=07, Q=10
964			::		BIT 10=REAL TIME CLOCK
965			::		BIT 9=FLOATING POINT PROCESSOR
966			::		BIT 8=MEMORY MANAGEMENT
967	001224	000	\$MAMS1: .BYTE	AMAMS1	:: HIGH ADDRESS, M.S. BYTE
968	001225	000	\$MTYP1: .BYTE	AMTYP1	:: MEM. TYPE, BLK#1
969			::		MEM. TYPE BYTE -- (HIGH BYTE)
970			::		900 NSEC CORE=001
971			::		300 NSEC BIPOLAR=002
972			::		500 NSEC MOS=003
973	001226	000000	\$MAOR1: .WORD	AMAOR1	:: HIGH ADDRESS, BLK#1
974			::		MEM. LAST ADDR.=3 BYTES, THIS WORD AND LOW OF "TYPE" ABOVE
975	001230	000	\$MAMS2: .BYTE	AMAMS2	:: HIGH ADDRESS, M.S. BYTE
976	001231	000	\$MTYP2: .BYTE	AMTYP2	:: MEM. TYPE, BLK#2
977	001232	000000	\$MAOR2: .WORD	AMAOR2	:: MEM. LAST ADDRESS, BLK#2
978	001234	000	\$MAMS3: .BYTE	AMAMS3	:: HIGH ADDRESS, M.S. BYTE
979	001235	000	\$MTYP3: .BYTE	AMTYP3	:: MEM. TYPE, BLK#3
980	001236	000000	\$MAOR3: .WORD	AMAOR3	:: MEM. LAST ADDRESS, BLK#3
981	001240	000	\$MAMS4: .BYTE	AMAMS4	:: HIGH ADDRESS, M.S. BYTE
982	001241	000	\$MTYP4: .BYTE	AMTYP4	:: MEM. TYPE, BLK#4
983	001242	000000	\$MAOR4: .WORD	AMAOR4	:: MEM. LAST ADDRESS, BLK#4
984	001244	000300	\$VECT1: .WORD	AVECT1	:: INTERRUPT VECTOR#1, BUS PRIORITY#1
985	001246	000000	\$VECT2: .WORD	AVECT2	:: INTERRUPT VECTOR#2, BUS PRIORITY#2
986	001250	175610	\$BASE: .WORD	ABASE	:: BASE ADDRESS OF EQUIPMENT UNDER TEST
987	001252	100000	\$DEVH: .WORD	ADEVH	:: DEVICE MAP
988	001254		\$ETEND:		
989			.MEXIT		

J02

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 23
 DVDVCA.P11 08-AUG-77 09:16 ERROR POINTER TABLE

SEQ 0022

990
 991
 992
 993
 994
 995
 996
 997
 998
 999
 1000
 1001
 1002
 1003
 1004
 1005
 1006
 1007
 1008
 1009
 1010
 1011
 1012
 1013
 1014
 1015

001254

001254 175610
 001256 000300
 001260 175610
 001262 175612
 001264 175614
 001266 175615
 001270 175616
 001272 000000
 001274 000020
 001334 000000

.SBTTL ERROR POINTER TABLE

;;*THIS TABLE CONTAINS THE INFORMATION FOR EACH ERROR THAT CAN OCCUR.
 ;;*THE INFORMATION IS OBTAINED BY USING THE INDEX NUMBER FOUND IN
 ;;*LOCATION \$ITEMB. THIS NUMBER INDICATES WHICH ITEM IN THE TABLE IS PERTINENT.
 ;;*NOTE1: IF \$ITEMB IS 0 THE ONLY PERTINENT DATA IS (\$ERRPC).
 ;;*NOTE2: EACH ITEM IN THE TABLE CONTAINS 4 POINTERS EXPLAINED AS FOLLOWS:

;;* EM ;;POINTS TO THE ERROR MESSAGE
 ;;* DM ;;POINTS TO THE DATA HEADER
 ;;* DT ;;POINTS TO THE DATA
 ;;* DF ;;POINTS TO THE DATA FORMAT

\$ERRTB:

;; GLOBAL DATA
 DLADD: DLADDR
 DLVEC: 300
 RCSR: DLADDR + 0
 RBUF: DLADDR + 2
 TCSR: DLADDR + 4
 TCSRHI: DLADDR + 5
 TBUF: DLADDR + 6
 I: 0
 .BLKW 20 ;FOR R5 STACK
 RSSTACK: .WORD 0

K02

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 24
 DVDVCA.P11 08-AUG-77 09:16 ERROR POINTER TABLE

SEQ 0023

```

1016 001336
1017
1018
1019 001336 012706 001100
1020 001342 005026
1021 001344 022706 001140
1022 001350 001374
1023 001352 012706 001100
1024
1025 001356 012737 013474 000020
1026 001364 012737 000340 000022
1027 001372 012737 013274 000030
1028 001400 012737 000340 000032
1029 001406 012737 014426 000034
1030 001414 012737 000340 000036
1031 001422 012737 011606 000024
1032 001430 012737 000340 000026
1033 001436 016767 010056 010046
1034 001444 005067 177510
1035 001450 005067 177506
1036 001454 112767 000001 177433
1037 001462 012767 001462 177416
1038 001470 012767 001470 177412
1039
1040
1041 001476 013746 000004
1042 001502 012737 001536 000004
1043 001510 012767 177570 177422
1044 001516 012767 177570 177416
1045 001524 022777 177777 177406
1046 001532 001012
1047
1048 001534 000403
1049 001536 012716 001544
1050 001542 000002
1051 001544 012767 000176 177366
1052 001552 012767 000174 177362
1053 001560 012637 000004
1054
1055 001564 005067 177412
1056 001570 132767 000200 177417
1057 001576 001403
1058 001600 012767 001216 177332
1059 001606
1060
1061
1062 001606 005227 177777
1063 001612 001037
1064 001614 022737 011552 000042
1065 001622 001433
1066 001624 104401 001672
1067
1068 001630 005737 000042
1069 001634 001012
1070 001636 126727 177352 000001
1071 001644 001406

START:
.SBTTL INITIALIZE THE COMMON TAGS
;;CLEAR THE COMMON TAGS ($CHTAG) AREA
MOV $CHTAG,R6 ;;FIRST LOCATION TO BE CLEARED
CLR (R6)+ ;;CLEAR MEMORY LOCATION
CMP $SWR,R6 ;;DONE?
BNE -6 ;;LOOP BACK IF NO
MOV $STACK,SP ;;SETUP THE STACK POINTER
;;INITIALIZE A FEW VECTORS
MOV $SCOPE,$IOTVEC ;;IOT VECTOR FOR SCOPE ROUTINE
MOV $340,$IOTVEC+2 ;;LEVEL 7
MOV $ERROR,$EMTVEC ;;EMT VECTOR FOR ERROR ROUTINE
MOV $340,$EMTVEC+2 ;;LEVEL 7
MOV $TRAP,$TRAPVEC ;;TRAP VECTOR FOR TRAP CALLS
MOV $340,$TRAPVEC+2 ;;LEVEL 7
MOV $SPWRON,$PWRVEC ;;POWER FAILURE VECTOR
MOV $340,$PWRVEC+2 ;;LEVEL 7
MOV $ENDCT,$EUPCT ;;SETUP END-OF-PROGRAM COUNTER
CLR $TIMES ;;INITIALIZE NUMBER OF ITERATIONS
CLR $ESCAPE ;;CLEAR THE ESCAPE ON ERROR ADDRESS
MOVB $1,$SERMAX ;;ALLOW ONE ERROR PER TEST
MOV $0,$SLPADR ;;INITIALIZE THE LOOP ADDRESS FOR SCOPE
MOV $0,$SLPERR ;;SETUP THE ERROR LOOP ADDRESS
;;SIZE FOR A HARDWARE SWITCH REGISTER. IF NOT FOUND OR IT IS
;;EQUAL TO A "-1" SETUP FOR A SOFTWARE SWITCH REGISTER.
MOV $ERRVEC,-(SP) ;;SAVE ERROR VECTOR
MOV $64,$ERRVEC ;;SET UP ERROR VECTOR
MOV $DSWR,$SWR ;;SETUP FOR A HARDWARE SWICH REGISTER
MOV $DDISP,$DISPLAY ;;AND A HARDWARE DISPLAY REGISTER
CMP $-1,$DSWR ;;TRY TO REFERENCE HARDWARE SWR
BNE 66$ ;;BRANCH IF NO TIMEOUT TRAP OCCURRED
AND THE HARDWARE SWR IS NOT = -1
BR 65$ ;;BRANCH IF NO TIMEOUT
SET UP FOR TRAP RETURN
64$: MOV $65$,(SP)
RTI
65$: MOV $SWREG,$SWR ;;POINT TO SOFTWARE SWR
MOV $DISPREG,$DISPLAY
66$: MOV (SP)+,$ERRVEC ;;RESTORE ERROR VECTOR
CLR $PASS ;;CLEAR PASS COUNT
BITB $APTSIZE,$ENVM ;;TEST USER SIZE UNDER APT
BEQ 67$ ;;YES,USE NON-APT SWITCH
MOV $SSWREG,$SWR ;;NO,USE APT SWITCH REGISTER
67$:
.SBTTL TYPE PROGRAM NAME
;;TYPE THE NAME OF THE PROGRAM IF FIRST PASS
INC $-1 ;;FIRST TIME?
BNE 68$ ;;BRANCH IF NO
CMP $SENDAD,$42 ;;ACT-11?
BEQ 68$ ;;BRANCH IF YES
TYPE 69$ ;;TYPE ASCIZ STRING
.SBTTL GET VALUE FOR SOFTWARE SWITCH REGISTER
TST $42 ;;ARE WE RUNNING UNDER XXDP/ACT?
BNE 70$ ;;BRANCH IF YES
CMPB $ENV,$1 ;;ARE WE RUNNING UNDER APT?
BEQ 70$ ;;BRANCH IF YES

```

L02

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 25
 DVDVCA.P11 08-AUG-77 09:16 GET VALUE FOR SOFTWARE SWITCH REGISTER

SEQ 0024

1072	001546	026727	177266	000176		CMP	SWR,#SWREG	;;SOFTWARE SWITCH REG SELECTED?
1073	001654	001005				BNE	71\$	;;BRANCH IF NO
1074	001656	104406				GTSWR		;;GET SOFT-SWR SETTINGS
1075	001660	000403				BR	71\$	
1076	001662	112767	000001	177244	70\$:	MOVB	#1,\$AUTOB	;;SET AUTO-MODE INDICATOR
1077	001670				71\$:			
1078	001670	000410				BR	68\$	;;GET OVER THE ASCIZ
1079					;;69\$:	.ASCIZ	<CRLF>*MD-11-DVDVC-A*<CRLF>	
1080	001712				68\$:			

M02

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 26
 DVDVCA.P11 08-AUG-77 09:16 GET VALUE FOR SOFTWARE SWITCH REGISTER

SEQ 0025

```

1081 001712                                WHILE $DEVN EQ #0 DO
1082 001712
1083 001712 005767 177334          $1:      TST      $DEVN
1084 001716 001101                BNE      $2
1085 001720                        TYPTXT <<CRLF>!!I HAVE NO DEVICE TO TEST.!!>
1086 001762                        TYPTXT <<CRLF>!!SET UP $DEVN TO INDICATE ACTUAL CONFIGURATION.!!>
1087 002050                        TYPTXT <<CRLF>!!TYPE PROCEED (P) TO CONTINUE.!!>
1088 002116 000000                HALT
1089 002120                                ENDDO
1090 002120 000674          $2:      BR      $1
1091 002122
1092 002122                                LET INITFLAG := #1
1093 002122 012767 000001 007122      MOV      #1,INITFLAG
1094 002130                                LET BITMASK := #BIT15 ; START AT CONSOLE
1095 002130 012767 100000 007112      MOV      #BIT15,BITMASK
1096 002136          LOOP:      CALL     CYCLE      ; NO ARGUMENTS--ADDRS -> NEXT ADDRESS
1097 002136 004767 006666      JSR      PC,CYCLE      ;
1098 002136                                ADDR+2 -> NEXT VECTOR
1099 002136                                ;GET UNIT ADDRESS
1100 002142                                DLADD := (ADDRS)+
1101 002142 012167 177106      MOV      (ADDRS)+,DLADD
1102 002142                                ;GET UNIT VECTOR
1103 002146                                DLVEC := (ADDRS)
1104 002146 011167 177104      MOV      (ADDRS),DLVEC
1105 002152                                LET     ADDRS := DLADD
1106 002152 016701 177076      MOV      DLADD,ADDRS
1107 002156                                ;RCSR = DLADD + 0
1108 002156 016767 177072 177074      MOV      DLADD,RCSR
1109 002164                                LET     RBUF := DLADD + #2
1110 002164 016767 177064 177070      MOV      DLADD,RBUF
1111 002172 062767 000002 177062      ADD      #2,RBUF
1112 002200                                LET     TCSR := DLADD + #4
1113 002200 016767 177050 177056      MOV      DLADD,TCSR
1114 002206 062767 000004 177050      ADD      #4,TCSR
1115 002214                                LET     TCSRHI := DLADD + #5
1116 002214 016767 177034 177044      MOV      DLADD,TCSRHI
1117 002222 062767 000005 177036      ADD      #5,TCSRHI
1118 002230                                LET     TBUF := DLADD + #6
1119 002230 016767 177020 177032      MOV      DLADD,TBUF
1120 002236 062767 000006 177024      ADD      #6,TBUF
1121 002244                                LET     RS := #RSSTACK
1122 002244 012705 001334      MOV      #RSSTACK,RS
1123 002250                                ;;BRESET
1124 002250 000005      RESET
1125
1126

```

N02

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 27
 DVDVCA.P11 08-AUG-77 09:16 T1 ADDRESSABILITY

SEQ 0026

```

1127 *****
1128 ;TEST 1 ADDRESSABILITY
1129 ;
1130 ; THIS TEST VERIFIES THAT THE ADDRESS AS PLACED IN
1131 ; THE HARDWARE P-TABLE TO BE CORRECT AND THE DLV11-F RESPONDS
1132 ; TO THAT ADDRESS SPACE
1133 *****
1134 TST1: SCOPE
1135 MOV #2,STIMES ;;DO 2 ITERATIONS
1136 MOV #1,STESTN ;;SET TEST NUMBER IN APT MAIL BOX
1137 LET ADRS := DLADD
1138
1139 ; SET UP INTERRUPT
1140 SETVEC ILLMEM,#INTSRV,#PR7
1141 MOV R1,-(SP)
1142 MOV ILLMEM,R1
1143 MOV INTSRV(R1)+
1144 MOV #PR7(R1)
1145 MOV (SP)+,R1
1146 LET I := #0
1147 REPEAT
1148
1149 $3: BGNSUB
1150 MOV #64$,$LPERR
1151 ;CLEAR FLAG
1152 LET INTFLAG := #0
1153 CLR INTFLAG
1154 ;READ FLAG
1155 IF INTFLAG NE #0 THEN
1156 ; FATAL ERROR
1157 ERRORF 1,,NODL
1158 ENDIF
1159 ENDSUB
1160 LET I := I + #2
1161 LET ADRS := DLADD + I
1162 UNTIL I EQ #8.
1163
1164 CLAVEC ILLMEM
1165 ;;PUSH R1 ON STACK
1166 ;;PUSH R2 ON STACK
1167 MOV R1,-(SP)
1168 MOV R2,-(SP)
1169 MOV ILLMEM,R1
1170 MOV R1,R2
1171 ADD #2,R2
1172 MOV R2,(R1)+
1173 CLR (R1)
1174 MOV (SP)+,R2 ;;POP STACK INTO R2
1175
1176 002252 000004
1177 002254 012767 000002 176676
1178 002262 012767 000001 176710
1179 002270
1180 002270 016701 176760
1181
1182 002274
1183 002274 010146
1184 002276 012701 000004
1185 002302 012721 011020
1186 002306 012711 000340
1187 002312 012601
1188 002314
1189 002314 005067 176752
1190 002320
1191 002320 $3:
1192 002320 012767 002326 176562
1193 002326
1194 002326 005067 006474
1195
1196 002332 005711
1197 002334
1198 002334 005767 006466
1199 002340 001401
1200
1201 002342
1202 002342 104001
1203 002344
1204 002344
1205 002344
1206 002344
1207 002344 062767 000002 176720
1208 002352
1209 002352 016701 176676
1210 002352 066701 176710
1211 002352
1212 002352 026727 176704 000010
1213 002353 001353
1214 002372
1215 002372 010146
1216 002374 010246
1217 002376 012701 000004
1218 002376 010102
1219 002404 066702 000002
1220 002410
1221 002412 002411
1222 002414 012602

```

B03

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 28
DVDVCA.P11 08-AUG-77 09:16 T1 ADDRESSABILITY

SEQ 0027

1183 002416 012601

1184

1185 002420

MOV (SP)+,R1

;;POP STACK INTO R1
;END OF TEST

ENDTST

C03

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 29
 DVDVCA.P11 08-AUG-77 09:16 T1 ADDRESSABILITY

SEQ 0028

```

1186 .....
1187 * THE FOLLOWING 8 TESTS TEST ALL 'READ WRITE' BITS
1188 .....
1189 .....
1190 *TEST 2 BREAK - TCSR0 SET, CLEAR, RESET
1191 * THIS BIT IS THE ONLY ONE IN THIS POSITION
1192 * THAT IS READ AND WRITE.
1193 .....
1194 TST2: SCOPE
1195 002420 000004 MOV #10,$TIMES ;;DO 10 ITERATIONS
1196 002422 012767 000010 176530 MOV #2,$TESTN ;;SET TEST NUMBER IN APT MAIL BOX
1197 002430 012767 000002 176542
1198 002436 IF #BRK NOTSETIN $USWR THEN
1199 002436 032767 010000 176554 BIT #BRK,$USWR
1200 002444 001004 BNE $5
1201 002446 EXIT TEST ; BREAK NOT INSTALLED
1202 002446 012767 000001 176504 MOV #1,$TIMES
1203 002454 000452 BR TST3 ;;EXIT THIS TEST
1204 002456 ENDF
1205 002456 $5:
1206
1207 ; SEE IF IT IS CLEAR
1208 002456 BGNSUB
1209 002456 012767 002464 176424 MOV #64,$LPERR
1210
1211 IF #BREAK SETIN @TCSR THEN
1212 002464 032777 000001 176572 BIT #BREAK,@TCSR
1213 002472 001401 BEQ $6
1214 ; BREAK DID NOT RESET IN TCSR
1215 002474 104002 ERROR 2 ERRHRD 2,,DIDNOT
1216 002474
1217 002476 ENDF
1218 002476 $6:
1219 002476 ENDSUB
1220
1221 ; TRY TO SET BREAK BIT
1222 002476 BGNSUB
1223 002476 012767 002504 176404 MOV #64,$LPERR
1224 002504 LET @TCSR := @TCSR SET.BY #BREAK
1225 002504 052777 000001 176552 BIS #BREAK,@TCSR
1226 ; STUCK TO 0
1227 002512 IF #BREAK NOTSETIN @TCSR THEN
1228 002512 032777 000001 176544 BIT #BREAK,@TCSR
1229 002520 001001 BNE $7
1230 ; BREAK DID NOT SET IN TCSR
1231 002522 104003 ERROR 3 ERRHRD 3,,DIDNOT
1232 002522
1233 002524 ENDF
1234 002524 $7:
1235 002524 ENDSUB
1236
1237 ; TRY TO CLEAR A SET BIT
1238 002524 BGNSUB
1239 002524 012767 002532 176356 MOV #64,$LPERR
1240
1241 LET @TCSR := @TCSR CLR.BY #BREAK

```

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 30
DVDVCA.P11 08-AUG-77 09:16 T2 BREAK - TCSRO SET, CLEAR, RESET

Address	Hex	Mask	Value	Field	Comment
1242	002532	042777	000001	176524	BIC #BREAK,@TCSR
1243					
1244	002540				
1245	002540	032777	000001	176516	BIT #BREAK,@TCSR
1246	002546	001401			BEQ \$10
1247					
1248	002550				
1249	002550	104004			ERROR 4
1250	002552				
1251	002552				
1252	002552				
1253					
1254					
1255	002552				
1256	002552	012767	002560	176330	MOV #64,\$LPERR
1257					
1258	002560				
1259	002560	052777	000001	176476	BIS #BREAK,@TCSR
1260					
1261	002566				
1262	002566	000005			RESET
1263	002570				
1264	002570	032777	000001	176466	BIT #BREAK,@TCSR
1265	002576	001401			BEQ \$11
1266					
1267	002600				
1268	002600	104005			ERROR 5
1269	002602				
1270	002602				
1271	002602				
1272	002602				
1273					
1274					

E03

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 31
 DVDVCA.P11 08-AUG-77 09:16 T2 BREAK - TCSR0 SET, CLEAR, RESET

SEQ 0030

```

1275 ;*****
1276 ;*****
1277 ;TEST 3      MAINT - TCSR2 SET, CLEAR, RESET
1278 ;*****
1279 002602 000004 ST3: SCOPE
1280 002604 012767 000010 176346 MOV #10,$TIMES ;DO 10 ITERATIONS
1281 002612 012767 000003 176360 MOV #3,$TESTN ;SET TEST NUMBER IN APT MAIL BOX
1282
1283 002620 IF #MAINTJUMP NOTSETIN $USWR THEN
1284 002620 032767 040000 176372 BIT #MAINTJUMP,$USWR
1285 002626 001004 BNE $12
1286 002630 EXIT TEST
1287 002630 012767 000001 176322 MOV #1,$TIMES
1288 002636 000452 BR TST4 ;;;EXIT THIS TEST
1289 002640 ENDF
1290 002640 $12:
1291
1292 ; SEE IF IT IS CLEAR
1293 002640 BGNSUB
1294 002640 012767 002646 176242 MOV #64,$SLPERR
1295
1296 002646 IF #MAINT SETIN @TCSR THEN
1297 002646 032777 000004 176410 BIT #MAINT,@TCSR
1298 002654 001401 BEQ $13
1299
1300 ; MAINT DID NOT RESET IN TCSR
1301 002656 104006 ERROR 6 ERRHRD 6,,DIDNOT
1302 002660 ENDF
1303 002660 $13:
1304 002660 ENDSUB
1305
1306 ; TRY TO SET MAINT BIT
1307 002660 BGNSUB
1308 002660 012767 002666 176222 MOV #64,$SLPERR
1309 002666 LET @TCSR := @TCSR SET.BY #MAINT
1310 002666 052777 000004 176370 BIS #MAINT,@TCSR
1311
1312 ; STUCK TO 0
1313 002674 IF #MAINT NOTSETIN @TCSR THEN
1314 002674 032777 000004 176362 BIT #MAINT,@TCSR
1315 002702 001001 BNE $14
1316
1317 ; MAINT DID NOT SET IN TCSR
1318 002704 104007 ERROR 7 ERRHRD 7,,DIDNOT
1319 002706 ENDF
1320 002706 $14:
1321 ENDSUB
1322
1323 ; TRY TO CLEAR A SET BIT
1324 002706 BGNSUB
1325 002706 012767 002714 176174 MOV #64,$SLPERR
1326
1327 002714 LET @TCSR := @TCSR CLR.BY #MAINT
1328 002714 042777 000004 176342 BIC #MAINT,@TCSR
1329
1330 IF #MAINT SETIN @TCSR THEN
1330 002722 032777 000004 176334 RIT #MAINT,@TCSR

```

F03

MAINDEC-11-DVDC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 32
 DVDVCA.P11 08-AUG-77 09:16 T3 MAINT - TCSR2 SET, CLEAR, RESET

SEQ 0031

```

1331 002730 001401          BEQ      $15
1332                                     ; MAINT DID NOT CLEAR INTCSR
1333                                     ERRHRD 10,,DIDNOT
1334 002732 104010          ERROR    10
1335 002734                                     ENDIF
1336 002734          $15:                                     ENDSUB
1337 002734                                     ; NOW SEE IF RESET CLEARS IT
1338                                     BGNSUB
1339 002734 012767 002742 176146      MOV      @645,$LPERR
1340 002734 052777 000004 176314      BIS      @MAINT,@TCSR
1341                                     LET      @TCSR := @TCSR SET.BY @MAINT
1342                                     ; ISSUE BUS RESET
1343 002742 000005          RESET
1344 002742 032777 000004 176304      BIT      @MAINT,@TCSR
1345                                     BRESET
1346 002750 001401          IF      @MAINT SETIN @TCSR THEN
1347 002750 000005          ; MAINT DID NOT RESET IN TCSR
1348 002752 001401          ERRHRD 11,,DIDNOT
1349 002752 104011          ERROR    11
1350 002760          $16:                                     ENDIF
1351 002762                                     ENDSUB
1352 002762 104011          ENOTST
1353 002764
1354 002764
1355 002764
1356 002764
1357 002764
1358
1359
1360

```

G03

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 33
 DVDVCA.P11 08-AUG-77 09:16 T3 MAINT - TCSR2 SET, CLEAR, RESET

SEQ 0032

```

1361
1362
1363
1364
1365 002764 000004
1366 002766 012767 000010 176164
1367 002774 012767 000004 176176
1368
1369 003002 012746 000340
1370 003006 012746 003014
1371 003012 000002
1372 003014
1373
1374
1375 003014
1376 003014 012767 003022 176066
1377
1378 003022
1379 003022 032777 000100 176234
1380 003030 001401
1381
1382 003032
1383 003032 104012
1384 003034
1385 003034
1386 003034
1387
1388
1389 003034
1390 003034 012767 003042 176046
1391 003042
1392 003042 052777 000100 176214
1393
1394 003050
1395 003050 032777 000100 176206
1396 003056 001001
1397
1398 003060
1399 003060 104013
1400 003062
1401 003062
1402 003062
1403
1404
1405 003062
1406 003062 012767 003070 176020
1407
1408 003070
1409 003070 042777 000100 176166
1410
1411 003076
1412 003076 032777 000100 176160
1413 003104 001401
1414
1415 003106
1416 003106 104014

```

```

*****
*****
TEST 4 XMITIE - TCSR6 SET, CLEAR, RESET
*****
TST4: SCOPE
      MOV #10,$TIMES      ;; DO 10 ITERATIONS
      MOV #4,$TESTN      ;; SET TEST NUMBER IN APT MAIL BOX
                          ;; USE PRIORITY OF 7
      MOV #PR7,-(SP)      ;; PUT NEW PS ON STACK
      MOV #64$,-(SP)      ;; PUT NEW PC ON STACK
      RTI                 ;; POP NEW PC AND PS
64$:
      ; SEE IF IT IS CLEAR
      BGNSUB
      MOV #65$,$LPERR
      IF #XMITIE SETIN @TCSR THEN
      BIT #XMITIE,@TCSR
      BEQ $17
      ; XMITIE DID NOT RESET IN TCSR
      ERRHRD 12,,DIDNOT
      ENDIF
      $17:
      ENDSUB
      ; TRY TO SET XMITIE BIT
      BGNSUB
      MOV #64$,$LPERR
      BIS #XMITIE,@TCSR
      LET @TCSR := @TCSR SET.BY #XMITIE
      IF ; STUCK TO 0
      IF #XMITIE NOTSETIN @TCSR THEN
      ; XMIT DID NOT RESET IN TCSR
      ERRHRD 13,,DIDNOT
      ENDIF
      $20:
      ENDSUB
      ; TRY TO CLEAR A SET BIT
      BGNSUB
      MOV #64$,$LPERR
      BIC #XMITIE,@TCSR
      LET @TCSR := @TCSR CLR.BY #XMITIE
      IF ; SHOULD HAVE CLEARED
      IF #XMITIE SETIN @TCSR THEN
      ; XMIT DID NOT CLEAR IN TCSR
      ERRHRD 14,,DIDNOT
      ERROR 14

```

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 34
DVDVCA.P11 08-AUG-77 09:16 T4 XMITIE - TCSR6 SET, CLEAR, RESET

```

1417      003110      ENDIF
1418      003110      $21:
1419      003110      ENDSUB
1420
1421      ; NOW SEE IF RESET CLEARS IT
1422      003110      BGNSUB
1423      003110      012767      003116      175772      MOV      #64$,SLPERR
1424
1425      003116      LET      @TCSR := @TCSR SET.BY #XMITIE
1426      003116      052777      000100      176140      BIS      #XMITIE,@TCSR
1427
1428      003124      ; ISSUE BUS RESET
1429      003124      000005      RESET      BRESET
1430      003126      IF      #XMITIE SETIN @TCSR THEN
1431      003126      032777      000100      176130      BIT      #XMITIE,@TCSR
1432      003134      001401      BEQ      $22
1433
1434      003136      ; XMIT DID NOT RESET IN TCSR
1435      003136      104015      ERROR      15      ERRHRD 15,,DIDNOT
1436      003140      ENDIF
1437      003140      $22:
1438      003140      ENDSUB
1439      003140      ENDTST
1440
1441
1442

```

I03

 MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 35
 DVDVCA.P11 08-AUG-77 09:16

T4 XMITIE - TCSR6 SET, CLEAR, RESET

SEQ 0034

```

1443 ;*****
1444 ;*****
1445 ;TEST 5          RCVRIE - RCSR6 SET, CLEAR, RESET
1446 ;              THIS BIT IS THE ONLY ONE IN THIS POSITION
1447 ;              THAT IS READ AND WRITE.
1448 ;*****
1449 003140 000004      STS: SCOPE
1450 003142 012767 000010 176010      MOV      #10,STIMES      ;;DO 10 ITERATIONS
1451 003150 012767 000005 176022      MOV      #5,$TESTN      ;;SET TEST NUMBER IN APT MAIL BOX
1452 ;                      ; SEE IF IT IS CLEAR
1453 003156          BGNSUB
1454 003156 012767 003164 175724      MOV      #64$,SLPERR
1455
1456 003164          IF      #RCVRIE SETIN @RCSR THEN
1457 003164 032777 000.00 176066      BIT      #RCVRIE,@RCSR
1458 003172 001401      BEQ      $23
1459
1460 003174          ; RCVRIE DID NOT RESET IN RCSR
1461 003174 104035      ERROR    35      ERRHRD 35,,DIDNOT
1462 003176          ENDIF
1463 003176      $23:
1464 003176          ENDSUB
1465
1466          ; TRY TO SET RCVRIE BIT
1467 003176          BGNSUB
1468 003176 012767 003204 175704      MOV      #64$,SLPERR
1469 003204          LET      @RCSR := @RCSR SET.BY #RCVRIE
1470 003204 052777 000100 176046      BIS      #RCVRIE,@RCSR
1471
1472          ; STUCK TO 0
1473 003212          IF      #RCVRIE NOTSETIN @RCSR THEN
1474 003220 032777 000100 176040      BIT      #RCVRIE,@RCSR
1475 003220 001001      BNE      $24
1476
1477 003222          ; RCVRIE DID NOT SET IN RCSR
1478 003222 104036      ERROR    36      ERRHRD 36,,DIDNOT
1479 003224          ENDIF
1480 003224      $24:
1481          ENDSUB
1482
1483          ; TRY TO CLEAR A SET BIT
1484 003224          BGNSUB
1485 003224 012767 003232 175656      MOV      #64$,SLPERR
1486
1487 003232          LET      @RCSR := @RCSR CLR.BY #RCVRIE
1488 003232 042777 000100 176020      BIC      #RCVRIE,@RCSR
1489
1490          ; SHOULD HAVE CLEARED
1491 003240          IF      #RCVRIE SETIN @RCSR THEN
1492 003240 032777 000100 176012      BIT      #RCVRIE,@RCSR
1493 003246 001401      BEQ      $25
1494
1495 003250          ; RCVRIE DID NOT CLEAR IN RCSR
1496 003250 104037      ERROR    37      ERRHRD 37,,DIDNOT
1497 003252          ENDIF
1498 003252      $25:
          ENDSUB

```

J03

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 36
 DVDVCA.P11 08-AUG-77 09:16 TS RCVRIE - RCSR6 SET, CLEAR, RESET

SEQ 0035

; NOW SLE IF RESET CLEARS IT
 BGNSUB

1499							
1500	003252				MOV	#64\$,SLPERR	
1501	003252	012767	003260	175630			
1502							
1503	003260						LET @RCSR := @RCSR SET.BY #RCVRIE
1504	003260	052777	000100	175772	BIS	#RCVRIE,@RCSR	; ISSUE BUS RESET
1505							BRESET
1506	003266						
1507	003266	000005			RESET		
1508	003270						IF #RCVRIE SETIN @RCSR THEN
1509	003270	032777	000100	175762	BIT	#RCVRIE,@RCSR	
1510	003276	001401			BEQ	\$26	
1511							; RCVRIE DID NOT RESET IN RCSR
1512	003300						ERRHRD 40,,DIDNOT
1513	003300	104040			ERROR	40	
1514	003302						ENDIF
1515	003302				\$26:		
1516	003302						CKLOOP
1517	003302						ENDSUB
1518	003302						ENDTST
1519							
1520							
1521							
1522							

K03

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 37
 DVDVCA.P11 08-AUG-77 09:16 T5 RCVRIE - RCSR6 SET, CLEAR, RESET

SEQ 0036

```

1523 *****
1524 * THE FOLLOWING 4 TESTS VERIFY
1525 * THAT RESET (INIT) INITIALIZES READ ONLY BITS.
1526 *****
1527 *****
1528 *TEST 6 TEST THAT RCVRDONE - RCSR 7 - IS CLEARED BY INIT
1529 *****
1530 TST6: SCOPE
1531 MOV #10,$TIMES ;;DO 10 ITERATIONS
1532 MOV #6,$TESTN ;;SET TEST NUMBER IN APT MAIL BOX
1533
1534
1535
1536
1537 003320 BGNSUB
1538 003320 012767 003326 175562 MOV #64$,$LPERR
1539 003326 IF #RCVRDONE SETIN @RCSR THEN
1540 003326 032777 000200 175724 BIT #RCVRDONE,@RCSR
1541 003334 001402 BEQ $27
1542
1543 ;RCVRDONE SHOULD HAVE CLEARED BY INIT
1544 ; RCVRDONE DID NOT CLEAR IN RCSR
1545 003336 ERROR 41 ERRHAD 41,HRESET, DIDNOT
1546 003336 104041
1547 ;REISSUE RESET
1548 003340 BRESET
1549 003340 000005 RESET
1550 003342 ENDIF
1551 003342 $27:
1552 ;ALLOW LOOPING AFTER ERROR
1553 003342 CKLOOP
1554 003342 ENDSUB
1555 003342 ENDTST
1556
1557
1558
  
```

L03

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 38
DVDVCA.P11 08-AUG-77 09:16

T6 TEST THAT RCVRDONE - RCSR 7 - IS CLEARED BY INIT

SEQ 0037

```

1559 ;*****
1560 ;*****
1561 ;TEST 7 TEST THAT RCVRACT - RCSR 11 - IS CLEARED BY INIT
1562 ;*****
1563 003342 000004 TST7: SCOPE
1564 003344 012767 000010 175606 MOV #10,$TIMES ;;DO 10 ITERATIONS
1565 003352 012767 000007 175620 MOV #7,$TESTN ;;SET TEST NUMBER IN APT MAIL BOX
1566
1567
1568
1569
1570 003360 IF CONSOLE EQ #TRUE THEN
1571 003360 026727 005703 000001 CMP CONSOLE,#TRUE
1572 003366 001001 BNE $30
1573 ELSE ;; EXECUTE TEST
1574 003370
1575 003370 000416 BR $31
1576 003372 $30:
1577 003372 IF #WRAP SETIN $USWR THEN
1578 003372 032767 020000 175620 BIT #WRAP,$USWR
1579 003400 001401 BEQ $32
1580 ELSE ;; EXECUTE TEST
1581 003402
1582 003402 000411 BR $33
1583 003404 $32:
1584 003404 IF #MAINT SETIN $USWR THEN
1585 003404 032767 000004 175606 BIT #MAINT,$USWR
1586 003412 001401 BEQ $34
1587 ELSE ;;EXECUTE TEST
1588 003414
1589 003414 000404 BR $35
1590 003416 $34:
1591 003416 EXIT TEST ; LINE MUST BE TERMINATED
1592 003416 012767 000001 175534 MOV #1,$TIMES
1593 003424 000414 BR TST10 ;;EXIT THIS TEST
1594 003426 ENDF
1595 003426 $35:
1596 003426 ENDF
1597 003426 $33:
1598 003426 ENDF
1599 003426 $31:
1600
1601 BGNSUB
1602 003426 012767 003434 175454 MOV #64,$SLPERR
1603
1604 IF #RCVRACT SETIN #RCSR THEN
1605 003434 032777 004000 175616 BIT #RCVRACT,#RCSR
1606 003442 001405 BEQ $36
1607
1608 ;RESET SHOULD HAVE CLEARED RCVRACT
1609 003444 LET #RCSR := #RCSR CLR.BY #MAINT
1610 003444 042777 000004 175612 BIC #MAINT,#RCSR
1611 003452 ERRHRD 44, HRESET, DIDNOT
1612 003452 104044 ERROR 44
1613
1614 ;TESTING EFFECT OF RESET ON BIT

```

M03

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 39

DVDVCA.P11 08-AUG-77 09:16

T7 TEST THAT RCVRCT - RCSR 11 - IS CLEARED BY INIT

SEQ 0038

1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627

003454
003454 000005
003456
003456
003456
003456
003456
003456

RESET
\$36:

;RCVRCT DID NOT CLEAR IN RCSR
;ALLOW ANOTHER TRY
BRESE
ENDIF
;ALLOW LOOPING ON ERROR
CKLOOP
ENDSUB
ENDTST

N03

MAINDEC-11-DVDYC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 40
 DVDCA.P11 08-AUG-77 09:16

T7 TEST THAT ACVRCT - RCSR 11 - IS CLEARED BY INIT

SEQ 0039

```

1628 ;*****
1629 ;*****
1630 ;TEST 10 TEST THAT XMITRDY - TCSR 7 - IS SET BY INIT
1631 ;*****
1632 ST10: SCOPE
1633 MOV #10,$TIMES ;;DO 10 ITERATIONS
1634 MOV #10,$TESTN ;;SET TEST NUMBER IN APT MAIL BOX
1635
1636
1637
1638
1639 003474 BGNSUB
1640 003474 012767 003502 175406 MOV #64$,$LPERR
1641
1642 IF #XMITRDY NOTSETIN @TCSR THEN
1643 003502 032777 000200 175554 BIT #XMITRDY,@TCSR
1644 003510 001002 BNE $37
1645
1646 ;RESET SHOULD HAVE SET BIT.
1647 ;XMITRDY DID NOT SET IN TCSR (AFTER RESE
1648 003512 ERRHRD 42,HRESET,DIDNOT
1649 003512 104042 ERROR 42
1650
1651 ;ISSUE ANOTHER RESET
1652 003514 BCLR SET
1653 003514 000005 RESET
1654 003516 $37:
1655
1656 ENDIF
1657 003516 CKLOOP
1658 003516 ENDOSUB
1659
1660 ENDTST
1661
  
```

804

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 41
 DVDVCA.P11 08-AUG-77 09:16 T10 TEST THAT XMITRDY - TCSR 7 - IS SET BY INIT

SEQ 0040

```

1662 ;*****
1663 ;*****
1664 ;TEST 11 TEST THAT XMIT RDY - TCSR 7 - CLEARS
1665 ; WHEN TBUF IS LOADED WITH A CHARACTER
1666 ; AND THAT IT SETS WITHIN A REASONABLE AMOUNT OF TIME.
1667 ;*****
1668 003516 000004 ST11: SCOPE
1669 003520 012767 000010 175432 MOV #10,STIMES ;;DO 10 ITERATIONS
1670 003526 012767 000011 175444 MOV #11,STESTN ;;SET TEST NUMBER IN APT MAIL BOX
1671
1672 003534 IFB APTCON EQ #TRUE THEN
1673 003534 126727 005526 000001 CMPB APTCON,#TRUE
1674 003542 001004 BNE $40
1675 003544 EXIT TEST
1676 003544 012767 000001 175406 MOV #1,STIMES
1677 003552 000513 BR TS12 ;;EXIT THIS TEST
1678 003554 ENDF
1679 003554 $40:
1680
1681 003554 LET PASS := #1 ;INIT COUNT OF TIMES THRU
1682 003554 012767 000001 000212 MOV #1,PASS
1683 003562 LOOP ; START OF LOOP
1684 003562 $41:
1685 ; MAX OF 2 TIMES THRU
1686 003562 LET ERRORFLAG := #CLR
1687 003562 012767 000000 000206 MOV #CLR,ERRORFLAG
1688 003570 LET EXITFLAG := #CLR
1689 003570 012767 000000 000202 MOV #CLR,EXITFLAG
1690 ; LOAD TBUF WITH ONE CHARACTER
1691 ; WAIT FOR READY TO SET
1692 ; (SHOULD BE VERY SHORT WAIT
1693 ; SINCE UART DOUBLE BUFFERS ITS INPUT)
1694
1695 ; SEND A CHARACTER
1696 003576 LET #TBUF := #0
1697 003576 105077 175466 CLRB #TBUF
1698 ; WAIT A MAXIMUM
1699 ; OF 50 MSEC FOR
1700 ; XMIT RDY TO SET IN TCSR
1701 CALL TIMER IN (<#5,#XMITRDY,TCSR,#SET)
1702 003602 MOV RS,-(SP)
1703 003604 010546 MOV #SET,-(RS)
1704 003610 012745 177777 MOV TCSR,-(RS)
1705 003614 016745 175450 MOV #XMITRDY,-(RS)
1706 003620 012745 000200 MOV #5,-(RS)
1707 003624 004767 000005 MOV PC,TIMER
1708 003630 012605 MOV (SP)+,RS
1709
1710 ; TIMER RETURNS AN ERROR IF BIT DID
1711 ; NOT MEET CONDITION WITHIN TIME LIMIT
1712 003632 103001 BCC $43 IF.ERROR THEN
1713
1714 003634 ; XMIT RDY DID NOT SET IN TCSR
1715 003634 104066 ERROR 66, DIDNOT
1716 003636 ENDF
1717 003636 $43:

```

C04

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 42
 DVDVCA.P11 08-AUG-77 09:16 T11 TEST THAT XMIT ROY - TCSR 7 - CLEARS

SEQ 0041

```

1718
1719
1720
1721
1722
1723
1724 003636 105077 175426 CLR8 2TBUF
1725 003636 105077 175426 NOP
1726 003642 000240
1727
1728
1729 003644 032777 000200 175412 BIT #XMITRDY,2TCSR
1730 003644 032777 000200 175412 BEQ $44
1731 003652 001404
1732
1733 003654 012767 177777 000114 MOV #SET,ERRORFLAG
1734 003654 012767 177777 000114
1735
1736
1737 003662 000416 $44: BR $45
1738 003662 000416
1739 003664
1740
1741
1742
1743 003664 010546 MOV R5,-(SP)
1744 003664 012745 177777 MOV #SET,-(R5)
1745 003666 016745 175366 MOV TCSR,-(R5)
1746 003672 012745 000200 MOV #XMITRDY,-(R5)
1747 003676 012745 000005 MOV #5,-(R5)
1748 003702 012745 000005 MOV PC,TIMER
1749 003706 004767 004550 JSR PC,TIMER
1750 003712 012605 MOV (SP)+,R5
1751 003714
1752 003714 103001 BCC $46
1753
1754 003716 104070 ERROR 70
1755 003716 104070
1756 003720 $46:
1757 003720 $45:
1758 003720
1759 003720
1760 003720 026727 000052 177777 CMP ERRORFLAG,#SET
1761 003720 001011 BNE $47
1762 003726
1763 003730
1764 003730 026727 000040 000001 CMP PASS,#1
1765 003736 003404 BLE $50
1766
1767 003740 104067 ERROR 67
1768 003740 104067
1769 003742 012767 177777 000030 MOV #SET,EXITFLAG
1770 003742 012767 177777 000030
1771 003750 $50:
1772 003750
1773 003750

```

; LOAD TBUF WITH A SECOND CHARACTER
 ; CHECK IMMEDIATELY THAT XMITRDY IS CLEAR
 ; AND THEN WAIT FOR IT TO SET

; SEND SECOND CHARACTER
 LET 2TBUF:B= #0

; GIVE IT TIME TO CLEAR
 ; XMITRDY SHOULD HAVE CLEARED UPON
 ; RECEIPT OF A CHARACTER
 IF #XMITRDY SET IN 2TCSR THEN

; XMITRDY DID NOT CLEAR IN TCSR
 LET ERRORFLAG := #SET

; DEFER ERROR TIMEOUT

ELSE

; WAIT A MAXIMUM
 ; OF 50 MSEC FOR
 ; XMIT RDY TO SET IN TCSR
 CALL TIMER IN (<#5,#XMITRDY,TCSR,#SET>)

IF.ERROR THEN

; XMIT RDY DID NOT SET IN TCSR
 ERRHRD 70,,DIDNOT

ENDIF

ENDIF ; OF DEFERED ERROR CALL

IF ERRORFLAG EQ #SET THEN

IF PASS GT #1 THEN

; CALL ERROR IF 2ND TRY
 ERRHRD 67,,DIDNOT

LET EXITFLAG := #SET

ENDIF

ELSE ; NO ERROR

004

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 43
 DVDVCA.P11 08-AUG-77 09:16 T11 TEST THAT XMIT RDY - TCSR 7 - CLEARS

SEQ 0042

1774	003750	000403		BR	\$51	
1775	003752		\$47:			LET EXITFLAG := #SET
1776	003752					
1777	003752	012767	177777	000020	MOV	#SET,EXITFLAG
1778	003760					ENDIF
1779	003760			\$51:		
1780	003760					EXIF EXITFLAG EQ #SET
1781	003760	026727	000014	177777	CMP	EXITFLAG,#SET
1782	003766	001401			BEQ	\$42
1783	003770					ENDLOOP
1784	003770	000674			BR	\$41
1785	003772			\$42:		
1786	003772					EXIT : SKIP AROUND FLAG WORDS
1787	003772	000403		BR	TST12	;;;EXIT THIS TEST
1788	003774	000000		PASS:	0	
1789	003776	000000		ERRORFLAG:	0	
1790	004000	000000		EXITFLAG:	0	
1791	004002					ENDTST

E04

 MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 44
 DVDVCF.P11 08-AUG-77 09:16

T11 TEST THAT XMIT RDY - TCSR 7 - CLEARS

SEQ 0043

```

1792 ;*****
1793 ;*****
1794 ;TEST 12      TEST THAT OUTPUTTING A CHAR FROM TBUF (WITH MAINT SET)
1795 ;            RESULTS IN RCVRDONE SETTING WITHIN A REASONABLE AMOUNT OF TIME
1796 ;            AND THAT RESET CLEARS THE BIT.
1797 ;*****
1798 004002 000004 1ST12: SCOPE
1799 004004 012767 000010 175146      MOV      #10,$TIMES      ;;DO 10 ITERATIONS
1800 004012 012767 000012 175160      MOV      #12,$TESTN      ;;SET TEST NUMBER IN APT MAIL BOX
1801
1802 004020                                IF #MAINTJUMP NOTSETIN SUSWR ORB APTCON EQ #TRUE
1803 004020 032767 040000 175172      BIT      #MAINTJUMP,$USWR
1804 004026 001404                                BEQ      $52
1805 004030 126727 005232 000001      CMPB     APTCON,#TRUE
1806 004036 001004                                DNE      $53
1807
1808 004040                                $52:                                EXIT TEST
1809 004040 012767 000001 175112      MOV      #1,$TIMES
1810 004046 000442                                BR       TS13              ;;EXIT THIS TEST
1811
1812                                $53:                                ENDF
1813
1814                                ; SET THE MAINTENANCE BIT
1815 004050                                LET @TCSR := @TCSR SET.BY #MAINT
1816 004050 052777 000004 175206      BIS      #MAINT,@TCSR
1817
1818 004056                                BGNSUB
1819 004056 012767 004064 175024      MOV      #64,$SLPERR
1820                                ; SEND A CHARACTER AND LET IT WRAP AROUND
1821
1822 004064                                LET @TBUF :B= #0
1823 004064 105077 175200      CLR      @TBUF
1824
1825                                ; WAIT A MAXIMUM OF 50 MSEC
1826                                ; FOR RCVR DONE TO SET IN
1827                                ; RCSR
1828                                CALL TIMER IN (<#5,#RCVRDONE,RCSR,#SET)
1829 004070
1830 004070 010546      MOV      RS,-(SP)
1831 004072 012745      MOV      #SET,-(RS)
1832 004076 016745      MOV      RCSR,-(RS)
1833 004102 012745      MOV      #RCVRDONE,-(RS)
1834 004106 012745      MOV      #5,-(RS)
1835 004112 004767 004344      JSR      PC,TIMER
1836 004116 012605      MOV      (SP)+,RS
1837
1838 004120                                ;DIDN'T SET IN TIME
1839 004120 103004      BCC      $54      IF.ERROR THEN
1840
1841                                ; RCVRDONE DID NOT SET IN RCSR
1842                                ; CAN NOT LEAVE WITH MAINT SET
1843 004122 042777 000004 175134      BIC      #MAINT,@TCSR
1844 004130                                LET @TCSR := @TCSR CLR.BY #MAINT
1845 004130 104071      ERROR      71      ERRARD 71,,DIDNOT
1846 004132
1847                                $54:                                ENDF

```

F04

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 45
 DVDVCA.P11 08-AUG-77 09:16 T12 TEST THAT OUTPUTTING A CHAR FROM TBUF (WITH MAINT SET)

SEQ 0044

```

1848
1849 004132                                ENDSUB
1850
1851 004132                                BGNSUB
1852 004132 012767 004140 174750          MOV    #64$,$LPERG
1853                                     ; NOW THAT IT IS SET SEE IF IT CAN BE RESET
1854                                     ; THIS ALSO WILL CLEAR THE MAINT. BIT
1855                                     BRESET
1856 004140 000005                          RESET
1857
1858 004142                                IF #RCVRDONE SET IN #RCSR THEN
1859 004142 032777 000200 175110          BIT    #RCVRDONE,#RCSR
1860 004150 001401                          BEQ    $55
1861                                     ; RCVRDONE DID NOT RESET IN RCSR.
1862                                     ERRHRD 72,,DIDNOT
1863 004152                                ERROR 72
1864 004154                                ENDIF
1865 004154                                $55.
1866 004154                                ENDSUB
1867 004154                                ENDTST
  
```

G04

 MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 46
 DVDVCA.P11 08-AUG-77 09:16

T12 TEST THAT OUTPUTTING A CHAR FROM TBUF (WITH MAINT SET)

SEQ 0045

```

1868
1869
1870
1871
1872 004154 000004
1873 004156 012767 000010 174774
1874 004164 012767 000013 175006
1875
1876 004172
1877 004172 032767 040000 175020
1878 004200 001404
1879 004202 126727 005060 000001
1880 004210 001004
1881 004212
1882 004212
1883 004212 012767 000001 174740
1884 004220 000451
1885 004222
1886
1887
1888
1889 004222
1890 004222 052777 000004 175034
1891 004230
1892 004230 012767 004236 174652
1893
1894
1895
1896
1897 004236
1898 004236 105077 175026
1899
1900
1901
1902 004242
1903 004242 010546
1904 004244 012745 177777
1905 004250 016745 175004
1906 004254 012745 000200
1907 004260 012745 000050
1908 004264 004767 004172
1909 004270 012605
1910
1911 004272
1912 004272 103007
1913
1914
1915 004274
1916 004274 042777 000004 174762
1917 004302
1918 004302 104073
1919
1920 004304
1921 004304 052777 000004 174752
1922 004312
1923 004312

```

```

*****
*****
TEST 13      TEST THAT RCVRDONE IS CLEARED BY READING RBUF
*****
TST13:  SCOPE
        MOV     #10,$TIMES      ;;DO 10 ITERATIONS
        MOV     #13,$TESTN     ;;SET TEST NUMBER IN APT MAIL BOX

                                IF #MAINTJUMP NOTSETIN SUSWR ORB APTCON EQ #TRUE
        BIT     #MAINTJUMP,$SUSWR
        BEQ     $56
        CMPB    APTCON,#TRUE
        BNE     $57
$56:
                                EXIT TEST
        MOV     #1,$TIMES
        BR      TST14           ;;EXIT THIS TEST
                                ENDIF
$57:
                                ; SET MAINT. BIT
                                LET @TCSR := @TCSR SET.BY #MAINT
                                BGNSUB
        MOV     #64,$SLPERR
                                ; OUTPUT A CHARACTER WITH MAINTENANCE
                                ; SET, AND WAIT FOR XMITRDY TO SET.
                                ; OUTPUT A CHARACTER
                                LET @TBUF :B= #0
                                ; WAIT MAXIMUM OF 500 MSEC
                                ; FOR RCVRDONE TO SET IN
                                ; RCSR
                                CALL TIMER IN (<#50,#RCVRDONE,RCSR,#SET)
        MOV     RS,-(SP)
        MOV     #SET,-(RS)
        MOV     RCSR,-(RS)
        MOV     #RCVRDONE,-(RS)
        MOV     #50,-(RS)
        JSR     PC,TIMER
        MOV     (SP)+,RS
                                ; DID IT BECAME READY?
                                IF.ERROR THEN
                                ; RCVRDONE DID NOT SET IN RCSR
                                ; CAN NOT LEAVE WITH MAINT SET
                                LET @TCSR := @TCSR CLR.BY #MAINT
                                ERRHRD 73,, DIDNOT
                                ; SET IT BACK TO CONTINUE
                                LET @TCSR := @TCSR SET.BY #MAINT
                                ENDIF
$60:

```

H04

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 47
 DVDVCA.P11 08-AUG-77 09:16 T13 TEST THAT RCVRDONE IS CLEARED BY READING RBUF

SEQ 0046

1924	004312					ENDSUB
1925						
1926						; NOW THAT IT IS SET LETS SEE IF READING THE
1927						; BUFFER CLEARS RCVRDONE.
1928						
1929						; READ BUFFER
1930	004312					LET RD :B= @RBUF
1931	004312	117700	174744			
1932						
1933	004316					IF #RCVRDONE SETIN @RCSR THEN
1934	004316	032777	000200	174734		
1935	004324	001407				
1936						; RCVRDONE DID NOT CLEAR IN RCSR
1937						; CAN NOT LEAVE WITH MAINT SET
1938	004326					LET @TCSR := @TCSR CLR.BY #MAINT
1939	004326	042777	000004	174730		
1940	004334					ERRHRD 74,,DIDNOT
1941	004334	104074				
1942						; SET IT BACK TO CONTINUE
1943	004336					LET @TCSR := @TCSR SET.BY #MAINT
1944	004336	052777	000004	174720		
1945	004344					ENDIF
1946	004344				\$61:	
1947	004344					ENDTST

104

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 48
 DVDVCA.P11 08-AUG-77 09:16

T13 TEST THAT RCVRDONE IS CLEARED BY READING RBUF

SEQ 0047

```

1948 .....
1949 .....
1950 .....
1951 *TEST 14 TEST THAT RCVRACT - RCSR 11 - SETS
1952 * WHEN A START BIT IS RECEIVED AND
1953 * CLEARS WHEN RCVRDONE - RCSR 7 - SETS
1954 .....
1954 004344 000004 TST14: SCOPE
1955 004346 012767 000010 174604 MOV #10,STIMES ;;DO 10 ITERATIONS
1956 004354 012767 000014 174616 MOV #14,STESTN ;;SET TEST NUMBER IN APT MAIL BOX
1957 004362 IFB APTCON EQ #FALSE AND #MAINTJUMP SETIN $USWR
1958 004362 126727 004700 000000 CMPB APTCON,#FALSE
1959 004370 001005 BNE $62
1960 004372 032767 040000 174620 BIT #MAINTJUMP,$USWR
1961 004400 001401 BEQ $62
1962 .....
1963 004402 ELSE ;; EXECUTE TEST
1964 004402 000404 BR $63
1965 .....
1966 004404 $62: EXIT TEST
1967 004404 012767 000001 174546 MOV #1,STIMES
1968 004412 000526 BR TST15 ;;;EXIT THIS TEST
1969 .....
1970 004414 $63: ENDF
1971 004414 LET @TCSR := @TCSR SET.BY #MAINT
1972 004414 052777 000004 174642 BIS #MAINT,@TCSR
1973 004422 LET FLAG :B= #CLR
1974 004422 112767 000000 004202 MOVB #CLR,FLAG
1975 004430 LET COUNT := #0
1976 004430 005067 000232 CLR COUNT
1977 .....
1978 ;LOAD A CHARACTER INTO TBUF
1979 ;WAIT FOR RCVRACT TO SET
1980 .....
1981 004434 ;SEND A CHARACTER
1982 004434 105077 174630 CLRB @TBUF LET @TBUF :B= #0
1983 .....
1984 004440 REPEAT
1985 004440 IF #RCVRACT SETIN @RCSR THEN
1986 004440 032777 004000 174612 BIT #RCVRACT,@RCSR
1987 004446 001404 BEQ $65
1988 .....
1989 004450 LET FLAG :B= #SET
1990 004456 ELSE
1991 004456 000402 BR $66
1992 .....
1993 004460 $65: LET COUNT := COUNT + #1
1994 004460 005267 000202 INC COUNT
1995 .....
1996 004464 $66: ENDF
1997 004464 UNTILB FLAG EQ #SET OR COUNT HI MAX
1998 004464 126727 004142 177777 CMPB FLAG,#SET
1999 004472 001404 BEQ $67
2000 004474 026767 000166 000162 CMP COUNT,MAX
2001 004502 101756 BLOS $64
2002 .....
2003 004504 $67: IF COUNT HI MAX THEN
  
```

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 49
DVDVCA.P11 08-AUG-77 09:16 T14 TEST THAT RCVRCT - RCSR 11 - SETS

2004	004504	026767	000156	000152
2005	004512	101410		
2006				
2007				
2008				
2009	004514			
2010	004514	042777	000004	174542
2011	004522			
2012	004522	104075		
2013	004524			
2014	004524	012767	000001	174426
2015	004532	000456		
2016	004534			
2017	004534			

CMP	COUNT, MAX
BLOS	\$70

```

:IT NEVER SET
:ACVRACT DID NOT SET IN RCSR.
: CAN NOT LEAVE WITH MAINT SET
LET  @TCSR := @TCSR CLR.BY @MAINT

ERRHRD 75,, DIDNOT

EXIT TEST

S TEST
ENDIF

```

\$70:

```
;CHECK FOR TIMING OF RCVRACT. CLEARING
;VS RCVRDONE SETTING
```

```
WHILE #RCVRACT SETIN #RCSR DO
```

571:

BIT #RCVRACT, JRCSS
BEQ \$72

IF RCVRDONE SETIN RCSCR THEN

```

BIT      #RCVRDONE, 2RCSR
BEQ      $73

```

```
IF @RCVRACT SET IN @RCSCR THEN
```

BIT RCYRACT, JRCSR
BEQ \$74

```
;RCVRDONE AND RCVRACT
: BOTH SET
```

```

; CAN NOT LEAVE WITH MAINT SET
LET 2TCSR := 2TCSR CLR.BY 2MAINT

```

ERRRD 76. DONEACT

```
;NO USE CONTINUING
EXIT 151
```

```
;;;EXIT THIS TEST
ENDIF
```

ENDIF

ENDO

\$74:

BR 571

\$72:

```

:RCVRACT = 0 NOW.
IF #RCVRDONE NOTSETIN #RCSR THEN

```

```

BIT      #RCVRDONE, #RCSR
BNE      $75

```

```
;RCVRDONE DID NOT SET IN RCSR
: CAN NOT LEAVE WITH MAINT SET
```

2058
2059

K04

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 50
 DVDVCA.P11 08-AUG-77 09:16 T14 TEST THAT RCVRCT - RCSR 11 - SETS

SEQ 0049

```

2060 004616          LET  @TCSR := @TCSR CLR.BY #MAINT
2061 004616 042777 000004 174440      BIC  #MAINT,@TCSR
2062 004624          ERRHRD 77,,DIDNOT
2063 004624 104077          ERROR  77
2064 004626          EXIT TEST
2065 004626 012767 000001 174324      MOV  #1,$TIMES
2066 004634 000415      BR      TST15          ;;;EXIT THIS TEST
2067 004636          ENDIF
2068 004636          $75:
2069          ;TEST THAT READING THE RECEIVER
2070          ;BUFFER CLEARS RCVRDONE
2071
2072
2073          ;READ CHAR.
2074 004636          LET  @R := @RBUF
2075 004636 017700 174420      MOV  @RBUF,@R
2076
2077          IF #RCVRDONE SETIN @RCSR THEN
2078 004642 032777 000200 174410      BIT  #RCVRDONE,@RCSR
2079 004650 001404          BEQ  $76
2080          ;RCVRDONE DID NOT CLEAR IN RCSR
2081          ; CAN NOT LEAVE WITH MAINT SET
2082 004652          LET  @TCSR := @TCSR CLR.BY #MAINT
2083 004652 042 000004 174404      BIC  #MAINT,@TCSR
2084 004660          ERRHRD 100,,DIDNOT
2085 004660 104100          ERROR  100
2086 004662          ENDIF
2087 004662          $76:
2088
2089          EXIT
2090 004662 000402      BR      TST15          ;;;EXIT THIS TEST
2091 004664 070000      MAX:70000
2092 004666 000000      COUNT: 0
2093
2094          ENDTST
2095

```

L04

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 51
 DVDVCA.P11 08-AUG-77 09:16 T14 TEST THAT RCVRCT - RCSR 11 - SETS

SEQ 0050

```

2096
2097
2098
2099
2100
2101 004670 000004
2102 004672 012767 000010 174260
2103 004700 012767 000015 174272
2104
2105 004706
2106 004706 032767 100000 174304
2107 004714 001416
2108 004716 126727 004344 000000
2109 004724 001012
2110 004726
2111 004726 032767 040000 174264
2112 004734 001401
2113
2114 004736
2115 004736 000404
2116 004740
2117 004740
2118 004740 012767 000001 174212
2119 004746 000567
2120 004750
2121 004750
2122 004750
2123 004750 000404
2124 004752
2125 004752
2126 004752 012767 000001 174200
2127 004760 000562
2128 004762
2129 004762
2130
2131 004762
2132 004762 052777 000004 174274
2133
2134
2135
2136 004770
2137 004770 012767 004776 174112
2138
2139
2140
2141
2142
2143 004776
2144 004776 105077 174266
2145
2146 005002
2147 005002 010546
2148 005004 012745 000050
2149 005010 004767 003724
2150 005014 012605
2151

;*****
;*****
;TEST 15 TEST THE OVERRUN BIT - RBUF 14
;*****
TST15: SCOPE
MOV #10,STIMES ;;DO 10 ITERATIONS
MOV #15,STESTN ;;SET TEST NUMBER IN APT MAIL BOX

IF #ERRBITS SETIN SUSWR ANDB APTCON EQ #FALSE TH

BIT #ERRBITS,SUSWR
BEQ $77
CMPB APTCON,#FALSE
BNE $77

IF #MAINTJUMP SETIN SUSWR THEN

BIT #MAINTJUMP,SUSWR
BEQ $100

;; NULL ---EXECUTE TEST
ELSE

BR $101

$100: BR $101

EXIT TEST

MOV #1,STIMES
BR TST16 ;;;EXIT THIS TEST
ENDIF

$101: ELSE

BR $102

$77: BR $102

EXIT TEST

MOV #1,STIMES
BR TST16 ;;;EXIT THIS TEST
ENDIF

$102:

LET @TCSR := @TCSR SET.BY #MAINT

BIS #MAINT,@TCSR

BGNSUB

MOV #64,$LPERR
;OUTPUT 2 CHARACTERS WITH
;AMPLE DELAYS BETWEEN FOR RECEPTION.
;THIS SHOULD AN CAUSE OVERRUN ERROR.

;OUTPUT 1 CHARACTER
LET @TBUF :B= #0

;GO AWAY FOR 500 M SEC
WAITUS 50

MOV R5,-(SP)
MOV #50,-(R5)
JSR PC,WAIT
MOV (SP)+,R5

```

M04

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 52
 DVDVCA.P11 08-AUG-77 09:16 T15 TEST THE OVERRUN BIT - RBUF 14

SEQ 0051

```

2152                                     ;OUTPUT 2ND CHARACTER
2153 005016                               LET @TBUF :B= #0
2154 005016 105077 174246                 CLRB  @TBUF
2155                                     ;LET OVERRUN HAPPEN
2156 005022                               WAITUS 50
2157 005022 010546                       MOV  R5, -(SP)
2158 005024 012745 000050                 MOV  #50, -(R5)
2159 005030 004767 003704                 JSR  PC, WAIT
2160 005034 012605                       MOV  (SP)+, R5
2161
2162                                     ;READ BUFFER AND ERRGR BITS
2163 005036                               LET R4 := @RBUF
2164 005036 017704 174220                 MOV  @RBUF, R4
2165
2166                                     ;IT DIDN'T SET
2167 005042                               IF #ORERR NOTSETIN R4 THEN
2168 005042 032704 040000                 BIT   #ORERR, R4
2169 005046 001010                       BNE   $103
2170
2171                                     ;ORERR DID NOT SET IN RBUF
2172                                     ; CAN NOT LEAVE WITH MAINT SET
2173 005050                               LET  @TCSR := @TCSR CLR.BY #MAINT
2174 005050 042777 000004 174206          BIC   #MAINT, @TCSR
2175 005056 104101                       ERRHRD 101,,DIDNOT
2176
2177                                     ;NO USE COMPOUNDING ERRORS
2178 005060                               EXIT TST
2179 005060 012767 000001 174072          MOV  #1, $TIMES
2180 005066 000517                       BR    TST16      ;;;EXIT THIS TEST
2181 005070                               ENDIF
2182 005070                               $103:
2183 005070                               ENDSUB
2184
2185                                     ;NOW SEE IF ERROR BIT SET WITH OVERRUN ERROR:
2186 005070                               BGNSUB
2187 005070 012767 005076 174012          MOV  #64$, $LPERR
2188 005076                               IF #ERROR NOTSETIN R4 THEN
2189 005076 032704 100000                 BIT   #ERROR, R4
2190 005102 001010                       BNE   $104
2191
2192                                     ;ERROR DID NOT SET IN RBUF
2193                                     ; CAN NOT LEAVE WITH MAINT SET
2194 005104                               LET  @TCSR := @TCSR CLR.BY #MAINT
2195 005104 042777 000004 174152          BIC   #MAINT, @TCSR
2196 005112                               ERRHRD 102,,DIDNOT
2197 005112 104102                       ERROR 102
2198
2199                                     ;--WHEN ORERR SET.
2200                                     ;GET OUT NOW.
2201 005114                               EXIT TST
2202 005114 012767 000001 174036          MOV  #1, $TIMES
2203 005122 000501                       BR    TST16      ;;;EXIT THIS TEST
2204 005124                               ENDIF
2205 005124                               $104:
2206 005124                               ENDSUB
2207

```

N04

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 53
 DVDVCA.P11 08-AUG-77 09:16 T15 TEST THE OVERRUN BIT - RBUF 14

SEQ 0052

```

2208 005124                                BGNSUB
2209 005124 012767 005132 173756      MOV    #64$,$LPERR
2210                                ;CHECK REAL RBUF TO SEE IF ORERR IS STILL SET.
2211
2212                                IF #ORERR NOTSETIN @RBUF THEN
2213 005132 032777 040000 174122      BIT     #ORERR,@RBUF
2214 005140 001010                      BNE     $105
2215
2216                                ;READING RBUF CLEARED ORERR.
2217                                ; CAN NOT LEAVE WITH MAINT SET
2218                                LET    @TCSR := @TCSR CLR.BY #MAINT
2219 005142 042777 000004 174114      BIC     #MAINT,@TCSR
2220 005150 104103                      ERROR   103
2221                                ;SKIP REST OF TEST
2222                                EXIT TEST
2223 005152
2224 005152 012767 000001 174000      MOV     #1,$TIMES
2225 005160 000462                      BR      T$16
2226                                ;;;EXIT THIS TEST
2227                                ENOIF
2228                                $105:
2229                                ENDSUB
2230                                BGNSUB
2231 005162 012767 005170 173720      MOV     #64$,$LPERR
2232                                ;NOW SEE IF THEY CLEAR WHEN ANOTHER CHAR. IS RECEIVED
2233
2234                                ;SEND A CHARACTER AROUND.
2235                                LET    @TBUF :B= #0
2236 005170 105077 174074      CLRB    @TBUF
2237                                ;LET IT CIRCULATE
2238                                WAITUS 50
2239 005174 010546
2240 005176 012745 000050      MOV     R5,-(SP)
2241 005202 004767 003532      MOV     #50,-(R5)
2242 005206 012605      JSR     PC,WAIT
2243                                (SP)+,R5
2244
2245                                IF #ORERR SETIN @RBUF THEN
2246 005210 032777 040000 174044      BIT     #ORERR,@RBUF
2247 005216 001410                      BEQ     $106
2248
2249                                ;ORERR DID NOT CLEAR IN @RBUF
2250                                ; CAN NOT LEAVE WITH MAINT SET
2251                                LET    @TCSR := @TCSR CLR.BY #MAINT
2252 005220 042777 000004 174036      BIC     #MAINT,@TCSR
2253 005226 104104                      ERROR   104
2254                                ;-AFTER RECEIVING ANOTHER CHAR
2255                                ;SKIP AROUND REST
2256                                EXIT TST
2257 005230 012767 000001 173722      MOV     #1,$TIMES
2258 005236 000433                      BR      T$16
2259                                ;;;EXIT THIS TEST
2260                                ENOIF
2261                                $106:
2262                                IF #ERROR SETIN @RBUF THEN
2263 005240 032777 100000 174014      BIT     #ERROR,@RBUF
  
```

B05

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 54
 DVDVCA.P11 08-AUG-77 09:16 T15 TEST THE OVERRUN BIT -- RBUF 14

SEQ 0053

```

2264 005246 001404          BEQ      $107
2265
2266
2267 005250
2268 005250 042777 000004 174006      BIC      #MAINT,@TCSR
2269 005256
2270 005256 104105          ERROR    105
2271
2272 005260
2273 005260          $107:
2274 005260
2275 005260
2276 005260 000422          BR       TST16
2277 005262 042522 042101 047111      ;;;EXIT THIS TEST
2278 005270 020107 041122 043125      ITCLRED:      .ASCIZ  /READING RBUF CLEARED OV
2279 005276 041440 042514 051101
2280 005304 042105 047440 042526
2281 005312 051122 047125 042440
2282 005320 051122 051117 000056
2283
2284 005326          .EVEN
2285          ENDTST
  
```

;ERROR DID NOT CLEAR IN RBUF
 ; CA: NOT LEAVE WITH MAINT SET
 LET @TCSR := @TCSR CLR.BY #MAINT

ERRHRD 105,,DIDNOT

ENDIF

ENDSUB

EXIT

ITCLRED:

.ASCIZ /READING RBUF CLEARED OV

C05

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 55
 DVDVCA.P11 08-AUG-77 09:16 T15 TEST THE OVERRUN BIT - RBUF 14

SEQ 0054

```

2286
2287
2288
2289
2290
2291
2292
2293
2294 005326 000004
2295 005330 012767 000010 173622
2296 005336 012767 000016 173634
2297
2298
2299
2300
2301
2302
2303 005344
2304 005344 032767 000200 173646
2305 005352 001416
2306 005354 032767 040000 173636
2307 005362 001412
2308 005364
2309 005364 126727 003676 000000
2310 005372 001001
2311
2312 005374
2313 005374 000404
2314 005376
2315 005376
2316 005376 012767 000001 173554
2317 005404 000565
2318 005406
2319 005406
2320 005406
2321 005406 000404
2322 005410
2323 005410
2324 005410 012767 000001 173542
2325 005416 000560
2326 005420
2327 005420
2328
2329 005420
2330 005420 005067 002710
2331 005424
2332 005424 012767 177777 000322
2333 005432
2334 005432 012767 177777 000316
2335 005440
2336 005440 052777 000004 173616
2337
2338 005446
2339 005446 005003
2340 005450 000401
2341 005452

*****
*****
*TEST 16 PROGRAMMABLE BAUD RATE TEST
* TEST AT ALL SPEEDS AVAILABLE
* ? COMPARISON WILL BE MADE TO SEE
* IF NEW TIME IS LESS THAN PREVIOUS.
*****
TST16: SCOPE
MOV #10,$TIMES ;;DO 10 ITERATIONS
MOV #16,$TESTN ;;SET TEST NUMBER IN APT MAIL BOX

;;NOTE: IF PROGRAMMABLE BAUD RATE TEST IS TO BE RUN ON THE APT CONSOLE
;; THE OPERATOR MUST MAKE CERTAIN THAT LOCATION $USWR IS
;; SET WITH #16 IN BITS <11:8>(9600 BAUD).

IF #PBR SETIN $USWR AND #MAINTJUMP SETIN $USWR T
BIT #PBR,$USWR
BEQ $110
BIT #MAINTJUMP,$USWR
BEQ $110
IFB APTCON EQ #FALSE THEN
ELSE;;; NULL --- EXECUTE TEST

$111: BR $112
EXIT TST
MOV #1,$TIMES
BR TST17 ;;;EXIT THIS TEST
ENDIF
$112: ELSE ; CATCH IT HERE

$110: EXIT TEST ; IF EITHER OF CASES FAIL
MOV #1,$TIMES
BR TST17 ;;;EXIT THIS TEST
ENDIF
$113:

LET ERRCHK := #0 ; CLEAR ERROR WORD
LET OLD := #-1
LET OLD+2 := #-1
LET @TCSR := @TCSR SET.BY #MAINT
;EACH BAUD RATE
INCR R3 FROM #0 TO #15. BY #1

$115:

```

D05

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 56
 DVDVCA.P11 08-AUG-77 09:16 T16 PROGRAMMABLE BAUD RATE TEST

SEQ 0055

2342	005452	005203			INC	R3	
2343	005454			\$114:			
2344	005454	020327	000017		CMP	R3,#15.	
2345	005460	003062			BGT	\$116	
2346	005462						LET R0 := @RBUF
2347	005462	017700	173574		MOV	@RBUF,R0	
2348							;CHANGE BAUD RATE
2349	005466						LET @TCSRHI := B=RATES(R3)
2350	005466	116377	005676 173572		MOVB	RATES(R3),@TCSRHI	
2351							;FLAG
2352	005474						LET BIT := #0
2353	005474	005002			CLR	BIT	
2354							;OUTPUT THE CHARACTER
2355	005476						LET @TBUF := #0
2356	005476	005077	173566		CLR	@TBUF	
2357							;INITIALIZE COUNTER
2358	005502						LET NEW := #0
2359	005502	005067	000242		CLR	NEW	
2360	005506						LET NEW+2 := #0
2361	005506	005067	000240		CLR	NEW+2	
2362	005512						WHILE BIT EQ #0 DO
2363	005512			\$117:			
2364	005512	005702			TST	BIT	
2365	005514	001014			BNE	\$120	
2366	005516						IF @RCVRDONE SETIN @RCSR THEN
2367	005516	032777	000200 173534		BIT	@RCVRDONE,@RCSR	
2368	005524	001403			BEQ	\$121	
2369							;DONE - ITS READY
2370	005526						LET BIT := #1
2371	005526	012702	000001		MOV	#1,BIT	
2372	005532						ELSE
2373	005532	000404			BR	\$122	
2374	005534			\$121:			
2375							;OTHERWISE-INCREMENT TIME
2376	005534						LET NEW := NEW + #1
2377	005534	005267	000210		INC	NEW	
2378	005540						LET NEW+2 := NEW+2 + CARRY
2379	005540	005567	000206		ADC	NEW+2	
2380	005544						ENDIF
2381	005544			\$122:			;SIGNALS DONE
2382							ENDDO
2383	005544						
2384	005544	000762			BR	\$117	
2385	005546			\$120:			
2386							IF NEW+2 LO OLD+2 THEN
2387	005546						
2388	005546	026767	000200 000202		CMP	NEW+2,OLD+2	
2389	005554	103001			BHIS	\$123	
2390							; OK
2391	005556						ELSE
2392	005556	000414			BR	\$124	
2393	005560			\$123:			
2394							; NEW+2 >= OLD+2
2395	005560						IF NEW+2 EQ OLD+2 AND NEW LO OLD THEN
2396	005560	026767	000166 000170		CMP	NEW+2,OLD+2	
2397	005566	001005			BNE	\$125	

E05

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 57
 DVDVCA.P11 08-AUG-77 09:16 T16 PROGRAMMABLE BAUD RATE TEST

SEQ 0056

```

2398 005570 026767 000154 000156      CMP      NEW,OLD
2399 005576 103001                      BHIS     $125
2400                                     ;OK
2401 005600                                     ELSE
2402 005600 000403                      BR       $126
2403 005602                      $125:
2404                                     ;NEW+2 > OLD+2 OR
2405                                     ; (NEW+2 = OLD+2 AND
2406                                     ; NEW >= OLD)
2407                                     ;BAUD RATE DIDN'T CHANGE
2408 005602                                     LET ERRCHK := #BIT2 ; SET ERROR INDICATOR
2409 005602 012767 000004 002524      MOV      #BIT2,ERRCHK
2410 005610                                     ENDIF
2411 005610                      $126:
2412 005610                                     ENDIF
2413 005610                      $124:
2414                                     ;UPDATE OLD TIME
2415 005610                                     LET OLD := NEW
2416 005610 016767 000134 000136      MOV      NEW,OLD
2417 005616 016767 000130 000132      MOV      NEW+2,OLD+2
2418 005616
2419                                     ENDINC ;BAUD RATE
2420 005624
2421 005624 000712                      BR       $115
2422 005626                      $116:
2423 005626                                     LET R3 :B= $USWR+1 AND #17 ; PUT BAUD BACK
2424 005626 116703 173367      MOV      $USWR+1,R3
2425 005632 110346      MOV      R3,-(SP)
2426 005634 142716 000017      BICB     #17,(SP)
2427 005640 142603      BICB     (SP)+,R3
2428 005642                                     LET R3 := R3 CLR.BY #177400
2429 005642 042703 177400      BIC      #177400,R3
2430 005646                                     LET @TCSRHI :B= RATES(R3) ; LIKE HE WANTED IT
2431 005646 116377 005676 173412      MOV      RATES(R3),@TCSRHI
2432
2433                                     ; CAN NOT LEAVE WITH MAINT SET
2434 005654                                     LET @TCSR := @TCSR CLR.BY #MAINT
2435 005654 042777 000004 173402      BIC      #MAINT,@TCSR
2436 005662                                     IF #BIT2 SETIN ERRCHK THEN
2437 005662 032767 000004 002444      BIT      #BIT2,ERRCHK
2438 005670 001401                      BEQ      $127
2439                                     ; REPORT DEFERED ERROR
2440 005672                                     ERRHRO 126
2441 005672 104126                      ERROR 126
2442 005674                                     ENDIF
2443 005674                      $127:
2444 005674                                     EXIT ;SKIP TABLE
2445 005674 000431                      BR       TST17 ;;;EXIT THIS TEST
2446
2447 005676
2448
2449
2450
2451
2452
2453
  
```

RATES: ; A TABLE OF THE ACTUAL BYTES TO MOVE INTO THE
 ; UPPER BYTE OF XCSR FOR EACH BAUD RATE
 ; ** NOTE: : THE VALUE INDICATED IN THE COLUMN 'OFFSET
 ; ** INTO TABLE' CAN BE PLACED INTO BITS(11:8)
 ; ** OF LOCATION 'SUSWR' TO CAUSE THE CORRESPONDING
 ; ** BAUD TO BE SELECTED IN THE DLV11-F UPON
 ; ** COMPLETION OF THIS TEST.

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 58
DVDVCA.P11 08-AUG-77 09:16 T16 PROGRAMMABLE BAUD RATE TEST

```

2454
2455
2456 005676 010 R0050: .BYTE 010 BAUD 50 OFFSET INTO TABLE 0
2457 005677 030 R0070: .BYTE 030 70 1
2458 005700 050 R0110: .BYTE 050 110 2
2459 005701 070 R0135: .BYTE 070 135 3
2460 005702 110 R0150: .BYTE 110 150 4
2461 005703 130 R0300: .BYTE 130 300 5
2462 005704 150 R0600: .BYTE 150 600 6
2463 005705 170 R0200: .BYTE 170 1200 7
2464 005706 210 R1800: .BYTE 210 1800 10
2465 005707 230 R2000: .BYTE 230 2000 11
2466 005710 250 R2400: .BYTE 250 2400 12
2467 005711 270 R3600: .BYTE 270 3600 13
2468 005712 310 R4800: .BYTE 310 4800 14
2469 005713 330 R7200: .BYTE 330 7200 15
2470 005714 350 R9600: .BYTE 350 9600 16
2471 005715 370 R10000: .BYTE 370 19200 17
2472
2473 005716 040502 042125 051040 BAUDRATE: .ASCIZ /BAUD RATE DIDN'T CHANGE./
2474 005724 052101 020105 044504
2475 005732 047104 052047 041440
2476 005740 040510 043516 027105
2477 005746 000
2478 005750
2479 005750 000000 000000 .EVEN
2480 005754 000000 000000 NEW: 0,0
2481 005760 000000 000000 OLD: 0,0
2482
2483
2484
ENDST

```

G05

MAINDEC-11-DVDVC-A
DVDVCA.P11MACY11 27(1006)
08-AUG-77 09:1608-AUG-77 09:20 PAGE 59
T16 PROGRAMMABLE BAUD RATE TEST

SEQ 0058

2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540

```

005760 000004
005762 012767 000010 173170
005770 012767 000017 173202

005776
005776 005067 003024

006002
006002 016703 173250

006006
006006 062703 000004

006012
006012 010146
006014 010301
006016 012721 011020
006022 012711 000340
006026 012601

006030
006030 012767 006036 173052

006036
006036 010546
006040 012745 177777
006044 016745 173214
006050 012745 000200
006054 012745 000050
006060 004767 00237F
006064 012605

006066
006066 042777 000100 173170

006074 012746 000000
006100 012746 006106
006104 000002
006106

```

```

*****
*****
TEST 17 TRANSMITTER INTERRUPT LOGIC TEST
LOGICALLY THIS IS 4 SEPARATE TESTS
A) DOES TRANSMITTER INTERRUPT LOGIC WORK
B) AT PRIORITY OF 0
C) AND ONLY ONCE
D) BUT NOT WITH INTERRUPT ENABLE CLEAR
*****

```

```

TST17: SCOPE
MOV #10, $TIMES ;; DO 10 ITERATIONS
MOV #17, $TESTN ;; SET TEST NUMBER IN APT MAIL BOX
;; CLEAR 'INTERRUPT OCCURED' FLAG
LET INTFLAG := #0

;; GET VECTOR ADDRESS
LET R3 := DLVEC

;; FOR THE TRANSMITTER
LET R3 := R3 + #4

;; SET VECTOR TO POINT TO TRANS.SRV AT PRI
SETVEC R3, #INTSRV, #PR7

BGNSUB

;; MAKE SURE THAT TRANSMITTER READY IS SET
CALL TIMER IN (<#50, #XMITRDY, TCSR, #SET>)

;; CLEAR INTERRUPT ENABLE
LET @TCSR := @TCSR CLR.BY #XMITIE

;; SET IT TO 0
MOV #PRO, -(SP) ;; PUT NEW PS ON STACK
MOV #65$, -(SP) ;; PUT NEW PC ON STACK
RTI ;; POP NEW PC AND PS

65$:

;; NOW SET I.E. BIT
LET @TCSR := @TCSR SET.BY #XMITIE

```

;LET INTERRUPT HAVE TIME TO OCCUR

H05

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 60
 DVDVCA.P11 08-AUG-77 09:16 T17 TRANSMITTER INTERRUPT LOGIC TEST

SEQ 0059

```

2541 006114                                WAITUS 10
2542 006114 010546                        MOV    RS,-(SP)
2543 006116 012745 000010                MOV    #10,-(RS)
2544 006122 004767 002612                JSR    PC,WAIT
2545 006126 012605                        MOV    (SP)+,RS
2546
2547                                     ;DID EXACTLY 1 INTERRUPT OCCUR
2548                                     IF INTFLAG NE #1 THEN
2549 006130                                ;NO - WAS IT 0 OR MORE THAN ONCE
2550 006130 026727 002672 000001          CMP    INTFLAG,#1
2551 006136 001406                        BEQ    $13C
2552                                     ;IF INTFLAG EQ #0 THEN
2553 006140                                ;TRANSMITTER DID NOT INTERRUPT IN TIME
2554 006140 005767 002662                TST    INTFLAG
2555 006144 001002                        BNE    $131
2556                                     ;ERRHRD 106,,DIDNOT
2557 006146 104106                        ERROR   106
2558 006150                                ELSE
2559 006150 000401                        BR      $132
2560 006152                                ;TWICE
2561                                     ;TRANSMITTER INTERRUPTED TWICE
2562                                     ;ERRHRD 107,,TWICE
2563 006152 104107                        ERROR   107
2564 006152                                ENDF
2565 006154                                ENDF
2566 006154                                ENDF
2567 006154                                ENDF
2568 006154                                ENDF
2569 006154                                ENDF
2570                                     ;INTERRUPT WITHOUT INTERRUPT ENABLE SET
2571 006154                                BGNSUB
2572 006154 012767 006162 172726          MOV    #64$,$LPERR
2573                                     ;CLEAR 'INTERRUPT OCCURED' FLAG
2574 006162                                LET INTFLAG := #0
2575 006162 005067 002640                CLR    INTFLAG
2576                                     ;CLEAR INTERRUPT ENABLE
2577 006166                                LET @TCSR := @TCSR CLR.BY #XMITIE
2578 006166 042777 000100 173070          BIC    #XMITIE,@TCSR
2579                                     ;NO INTERRUPTS SHOULD OCCUR.
2580 006174 012746 000000                MOV    #PRO,-(SP)
2581 006200 012746 006206                MOV    #65$,-(SP)
2582 006204 000002                        RTI
2583 006206                                ;;PUT NEW PS ON STACK
2584                                     ;;PUT NEW PC ON STACK
2585                                     ;;POP NEW PC AND PS
2586 006206                                ;DARE IT TO HAPPEN
2587 006206 010546                        WAITUS 10
2588 006210 012745 000010                MOV    RS,-(SP)
2589 006214 004767 002520                MOV    #10,-(RS)
2590 006220 012605                        JSR    PC,WAIT
2591 006222 005767 002600                MOV    (SP)+,RS
2592 006226 001401                        TST    INTFLAG
2593                                     IF INTFLAG NE #0 THEN
2594 006230                                ;INTERRUPT OCCURED WITH I E CLEARED
2595 006230 104110                        ERRHRD 110,NOTENAB
2596 006232                                ENDF

```

MM:NDCC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 61
 DVDVCA.P11 08-AUG-77 09:16 T17 TRANSMITTER INTERRUPT LOGIC TEST

SEQ 0060

```

2597 006232          $133:
2598 006232
2599 006232 000005      RESET
2600 006234
2601
2602 006234
2603 006234 010146      MOV    R1,-(SP)
2604 006236 010246      MOV    R2,-(SP)
2605 006240 012701 000003  MOV    #R3,R1
2606 006244 010102      MOV    R1,R2
2607 006246 062702 000002  ADD    #2,R2
2608 006252 010221      MOV    R2,(R1)+
2609 006254 005011      CLR    (R1)
2610 006256 012602      MOV    (SP)+,R2
2611 006260 012601      MOV    (SP)+,R1
2612
2613 006262
2614
2615
2616
2617
2618
2619

```

BRESET
 ENDSUB
 ;RESTORE VECTOR AREA
 CLRVEC R3
 ;;PUSH R1 ON STACK
 ;;PUSH R2 ON STACK
 ;;POP STACK INTO R2
 ;;POP STACK INTO R1
 ENDTST

J05

 MAINDEC-11-DVDVC-A
 DVDVCA.P11 08-AUG-77 09:16

MACY11 27(1006)

 08-AUG-77 09:20 PAGE 62
 T17 TRANSMITTER INTERRUPT LOGIC TEST

SEQ 0061

```

2620 ;*****
2621 ;*****
2622 ;*****
2623 ;TEST 20 RECEIVER INTERRUPT LOGIC TEST
2624 ; THIS TEST COVERS ALL OF THE RECEIVER
2625 ; SIDE OF THE INTERRUPT LOGIC IN
2626 ; CHARACTER MODE.
2627 ;*****
2627 006262 000004 TST21 SCOPE
2628 006264 012767 000010 172666 MOV #10,$TIMES ;DO 10 ITERATIONS
2629 006272 012767 000020 172700 MOV #20,$TESTN ;SET TEST NUMBER IN APT MAIL BOX
2630 006300 IF #MAINTJUMP NOTSETIN $USWR ORB APTCON EQ #TRUE
2631 006300 032767 040000 172712 BIT #MAINTJUMP,$USWR
2632 006306 001404 BEQ $134
2633 006310 126727 002752 000001 CMPB APTCON,#TRUE
2634 006316 001002 BNE $135
2635 006320 $134:
2636 006320 000167 000422 JMP TST21 ; EXIT TEST
2637 006324 $135: ENDF
2638 006324
2639 ;
2640 ;CLEAR INTERRUPT OCCURED FLAG
2641 ;SET UP RECEIVER INTER.VECTOR
2642 006324 SETVEC DLVEC,#INTSRV,#PR7
2643 006324 010146 MOV R1,-(SP)
2644 006326 016701 172724 MOV DLVEC,R1
2645 006332 012721 011020 MOV #INTSRV,(R1)+
2646 006336 012711 000340 MOV #PR7,(R1)
2647 006342 012601 MOV (SP),R1
2648 ;PRIORITY 0 AND MULTIPLE INTERRUPT TEST.-RCVRIE
2649 006344 BGNSUB
2650 006344 012767 006352 172536 MOV #64,$SLPERR
2651 006352 LET INTFLAG := #0
2652 006352 005067 002450 CLR INTFLAG
2653 ;SET MAINT. BIT
2654 006356 LET @TCSR := @TCSR SET.BY #MAINT
2655 006356 052777 000004 172700 BIS #MAINT,@TCSR
2656 ;CLEAR INTERRUPTS
2657 006364 LET @RCR := @RCR CLR.BY #RCVRIE
2658 006364 042777 000100 172666 BIC #RCVRIE,@RCR
2659 ;CHANGE PRIORITY
2660 ;...TO 0
2661 006372 012746 000000 MOV #PR0,-(SP) ;PUT NEW PS ON STACK
2662 006376 012746 006404 MOV #65,-(SP) ;PUT NEW PC ON STACK
2663 006402 000002 RTI ;POP NEW PC AND PS
2664 006404 65$:
2665 ;
2666 ;SEND A CHARACTER
2667 006404 LET @TBUF := #0
2668 006404 105077 172660 CLRB @TBUF
2669 ;WAIT A MAXIMUM
2670 ;OF 50 MSEC FOR
2671 ;XMIT RDY TO SET IN TCSR
2672 006410 CALL TIMER IN (<#50,#XMITRDY,TCSR,#SET>)
2673 006410 010546 MOV R5,-(SP)
2674 006412 012745 177777 MOV #SET,-(R5)
2675 006416 016745 172642 MOV TCSR,-(R5)

```

KOL

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 63
 DVDVCA.P11 08-AUG-77 09:16 T20 RECEIVER INTERRUPT LOGIC TEST

SEQ 0062

```

2676 006422 012745 000200      MOV      #XMITRDY, -(RS)
2677 006426 012745 000050      MOV      #50, -(RS)
2678 006432 004767 002024      JSR      PC, TIMER
2679 006436 012605              MOV      (SP)+, RS
2680
2681 006440              ;SET INTERRUPT ENABLE
2682 006440 052777 000100 172612  BIS      #RCVRIE, @RCR
2683
2684 006446              ;LET IT COME IN.
2685 006446 010546              WAITUS 10
2686 006450 012745 000010      MOV      RS, -(SP)
2687 006454 004767 002260      MOV      #10, -(RS)
2688 006460 012605              JSR      PC, WAIT
2689
2690
2691 006462              MOV      (SP)+, RS
2692 006462 026727 002340 000001  CMP      INTFLAG, #1
2693 006470 001411              IF INTFLAG NE #1 THEN
2694
2695              ;DID HE DO IT RIGHT?
2696 006472              IF INTFLAG NE #1 THEN
2697 006472 042777 000004 172564  BIC      #MAINT, @TCSR
2698 006500              ;NONE OCCURED
2699 006500 005767 002322              ; CAN NOT LEAVE WITH MAINT SET
2700 006504 001002              LET      @TCSR := @TCSR CLR.BY #MAINT
2701
2702              IF INTFLAG EQ #0 THEN
2703 006506              ;RECEIVER DID NOT INTERRUPT IN TIME
2704 006506 104111              ERRHRD 111,,DIDNOT
2705
2706 006510              ;TWICE OR MORE
2707 006510 000401              ELSE
2708
2709              $137:
2710 006512              ;RECEIVER INTERRUPTED TWICE
2711 006512 104112              ERRHRD 112,,TWICE
2712
2713 006514              ENDIF
2714 006514              $140:
2715 006514              $136:
2716 006514              ENDIF
2717 006514 042777 000004 172542  BIC      #MAINT, @TCSR
2718 006522              ;RESET MAINT. BIT.
2719
2720              LET @TCSR := @TCSR CLR.BY #MAINT
2721
2722              ENDSUB
2723
2724
2725              ;INTERUPT WITHOUT I E SET.
2726 006522              BGNSUB
2727 006522 012767 006530 172360  MOV      #64$, $LPERR
2728 006530              LET @TCSR := @TCSR SET.BY #MAINT
2729 006530 052777 000004 172526  BIS      #MAINT, @TCSR
2730
2731              ;CLEAR INTERRUPT FLAG

```


M05

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 65
 DVDVCA.P11 08-AUG-77 09:16 T20 RECEIVER INTERRUPT LOGIC TEST

SEQ 0064

```

2788 006702 104115          ERROR 115
2789 006704
2790 006704          $142:
2791 006704
2792 006704
2793 006704 000005          RESET
2794
2795
2796 006706
2797 006706 042777 000004 172350      BIC      #MAINT,@TCSR
2798 006714
2799
2800 006714
2801 006714 017704 172336      MOV      @DLVEC,R4
2802 006720
2803 006720 010146      MOV      R1,-(SP)      ;; PUSH R1 ON STACK
2804 006722 010246      MOV      R2,-(SP)      ;; PUSH R2 ON STACK
2805 006724 012701 000004      MOV      #R4,R1
2806 006730 010102      MOV      R1,R2
2807 006732 062702 000002      ADD      #2,R2
2808 006736 010221      MOV      R2,(R1)+
2809 006740 005011      CLR      (R1)
2810 006742 012602      MOV      (SP)+,R2      ;; POP STACK INTO R2
2811 006744 012601      MOV      (SP)+,R1      ;; POP STACK INTO R1
2812 006746
  
```

```

ENDIF
;CLEAR THE WORLD
BRESET
;RESET MAINT. BIT.
LET @TCSR := @TCSR CLR.BY #MAINT
ENDSUB
LET R4 := @DLVEC
CLRVEC R4
ENDTST
  
```

N05

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 66
 DVDVCA.P11 08-AUG-77 09:16 T20 RECEIVER INTERRUPT LOGIC TEST

SEQ 0065

```

2813 ;*****
2814 ;*****
2815 ;*****
2816 *TEST 21 TEST ACTUAL DATA TRANSFERRED
2817 *NON-INTERRUPT MAINTENANCE BIT SET
2818 ;*****
2818 006746 000004 TST21: SCOPE
2819 006750 012767 000001 172202 MOV #1,$TIMES ;;DO 1 ITERATION
2820 006756 012767 000021 172214 MOV #21,$TESTN ;;SET TEST NUMBER IN APT MAIL BOX
2821 006764 032767 040000 172226 BIT #MAINTJUMP,$USWR IF #MAINTJUMP NOTSETIN $USWR ORB APTCON EQ #TRUE
2822 006764 001404 BEQ $143
2823 006772 126727 002266 000001 CMPB APTCON,#TRUE
2824 007002 001004 BNE $144
2825 007004 $143:
2826 007004 EXIT TEST
2827 007004
2828 007004 012767 000001 172146 MOV #1,$TIMES
2829 007012 000520 BR TST22 ;;EXIT THIS TEST
2830 007014 $144: ENDF
2831 007014
2832 007014 LET ERRCHK := #0
2833 007014 005067 001314 CLR ERRCHK
2834 007020 ;SET MAINT. BIT
2835 007020 052777 000004 172236 BIS #MAINT,$TCSR LET $TCSR := $TCSR SET.BY #MAINT
2836 007026 ;CHANGE PRIORITY
2837 007032 012746 000000 007040 MOV #PRO,-(SP) ;;PUT NEW PS ON STACK
2838 007036 000002 007040 MOV #64$,-(SP) ;;PUT NEW PC ON STACK
2839 007040 64$: RTI ;;POP NEW PC AND PS
2840 ;GET DATA MASK.
2841 007040 CALL DATLNG OUT (R1)
2842 007040 162705 000002 SUB #1*2,$R5
2843 007044 004767 001570 JSR PC,DATLNG
2844 007050 012501 MOV (R5)+,$R1
2845 ;ALL BINARY CHAR.
2846 007052 INCR R2 FROM #0 TO #377 BY #1
2847 007052 005002 CLR R2
2848 007054 000401 BR $145
2849 007056 $146: INC R2
2850 007056 005202 $145: CMP R2,$377
2851 007060 020227 000377 BGT $147
2852 007064 003056
2853 ;TRANSMIT CHAR IN R2
2854 007066 CALL TIMER IN (<$50,$XMITRDY,$TCSR,$SET)
2855 007066 010546 MOV $R5,-(SP)
2856 007070 012745 177777 MOV #SET,-($R5)
2857 007074 016745 172164 MOV $TCSR,-($R5)
2858 007100 012745 000200 MOV #XMITRDY,-($R5)

```

MAINDEC-11-DVDC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 67
 DVDVCA.P11 08-AUG-77 09:16 T21 TEST ACTUAL DATA TRANSFERED

SEQ 0066

```

2869 007104 012745 000050      MOV    #50,-(R5)
2870 007110 004767 001346      JSR    PC,TIMER
2871 007114 012605              MOV    (SP)+,R5
2872 007116              BCC    $150
2873 007116 103003              BIS    #BIT3,ERRCHK
2874 007120 052767 000010 001206
2875 007120 052767 000010 001206
2876 007126              $150:
2877 007126
2878
2879
2880 007126              MOV    R2,@TBUF
2881 007126 110277 172136
2882
2883 007132              MOV    R5,-(SP)
2884 007132 010546              MOV    #SET,-(R5)
2885 007134 012745 177777      MOV    RCSR,-(R5)
2886 007140 016745 172114      MOV    #RCVRDONE,-(R5)
2887 007144 012745 000200      MOV    #50,-(R5)
2888 007150 012745 000050      JSR    PC,TIMER
2889 007154 004767 001302      MOV    (SP)+,R5
2890 007160 012605
2891 007162              BCC    $151
2892 007162 103003              BIS    #BIT4,ERRCHK
2893 007164 052767 000020 001142
2894 007164 052767 000020 001142
2895 007172              $151:
2896 007172
2897
2898 007172              MOV    @RBUF,R3
2899 007172 017703 172064
2900
2901
2902
2903
2904
2905
2906 007176              MOV    R2,R4
2907 007176 010204              BIC    R1,R4
2908 007200 040104
2909 007202              BIC    R1,R3
2910 007202 040103
2911
2912
2913 007204              CMP    R4,R3
2914 007204 020403              BEQ    $152
2915 007206 001404
2916
2917
2918 007210              BIC    #MAINT,@TCSR
2919 007210 042777 000004 172046
2920 007216
2921 007216 104116              ERROR 116
2922
2923 007220              $152:
2924 007220

```

IF.ERROR THEN

LET ERRCHK := ERRCHK SET.BY #BIT3

ENDIF

;TRANSMIT IT
 LET @TBUF := R2

CALL TIMER IN (<#50,#RCVRDONE,RCSR,#SET>)

IF.ERROR THEN

LET ERRCHK := ERRCHK SET.BY #BIT4

ENDIF

;AND SAVE IT
 LET R3 := @RBUF

;COMPARE TO SEE IF WE RECEIVED IT ALL

;CLEAN OFF NON-DATA BITS
 ;ON BOTH TRANSMITTED AND
 LET R4 := R2 CLR.BY R1

LET R3 := R3 CLR.BY R1

;RECEIVED DATA
 IF R4 NE R3 THEN

;DATA COMPARE ERROR
 ;CAN NOT LEAVE WITH MAINT SET
 LET @TCSR := @TCSR CLR.BY #MAINT

ERRHRD 116,COMP,SBWAS

;(<TRANSMITTED> <RECEIVED>)
 ENDIF

C06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 68
 DVDVCA.PI: 08-AUG-77 09:16 T2! TEST ACTUAL DATA TRANSFERED

SEQ 0067

2925	007220					ENDINC ; R2
2926	007220	000716				
2927	007222			\$147:	BR	\$146
2928						
2929						RESET MAINT. BIT.
2930	007222					LET ATCSR := ATCSR CLR.BY #MAINT
2931	007222	04277'	000004	172034	BIC	#MAINT,ATCSR
2932	007230					IF #BIT3 SETIN ERRCHK THEN
2933	007230	032767	000010	00 J76	BIT	#BIT3,ERRCHK
2934	007236	001401			BEQ	\$153
2935	007240					ERRHRD 130
2936	007240	104130			ERROR	130
2937	007242					ENDIF
2938	007242			\$153:		
2939	007242					IF #BIT4 SETIN ERRCHK THEN
2940	007242	032767	000020	00:064	BIT	#BIT4,ERRCHK
2941	007250	001401			BEQ	\$154
2942	007252					ERRHRD 131
2943	007252	104131			ERROR	131
2944	007254					ENDIF
2945	007254			\$154:		
2946	007254					ENDTST
2947						
2948						
2949						

D06

MAINDEC-11-DVDC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 69
 DVDCR.P11 08-AUG-77 09:16 T21 TEST ACTUAL DATA TRANSFERED

SEQ 0068

```

2950          ;*****
2951          ;*****
2952          ;TEST 22      TEST DATA THROUGH WRAP
2953          ;*****
2954 007254 000004          TST2: SCOPE
2955 007256 012767 000001 171674      MOV      #1,STIMES      ;;DO 1 ITERATION
2956 007264 012767 000022 171706      MOV      #22,STIMN      ;;SET TEST NUMBER IN APT MAIL BOX
2957 007272          IF #WRAP NOTSETIN SUSWR OR #COMSPD NOTSETIN SUSW
2958 007272 032767 020000 171720      BIT      #WRAP,SUSWR
2959 007300 001104          BEQ      $155
2960 007302 032767 000100 171710      BIT      #COMSPD,SUSWR
2961 007310 001014          BNE      $156
2962          $155:
2963          ;CAN'T TEST WITHOUT A WRAP
2964          EXIT TST
2965 007312 012767 000001 171640      MOV      #1,STIMES
2966 007320 000522          BR       TST23      ;;EXIT THIS TEST
2967 007322          IFB APTCON EQ #TRUE THEN
2968 007322 126727 001740 000001      CMPB     APTCON,#TRUE
2969 007330 001004          BNE      $157
2970 007332          EXIT TEST
2971 007332 012767 000001 171620      MOV      #1,STIMES
2972 007340 000512          BR       TST23      ;;EXIT THIS TEST
2973 007342          ENDF
2974 007342          $157:
2975 007342          $156:
2976 007342          ENDF
2977          ;DON'T USE MAINT.
2978 007342 042777 000004 171714      BIC      #MAINT,@TCSR      LET @TCSR := @TCSR CLR.BY #MAINT
2979 007342
2980          ; IF A SPECIAL TURN AROUND CARD IS
2981          ; CONNECTED IN PLACE OF THE WRAP
2982          ; SETTING READER RUN WILL ENABLE IT.
2983          ; THIS MODULE IS ONLY USED IN MANUFACTUR
2984          ; AND ONLY ON THE CONSOLE DLV11-F.
2985          ; IF NO SPECIAL MODULE IS AVAILABLE,
2986          ; AND THE WRAP BIT IS SET IN SUSWR
2987          ; THEN THIS TEST WILL ERROR ON THE CONSO
2988
2989 007350          LET @RCSR := @RCSR SET.BY #11
2990 007350 052777 000011 171702      BIS      #11,@RCSR
2991          ;CHANGE PRIORITY
2992          ; TO 0
2993 007356 012746 000000          MOV      #PRO,-(SP)      ;;PUT NEW PS ON STACK
2994 007362 012746 007370          MOV      #64$,-(SP)      ;;PUT NEW PC ON STACK
2995 007366 000002          RTI          ;;POP NEW PC AND PS
2996 007370          64$:
2997          ;GET DATA MASK
2998 007370          CALL DATLNG OUT (R1)
2999 007370 162705 000002          SUB      #1*2,R5
3000 007374 004767 001240          JSR      PC,DATLNG
3001 007400 012501          MOV      (R5)+,R1
3002 007402          LET R0 := @RBUF ; START CLEAN
3003 007402 017700 171654          MOV      @RBUF,R0
3004          ;BINARY COUNT PATTERN
3005 007406          INCR R2 FROM #0 TO #377 BY #1

```

E06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 70
 DVDVCA.P11 08-AUG-77 09:16 T22 TEST DATA THROUGH WRAP

SEQ 0069

```

3006 007406 005002          CLR    R2
3007 007410 000401          BR     $160
3008 007412          $161:    INC     R2
3009 007412 005202          $160:    CMP     R2,#377
3010 007414          BGT     $162
3011 007414 020227 000377
3012 007420 003057
3013
3014
3015
3016
3017
3018 007422
3019 007422 010546          MOV     R5,-(SP)
3020 007424 012745 177777    MOV     #SET,-(R5)
3021 007430 016745 171630    MOV     TCSR,-(R5)
3022 007434 012745 000200    MOV     #XMITRDY,-(R5)
3023 007440 012745 000050    MOV     #50,-(R5)
3024 007444 004767 001012    JSR     PC,TIMER
3025 007450 012605          MOV     (SP)+,R5
3026 007452
3027 007452 103005          BCC     $163
3028 007454 104123          ERROR   123
3029 007456
3030 007456 012767 000001 171474    MOV     #1,$TIMES
3031 007464 000440          BR     T$123          ;;EXIT THIS TEST
3032 007466
3033 007466          $163:
3034
3035
3036 007466
3037 007466 110277 171576          MOVB    R2,@TBUF
3038
3039 007472
3040 007472 010546          MOV     R5,-(SP)
3041 007474 012745 177777    MOV     #SET,-(R5)
3042 007500 016745 171554    MOV     RCSR,-(R5)
3043 007504 012745 000200    MOV     #RCVRDOI,-(R5)
3044 007510 012745 000050    MOV     #50,-(R5)
3045 007514 004767 000742    JSR     PC,TIMER
3046 007520 012605          MOV     (SP)+,R5
3047 007522
3048 007522 103005          BCC     $164
3049 007524          ERROR   124
3050 007524 104124
3051
3052 007526
3053 007526 012767 000001 171424    MOV     #1,$TIMES
3054 007534 000414          BR     T$123          ;;EXIT THIS TEST
3055 007536
3056 007536          $164:
3057
3058
3059 007536
3060 007536 017703 171520          MOV     @RBUF,R3
3061

```

;TRANSMIT THE CHAR. IN R2.

; MAKE SURE IT'S READY
 CALL TIMER IN (<#50,#XMITRDY,TCSR,#SET>)

IF.ERROR THEN
 ; TRANSMITTER NEVER BECAME READY
 EXIT TEST

;;EXIT THIS TEST
 ENDIF

;START IT ON ITS WAY
 LET @TBUF:=R2

; NOW WAIT FOR RECIEVER DONE
 CALL TIMER IN (<#50,#RCVRDONE,RCSR,#SET>)

IF.ERROR THEN
 ERRHRO 124
 ; RECIEVER NEVER BECAME READY
 EXIT TEST

;;EXIT THIS TEST
 ENDIF

;RETRIEVE
 LET R3:=@RBUF

F06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 71
 DVDVCA.P11 08-AUG-77 09:16 T22 TEST DATA THROUGH WRAP

SEQ 0070

```

3062
3063 007542
3064 007542 010204      MOV    R2,R4
3065 007544 040104      BIC    R1,R4
3066 007546
3067 007546 040103      BIC    R1,R3
3068
3069
3070 007550
3071 007550 020403      CMP    R4,R3
3072 007552 001401      BEQ    $165
3073
3074 007554
3075 007554 104117      ERROR  117
3076
3077 007556
3078 007556      $165:
3079
3080 007556
3081 007556 000715      BR     $161
3082 007560      $162:
3083
3084
3085
3086 007560
3087 007560 052777 000011 171472  BIS    #11,ARCSR
3088
3089
3090
3091 007566
3092
3093
3094
3095

```

```

;STRIP OFF JUNK ON BOTH
LET R4 := R2 CLR.BY R1

```

```

LET R3 := R3 CLR.BY R1

```

```

;WE HAVE TROUBLE
IF R4 NE R3 THEN

```

```

;DATA COMPARE ERROR
ERRHRD 117,COMP,SBUAS

```

```

; <R2> <R3>
ENDIF

```

```

ENDING ; R2

```

```

; NOW THAT THE TEST IS DONE
; WE WILL TOGGLE READER RUN
; TO TURN OFF THE SPECIAL MODULE.
LET ARCSR := ARCSR SET.BY #11

```

```

ENDTST

```

G06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 72
 DVDVCA.P11 08-AUG-77 09:16 T22 TEST DATA THROUGH WRAP

SEQ 0071

3096
 3097
 3098
 3099
 3100
 3101
 3102
 3103
 3104
 3105
 3106
 3107
 3108
 3109
 3110
 3111
 3112
 3113
 3114
 3115
 3116
 3117
 3118
 3119
 3120
 3121
 3122
 3123
 3124
 3125
 3126
 3127
 3128
 3129
 3130
 3131
 3132
 3133
 3134
 3135
 3136
 3137
 3138
 3139
 3140
 3141
 3142
 3143
 3144
 3145
 3146
 3147
 3148
 3149
 3150
 3151

007566 000004
 007570 012767 000001 171362
 007576 012767 000023 171374
 007604
 007604 032767 040000 171406
 007612 001404
 007614 126727 001446 000001
 007622 001004
 007624
 007624 012767 000001 171326
 007632 000540
 007634
 007634
 007634
 007634 162705 000002
 007640 004767 000774
 007644 012503
 007646 012746 000000
 007652 012746 007660
 007656 000002
 007660
 007660
 007660 016701 171372
 007664
 007664 012721 010040
 007670
 007670 012721 000340
 007674
 007674 012721 007776
 007700
 007700 012711 000340

 TEST 23 FULL DATA TRANSFER WITH INTERRUPTS
 AND MAINTENANCE MODE.

TST23: SCOPE
 MOV #1,STIMES ;DO 1 ITERATION
 MOV #23,STESTN ;SET TEST NUMBER IN APT MAIL BOX

IF #MAINTJUMP NOTSETIN SUSWR ORB APTCON EQ #TRUE

BIT #MAINTJUMP,SUSWR
 BEQ \$166
 CMPB APTCON,#TRUE
 BNE \$167

\$166:

EXIT TEST

MOV #1,STIMES
 BR TST24

;;;EXIT THIS TEST
 ENOIF

\$167:

;GET DATA MASK
 CALL DATLNG OUT <R3>

SUB #1*2,R5
 JSR PC,DATLNG
 MOV (R5)+,R3

;SINCE THIS WILL BE A FULL SPEED INTERRUPT TEST
 ;IF A DATA COMPARE ERROR OCCURS THE REPORT
 ;WILL BE DEFERRED UNTIL ALL DATA PATTERNS HAVE
 ;BEEN TRANSFERED. IF THE NUMBER OF CHAR. TRANSMITTED
 ;DOES NOT EQUAL THE NUMBER RECEIVED THIS ALSO WILL
 ;BE REPORTED.

;CHANGE PRIORITY
 ;...TO 0

MOV #PRO,-(SP) ;PUT NEW PS ON STACK
 MOV #64\$,-(SP) ;PUT NEW PC ON STACK
 RTI ;POP NEW PC AND PS

64\$:

;GET VECTOR ADDRESS
 LET R1 := DLVEC

;RCVR VECTOR
 LET (R1)+ := #REC

LET (R1)+ := #PR7

MOV #PR7,(R1)+

;POINT TO TRANSMITTER VECTOR
 AND SET IT UP ALSO

MOV #TRAN,(R1)+

LET (R1) := #PR7

MOV #PR7,(R1)

H06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 73
 DVDVCA.P11 08-AUG-77 09:16 T23 FULL DATA TRANSFER WITH INTERRUPTS

SEQ 0072

```

3152          ; CLEAR ERROR COUNTER
3153          LET ERRCNT := #0
3154 007704 005067 000060      CLR      ERRCNT
3155          ; START COUNT AT 0
3156          LET R1 := #0
3157 007710 005001      CLR      R1
3158          ; RECEIVER STORAGE
3159          LET R2 := #0
3160 007712 005002      CLR      R2
3161          ; # OF RECEIVED CHAR. COUNT.
3162          LET R4 := #0
3163 007714 005004      CLR      R4
3164
3165          BRESET ; SET UP ALL REGISTERS
3166 007716 000005      RESET
3167          ; SET UP MAINTENANCE
3168          LET @TCSR := @TCSR SET.BY #MAINT
3169 007720 052777 000004 171336      BIS      #MAINT,@TCSR
3170
3171          ; SET I.E. IN TRANSMITTER
3172          LET @TCSR := @TCSR SET.BY #XMITIE
3173 007726 052777 000100 171330      BIS      #XMITIE,@TCSR
3174          ; AND RECEIVER
3175          LET @RCR := @RCR SET.BY #RCVRIE
3176 007734 052777 000100 171316      BIS      #RCVRIE,@RCR
3177
3178          ; CLEAR OUR DONE AND ERROR FLAG WORD
3179          ; NOW WE WAIT UNTIL R4 COUNT (RECEIVED) IS EQUAL
3180          REPEAT
3181          UNTIL R4 HIS NUMBER
3182          $170:
3183          007742
3184 007742 020467 000024      CMP      R4,NUMBER
3185 007746 103775      BLO      $170
3186
3187          LET @TCSR := @TCSR CLR.BY #MAINT
3188 007750 042777 000004 171306      BIC      #MAINT,@TCSR
3189
3190          ; CHECK FOR DATA COMPARE ERRORS.
3191          IF ERRCNT NE #0 THEN
3192          007756 005767 000006      TST      ERRCNT
3193          007762 001401      BEQ      $171
3194
3195          ; DATA COMPARE ERROR
3196          ERRHAD 120,COMP,FIRST
3197          ; <R3> OCCURED, FIRST: SB <S3>, WAS <WAS>
3198          $171:
3199          007766
3200          007766 000462      BR      TST24
3201          ;;;EXIT IS TEST
3202          EXIT ; SKIP OVER SUPPORT ROUTINES & STORAGE
3203          ERRCNT: 0
3204          NUMBER: 1000
3205          SB: .BYTE 0
3206          WAS: .BYTE 0
3207

```

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 74
 DVDVCA.P11 08-AUG-77 09:16 T23 FULL DATA TRANSFER WITH INTERRUPTS

SEQ 0073

```

3208
3209
3210
3211 007776
3212 007776
3213
3214
3215 007776
3216 007776 010167 000032
3217 010002 040367 000026
3218
3219 010006
3220 010006 016777 000022 171254
3221
3222 010014
3223 010014 005201
3224
3225 010016
3226 010016 020167 177750
3227 010022 001003
3228
3229 010024
3230 010024 042777 000100 171232
3231 010032
3232 010032
3233
3234 010032 000401
3235
3236 010034 000000
3237
3238 010036
3239 010036 000002
3240
3241
3242
3243
3244 010040
3245 010040
3246
3247
3248 010040
3249 010040 017702 171216
3250 010044 040302
3251
3252 010046
3253 010046 010467 000056
3254 010052 040367 000052
3255
3256
3257 010056
3258 010056 020267 000046
3259 010062 001412
3260
3261 010064
3262 010064 005767 177700
3263 010070 001005

```

TRAN:

BGNSRV : TRANSMIT INTERRUPT HANDLER
 TRAN

: SET UP FOR TRANSFER
 LET HOLD := R1 CLR.BY R3

: AND SEND.
 LET @TBUF := HOLD

: INCREMENT CHAR COUNT
 LET R1 := R1 + #1

: ALL DONE
 IF R1 EQ NUMBER THEN

: STOP INTERRUPT PROCESSING
 LET @TCSR := @TCSR CLR.BY @XMITIE

ENDIF

\$172:

BR ZZZ

HOLD: 0

ZZZ:

RTI

ENDSRV

BGNSRV : RECEIVER INTERRUPT HANDLER
 REC

: GET CHAR IN + MASK IT
 LET R2 := @RBUF CLR.BY R3

: RHL D WILL CONTAIN EXPECTED INPUT
 LET RHL D := R4 CLR.BY R3

: DO THEY COMPARE
 IF R2 NE RHL D THEN

: FIRST ERROR
 IF ERRCNT EQ #0 THEN

TST ERRCNT
 BNE \$174

J06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 75
 DVDVCA.P11 08-AUG-77 09:16 T23 FULL DATA TRANSFER WITH INTERRUPTS

SEQ 0074

```

3264                                     ;SAVE RECORD OF FIRST MISS
3265 010072                             LET SB :B= RHL D
3266 010072 116767 000032 177674      MOV B  RHL D, SB
3267 010100                             LET WAS :B= R2
3268 010100 110267 177671      MOV B  R2, WAS
3269 010104
3270 010104      $174:
3271
3272 010104                             ;COUNT IT.
3273 010104 005267 177660      INC  ERR CNT
3274 010110
3275 010110      $173:
3276
3277
3278 010110                             ;COUNT THIS CHAR.
3279 010110 005204      INC  R4
3280
3281 010112                             ;ALL DONE?
3282 010112 020467 177654      CMP  R4, NUMBER
3283 010116 001003      BNE  $175
3284
3285 010120                             ;STOP RECEIVER INTERRUPTS
3286 010120 042777 000100 171132      BIC  #RCVRIE, #RCSR
3287
3288
3289 010126                             ;INDICATE ALL DONE TO TIMER
3290 010126      $175:      ;MAIN REPEAT LOOP IS CHECKING
3291
3292
3293 010126 000401      BR  ZZZZ
3294
3295 010130 000000
3296 010132      ZZZZ:
3297 010132
3298 010132 000002      RTI
3299
3300 010134
3301
3302
3303
  
```

ENDTST

K06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 76
 DVDVCA.P11 08-AUG-77 09:16 T23 FULL DATA TRANSFER WITH INTERRUPTS

SEQ 0075

```

3304 ;*****
3305 ;*****
3306 ;TEST 24 TEST BREAK GENERATION LOGIC
3307 ; TRANSMIT KNOWN CHAR WITH BREAK SET
3308 ; AND COMPARE RECEIVED WITH 0.
3309 ; FRAMING ERROR WILL ALSO BE CHECKED
3310 ; IF ERROR BITS ARE ENABLED.
3311 ;*****
3312 010134 000004 TST24: SCOPE
3313 010136 012767 000010 171014 MOV #10,$TIMES ;;DO 10 ITERATIONS
3314 010144 012767 000024 171026 MOV #24,$TESTN ;;SET TEST NUMBER IN APT MAIL BOX
3315 010152 IF #MAINTJUMP NOTSETIN $USWR OR #BRK NOTSETIN $U
3316 010152 032767 040000 171040 BIT #MAINTJUMP,$USWR
3317 010160 001404 BEQ $176
3318 010162 032767 010000 171030 BIT #BRK,$USWR
3319 010170 001004 BNE $177
3320 010172 $176:
3321 010172 EXIT TEST
3322 010172 012767 000001 170760 MOV #1,$TIMES
3323 010200 000456 BR TST25 ;;EXIT THIS TEST
3324 010202 ENDF
3325 010202 $177:
3326 010202 LET ERRCHK := #0 ; CLEAR ERROR WORD
3327 010202 005067 000126 CLR ERRCHK
3328 010202 ; SET MAINTENANCE BIT
3329 010206 LET @TCSR := @TCSR SET.BY #MAINT
3330 010206 052777 000004 171050 BIS #MAINT,@TCSR
3331 010206 ; SET BREAK BIT
3332 010214 LET @TCSR := @TCSR SET.BY #BREAK
3333 010214 052777 000001 171042 BIS #BREAK,@TCSR
3334 010222 ; NON-ZERO CHAR. '*'
3335 010222 LET @TBUF := #252
3336 010222 012777 000252 171040 MOV #252,@TBUF
3337 010230 REPEAT ;WAIT FOR DONE
3338 010230 $200:
3339 010230 ; WAIT FOR DONE
3340 010230 UNTIL #RCVRDONE SETIN @RCR
3341 010230 032777 000200 171022 BIT #RCVRDONE,@RCR
3342 010230 BEQ $200
3343 010236 001774
3344 010240 IFB @RBUF NE #0 THEN
3345 010240 105777 171016 TSTB @RBUF
3346 010244 001404 BEQ $201
3347 010246 ; BREAK DID NOT EQUAL 0
3348 010246 052767 000001 000060 BIS #BIT0,ERRCHK LET ERRCHK := ERRCHK SET.BY #BIT0
3349 010254 ELSE
3350 010254 000413 BR $202
3351 010256 $201:
3352 010256 IF #ERRBITS SETIN $USWR THEN
3353 010256 032767 100000 170734 BIT #ERRBITS,$USWR
3354 010264 BEQ $203
3355 010266 IF #FRERR NOTSETIN @RBUF THEN
3356 010266 032777 020000 170766 BIT #FRERR,@RBUF

```

L06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 77
 DVDVCA.P11 08-AUG-77 09:16 T24 TEST BREAK GENERATION LOGIC

SEQ 0076

```

3360 010274 001003      BNE      $204
3361 010276
3362 010276 052767 000002 000030      BIS      #BIT1,ERRCHK
3363 010304
3364 010304      $204:
3365 010304
3366 010304      $203:
3367 010304
3368 010304      $202:
3369
3370 010304
3371 010304 000005      RESET
3372
3373 010306
3374 010306 032767 000001 000020      BIT      #BIT0,ERRCHK
3375 010314 001401      BEQ      $205
3376 010316
3377 010316 104121      ERROR    121
3378 010320
3379 010320      $205:
3380 010320
3381 010320 032767 000002 000006      BIT      #BIT1,ERRCHK
3382 010326 001401      BEQ      $206
3383 010330
3384 010330 104122      ERROR    122
3385 010332
3386 010332      $206:
3387 010332
3388 010332 000401
3389 010334 000000      ERRCHK: BR      TST25
3390 010336
3391

```

LET ERRCHK := ERRCHK SET.BY #BIT1
 ENDIF
 ENDIF
 ENDIF
 BRESET ;CLEAN UP
 IF #BIT0 SETIN ERRCHK THEN
 ERRHRD 121 ;BREAK ERROR
 ENDIF
 IF #BIT1 SETIN ERRCHK THEN
 ERRHRD 122 ; FRAMING ERROR
 ENDIF
 EXIT
 ;;;EXIT THIS TEST
 ENDTST

MOS

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 78
 DVDVCA.P11 08-AUG-77 09:16 T24 TEST BREAK GENERATION LOGIC

SEQ 0077

3392					*****
3393					TEST 25 NOT A TEST - SEND BACK TO LOOP
3394					*****
3395					*****
3396	010336	000004			1ST25: SCOPE
3397	010340	012767	000001	170612	MOV #1,STIMES ;DO 1 ITERATION
3398	010346	104401	010354		TYPE 65\$;TYPE ASCIZ STRING
3399	010352	000404			BR 64\$;GET OVER THE ASCIZ
3400					;;65\$: .ASCIZ <CRLF>*CSR: *
3401	010364				64\$: MOV DLADD,-(SP) ;SAVE DLADD FOR TYPEOUT
3402	010364	016746	170664		TYPOC ;GO TYPE--OCTAL ASCII(ALL DIGITS)
3403	010370	104402			TYPE 67\$;TYPE ASCIZ STRING
3404	010372	104401	010400		BR 66\$;GET OVER THE ASCIZ
3405	010376	000405			;;67\$: .ASCIZ *,VECTOR: *
3406					66\$: MOV DLVEC,-(SP) ;SAVE DLVEC FOR TYPEOUT
3407	010412				TYPOC ;GO TYPE--OCTAL ASCII(ALL DIGITS)
3408	010412	016746	170640		TYPE 69\$;TYPE ASCIZ STRING
3409	010416	104402			BR 68\$;GET OVER THE ASCIZ
3410	010420	104401	010426		;;69\$: .ASCIZ *,ERRORS: *
3411	010424	000405			68\$: MOV \$ERTTL,-(SP) ;SAVE \$ERTTL FOR TYPEOUT
3412					TYPOC ;GO TYPE--DECIMAL ASCII WITH SIGN
3413	010440				CLR \$ERTTL ;RESET FOR NEXT DEVICE/PASS
3414	010440	016746	170446		TYPE \$CRLF ; BACK UP TO THE BEGINNING
3415	010444	104405			
3416	010446	005067	170440		
3417	010452	104401	001171		
3418	010456	000167	171454		

N06

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 79
 DVDVCA.P11 08-AUG-77 09:16 T25 NOT A TEST - SEND BACK TO LOOP

SEQ 0078

3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3470
3471
3472
3473
3474

010462
010462

000001
000000

010462
010462
010470
010470
010476
010476

010504
010504

010504
010504
010512
010514
010514
010522
010522
010524
010524
010532
010532

010532

016567 000004 000136
016567 000000 000132
112767 000000 000126

036577 000002 000114
112767 000000 000111
000403
112767 177777 000101

```

;: BGNMOD SUBS
;: *****
ROUTINE TIMER <HOWLONG,WHICHBIT,REG,SETCLR>
TIMER:
* ROUTINE:TIMER
* THIS ROUTINE IS USED TO TEST THE STATUS OF ANY BIT
* IN ANY REGISTER.
* INPUTS:
* HOWLONG THE MAXIMUM AMOUNT OF TIME TO SPEND IN
* THIS ROUTINE.
* WHICHBIT A MASK WITH THE BIT(S) SET THAT ARE
* TO BE CHECKED.
* REG A POINTER TO THE REGISTER TO BE CHECKED
* SETCLR THE DESIRED RESULTS
* EITHER #SET OR #CLR
* OUTPUT:
* THE 'C' BIT IS SET TO INDICATE AN ERROR
* BUT IT IS TESTED BY THE IF.ERROR STATEMENT
*
* NOTE:: THE USE OF (R5) IS PART OF THE LINKAGE
* MECHANISM BETWEEN THE CALLER AND THE CALLED
;: *****

```

```

TRUE= 1
FALSE= 0

LET REGSAV := REG(R5) ; GET POINTER TO REGIST
LET TIMSAV := HOWLONG(R5) ; SAVE HOWLONG FOR
LET FLAG :B= #FALSE ; INITIALIZE THE EXIT FLA

; START OF AN INFINITE LOOP
LOOP
$211:
; TEST TO SEE IF WHICHBIT IS SET
IF WHICHBIT(R5) NOTSETIN @REGSAV THEN
LET HOLDSC :B= #CLR
ELSE
LET HOLDSC :B= #SET ; REMEMBER THIS
ENDIF
; NOW SEE IF THAT WAS WHAT WE WANTED
IFB HOLDSC EQ SETCLR(R5) THEN

```

\$211:

\$213:

\$214:

```

3475 010532 126765 000075 000006      CMPB   HOLDSC,SETCLR(R5)
3476 010540 001003                      BNE     $215
3477                                     ; JUST THE THING WE NEEDED
3478 010542 112767 000001 000062      MOVB   #TRUE,FLAG
3479 010550                      ENDIF
3480 010550                      $215:
3481 010550                      EXIFB   FLAG EQ #TRUE OR TIMSAV LE #0
3482 010550
3483 010550 126727 000056 000001      CMPB   FLAG,#TRUE
3484 010556 001414                      BEQ     $212
3485 010560 005767 000044                      TST     TIMSAV
3486 010564 003411                      BLE     $212
3487
3488                                     ; ONE WAY OR THE OTHER, WE ARE DONE
3489                                     ; IF WE ARE STILL HERE THEN HANG AROUND A WHILE
3490
3491 010566                                     WAITUS 10          ;WAIT FOR 10 MILLI-SECONDS
3492 010566 010546                      MOV     R5, -(SP)
3493 010570 012745 000010                      MOV     #10, -(R5)
3494 010574 004767 000140                      JSR     PC, WAIT
3495 010600 012605                      MOV     (SP)+, R5
3496 010602
3497 010602 005367 000022                      DEC     TIMSAV
3498 010606                                     LET     TIMSAV := TIMSAV - #1 ; COUNTING DOWN
3499 010606 000736                                     ENDLOOP          ; CONTINUED AT THE TOP
3500 010610                      $212:
3501
3502                                     ; ONLY 2 WAYS TO GET HERE
3503                                     ; 1). WE RAN OUT OF TIME---ERROR !!
3504                                     ; 2). THE BIT IS IN THE CORRECT CONDITION--GOOD !!
3505
3506 010610 126727 000016 000001      CMPB   FLAG,#TRUE
3507 010616 001001                      BNE     $216
3508 010620 000405                      BR      $207
3509
3510                                     IFB     FLAG EQ #TRUE THEN
3511                                     RETURN  NO.ERROR      ; GOOD
3512                                     ENDIF
3513                                     RETURN  ERROR        ; BAD
3514 010622 000261                      SEC
3515 010624 000404                      BR      $210
3516
3517 010626 000000                      REGSAV: .WORD 0
3518 010630 000000                      TIMSAV: .WORD 0
3519 010632 000                      FLAG:   .BYTE 0
3520 010633 000                      HOLDSC: .BYTE 0
3521                                     ; WE ARE DONE GO BACK HOME
3522 010634                                     ENDRTN
3523 010634                      $207:
3524 010634 000241                      CLC
3525 010636                      $210:
3526 010636 000207                      RTS     PC

```

C07

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 81
 DVDVCA.P11 08-AUG-77 09:16 ROUTINE - TIMER

SEQ 0080

```

3527
3528
3529 010640
3530 010640
3531
3532
3533
3534
3535
3536
3537
3538
3539
3540
3541
3542
3543
3544 010640
3545 010640 005065 000000
3546 010644
3547 010644 016767 170350 000062
3548 010652 016746 000056
3549 010656 042716 000017
3550 010662 042667 000046
3551
3552 010666
3553 010666 012767 000001 170376
3554 010674 000402
3555 010676
3556 010676 005267 170370
3557 010702
3558 010702 026767 170364 000024
3559 010710 003006
3560 010712
3561 010712 006365 000000
3562 010716
3563 010716 052765 000001 000000
3564 010724
3565 010724 000764
3566 010726
3567 010726
3568 010726 005165 000000
3569 010732
3570 010732 000401
3571 010734 000000
3572 010736
3573 010736
3574 010736
3575 010736 000207

;*****
ROUTINE DATLNG <MASK>
DATLNG:
; ROUTINE:DATLNG
; THIS ROUTINE SETS UP A MASK FOR DATA, WITH
; INPUT - NOTHING IS PASSED TO THIS ROUTINE
; BUT GLOBAL INFORMATION IS ASSUMED TO EXIST:
; SUSWR-- THE WORD FOR SOFTWARE PARAMETERS
; DATA-- A MASK FOR THE LOCATION OF THE OCTAL
; NUMBER OF DATA BITS
; OUTPUT----
; MASK-- A MASK OF BINARY ONES RIGHT-JUSTIFIED
; THE NUMBER OF WHICH IS DEFINED IN SUSWR WORD.
;*****

;*****
LET MASK(R5) := #0 ; START
LET NUMBR := SUSWR AND #DATA
MOV SUSWR, NUMBR
MOV NUMBR, -(SP)
BIC #DATA, (SP)
BIC (SP)+, NUMBR

INCR I FROM #1 TO NUMBR BY #1

$222: INC I
$221: CMP I, NUMBR
BGT $223
ASL MASK(R5)
LET MASK(R5) := MASK(R5) SHIFT 1
LET MASK(R5) := MASK(R5) SET.BY #1
ENDING
BR $222
$223: COM MASK(R5)
RETURN
BR $217
NUMBR:0
$217:
$220: RTS PC
ENDRTN

```

007

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 82
 DVDVCA.P11 08-AUG-77 09:16 ROUTINE - DATLNG

SEQ 0081

3576				*****
3577				ROUTINE WAIT (TIME)
3578	010740			WAIT:
3579	010740			ROUTINE WAIT
3580				THIS ROUTINE IS USED TO DELAY EXECUTION OF THE
3581				MAIN PROGRAM FOR A SPECIFIED AMOUNT OF TIME.
3582				THIS IS ACCOMPLISHED BY INCREMENTING A
3583				REGISTER UP TO A LIMIT. THE INNER LOOP IS SET
3584				TO APPROXIMATE 1 MICRO SEC.
3585				*****
3586				MOV R1, -(SP) ; PUSH R1 ON STACK
3587	010740	010146		MOV R2, -(SP) ; PUSH R2 ON STACK
3588	010742	010246		MOV R3, -(SP) ; PUSH R3 ON STACK
3589	010744	010346		LET R1 := TIME(R5)
3590	010746			MOV TIME(R5), R1
3591	010746	016501	000000	INCRU R2 FROM #1 TO R1 BY #1
3592	010752			MOV #1, R2
3593	010752	012702	000001	BR \$225
3594	010756	000402		\$227: ADD #01, R2
3595	010760			\$226: CMP R2, R1
3596	010760	062702	000001	BHI \$230
3597	010764			INCR R3 FROM #0 TO #700 BY #1
3598	010764	020201		CLR R3
3599	010766	101010		BR \$231
3600	010770			\$232: INC R3
3601	010770	005003		\$231: CMP R3, #700
3602	010772	000401		BGT \$233
3603	010774			ENDINC
3604	010774	005203		BR \$232
3605	010776			\$233: ENDINC
3606	010776	020327	000700	BR \$227
3607	011002	003001		\$230: MOV (SP)+, R3 ; POP STACK INTO R3
3608	011004			MOV (SP)+, R2 ; POP STACK INTO R2
3609	011004	000773		MOV (SP)+, R1 ; POP STACK INTO R1
3610	011006			ENDRTN
3611	011006			\$224:
3612	011006	000764		\$225:
3613	011010			RTS PC
3614	011010	012603		
3615	011012	012602		
3616	011014	012601		
3617	011016			
3618	011016			
3619	011016			
3620	011016	000207		

E07

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 83
 DVDVCA.P11 08-AUG-77 09:16 ROUTINE - WAIT

SEQ 0082

3621						
3622						.SBTTL INTSRV INTERRUPT SERVICE ROUTINE
3623						*****
3624	011020					INTSRV:
3625						* SERVICE ROUTINE: INTSRV
3626						* THIS GLOBAL ROUTINE DOES NOTHING BUT INCREMENT
3627						* 'INTFLAG' EACH TIME IT IS CALLED. IT ASSUMES
3628						* THAT THE MAIN CALLING ROUTINE WILL KNOW WHAT
3629						* TO LOOK FOR.
3630						*****
3631						
3632						;ADD 1 TO 'INTERRUPT OCCURED' FLAG
3633	011020					LET INTFLAG := INTFLAG + #1
3634	011020	005267	000002			INC INTFLAG
3635	011024					ENDSRV ;THAT'S ALL
3636	011024	000002				RTI
3637	011026	000000				INTFLAG: 0

F07

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 84
 DVDVCA.P11 08-AUG-77 09:16 INTSRV INTERRUPT SERVICE ROUTINE

SEQ 0083

3638 011030
 3639 011030
 3640
 3641
 3642
 3643
 3644
 3645
 3646
 3647
 3648
 3649

ROUTINE CYCLE

CYCLE:

```
*****
ROUTINE:  CYCLE
THIS ROUTINE CAUSES ADRS TO POINT TO THE
ADDRESS OF DLV11-F UNDER TEST, ADRS +2 TO
POINT TO THE VECTOR OF THE DLV11-F UNDER TEST.
IT KEEPS TRACK OF THE CURRENT DEVICE AND BIT
MASKS. THE CONSOLE IS TREATED SPECIAL BY THIS ROUTINE.
IT IS ONLY TESTED ONCE IF UNDER APT. IF NOT UNDER APT
ALL TESTS THAT REQUIRE THE MAINT BIT ARE NOT RUN.
*****
```

3650 011030
 3651 011030 112767 000000 000230
 3652 011036
 3653 011036 112767 000000 000223
 3654 011044
 3655 011044
 3656 011044
 3657 011044 005767 000200
 3658 011050 001027
 3659 011050
 3660 011050 026727 000174 000001
 3661 011060 701003
 3662 011060
 3663 011060 005067 000164
 3664 011066
 3665 011066 000403
 3666 011070
 3667 011070
 3668 011070 004767 000370
 3669
 3670 011074
 3671 011074
 3672 011074 012600
 3673 011076
 3674 011076
 3675 011076
 3676 011076 012767 000001 000144
 3677 011104
 3678 011104 012767 000001 170072
 3679 011112
 3680 011112 016767 170132 000134
 3681 011120
 3682 011120 016767 170120 000130
 3683 011126
 3684 011126 000410
 3685 011130
 3686 011130
 3687 011130 012704 000010
 3688 011134
 3689 011134 006167 000110
 3690 011140
 3691 011140 060467 000110
 3692 011144
 3693 011144 060467 000106

```
*****
; LET APTCON :B= #FALSE ; SET DEFAULT VALUE
; LET CONSOLE :B= #FALSE
REPEAT ; UNTIL BITMASK SETIN $DEVH
IF BITMASK EQ #0 THEN
  TST BITMASK
  BNE $237
  IF INITFLAG EQ #1 THEN
    CMP INITFLAG, #1
    BNE $240
    LET INITFLAG := #0
  ELSE
    BR $241
$240:
  CALL $EOP ; AS A SUBROUTINE
  ; BECAUSE $EOP RETURNS AS A JUMP
  LET RO := POP
ENDIF
LET BITMASK := #1
LET $DEVCT := #1
LET ADDRESS := $BASE
LEI VECTOR := $VECT1
ELSE
  BR $242
$237:
  LET R4 := #10
  MOV #10, R4
  ROL BITMASK
  LET BITMASK := BITMASK ROTATE 1
  LET ADDRESS := ADDRESS + R4
  LET VECTOR := VECTOR + R4
  ADD R4, VECTOR
```

G07

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 85
 DVDVCA.P11 08-AUG-77 09:16 ROUTINE - CYCLE

SEQ 0084

```

3694 011150
3695 011150
3696 011150
3697 011150 036767 000074 170074 BIT BITMASK,$DEVN
3698 011156 001732 BEQ $236
3699 011160 IF BITMASK EQ #BIT15 THEN
3700 011160 026727 000064 100000 CMP BITMASK,#BIT15
3701 011166 001023 BNE $243
3702 011170 LET CONSOLE :B= #TRUE
3703 011170 112767 000001 000071 MOVB #TRUE,CONSOLE
3704 011176 LET ADDRESS := CONADR
3705 011176 016767 000060 000050 MOV CONADR,ADDRESS
3706 011204 LET VECTOR := CONVECT
3707 011204 016767 000054 000044 MOV CONVECT,VECTOR
3708
3709
3710
3711 011212
3712 011212 032767 000001 167774 BIT #APTENV,$ENV
3713 011220 001406 BEQ $244
3714 011222
3715 011222 005767 167754 TST $PASS
3716 011226 001403 BEQ $245
3717
3718 011230
3719 011230 112767 000001 000030 MOVB #TRUE,APTCON
3720 011236
3721 011236
3722 011236
3723 011236
3724 011236
3725 011236
3726
3727 011236 LET ADRS := #ADDRESS
3728 011236 012701 011254 MOV #ADDRESS,ADRS
3729 011242 LET $DEVCT := $DEVCT + #1
3730 011242 005267 167736 INC $DEVCT
3731 011246 RETURN
3732 011246 000411 BR $234
3733 011250 100000 BITMASK: 100000 ; CONSOLE FIRST
3734 011252 000001 INITFLAG: 1
3735 011254 000000 ADDRESS: 0
3736 011256 000000 VECTOR: 0
3737 011260 000000 OK: 0
3738 011262 177560 CONADR: 177560 ; CONSOLE ADDRESS
3739 011264 000060 CONVECT: 60 ; CONSOLE VECTOR
3740 011266 000 APTCON: .BYTE 0
3741 011267 000 CONSOLE: .BYTE 0
3742 011270 000 NOCONMANT: .BYTE 0
3743 011272 .EVEN
3744
3745
3746
3747
3748
3749

```

\$234:
 \$235:

RTS PC

ENDRTN

H07

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 86
DVDVCA.P11 08-AUG-77 09:16 ROUTINE - CYCLE

SEQ 0085

3750

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 87
 DVDVCA.P11 08-AUG-77 09:16 ROUTINE - CYCLE

SEQ 0086

```

3751
3752 011274
3753 011274
3754
3755 011274 104401 011302
3756 011300 000405
3757
3758 011314
3759 011314 016746 167660
3760 011320 104402
3761 011322 104401 011330
3762 011326 000405
3763
3764 011342
3765 011342 116767 167546 167626
3766 011350 016746 167622
3767 011354 104403
3768 011356 006
3769 011357 000
3770 011360 104401 011366
3771 011364 000404
3772
3773 011376
3774 011376 016746 167514
3775 011402 104402
3776 011404 104401 011412
3777 011410 000404
3778
3779 011422
3780 011422 016746 167626
3781 011426 104402
3782 011430 104401 011436
3783 011434 000405
3784
3785 011450
3786 011450 016746 167602
3787 011454 104402
3788 011456 104401 001171
3789 011462
3790 011462
3791 011462
3792 011462 000207

ROUTINE MYTYPE
MYTYPE:
;;*****
TYPE 65$ ;;TYPE ASCIZ STRING
BR 64$ ;;GET OVER THE ASCIZ
;;65$: .ASCIZ <CRLF>*TEST * *
64$:
MOV $TESTN,-(SP) ;;SAVE $TESTN FOR TYPEOUT
TYPOC ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
TYPE 67$ ;;TYPE ASCIZ STRING
BR 66$ ;;GET OVER THE ASCIZ
;;67$: .ASCIZ *,ERROR * *
66$:
MOVB $ITEMB,$FATAL ;;APT FATAL ERROR NUMBER
MOV $FATAL,-(SP) ;;SAVE $FATAL FOR TYPEOUT
TYPOS ;;GO TYPE--OCTAL ASCII
.BYTE 6 ;;TYPE 6 DIGITS
.BYTE 0 ;;SUPPRESS LEADING ZEROS
TYPE 69$ ;;TYPE ASCIZ STRING
BR 68$ ;;GET OVER THE ASCIZ
;;69$: .ASCIZ *,PC = *
68$:
MOV $ERRPC,-(SP) ;;SAVE $ERRPC FOR TYPEOUT
TYPOC ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
TYPE 71$ ;;TYPE ASCIZ STRING
BR 70$ ;;GET OVER THE ASCIZ
;;71$: .ASCIZ *,CSR: *
70$:
MOV DLADD,-(SP) ;;SAVE DLADD FOR TYPEOUT
TYPOC ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
TYPE 73$ ;;TYPE ASCIZ STRING
BR 72$ ;;GET OVER THE ASCIZ
;;73$: .ASCIZ *,VECTOR: *
72$:
MOV DLVEC,-(SP) ;;SAVE DLVEC FOR TYPEOUT
TYPOC ;;GO TYPE--OCTAL ASCII(ALL DIGITS)
TYPE ,SCRLF
ENORTN
$246:
$247:
RTS PC

```

J07

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 88
 DVDVCA.P11 08-AUG-77 09:16 END OF PASS ROUTINE

SEQ 0087

```

3793 .SBTTL END OF PASS ROUTINE
3794
3795 ;*****
3796 ;*INCREMENT THE PASS NUMBER ($PASS)
3797 ;*INDICATE END-OF-PROGRAM AFTER 1 PASSES THRU THE PROGRAM
3798 ;*TYPE "END PASS #XXXXX" (WHERE XXXXX IS A DECIMAL NUMBER)
3799 ;*IF THERES A MONITOR GO TO IT
3800 ;*IF THERE ISN'T JUMP TO SPECIALADDRESS
3801
3802 $EOP:
3803   SCOPE
3804   CLR $TSTNM ;:ZERO THE TEST NUMBER
3805   CLR $TIMES ;:ZERO THE NUMBER OF ITERATIONS
3806   INC $PASS ;:INCREMENT THE PASS NUMBER
3807   BIC #100000,$PASS ;:DON'T ALLOW A NEG. NUMBER
3808   DEC (PC)+ ;:LOOP?
3809   $EOPCT: .WORD 1
3810   BGT $DOAGN ;:YES
3811   MOV (PC)+,2(PC)+ ;:RESTORE COUNTER
3812   $ENDCT: .WORD 1
3813   $EOPCT
3814   TYPE $ENDMG ;:TYPE "END PASS #"
3815   MOV $PASS,-(SP) ;:SAVE $PASS FOR TYPEOUT
3816   TYPDS ;:GO TYPE--DECIMAL ASCII WITH SIGN
3817   TYPE $ENULL ;:TYPE A NULL CHARACTER
3818   $GET42: MOV 2#42,R0 ;:GET MONITOR ADDRESS
3819   BEQ $DOAGN ;:BRANCH IF NO MONITOR
3820   RESET ;:CLEAR THE WORLD
3821   $ENDAD: JSR PC,(R0) ;:GO TO MONITOR
3822   NOP ;:SAVE ROOM
3823   NOP ;:FOR
3824   NOP ;:ACT11
3825   $DOAGN:
3826   JMP 2(PC)+ ;:RETURN
3827   $RTNAD: .WORD SPECIALADDRESS
3828   $ENULL: .BYTE -1,-1,0 ;:NULL CHARACTER STRING
3829   $ENDMG: .ASCIIZ <15><12>/END PASS #/
3830
3831

```

011464	000004			
011466	005067	167410		
011472	005067	167462		
011476	005267	167500		
011502	042767	100000	167472	
011510	005327			
011512	000001			
011514	003022			
011516	012737			
011520	000001			
011522	011512			
011524	104401	011571		
011530	016746	167446		
011534	104405			
011536	104401	011566		
011542	013700	000042		
011546	001405			
011550	000005			
011552	004710			
011554	000240			
011556	000240			
011560	000240			
011562				
011562	000137			
011564	011074			
011566	377	377	000	
011571	015	042412	042116	
011576	050040	051501	020123	
011604	000043			

K07

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 89
 DVDVCA.P11 08-AUG-77 09:16 POWER DOWN AND UP ROUTINES

SEQ 0068

```

3832          .SBTTL  POWER DOWN AND UP ROUTINES
3833
3834          ;*****
3835          : POWER DOWN ROUTINE
3836          $PWRDN: MOV    $SILLUP, 2#$PWRVEC ; SET FOR FAST UP
3837                   MOV    #340, 2#$PWRVEC+2 ; PRI0:7
3838                   MOV    R0, -(SP) ; PUSH R0 ON STACK
3839                   MOV    R1, -(SP) ; PUSH R1 ON STACK
3840                   MOV    R2, -(SP) ; PUSH R2 ON STACK
3841                   MOV    R3, -(SP) ; PUSH R3 ON STACK
3842                   MOV    R4, -(SP) ; PUSH R4 ON STACK
3843                   MOV    R5, -(SP) ; PUSH R5 ON STACK
3844                   MOV    2$SWR, -(SP) ; PUSH 2$SWR ON STACK
3845                   MOV    SP, $SAVR6 ; SAVE SP
3846                   MOV    $SPWRUP, 2#$PWRVEC ; SET UP VECTOR
3847                   HALT
3848                   BR      .-2 ; HANG UP
3849
3850          ;*****
3851          : POWER UP ROUTINE
3852          $PWRUP: MOV    $SILLUP, 2#$PWRVEC ; SET FOR FAST DOWN
3853                   MOV    $SAVR6, SP ; GET SP
3854                   CLR    $SAVR6 ; WAIT LOOP FOR THE TTY
3855                   1$: INC    $SAVR6 ; WAIT FOR THE INC
3856                   BNE    1$ ; OF WORD
3857                   MOV    (SP)+, 2$SWR ; POP STACK INTO 2$SWR
3858                   MOV    (SP)+, R5 ; POP STACK INTO R5
3859                   MOV    (SP)+, R4 ; POP STACK INTO R4
3860                   MOV    (SP)+, R3 ; POP STACK INTO R3
3861                   MOV    (SP)+, R2 ; POP STACK INTO R2
3862                   MOV    (SP)+, R1 ; POP STACK INTO R1
3863                   MOV    (SP)+, R0 ; POP STACK INTO R0
3864                   MOV    $SPWRDN, 2#$PWRVEC ; SET UP THE POWER DOWN VECTOR
3865                   MOV    #340, 2#$PWRVEC+2 ; PRI0:7
3866                   TYPE    $PWRMG: .WORD    $POWER ; REPORT THE POWER FAILURE
3867                   MOV    (PC)+, (SP) ; POWER FAIL MESSAGE POINTER
3868                   $PWRAD: .WORD    LOOP ; RESTART AT LOOP
3869                   RTI ; RESTART ADDRESS
3870                   $SILLUP: HALT ; THE POWER UP SEQUENCE WAS STARTED
3871                   BR      .-2 ; BEFORE THE POWER DOWN WAS COMPLETE
3872                   $SAVR6: 0 ; PUT THE SP HERE
3873                   $POWER: .ASCIZ  <15><12>"POWER"
3874                   .EVEN
3875
3876

```

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 90
 DVDVCA.P11 08-AUG-77 09:16 TYPE ROUTINE

SEQ 0089

```

3877 .SBTTL TYPE ROUTINE
3878
3879 *****
3880 *ROUTINE TO TYPE ASCIZ MESSAGE. MESSAGE MUST TERMINATE WITH A 0 BYTE.
3881 *THE ROUTINE WILL INSERT A NUMBER OF NULL CHARACTERS AFTER A LINE FEED.
3882 *NOTE1: $NULL CONTAINS THE CHARACTER TO BE USED AS THE FILLER CHARACTER.
3883 *NOTE2: $FILLC CONTAINS THE NUMBER OF FILLER CHARACTERS REQUIRED.
3884 *NOTE3: $FILLC CONTAINS THE CHARACTER TO FILL AFTER.
3885
3886 *CALL:
3887 *1) USING A TRAP INSTRUCTION
3888 *      TYPE      ,MESADR      ;;MESADR IS FIRST ADDRESS OF AN ASCIZ STRING
3889 *OR
3890 *      TYPE
3891 *      MESADR
3892
3893
3894 011770 105767 167163 STYPE: TSTB $TPFLG      ;; IS THERE A TERMINAL?
3895 011774 100002      BPL 1$      ;; BR IF YES
3896 011776 000000      HALT      ;; HALT HERE IF NO TERMINAL
3897 012000 000430      BR 3$      ;; LEAVE
3898 012002 010046 1$: MOV RO,-(SP)      ;; SAVE RO
3899 012004 017600 000002 MOV 32(SP),RO      ;; GET ADDRESS OF ASCIZ STRING
3900 012010 122767 000001 167176 CMPB $APTENV,$ENV      ;; RUNNING IN APT MODE
3901 012016 001011      BNE 62$      ;; NO GO CHECK FOR APT CONSOLE
3902 012020 132767 000100 167167 BITB $APTSPool,$ENV      ;; SPOOL MESSAGE TO APT
3903 012026 001405      BEQ 62$      ;; NO GO CHECK FOR CONSOLE
3904 012030 010067 000004 MOV RO,61$      ;; SETUP MESSAGE ADDRESS FOR APT
3905 012034 004767 000774 JSR PC,$ATY3      ;; SPOOL MESSAGE TO APT
3906 012040 000000 61$: .WORD 0      ;; MESSAGE ADDRESS
3907 012042 132767 000040 167145 62$: BITB $APTCSUP,$ENV      ;; APT CONSOLE SUPPRESSED
3908 012050 001003      BNE 60$      ;; YES, SKIP TYPE OUT
3909 012052 112046 2$: MOVB (RO)+,-(SP)      ;; PUSH CHARACTER TO BE TYPED ONTO STACK
3910 012054 001005      BNE 4$      ;; BR IF IT ISN'T THE TERMINATOR
3911 012056 005726      TST (SP)+      ;; IF TERMINATOR POP IT OFF THE STACK
3912 012060 012600 60$: MOV (SP)+,RO      ;; RESTORE RO
3913 012062 062716 000002 3$: ADD #2,(SP)      ;; ADJUST RETURN PC
3914 012066 000002      RTI      ;; RETURN
3915 012070 122716 000011 4$: CMPB $HT,(SP)      ;; BRANCH IF <HT>
3916 012074 001430      BEQ 8$      ;;
3917 012076 122716 000200 CMPB $CRLF,(SP)      ;; BRANCH IF NOT <CRLF>
3918 012102 001006      BNE 5$      ;;
3919 012104 005726      TST (SP)+      ;; POP <CR><LF> EQUIV
3920 012106 104401      TYPE      ;; TYPE A CR AND LF
3921 012110 001171      SCRLF
3922 012112 105067 000130 CLRB $CHARCNT      ;; CLEAR CHARACTER COUNT
3923 012116 000755 2$: BR 2$      ;; GET NEXT CHARACTER
3924 012120 004767 000056 5$: JSR PC,$TYPEC      ;; GO TYPE THIS CHARACTER
3925 012124 126726 167026 6$: CMPB $FILLC,(SP)+      ;; IS IT TIME FOR FILLER CHARS.?
3926 012130 000750      BNE 2$      ;; IF NO GO GET NEXT CHAR.
3927 012132 016746 167016 MOV $NULL,-(SP)      ;; GET # OF FILLER CHARS. NEEDED
3928      AND THE NULL CHAR.
3929 012136 105366 000001 7$: DECB 1(SP)      ;; DOES A NULL NEED TO BE TYPED?
3930 012142 002770      BLT 6$      ;; BR IF NO--GO POP THE NULL OFF OF STACK
3931 012144 004767 000032 JSR PC,$TYPEC      ;; GO TYPE A NULL
3932 012150 105367 000072 DECB $CHARCNT      ;; DO NOT COUNT AS A COUNT

```

M07

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 91
 DVDVCA.P11 08-AUG-77 09:16 TYPE ROUTINE

SEQ 0090

```

3933 012154 000770          BR      7$          ;;LOOP
3934
3935          ;HORIZONTAL TAB PROCESSOR
3936
3937 012156 112716 000040      8$:  MOVB  #' (SP)          ;; REPLACE TAB WITH SPACE
3938 012162 004767 000014      9$:  JSR  PC,$TYPEC          ;; TYPE A SPACE
3939 012166 132767 000007 000052  BITB  #7,$CHARCNT          ;; BRANCH IF NOT AT
3940 012174 001372          BNE  9$          ;; TAB STOP
3941 012176 005726          TST  (SP)+          ;; POP SPACE OFF STACK
3942 012200 000724          BR  2$          ;; GET NEXT CHARACTER
3943 012202 105777 166742      $TYPEC: TSTB  @STPS          ;; WAIT UNTIL PRINTER IS READY
3944 012206 100375          BPL  $TYPEC
3945 012210 116677 000002 166734  MOVB  2(SP),@STPB          ;; LOAD CHAR TO BE TYPED INTO DATA REG.
3946 012216 122766 000015 000002  CMPB  #CR,2(SP)          ;; IS CHARACTER A CARRIAGE RETURN?
3947 012224 001003          BNE  1$          ;; BRANCH IF NO
3948 012226 105067 000014          CLRB  $CHARCNT          ;; YES--CLEAR CHARACTER COUNT
3949 012232 000406          BR  $TYPEX          ;; EXIT
3950 012234 122766 000012 000002  1$:  CMPB  #LF,2(SP)          ;; IS CHARACTER A LINE FEED?
3951 012242 001402          BEQ  $TYPEX          ;; BRANCH IF YES
3952 012244 105227          INCB  (PC)+          ;; COUNT THE CHARACTER
3953 012246 000000          $CHARCNT: .WORD 0          ;; CHARACTER COUNT STORAGE
3954 012250 000207          $TYPEX: RTS  PC
3955

```

NO7

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 92
 DVDVCA.P11 08-AUG-77 09:16 TTY INPUT ROUTINE

SEQ 0091

```

3956 .SBTTL TTY INPUT ROUTINE
3957
3958 ;*****
3959 .ENABL LSB
3960
3961 ;*****
3962 ;SOFTWARE SWITCH REGISTER CHANGE ROUTINE.
3963 ;ROUTINE IS ENTERED FROM THE TRAP HANDLER, AND WILL
3964 ;SERVICE THE TEST FOR CHANGE IN SOFTWARE SWITCH REGISTER TRAP CALL
3965 ;WHEN OPERATING IN TTY FLAG MODE.
3966 012252 022767 000176 166660 SCKSWR: CMP      $SWREG,SWR      ;IS THE SOFT-SWR SELECTED?
3967 012260 001074 BNE      15$          ;BRANCH IF NO
3968 012262 105777 TSTB     $STKS          ;CHAR THERE?
3969 012266 100071 BPL      15$          ;IF NO, DON'T WAIT AROUND
3970 012270 117746 MOVB     $STKB, -(SP) ;SAVE THE CHAR
3971 012274 042716 BIC      $1C177, (SP) ;STRIP-OFF THE ASCII
3972 012300 022726 CMP      $7, (SP)+      ;IS IT A CONTROL G?
3973 012304 001062 BNE      15$          ;NO, RETURN TO USER
3974 012306 126727 CMPB     $AUTOB, #1 ;ARE WE RUNNING IN AUTO-MODE?
3975 012314 001456 BEQ      15$          ;BRANCH IF YES
3976
3977 012316 104401 012777 SGTSWR: TYPE     , $CNTLG      ;ECHO THE CONTROL-G (1G)
3978 012322 104401 013004 TYPE     , $MSWR      ;TYPE CURRENT CONTENTS
3979 012326 016746 165644 MOV      $SWREG, -(SP) ;SAVE SWREG FOR TYPEOUT
3980 012332 104402 TYP0C     , $MNEW      ;GO TYPE--OCTAL ASCII(ALL DIGITS)
3981 012334 104401 013015 TYPE     , $MNEW      ;PROMPT FOR NEW SWR
3982 012340 005046 19$: CLR      -(SP)      ;CLEAR COUNTER
3983 012342 005046 7$: CLR      -(SP)      ;THE NEW SWR
3984 012344 105777 166574 TSTB     $STKS          ;CHAR THERE?
3985 012350 100375 BPL      7$          ;IF NOT TRY AGAIN
3986
3987 012352 117746 166570 MOVB     $STKB, -(SP) ;PICK UP CHAR
3988 012356 042716 177600 BIC      $1C177, (SP) ;MAKE IT 7-BIT ASCII
3989
3990
3991
3992 012362 021627 000025 9$: CMP      (SP), #25      ;IS IT A CONTROL-U?
3993 012366 001005 BNE      10$          ;BRANCH IF NOT
3994 012370 104401 012772 TYPE     , $CNTLU      ;YES, ECHO CONTROL-U (1U)
3995 012374 062706 000006 20$: ADD      $6, SP      ;IGNORE PREVIOUS INPUT
3996 012400 000757 BR       19$          ;LET'S TRY IT AGAIN
3997
3998
3999 012402 021627 000015 10$: CMP      (SP), #15      ;IS IT A <CR>?
4000 012406 001022 BNE      16$          ;BRANCH IF NO
4001 012410 005766 TST      4(SP)          ;YES, IS IT THE FIRST CHAR?
4002 012414 001403 BEQ      11$          ;BRANCH IF YES
4003 012416 016677 000002 166514 MOV      2(SP), $SWR      ;SAVE NEW SWR
4004 012424 062706 000006 11$: ADD      $6, SP      ;CLEAR UP STACK
4005 012430 104401 001171 14$: TYPE     , $CRLF      ;ECHO <CR> AND <LF>
4006 012434 126727 166475 000001 CMPB     $INTAG, #1      ;RE-ENABLE TTY KBD INTERRUPTS?
4007 012442 001003 BNE      15$          ;BRANCH IF NOT
4008 012444 012777 000100 166472 MOV      $100, $STKS      ;RE-ENABLE TTY KBD INTERRUPTS
4009 012452 000002 15$: RTI          ;RETURN
4010 012454 004767 177522 16$: JSR      PC, $TYPEC      ;ECHO CHAR
4011 012460 021627 000060 CMP      (SP), #60      ;CHAR < 0?

```

```

4012 012464 002420          BLT      18$          ;; BRANCH IF YES
4013 012466 021627 000067    CMP      (SP),#67      ;; CHAR > 7?
4014 012472 003015          BGT      18$          ;; BRANCH IF YES
4015 012474 042726 000060    BIC      #60,(SP)+      ;; STRIP-OFF ASCII
4016 012500 005766 000002    TST      2(SP),        ;; IS THIS THE FIRST CHAR
4017 012504 001403          BEQ      17$          ;; BRANCH IF YES
4018 012506 006316          ASL      (SP)          ;; NO, SHIFT PRESENT
4019 012510 006316          ASL      (SP)          ;; CHAR OVER TO MAKE
4020 012512 006316          ASL      (SP)          ;; ROOM FOR NEW ONE.
4021 012514 005266 000002    17$: INC      2(SP)      ;; KEEP COUNT OF CHAR
4022 012520 056616 177776    BIS      -2(SP),(SP)    ;; SET IN NEW CHAR
4023 012524 000707          BR       7$          ;; GET THE NEXT ONE
4024 012526 104401 001170    16$: TYPE    $QUES      ;; TYPE ?(CR)(LF)
4025 012532 000720          BR       20$          ;; SIMULATE CONTROL-U
4026
4027
4028
4029
4030
4031
4032
4033
4034
4035
4036
4037 012534 011646          $ROCHR: MOV      (SP),-(SP)    ;; PUSH DOWN THE PC
4038 012536 016666 000004 000002    MOV      4(SP),2(SP)    ;; SAVE THE PS
4039 012544 105777 166374 1$: TSTB      2$TKS      ;; WAIT FOR
4040 012550 100375          BPL      1$          ;; A CHARACTER
4041 012552 117766 166370 000004    MOVB      2$TKB,4(SP)    ;; READ THE TTY
4042 012560 042766 177600 000004    BIC      #1C(177),4(SP)    ;; GET RID OF JUNK IF ANY
4043 012566 026627 000004 000023    CMP      4(SP),#23      ;; IS IT A CONTROL-S?
4044 012574 001013          BNE      3$          ;; BRANCH IF NO
4045 012576 105777 146342 2$: TSTB      2$TKS      ;; WAIT FOR A CHARACTER
4046 012602 100375          BPL      2$          ;; LOOP UNTIL ITS THERE
4047 012604 117746 166336          MOVB      2$TKB,-(SP)    ;; GET CHARACTER
4048 012610 042716 177600          BIC      #1C177,(SP)    ;; MAKE IT 7-BIT ASCII
4049 012614 022627 000021          CMP      (SP)+,#21      ;; IS IT A CONTROL-Q?
4050 012620 001366          BNE      2$          ;; IF NOT DISCARD IT
4051 012622 000750          BR       1$          ;; YES, RESUME
4052 012624 026627 000004 000140 3$: CMP      4(SP),#140    ;; IS IT UPPER CASE?
4053 012632 002407          BLT      4$          ;; BRANCH IF YES
4054 012634 026627 000004 000175    CMP      4(SP),#175    ;; IS IT A SPECIAL CHAR?
4055 012642 003003          BGT      4$          ;; BRANCH IF YES
4056 012644 042766 000040 000004    BIC      #4C,4(SP)    ;; MAKE IT UPPER CASE
4057 012652 000002          RTI          ;; GO BACK TO USER
4058
4059
4060
4061
4062
4063
4064
4065 012654 010346          $RDLIN: MOV      R3, -(SP)    ;; SAVE R3
4066 012656 012703 012762 1$: MOV      #TTYIN,R3      ;; GET ADDRESS
4067 012662 022703 012772 2$: CMP      #TTYIN+8.,R3    ;; BUFFER FULL?

```

 THIS ROUTINE WILL INPUT A SINGLE CHARACTER FROM THE TTY
 *CALL:
 * ROCHR
 * RETURN HERE
 *

 INPUT A SINGLE CHARACTER FROM THE TTY
 CHARACTER IS ON THE STACK
 WITH PARITY BIT STRIPPED OFF

 PUSH DOWN THE PC
 SAVE THE PS
 WAIT FOR
 A CHARACTER
 READ THE TTY
 GET RID OF JUNK IF ANY
 IS IT A CONTROL-S?
 BRANCH IF NO
 WAIT FOR A CHARACTER
 LOOP UNTIL ITS THERE
 GET CHARACTER
 MAKE IT 7-BIT ASCII
 IS IT A CONTROL-Q?
 IF NOT DISCARD IT
 YES, RESUME
 IS IT UPPER CASE?
 BRANCH IF YES
 IS IT A SPECIAL CHAR?
 BRANCH IF YES
 MAKE IT UPPER CASE
 GO BACK TO USER

 THIS ROUTINE WILL INPUT A STRING FROM THE TTY
 *CALL:
 * RDLIN
 * RETURN HERE
 *

 INPUT A STRING FROM THE TTY
 ADDRESS OF FIRST CHARACTER WILL BE ON THE STACK
 TERMINATOR WILL BE A BYTE OF ALL 0'S

 SAVE R3
 GET ADDRESS
 BUFFER FULL?

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 94
 DVDVCA.P11 08-AUG-77 09:16 TTY INPUT ROUTINE

SEQ 0093

4068	012666	101405				BLOS	4S		:: BR IF YES
4069	012670	104410				ROCHR			:: GO READ ONE CHARACTER FROM THE TTY
4070	012672	112613				MOVB	(SP)+, (R3)		:: GET CHARACTER
4071	012674	122713	000177		10S:	CMPS	#177, (R3)		:: IS IT A RUBOUT
4072	012700	001003				BNE	3S		:: SKIP IF NOT
4073	012702	104401	001170		4S:	TYPE	\$QUES		:: TYPE A '?'
4074	012706	000763				BR	1S		:: CLEAR THE BUFFER AND LOOP
4075	012710	111367	000044		3S:	MOVB	(R3), 9S		:: ECHO THE CHARACTER
4076	012714	104401	012760			TYPE	9S		
4077	012720	122723	000015			CMPS	#15, (R3)+		:: CHECK FOR RETURN
4078	012724	001356				BNE	2S		:: LOOP IF NOT RETURN
4079	012726	105063	177777			CLRB	-1(R3)		:: CLEAR RETURN (THE 15)
4080	012732	104401	001172			TYPE	\$LF		:: TYPE A LINE FEED
4081	012736	012673				MOV	(SP)+, R3		:: RESTORE R3
4082	012740	0116-6				MOV	(SP), -(SP)		:: ADJUST THE STACK AND PUT ADDRESS OF THE
4083	012742	016666	000004	000002		MOV	4(SP), 2(SP)		:: FIRST ASCII CHARACTER ON IT
4084	012750	012766	012762	000004		MOV	#TTYIN, 4(SP)		
4085	012756	000002				RTI			:: RETURN
4086	012760	000			9S:	.BYTE	0		:: STORAGE FOR ASCII CHAR. TO TYPE
4087	012761	000				.BYTE	0		:: TERMINATOR
4088	012762	000010				.BLKB	8.		:: RESERVE 8 BYTES FOR TTY INPUT
4089	012772	052536	005015	000	\$TTYIN:	.ASCIZ	/↑U/⟨15⟩⟨12⟩		:: CONTROL "U"
4090	012777	136	006507	000012	\$CNTLG:	.ASCIZ	/↑G/⟨15⟩⟨12⟩		:: CONTROL "G"
4091	013004	005015	053523	020122	\$MSWR:	.ASCIZ	⟨15⟩⟨12⟩/SWR = /		
4092	013012	020075	000						
4093	013015	040	047040	053505	\$MNEW:	.ASCIZ	/ NEW = /		
4094	013022	036440	000040						

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 95
 DVDVCA.P11 08-AUG-77 09:16 APT COMMUNICATIONS ROUTINE

SEQ 0094

```

4095 .SBTTL APT COMMUNICATIONS ROUTINE
4096
4097 *****
4098 013026 112767 000001 000236 $ATY1: MOV8 #1,$FFLG ;;TO REPORT FATAL ERROR
4099 013034 112767 000001 000226 $ATY3: MOV8 #1,$MFLG ;;TO TYPE A MESSAGE
4100 013 42 000403 BR $ATYC
4101 013044 112767 000001 000220 $ATY4: MOV8 #1,$FFLG ;;TO ONLY REPORT FATAL ERROR
4102 013052 $ATYC:
4103 013052 010046 MOV R0,-(SP) ;;PUSH R0 ON STACK
4104 013 54 010146 MOV R1,-(SP) ;;PUSH R1 ON STACK
4105 013056 105767 000206 TSTB $MFLG ;;SHOULD TYPE A MESSAGE?
4106 013062 001450 BEQ 5$ IF NOT: BR
4107 013064 122767 000001 166122 CMPB #APTENV,$ENV ;;OPERATING UNDER APT?
4108 013072 001031 BNE 3$ IF NOT: BR
4109 013 74 132767 000100 166113 BITB #APTPOOL,$ENVM ;;SHOULD SPOOL MESSAGES?
4110 013102 001425 BEQ 3$ IF NOT: BR
4111 013104 017600 000004 MOV #4(SP),R0 ;;GET MESSAGE ADDR.
4112 013110 062766 000002 000004 ADD #2,4(SP) ;;BUMP RETURN ADDR.
4113 013116 005767 166052 1$: TST $MSGTYPE ;;SEE IF DONE W/ LAST XMISSION?
4114 013122 001375 BNE 1$ IF NOT: WAIT
4115 013124 010067 166060 MOV R0,$MSGAD ;;PUT ADDR IN MAILBOX
4116 013130 105720 2$: TSTB (R0)+ ;;FIND END OF MESSAGE
4117 013132 001376 BNE 2$
4118 013134 166700 166050 SUB $MSGAD,R0 ;;SUB START OF MESSAGE
4119 013140 006200 ASR R0 ;;GET MESSAGE LGTH IN WORDS
4120 013142 010067 166044 MOV R0,$MSGGLT ;;PUT LENGTH IN MAILBOX
4121 013146 012767 000004 166020 MOV #4,$MSGTYPE ;;TELL APT TO TAKE MSG.
4122 013154 000413 BR 5$
4123 013156 017667 000004 000016 3$: MOV #4(SP),4$ ;;PUT MSG ADDR IN JSR LINKAGE
4124 013164 062766 000002 000004 ADD #2,4(SP) ;;BUMP RETURN ADDRESS
4125 013172 016746 164600 MOV 177776,-(SP) ;;PUSH 177776 ON STACK
4126 013176 004767 176566 JSR PC,$TYPE ;;CALL TYPE MACRO
4127 013202 000000 4$: .WORD 0
4128 013204 5$:
4129 013204 105767 000062 10$: TSTB $FFLG ;;SHOULD REPORT FATAL ERROR?
4130 013210 001416 BEQ 12$ IF NOT: BR
4131 013212 005767 165776 TST $ENV ;;RUNNING UNDER APT?
4132 013216 001413 BEQ 12$ IF NOT: BR
4133 013220 005767 165750 11$: TST $MSGTYPE ;;FINISHED LAST MESSAGE?
4134 013224 001375 BNE 11$ IF NOT: WAIT
4135 013226 017667 000004 165742 MOV #4(SP),$FATAL ;;GET ERROR #
4136 013234 062766 000002 000004 ADD #2,4(SP) ;;BUMP RETURN ADDR.
4137 013242 005267 165726 INC $MSGTYPE ;;TELL APT TO TAKE ERROR
4138 013246 105067 000020 12$: CLRB $FFLG ;;CLEAR FATAL FLAG
4139 013252 105067 000013 CLRB $LFLG ;;CLEAR LOG FLAG
4140 013256 105067 000006 CLRB $MFLG ;;CLEAR MESSAGE FLAG
4141 013262 012601 MOV (SP)+,R1 ;;POP STACK INTO R1
4142 013264 012600 MOV (SP)+,R0 ;;POP STACK INTO R0
4143 013266 000207 RTS PC ;;RETURN
4144 013270 000 $MFLG: .BYTE 0 ;;MESSG. FLAG
4145 013271 000 $LFLG: .BYTE 0 ;;LOG FLAG
4146 013272 000 $FFLG: .BYTE 0 ;;FATAL FLAG
4147 013274 .EVEN
4148 000200 APTSIZE=200
4149 000001 APTENV=001
4150 000100 APTPOOL=100

```

E08

M2INDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 96
DVDVCA.P11 08-AUG-77 09:16 APT COMMUNICATIONS ROUTINE

SEQ 0095

4151

000040

APTC SUP=040

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 97
 DVDVCA.P11 08-AUG-77 09:16 ERROR HANDLER ROUTINE

SEQ 0096

```

4152 .SBTTL ERROR HANDLER ROUTINE
4153
4154 *****
4155 *THIS ROUTINE WILL INCREMENT THE ERROR FLAG AND THE ERROR COUNT,
4156 *SAVE THE ERROR ITEM NUMBER AND THE ADDRESS OF THE ERROR CALL
4157 *AND GO TO MYTYPE ON ERROR
4158 *THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
4159 *SW15=1 HALT ON ERROR
4160 *SW13=1 INHIBIT ERROR TYPEOUTS
4161 *SW10=1 BELL ON ERROR
4162 *SW09=1 LOOP ON ERROR
4163 *CALL
4164 * ERROR N ;;ERROR=EMT AND N=ERROR ITEM NUMBER
4165
4166 $ERROR:
4167 013274 104407 165601 7$: CKSWR INCB SERFLG ;;TEST FOR CHANGE IN SOFT-SWR
4168 013276 105267 165601 7$: BEQ 7$ SERFLG SET THE ERROR FLAG
4169 013302 001775 165572 165630 7$: MOV $TSTNM,2DISP; DON'T LET THE FLAG GO TO ZERO
4170 013304 016777 002000 165620 7$: BIT $BIT10,2SWR; DISPLAY TEST NUMBER AND ERROR FLAG
4171 013312 032777 001402 165552 7$: BEQ 1$ $BELL; BELL ON ERROR?
4172 013320 001402 001164 165542 7$: TYPE $BELL NO - SKIP
4173 013322 104401 001164 165542 7$: INC $ERTTL; RING BELL
4174 013326 005267 165560 165552 7$: MOV (SP), $ERRPC; COUNT THE NUMBER OF ERRORS
4175 013332 011667 165560 165552 7$: SUB $2, $ERRPC; GET ADDRESS OF ERROR INSTRUCTION
4176 013336 162767 000002 165542 7$: MOV $2, $ERRPC; STRIP AND SAVE THE ERROR ITEM CODE
4177 013344 117767 165546 165560 7$: MOV $2, $ERRPC; SKIP TYPEOUT IF SET
4178 013352 032777 020000 165560 7$: BIT $BIT13,2SWR; SKIP TYPEOUTS
4179 013360 001004 175706 20$: BNE 20$ PC,MYTYPE; GO TO USER ERROR ROUTINE
4180 013362 004767 001171 20$: JSR PC,MYTYPE
4181 013366 104401 001171 20$: TYPE $CRLF
4182 013372 122767 000001 165614 20$: CMPB $APTENV, $ENV; RUNNING IN APT MODE
4183 013372 001007 165506 000004 20$: BNE 2$ 2$; NO SKIP APT ERROR REPORT
4184 013400 001007 165506 000004 20$: MOV $ITEMB, 21$; SET ITEM NUMBER AS ERROR NUMBER
4185 013402 116767 177430 20$: JSR PC, $ATY4; REPORT FATAL ERROR TO APT
4186 013410 004767 000000 21$: .BYTE 0
4187 013414 000 000 21$: .BYTE 0
4188 013415 000 000 21$: .BYTE 0
4189 013416 000777 165514 22$: BR 22$; APT ERROR LOOP
4190 013420 005777 165514 22$: TST 2SWR; HALT ON ERROR
4191 013424 100002 3$: BPL 3$; SKIP IF CONTINUE
4192 013426 000000 3$: HALT; HALT ON ERROR!
4193 013430 104407 001000 165500 3$: CKSWR; TEST FOR CHANGE IN SOFT-SWR
4194 013432 032777 001000 165500 3$: BIT $BIT09,2SWR; LOOP ON ERROR SWITCH SET?
4195 013440 001402 165442 4$: BEQ 4$ $LPERK, (SP); BR IF NO
4196 013442 016716 165510 4$: MOV $ESCAPE, (SP); FUDGE RETURN FOR LOOPING
4197 013446 005767 165502 4$: TST $ESCAPE; CHECK FOR AN ESCAPE ADDRESS
4198 013452 001402 165502 4$: BEQ 5$; BR IF NONE
4199 013454 016716 165502 4$: MOV $ESCAPE, (SP); FUDGE RETURN ADDRESS FOR ESCAPE
4200 013460 022737 011552 000042 5$: CMP $SENDAD, 2$42; ACT-11 AUTO-ACCEPT?
4201 013460 001001 6$: BNE 6$; BRANCH IF NO
4202 013466 000000 6$: HALT; YES
4203 013470 000000 6$: RTI; RETURN
4204 013472 000002
4205 013472 000002

```

G08

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 98
 DVDVCA.P11 08-AUG-77 09:16 SCOPE HANDLER ROUTINE

SEQ 0097

```

4206 .SBTTL SCOPE HANDLER ROUTINE
4207
4208 *****
4209 *THIS ROUTINE CONTROLS THE LOOPING OF SUBTESTS. IT WILL INCREMENT
4210 *AND LOAD THE TEST NUMBER($STNM) INTO THE DISPLAY REG.(DISPLAY<7:0>)
4211 *AND LOAD THE ERROR FLAG ($ERFLG) INTO DISPLAY<15:08>
4212 *THE SWITCH OPTIONS PROVIDED BY THIS ROUTINE ARE:
4213 *SW14=1 LOOP ON TEST
4214 *SW11=1 INHIBIT ITERATIONS
4215 *SW09=1 LOOP ON ERROR
4216 *SW08=1 LOOP ON TEST IN SWR<7:0>
4217 *CALL
4218 * SCOPE ;;SCOPE=IOT
4219
4220 $SCOPE:
4221 013474 104407 CKSWR
4222 013476 032777 040000 165434 1$: BIT #BIT14,$SWR ;;TEST FOR CHANGE IN SOFT-SWR
4223 013504 001114 BNE $OVER ;;LOOP ON PRESENT TEST?
4224 *****
4225 013506 000416 $XTSTR: BR 6$ ;;YES IF SW14=1
4226 *****
4227 013510 013746 000004 MOV $ERRVEC,-(SP) ;;IF RUNNING ON THE "XOR" TESTER CHANGE
4228 013514 012737 013534 000004 MOV #5,$ERRVEC ;;THIS INSTRUCTION TO A "NOP" (NOP=240)
4229 013732 005737 177060 TST $BIT17060 ;;SAVE THE CONTENTS OF THE ERROR VECTOR
4230 013736 012637 000004 MOV (SP)+,$ERRVEC ;;SET FOR TIMEOUT
4231 013532 000463 BR $SVLAD ;;TIME OUT ON XOR?
4232 013534 022626 5$: CMP (SP)+,(SP)+ ;;RESTORE THE ERROR VECTOR
4233 013536 012637 000004 MOV (SP)+,$ERRVEC ;;GO TO THE NEXT TEST
4234 013542 000423 BR 7$ ;;CLEAR THE STACK AFTER A TIME OUT
4235 013544 032777 000400 165366 6$; *****END OF CODE FOR THE XOR TESTER***** ;;RESTORE THE ERROR VECTOR
4236 013552 001404 BEQ 2$ ;;LOOP ON THE PRESENT TEST
4237 013554 127767 165360 165320 CMPB $SWR,$STNM ;;LOOP ON SPEC. TEST?
4238 013562 001465 BEQ $OVER ;;BR IF NO
4239 013564 105767 165313 2$: TSTB $ERFLG ;;ON THE RIGHT TEST? SWR<7:0>
4240 013570 001421 BEQ 3$ ;;BR IF YES
4241 013572 126767 165317 165303 CMPB $ERMAX,$ERFLG ;;HAS AN ERROR OCCURRED?
4242 013600 101015 BHI 3$ ;;BR IF NO
4243 013602 032777 001000 165330 BIT #BIT09,$SWR ;;MAX. ERRORS FOR THIS TEST OCCURRED?
4244 013610 001404 BEQ 4$ ;;BR IF NO
4245 013612 016767 165272 165266 7$: MOV $LPERR,$LPADR ;;LOOP ON ERROR?
4246 013620 000446 BR $OVER ;;BR IF NO
4247 013622 105067 165255 4$: CLRB $ERFLG ;;SET LOOP ADDRESS TO LAST SCOPE
4248 013626 005067 165326 CLR $TIMES ;;ZERO THE ERROR FLAG
4249 013632 000415 BR 1$ ;;CLEAR THE NUMBER OF ITERATIONS TO MAKE
4250 013634 032777 004000 165276 3$: BIT #BIT11,$SWR ;;ESCAPE TO THE NEXT TEST
4251 013642 001011 BNE 1$ ;;INHIBIT ITERATIONS?
4252 013644 005767 165332 TST $PASS ;;BR IF YES
4253 013650 001406 BEQ 1$ ;;IF FIRST PASS OF PROGRAM
4254 013652 005267 165226 INC $ICNT ;;INHIBIT ITERATIONS
4255 013656 026767 165276 165220 CMP $TIMES,$ICNT ;;INCREMENT ITERATION COUNT
4256 013664 002024 BGE $OVER ;;CHECK THE NUMBER OF ITERATIONS MADE
4257 013666 012767 000001 165210 1$: MOV #1,$ICNT ;;BR IF MORE ITERATION REQUIRED
4258 013674 016767 000052 165256 MOV $MXCNT,$TIMES ;;REINITIALIZE THE ITERATION COUNTER
4259 013702 105267 165174 $SVLAD: INCB $STNM ;;SET NUMBER OF ITERATIONS TO DO
4260 013706 116767 165170 165264 MOV $STNM,$TESTN ;;COUNT TEST NUMBERS
4261 ;;SET TEST NUMBER IN APT MAILBOX

```

H08

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 99
 DVDVCA.P11 08-AUG-77 09:16 SCOPE HANDLER ROUTINE

SEQ 0098

4262	013714	011667	165166		MOV	(SP), \$LPADR	:: SAVE SCOPE LOOP ADDRESS
4263	013720	011667	165164		MOV	(SP), \$LPERR	:: SAVE ERROR LOOP ADDRESS
4264	013724	005077	165232		CLR	\$ESCAPE	:: CLEAR THE ESCAPE FROM ERROR ADDRESS
4265	013730	112767	000001	165157	MOVB	#1, \$ERMAX	:: ONLY ALLOW ONE(1) ERROR ON NEXT TEST
4266	013736	016777	165140	165176	\$OVER: MOV	\$TSTNM, @DISPLAY	:: DISPLAY TEST NUMBER
4267	013744	016716	165136		MOV	\$LPADR, (SP)	:: FUDGE RETURN ADDRESS
4268	013750	000002			RTI		:: FIXES PS
4269	013752	003720			\$MXCNT: 2000.		:: MAX. NUMBER OF ITERATIONS

MAINDEC-11-DVDVC-A
DVDVCA.P11

MACY11 27(1006)
08-AUG-77 09:16

08-AUG-77 09:20 PAGE 100
CONVERT BINARY TO DECIMAL AND TYPE ROUTINE

SEQ 0099

.SBTTL CONVERT BINARY TO DECIMAL AND TYPE ROUTINE

*THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 5-DIGIT
*SIGNED DECIMAL (ASCII) NUMBER AND TYPE IT. DEPENDING ON WHETHER THE
*NUMBER IS POSITIVE OR NEGATIVE A SPACE OR A MINUS SIGN WILL BE TYPED
*BEFORE THE FIRST DIGIT OF THE NUMBER. LEADING ZEROS WILL ALWAYS BE
*REPLACED WITH SPACES.

*CALL:

* MOV NUM, -(SP) ; PUT THE BINARY NUMBER ON THE STACK
* TYPDS ; GO TO THE ROUTINE

STYPDS:

MOV	R0, -(SP)	;	PUSH R0 ON STACK
MOV	R1, -(SP)	;	PUSH R1 ON STACK
MOV	R2, -(SP)	;	PUSH R2 ON STACK
MOV	R3, -(SP)	;	PUSH R3 ON STACK
MOV	R5, -(SP)	;	PUSH R5 ON STACK
MOV	#20200, -(SP)	;	SET BLANK SWITCH AND SIGN
MOV	20(SP), R5	;	GET THE INPUT NUMBER
BPL	1\$	;	BR IF INPUT IS POS.
NEG	R5	;	MAKE THE BINARY NUMBER POS.
MOVB	#'-, 1(SP)	;	MAKE THE ASCII NUMBER NEG.
CLR	R0	;	ZERO THE CONSTANTS INDEX
MOV	#\$DBLK, R3	;	SETUP THE OUTPUT POINTER
MOVB	#', (R3)+	;	SET THE FIRST CHARACTER TO A BLANK
CLR	R2	;	CLEAR THE BCD NUMBER
MOV	\$DTBL(R0), R1	;	GET THE CONSTANT
SUB	R1, R5	;	FORM THIS BCD DIGIT
BLT	4\$	;	BR IF DONE
INC	R2	;	INCREASE THE BCD DIGIT BY 1
BR	3\$		
ADD	R1, R5	;	ADD BACK THE CONSTANT
TST	R2	;	CHECK IF BCD DIGIT=0
BNE	5\$	;	FALL THROUGH IF 0
TSTB	(SP)	;	STILL DOING LEADING 0'S?
BMI	7\$	;	BR IF YES
ASLB	(SP)	;	MSD?
BCC	6\$	;	BR IF NO
MOVB	1(SP), -1(R3)	;	YES--SET THE SIGN
BIS	#'0, R2	;	MAKE THE BCD DIGIT ASCII
BIS	#', R2	;	MAKE IT A SPACE IF NOT ALREADY A DIGIT
MOVB	R2, (R3)+	;	PUT THIS CHARACTER IN THE OUTPUT BUFFER
TST	(R0)+	;	JUST INCREMENTING
CMP	R0, #10	;	CHECK THE TABLE INDEX
BLT	2\$	;	GO DO THE NEXT DIGIT
BGT	8\$	;	GO TO EXIT
MOV	R5, R2	;	GET THE LSD
BR	6\$	;	GO CHANGE TO ASCII
TSTB	(SP)+	;	WAS THE LSD THE FIRST NON-ZERO?
BPL	9\$	;	BR IF NO
MOVB	-1(SP), -2(R3)	;	YES--SET THE SIGN FOR TYPING
CLRB	(R3)	;	SET THE TERMINATOR
MOV	(SP)+, R5	;	POP STACK INTO R5
MOV	(SP)+, R3	;	POP STACK INTO R3
MOV	(SP)+, R2	;	POP STACK INTO R2

4270				
4271				
4272				
4273				
4274				
4275				
4276				
4277				
4278				
4279				
4280				
4281				
4282	013754			
4283	013754	010046		
4284	013756	010146		
4285	013760	010246		
4286	013762	010346		
4287	013764	010546		
4288	013766	012746	020200	
4289	013772	016605	000020	
4290	013776	100004		
4291	014000	005405		
4292	014002	112766	000055	000001
4293	014010	005000		
4294	014012	012703	014170	
4295	014016	112723	000040	
4296	014022	005002		
4297	014024	016001	014160	
4298	014030	160105		
4299	014032	002402		
4300	014034	005202		
4301	014036	000774		
4302	014040	060105		
4303	014042	005702		
4304	014044	001002		
4305	014046	105716		
4306	014050	100407		
4307	014052	106316		
4308	014054	103003		
4309	014056	116663	000001	177777
4310	014064	052702	000060	
4311	014070	052702	000040	
4312	014074	110223		
4313	014076	005720		
4314	014100	020027	000010	
4315	014104	002746		
4316	014106	003002		
4317	014110	010502		
4318	014112	000764		
4319	014114	105726		
4320	014116	100003		
4321	014120	116663	177777	177776
4322	014126	105013		
4323	014130	012605		
4324	014132	012603		
4325	014134	012602		

J08

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 101
 DVDVCA.P11 08-AUG-77 09:16 CONVERT BINARY TO DECIMAL AND TYPE ROUTINE

SEQ 0100

4326	014136	012601		MOV	(SP)+,R1	::POP STACK INTO R1
4327	014140	012600		MOV	(SP)+,R0	::POP STACK INTO R0
4328	014142	104401	014170	TYPE	\$DBLK	::NOW TYPE THE NUMBER
4329	014146	016666	000002 000004	MOV	2(SP),4(SP)	::ADJUST THE STACK
4330	014154	012616		MOV	(SP)+,(SP)	
4331	014156	000002		RTI		::RETURN TO USER
4332	014160	023420		\$DTBL:	10000.	
4333	014162	001750			1000.	
4334	014164	000144			100.	
4335	014166	000012			10.	
4336	014170	000004		\$DBLK:	.BLKW 4	

K08

MAINDEC-11-DVDVC-A
DVDVCA.P11MACY11 27(1006)
08-AUG-77 09:1608-AUG-77 09:20 PAGE 102
BINARY TO OCTAL (ASCII) AND TYPE

SEQ 0101

.SBTTL BINARY TO OCTAL (ASCII) AND TYPE

```

*****
;THIS ROUTINE IS USED TO CHANGE A 16-BIT BINARY NUMBER TO A 6-DIGIT
;OCTAL (ASCII) NUMBER AND TYPE IT.
;STYPOS---ENTER HERE TO SETUP SUPPRESS ZEROS AND NUMBER OF DIGITS TO TYPE
;CALL:
;      MOV      NUM,-(SP)      ;;NUMBER TO BE TYPED
;      TYPPOS      ;;CALL FOR TYPEOUT
;      .BYTE      N      ;;N=1 TO 6 FOR NUMBER OF DIGITS TO TYPE
;      .BYTE      M      ;;M=1 OR 0
;                               ;;1=TYPE LEADING ZEROS
;                               ;;0=SUPPRESS LEADING ZEROS

```

```

;STYPON---ENTER HERE TO TYPE OUT WITH THE SAME PARAMETERS AS THE LAST
;STYPOS OR STYPOC

```

```

;CALL:
;      MOV      NUM,-(SP)      ;;NUMBER TO BE TYPED
;      TYPON      ;;CALL FOR TYPEOUT

```

```

;STYPOC---ENTER HERE FOR TYPEOUT OF A 16 BIT NUMBER

```

```

;CALL:
;      MOV      NUM,-(SP)      ;;NUMBER TO BE TYPED
;      TYPOC      ;;CALL FOR TYPEOUT

```

4337									
4338									
4339									
4340									
4341									
4342									
4343									
4344									
4345									
4346									
4347									
4348									
4349									
4350									
4351									
4352									
4353									
4354									
4355									
4356									
4357									
4358									
4359									
4360									
4361									
4362	014200	017646	000000			STYPOS:	MOV	2(SP),-(SP)	;; PICKUP THE MODE
4363	014204	116667	000001	000211			MOVB	1(SP),SOFILL	;; LOAD ZERO FILL SWITCH
4364	014212	112667	000207				MOVB	(SP)+,SOMODE+1	;; NUMBER OF DIGITS TO TYPE
4365	014216	062716	000002				ADD	#2,(SP)	;; ADJUST RETURN ADDRESS
4366	014222	000406					BR	STYPON	
4367	014224	112767	000001	000171		STYPOC:	MOVB	#1,SOFILL	;; SET THE ZERO FILL SWITCH
4368	014232	112767	000006	000165			MOVB	#6,SOMODE+1	;; SET FOR SIX(6) DIGITS
4369	014240	112767	000005	000154		STYPON:	MOVB	#5,SOCNT	;; SET THE ITERATION COUNT
4370	014246	010346					MOV	R3, -(SP)	;; SAVE R3
4371	014250	010446					MOV	R4, -(SP)	;; SAVE R4
4372	014252	010546					MOV	R5, -(SP)	;; SAVE R5
4373	014254	116704	000145				MOVB	SOMODE+1,R4	;; GET THE NUMBER OF DIGITS TO TYPE
4374	014260	005404					NEG	R4	
4375	014262	062704	000006				ADD	#6,R4	;; SUBTRACT IT FOR MAX. ALLOWED
4376	014266	110467	000132				MOVB	R4,SOMODE	;; SAVE IT FOR USE
4377	014272	116704	000125				MOVB	SOFILL,R4	;; GET THE ZERO FILL SWITCH
4378	014276	016605	000012				MOV	12(SP),R5	;; PICKUP THE INPUT NUMBER
4379	014302	005003					CLR	R3	;; CLEAR THE OUTPUT WORD
4380	014304	006105				15:	ROL	R5	;; ROTATE MSB INTO "C"
4381	014306	000404					BR	35	;; GO DO MSB
4382	014310	006105				25:	ROL	R5	;; FORM THIS DIGIT
4383	014312	006105					ROL	R5	
4384	014314	006105					ROL	R5	
4385	014316	010503					MOV	R5,R3	
4386	014320	006103				35:	ROL	R3	;; GET LSB OF THIS DIGIT
4387	014322	105357	000076				DECB	SOMODE	;; TYPE THIS DIGIT?
4388	014326	100016					BPL	75	;; BR IF NO
4389	014330	042703	177770				BIC	#177770,R3	;; GET RID OF JUNK
4390	014334	001002					BNE	45	;; TEST FOR 0
4391	014336	005704					TST	R4	;; SUPPRESS THIS 0?
4392	014340	001403					BEQ	55	;; BR IF YES

L08

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 103
 DVDVCA.P11 08-AUG-77 09:16 BINARY TO OCTAL (ASCII) AND TYPE

SEQ 0102

4393	014342	005204		4\$:	INC	R4	:: DON'T SUPPRESS ANYMORE 0'S
4394	014344	052703	000060		BIS	#'0,R3	:: MAKE THIS DIGIT ASCII
4395	014350	052703	000040	5\$:	BIS	#' ,R3	:: MAKE ASCII IF NOT ALREADY
4396	014354	110367	000040		MOVB	R3,8\$	:: SAVE FOR TYPING
4397	014360	104401	014420		TYPE	8\$	:: GO TYPE THIS DIGIT
4398	014364	105367	000032	7\$:	DECB	\$OCNT	:: COUNT BY 1
4399	014370	003347			BGT	2\$	:: BR IF MORE TO DO
4400	014372	002402			BLT	6\$	:: BR IF DONE
4401	014374	005204			INC	R4	:: INSURE LAST DIGIT ISN'T A BLANK
4402	014376	000744			BR	2\$	:: GO DO THE LAST DIGIT
4403	014400	012605		6\$:	MOV	(SP)+,R5	:: RESTORE R5
4404	014402	012604			MOV	(SP)+,R4	:: RESTORE R4
4405	014404	012603			MOV	(SP)+,R3	:: RESTORE R3
4406	014406	016666	000002 000004		MOV	2(SP),4(SP)	:: SET THE STACK FOR RETURNING
4407	014414	012616			MOV	(SP)+,(SP)	
4408	014416	000002			RTI		:: RETURN
4409	014420	000		8\$:	.BYTE	0	:: STORAGE FOR ASCII DIGIT
4410	014421	000			.BYTE	0	:: TERMINATOR FOR TYPE ROUTINE
4411	014422	000		\$OCNT:	.BYTE	0	:: OCTAL DIGIT COUNTER
4412	014423	000		\$OFILL:	.BYTE	0	:: ZERO FILL SWITCH
4413	014424	000000		\$OMODE:	.WORD	0	:: NUMBER OF DIGITS TO TYPE

```

4414 .SBTTL TRAP DECODER
4415
4416 *****
4417 *THIS ROUTINE WILL PICKUP THE LOWER BYTE OF THE "TRAP" INSTRUCTION
4418 *AND USE IT TO INDEX THROUGH THE TRAP TABLE FOR THE STARTING ADDRESS
4419 *OF THE DESIRED ROUTINE. THEN USING THE ADDRESS OBTAINED IT WILL
4420 *GO TO THAT ROUTINE.
4421
4422 014426 010046 STRAP: MOV RO,-(SP) ;;SAVE RO
4423 014430 016600 000002 MOV 2(SP),RO ;;GET TRAP ADDRESS
4424 014434 005740 TST -(RO) ;;BACKUP BY 2
4425 014436 111000 MOV8 (RO),RO ;;GET RIGHT BYTE OF TRAP
4426 014440 006300 ASL RO ;;POSITION FOR INDEXING
4427 014442 016000 014462 MOV STRPAD(RO),RO ;;INDEX TO TABLE
4428 014446 000200 RTS RO ;;GO TO ROUTINE
4429
4430
4431 ;;THIS IS USE TO HANDLE THE "GETPRI" MACRO
4432
4433 014450 011646 STRAP2: MOV (SP),-(SP) ;;MOVE THE PC DOWN
4434 014452 016666 000004 000002 MOV 4(SP),2(SP) ;;MOVE THE PSW DOWN
4435 014460 000002 RTI ;;RESTORE THE PSW
4436
4437 .SBTTL TRAP TABLE
4438
4439 *THIS TABLE CONTAINS THE STARTING ADDRESSES OF THE ROUTINES CALLED
4440 *BY THE "TRAP" INSTRUCTION.
4441
4442 ROUTINE
4443 -----
4444 014462 014450 STRPAD: .WORD STRAP2
4445 014464 011770 $TYPE ;;CALL=TYPE TRAP+1(104401) TTY TYPEOUT ROUTINE
4446 014466 014224 $TYPOC ;;CALL=TYPOC TRAP+2(104402) TYPE OCTAL NUMBER (WITH LEADING ZEROS)
4447 014470 014200 $TYPOS ;;CALL=TYPOS TRAP+3(104403) TYPE OCTAL NUMBER (NO LEADING ZEROS)
4448 014472 014240 $TYPON ;;CALL=TYPON TRAP+4(104404) TYPE OCTAL NUMBER (AS PER LAST CALL)
4449 014474 013754 $TYPDS ;;CALL=TYPDS TRAP+5(104405) TYPE DECIMAL NUMBER (WITH SIGN)
4450
4451 014476 012322 $GTSWR ;;CALL=GTSWR TRAP+6(104406) GET SOFT-SWR SETTING
4452
4453 014500 012252 $CKSWR ;;CALL=CKSWR TRAP+7(104407) TEST FOR CHANGE IN SOFT-SWR
4454 014502 012534 $RDCHR ;;CALL=RDCHR TRAP+10(104410) TTY TYPEIN CHARACTER ROUTINE
4455 014504 012654 $RDLIN ;;CALL=RDLIN TRAP+11(104411) TTY TYPEIN STRING ROUTINE
4456 000001 .END

```

MACINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 106
DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

[illegible]

C09

M7INDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 103
 DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0106

[illegible]

009

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 109
 DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0107

PR7 = 000340	667#	1143	1369	2512	2646	2769	3144	3150						
PS = 177776	640#	641												
PSM = 177776	641#													
PMFVEC= 000024	732#	1031#	1032#	3836#	3837#	3846#	3852#	3864#	3865#					
RATES= 005676	2350	2431	2447#											
RBUF= 001262	1009#	1112#	1113#	1931	2075	2164	2213	2245	2263	2347	2899	3003	3060	
	3249	3347	3359											
RCSR= 001260	1008#	1110#	1457	1470#	1473	1487#	1490	1504#	1509	1540	1605	1831	1859	
	1905	1934	1986	2026	2030	2033	2056	2078	2367	2658#	2682#	2736#	2773#	
	2886	2990#	3042	3087#	3176#	3286#	3343							
RCVRAC= 004000	761#	1605	1986	2026	2033									
RCVR00= 000200	765#	1540	1832	1859	1906	1934	2030	2056	2078	2367	2887	3043	3343	
RCVRIE= 000100	766#	1457	1470	1473	1487	1490	1504	1509	2658	2682	2736	2773	3176	
	3286													
RDATA0= 000001	792#													
RDATA1= 000002	791#													
RDATA2= 000004	790#													
RDATA3= 000010	789#													
RDATA4= 000020	788#													
RDATA5= 000040	787#													
RDATA6= 000100	786#													
RDATA7= 000200	785#													
ROCHR = 104410	4069	4454#												
ROLIN = 104411	4455#													
RORRUN= 000001	772#													
REC = 010040	3142	3245#												
REG = 000004	3424#	3448												
REGSAV 010626	3448#	3461	3517#											
RESVEC= 000010	727#													
RHLD 010130	3253#	3254#	3258	3266	3295#									
R0050 005676	2456#													
R0070 005677	2457#													
R0110 005700	2458#													
R0135 005701	2459#													
R0150 005702	2460#													
R0200 005705	2463#													
R0300 005703	2461#													
R0600 005704	2462#													
R10000 005715	2471#													
R1800 005706	2464#													
R2000 005707	2465#													
R2400 005710	2466#													
R3600 005711	2467#													
R4800 005712	2468#													
R5STAC 001334	1015#	1124												
R7200 005713	2469#													
R9600 005714	2470#													
S8 007774	3205#	3266#												
SET = 177777	750#	1703	1734	1745	1761	1770	1777	1781	1830	1904	1989	1998	2519	
	2674	2866	2885	3020	3041	3469								
SETCLR= 000006	3424#	3475												
SPECIA 011074	3670#	3827												
STACK = 001100	631#	1023												
START 001336	864	1016#												
STKLMT= 177774	642#													
SWR 001140	925#	1021	1043#	1045	1051#	1058#	1072	3844	3857#	3966	4003#	4171	4178	

MACINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 P E 110
DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

		4190	4194	4222	4236	4238	4244	4251							
SWREG	000176	862#	1051	1072	3966	3979									
SW0	= 000001	695#													
SW00	= 000001	685#	695												
SW01	= 000002	684#	694												
SW02	= 000004	683#	693												
SW03	= 000010	682#	692												
SW04	= 000020	681#	691												
SW05	= 000040	680#	690												
SW06	= 000100	679#	689												
SW07	= 000200	678#	688												
SW08	= 000400	677#	687												
SW09	= 001000	676#	686												
SW1	= 000002	694#													
SW10	= 002000	675#													
SW11	= 004000	674#													
SW12	= 010000	673#													
SW13	= 020000	672#													
SW14	= 040000	671#													
SW15	= 100000	670#													
SW2	= 000004	693#													
SW3	= 000010	692#													
SW4	= 000020	691#													
SW5	= 000040	690#													
SW6	= 000100	689#													
SW7	= 000200	688#													
SW8	= 000400	687#													
SW9	= 001000	686#													
TBITVE=	000014	728#													
TBUF	001270	1012#	1121#	1122#	1697#	1725#	1823#	1898#	1982#	2144#	2154#	2236#	2356#	2668#	
		2739#	2775#	2881#	3037#	3220#	3337#								
TCSR	001264	1010#	1115#	1116#	1212	1225#	1228	1242#	1245	1259#	1264	1297	1310#	1313	
		1327#	1330	1344#	1349	1379	1392#	1395	1409#	1412	1426#	1431	1610#	1643	
		1704	1730	1746	1816#	1843#	1890#	1916#	1921#	1939#	1944#	1972#	2010#	2039#	
		2061#	2083#	2132#	2173#	2195#	2219#	2250#	2268#	2336#	2435#	2520	2528#	2538#	
		2578#	2655#	2675	2697#	2717#	2729#	2751#	2756#	2764#	2797#	2837#	2867	2919#	
		2931#	2979#	3021	3169#	3173#	3188#	3230#	3331#	3334#					
TCSRHI	001266	1011#	1118#	1119#	2350#	2431#									
TDATA0=	000001	834#													
TDATA1=	000002	833#													
TDATA2=	000004	832#													
TDATA3=	000010	831#													
TDATA4=	000020	830#													
TDATA5=	000040	829#													
TDATA6=	000100	828#													
TDATA7=	000200	827#													
TIME	= 000000	3580#	3591												
TIMER	010462	1707	1749	1834	1908	2523	2678	2870	2889	3024	3045	3423#			
TIMSAV	010630	3450#	3486	3497#	3518#										
TKVEC	= 000060	735#													

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 112
DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

H09

MAINDEC-11-DVDVC-A
DVDVCA.P11MACY11 27(1006)
08-AUG-77 09:1608-AUG-77 09:20 PAGE 113
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0111

		3510#	3511	3514#	3516	3545#	3546	3547#	3551	3553#	3554	3556	3560	3561#
		3562	3563#	3564	3568#	3569	3591#	3592	3593#	3594	3596	3600	3601#	3602
		3604	3608	3634#	3635	3651#	3652	3653#	3654	3663#	3664	3668#	3672#	3673
		3676#	3677	3678#	3679	3680#	3681	3682#	3683	3687#	3688	3689#	3690	3691#
		3692	3693#	3694	3703#	3704	3705#	3706	3707#	3708	3719#	3720	3728#	3729
		3730#	3731											
		1027	4166#											
		915#	3774	4175#	4176#	4177	4206							
		1004#												
		912#	3414	3416#	4174#	4206								
		936#	1035#	4197	4199	4206	4264#							
		955#												
		897	988#											
		948#	3765#	3766	4135#									
		4098#	4101#	4129	4138#	4146#								
		933#	3925	3956										
		932#	3956											
		1#	1085	1160	1201	1214	1230	1247	1266	1286	1299	1315	1332	1351
		1381	1397	1414	1433	1459	1475	1492	1511	1542	1573	1580	1587	1607
		1645	1675	1732	1763	1766	1805	1861	1879	1936	1960	1988	2006	2028
		2032	2035	2058	2080	2108	2113	2170	2191	2215	2247	2265	2306	2311
		2366	2369	2390	2398	2439	2551	2555	2593	2633	2694	2701	2749	2786
		2824	2916	2935	2942	2960	2970	3073	3108	3193	3228	3260	3264	3284
		3318	3349	3358	3361	3376	3383	3463	3477	3509	3659	3662	3702	3714
		3717												
		1#	1085	1160	1201	1214	1230	1247	1266	1286	1299	1315	1332	1351
		1381	1397	1414	1433	1459	1475	1492	1511	1542	1573	1580	1587	1607
		1645	1675	1732	1763	1766	1808	1861	1882	1936	1962	1988	2006	2028
		2032	2035	2058	2080	2110	2113	2170	2191	2215	2247	2265	2308	2311
		2366	2369	2390	2400	2439	2551	2555	2593	2636	2694	2701	2749	2786
		2827	2916	2935	2942	2963	2970	3073	3111	3193	3228	3260	3264	3284
		3321	3349	3358	3361	3376	3383	3463	3477	3509	3659	3662	3702	3714
		3717												
		1#	1082	1083	1158	1199	1212	1228	1245	1264	1284	1297	1313	1330
		1349	1379	1395	1412	1431	1457	1473	1490	1509	1540	1571	1578	1585
		1605	1643	1673	1712	1730	1752	1761	1764	1803	1839	1859	1877	1912
		1934	1958	1986	2004	2025	2026	2030	2033	2056	2078	2106	2111	2168
		2189	2213	2245	2263	2304	2309	2363	2364	2367	2388	2396	2437	2549
		2553	2591	2631	2692	2699	2747	2784	2822	2873	2892	2914	2933	2940
		2958	2968	3027	3048	3071	3106	3191	3226	3258	3262	3282	3316	3347
		3356	3359	3374	3381	3461	3475	3507	3657	3660	3700	3712	3715	
		1#	1158	1164	1199	1205	1212	1218	1228	1234	1245	1251	1264	1270
		1284	1290	1297	1303	1313	1319	1330	1336	1349	1355	1379	1385	1395
		1401	1412	1418	1431	1437	1457	1463	1473	1479	1490	1496	1509	1515
		1540	1551	1571	1575	1578	1582	1585	1589	1595	1597	1599	1605	1622
		1643	1654	1673	1679	1712	1717	1730	1738	1752	1757	1759	1761	1764
		1772	1774	1779	1803	1805	1812	1839	1847	1859	1865	1877	1879	1886
		1912	1923	1934	1946	1958	1960	1964	1970	1986	1991	1996	2004	2017
		2030	2033	2047	2049	2056	2068	2078	2087	2106	2108	2111	2115	2121
		2123	2129	2168	2182	2189	2205	2213	2227	2245	2260	2263	2273	2304
		2306	2309	2313	2319	2321	2327	2367	2373	2381	2388	2392	2396	2398
		2402	2411	2413	2437	2443	2549	2553	2559	2566	2568	2591	2597	2631
		2633	2638	2692	2699	2706	2712	2714	2747	2754	2784	2790	2822	2824

\$ERROR 013274
 \$ERRPC 001116
 \$ERRTB 001254
 \$ERTTL 001112
 \$ESCAP 001162
 \$ETABL 001214
 \$ETEND 001254
 \$FATAL 001176
 \$FFLG 013272
 \$FILLC 001156
 \$FILLS 001155
 \$FSAND= 000310

\$FSBAO= 000401

\$FSBLA= 000170
 \$FSRAS= 000150
 \$FSDEC= 000220
 \$FSG00= 000400

\$FSIF = 000110

MAINDEC-11-DVDC-A
 DVDVCA.P11

 MACY11 27(1006)
 08-AUG-77 09:16

 08-AUG-77 09:20 PAGE 114
 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0112

	2831	2873	2877	2892	2896	2914	2924	2933	2938	2940	2945	2958	2960
	2968	2974	2976	3027	3033	3048	3056	3071	3078	3106	3108	3115	3191
	3198	3226	3232	3258	3262	3270	3275	3282	3290	3316	3318	3325	3347
	3353	3356	3359	3364	3366	3368	3374	3379	3381	3386	3461	3466	3471
	3475	3481	3507	3512	3557	3660	3665	3674	3684	3695	3700	3712	3715
	3721	3723	3725										
SF\$INC= 000210	1#	2339	2421	2853	2926	3006	3081	3553	3565	3593	3601	3609	3612
SF\$LOO= 000200	1#	1684	1781	1784	3458	3484	3499						
SF\$NAM= 000160	1#												
SF\$NO = 000403	1#	1083	1158	1199	1212	1228	1245	1264	1284	1297	1313	1330	1349
	1379	1395	1412	1431	1457	1473	1490	1509	1540	1571	1578	1585	1605
	1643	1730	1761	1764	1803	1859	1877	1934	1960	1986	2004	2026	2030
	2033	2056	2078	2106	2111	2168	2189	2213	2245	2263	2304	2306	2364
	2367	2388	2396	2398	2437	2549	2553	2591	2631	2692	2699	2747	2784
	2822	2914	2933	2940	2958	2960	3071	3106	3191	3226	3258	3262	3282
	3316	3318	3356	3359	3374	3381	3461	3657	3660	3700	3712	3715	
SF\$OR = 000320	1#	1085	1160	1201	1214	1230	1247	1266	1286	1299	1315	1332	1351
	1381	1397	1414	1433	1459	1475	1492	1511	1542	1573	1580	1587	1607
	1645	1675	1732	1763	1766	1805	1861	1879	1936	1960	1962	1988	2006
	2028	2032	2035	2058	2080	2108	2110	2113	2170	2191	2215	2247	2265
	2306	2308	2311	2366	2369	2390	2398	2400	2439	2551	2555	2593	2633
	2694	2701	2749	2786	2824	2916	2935	2942	2960	2970	3073	3108	3193
	3228	3260	3264	3284	3318	3349	3358	3361	3376	3383	3463	3477	3509
	3659	3662	3702	3714	3717								
SF\$RTN= 000300	1#	3424	3523	3531	3573	3580	3618	3640	3746	3754	3790		
SF\$SEL= 000140	1#												
SF\$UNT= 000130	1#	1148	1172	1984	1998	3182	3184	3340	3343	3655	3697		
SF\$WHI= 000120	1#	1082	1083	1090	1805	1879	1900	2025	2026	2051	2108	2306	2363
	2364	2384	2398	2633	2824	2960	3108	3318					
SF\$YES= 000402	1#	1083	1158	1199	1212	1228	1245	1264	1284	1297	1313	1330	1349
	1379	1395	1412	1431	1457	1473	1490	1509	1540	1571	1578	1585	1605
	1643	1673	1730	1761	1764	1803	1805	1859	1877	1879	1934	1958	1960
	1986	2004	2026	2030	2033	2056	2078	2106	2108	2111	2168	2189	2213
	2245	2263	2304	2306	2309	2364	2367	2388	2396	2398	2437	2549	2553
	2591	2631	2633	2692	2699	2747	2784	2822	2824	2914	2933	2940	2958
	2960	2968	3071	3106	3108	3191	3226	3258	3262	3282	3316	3318	3347
	3356	3359	3374	3381	3461	3475	3507	3657	3660	3700	3712	3715	
SGDADR 001120	916#												
SGDOAT 001124	918#												
SGET42 011542	3818#												
SGTSMR 012322	3978#	4451											
\$HO = 000000	616												
\$HIBTS 001000	892#												
\$ICNT 001104	909#	4255#	4256	4258#	4269								
\$IFLEV= 177777	1#	1158#	1164#	1199#	1205#	1212#	1218#	1228#	1234#	1245#	1251#	1264#	1270#
	1284#	1290#	1297#	1303#	1313#	1319#	1330#	1336#	1349#	1355#	1379#	1385#	1395#
	1401#	1412#	1418#	1431#	1437#	1457#	1463#	1473#	1479#	1490#	1496#	1509#	1515#
	1540#	1551#	1571#	1578#	1585#	1595#	1597#	1599#	1605#	1622#	1643#	1654#	1673#
	1679#	1712#	1717#	1730#	1752#	1757#	1759#	1761#	1764#	1772#	1779#	1803#	1812#
	1839#	1847#	1859#	1865#	1877#	1886#	1912#	1923#	1934#	1946#	1958#	1970#	1986#
	1996#	2004#	2017#	2030#	2033#	2047#	2049#	2056#	2068#	2078#	2087#	2106#	2111#
	2121#	2129#	2168#	2182#	2189#	2205#	2213#	2227#	2245#	2260#	2263#	2273#	2304#
	2309#	2319#	2327#	2367#	2381#	2388#	2396#	2411#	2413#	2437#	2443#	2549#	2553#
	2566#	2568#	2591#	2597#	2631#	2638#	2692#	2699#	2712#	2714#	2747#	2754#	2784#
	2790#	2822#	2831#	2873#	2877#	2892#	2896#	2914#	2924#	2933#	2938#	2940#	2945#
	2958#	2968#	2974#	2976#	3027#	3033#	3048#	3056#	3071#	3078#	3106#	3115#	3191#

MAINDEC-11-DVDVC-A
DVDVCA.P11

MACY11 27(1006)
08-AUG-77 09:16

08-AUG-77 09:20 PAGE 115
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0113

\$ILLUP 011752
\$INTAG 001135
\$ISKO = 000001

\$ISK1 = 000001

\$ISK2 = 000001
\$ITEMB 001114
\$LF 001172
\$LFLG 013271
\$LOCTA= 177777

3198#	3226#	3232#	3258#	3262#	3270#	3275#	3282#	3290#	3316#	3325#	3347#	3356#
3359#	3364#	3366#	3368#	3374#	3379#	3381#	3386#	3461#	3471#	3475#	3481#	3507#
3512#	3557#	3660#	3674#	3695#	3700#	3712#	3715#	3721#	3723#	3725#		
3836#	3852#	3871#										
923#	4006#	4095#										
1158#	1164#	1199#	1205#	1212#	1218#	1228#	1234#	1245#	1251#	1264#	1270#	1284#
1290#	1297#	1303#	1313#	1319#	1330#	1336#	1349#	1355#	1379#	1385#	1395#	1401#
1412#	1418#	1431#	1437#	1457#	1463#	1473#	1479#	1490#	1496#	1509#	1515#	1540#
1551#	1571#	1599#	1605#	1622#	1643#	1654#	1673#	1679#	1712#	1717#	1730#	1759#
1761#	1779#	1803#	1812#	1839#	1847#	1859#	1865#	1877#	1886#	1912#	1923#	1934#
1946#	1958#	1970#	1986#	1996#	2004#	2017#	2030#	2049#	2056#	2068#	2078#	2087#
2106#	2129#	2168#	2182#	2189#	2205#	2213#	2227#	2245#	2260#	2263#	2273#	2304#
2327#	2367#	2381#	2388#	2413#	2437#	2443#	2549#	2568#	2591#	2597#	2631#	2638#
2692#	2714#	2747#	2754#	2784#	2790#	2822#	2831#	2873#	2877#	2892#	2896#	2914#
2924#	2933#	2938#	2940#	2945#	2958#	2976#	3027#	3033#	3048#	3056#	3071#	3078#
3106#	3115#	3191#	3198#	3226#	3232#	3258#	3275#	3282#	3290#	3316#	3325#	3347#
3368#	3374#	3379#	3381#	3386#	3461#	3471#	3475#	3481#	3507#	3512#	3657#	3695#
3700#	3725#											
1578#	1597#	1752#	1757#	1764#	1772#	2033#	2047#	2111#	2121#	2309#	2319#	2396#
2411#	2553#	2566#	2699#	2712#	2968#	2974#	3262#	3270#	3356#	3366#	3660#	3674#
3712#	3723#											
1585#	1595#	3359#	3364#	3715#	3721#							
913#	3765#	4177#	4185#	4206#								
940#	3956#	4080#	4089#	4206#								
4139#	4145#											
1#	1082#	1083#	1084#	1085#	1090#	1091#	1092#	1148#	1149#	1159#	1160#	1164#
1165#	1173#	1174#	1200#	1201#	1205#	1206#	1213#	1214#	1218#	1219#	1229#	1230#
1234#	1235#	1246#	1247#	1251#	1252#	1265#	1266#	1270#	1271#	1285#	1286#	1290#
1291#	1298#	1299#	1303#	1304#	1314#	1315#	1319#	1320#	1331#	1332#	1336#	1337#
1350#	1351#	1355#	1356#	1380#	1381#	1385#	1386#	1396#	1397#	1401#	1402#	1413#
1414#	1418#	1419#	1432#	1433#	1437#	1438#	1458#	1459#	1463#	1464#	1474#	1475#
1479#	1480#	1491#	1492#	1496#	1497#	1510#	1511#	1515#	1516#	1541#	1542#	1551#
1552#	1572#	1573#	1575#	1576#	1577#	1579#	1580#	1582#	1583#	1584#	1586#	1587#
1589#	1590#	1591#	1595#	1596#	1597#	1598#	1599#	1600#	1606#	1607#	1622#	1623#
1644#	1645#	1654#	1655#	1674#	1675#	1679#	1680#	1684#	1685#	1712#	1713#	1717#
1718#	1731#	1732#	1738#	1739#	1740#	1752#	1753#	1757#	1758#	1759#	1760#	1762#
1763#	1765#	1766#	1772#	1773#	1774#	1775#	1776#	1779#	1780#	1782#	1783#	1784#
1785#	1786#	1804#	1805#	1806#	1807#	1808#	1812#	1813#	1839#	1840#	1847#	1848#
1860#	1861#	1865#	1866#	1878#	1879#	1880#	1881#	1882#	1886#	1887#	1912#	1913#
1923#	1924#	1935#	1936#	1946#	1947#	1959#	1960#	1961#	1962#	1964#	1965#	1966#
1970#	1971#	1984#	1985#	1987#	1988#	1991#	1992#	1993#	1996#	1997#	1999#	2000#
2001#	2002#	2003#	2005#	2006#	2017#	2018#	2025#	2026#	2027#	2028#	2031#	2032#
2034#	2035#	2047#	2048#	2049#	2050#	2051#	2052#	2053#	2057#	2058#	2068#	2069#
2079#	2080#	2087#	2088#	2107#	2108#	2109#	2110#	2112#	2113#	2115#	2116#	2117#
2121#	2122#	2123#	2124#	2125#	2129#	2130#	2169#	2170#	2182#	2183#	2190#	2191#
2205#	2206#	2214#	2215#	2227#	2228#	2246#	2247#	2260#	2261#	2264#	2265#	2273#
2274#	2305#	2306#	2307#	2308#	2310#	2311#	2313#	2314#	2315#	2319#	2320#	2321#
2322#	2323#	2327#	2328#	2340#	2341#	2342#	2343#	2344#	2345#	2346#	2363#	2364#
2365#	2366#	2368#	2369#	2373#	2374#	2375#	2381#	2382#	2384#	2385#	2386#	2389#
2390#	2392#	2393#	2394#	2397#	2398#	2399#	2400#	2402#	2403#	2404#	2411#	2412#
2413#	2414#	2421#	2422#	2423#	2438#	2439#	2443#	2444#	2550#	2551#	2554#	2555#
2559#	2560#	2561#	2566#	2567#	2568#	2569#	2592#	2593#	2597#	2598#	2632#	2633#
2634#	2635#	2636#	2638#	2639#	2693#	2694#	2700#	2701#	2706#	2707#	2708#	2712#
2713#	2714#	2715#	2748#	2749#	2754#	2755#	2785#	2786#	2790#	2791#	2823#	2824#
2825#	2826#	2827#	2831#	2832#	2854#	2855#	2856#	2857#	2858#	2859#	2860#	2873#
2874#	2877#	2878#	2892#	2893#	2896#	2897#	2915#	2916#	2924#	2925#	2926#	2927#

K09

M3INDEC-11-DVDVC-A
DVDVCA.P11MACY11 27(1006)
08-AUG-77 09:1608-AUG-77 09:20 PAGE 116
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0114

SLPADR 001106
SLPERR 001110

SLSTCN= 177777

SLSTIN= 000000

2928	2934	2935	2938	2939	2941	2942	2945	2946	2959	2960	2961	2962
2963	2969	2970	2974	2975	2976	2977	3007	3008	3009	3010	3011	3012
3013	3027	3028	3033	3034	3048	3049	3056	3057	3072	3073	3078	3079
3081	3082	3083	3107	3108	3109	3110	3111	3115	3116	3182	3183	3185
3186	3192	3193	3198	3199	3227	3228	3232	3233	3259	3260	3263	3264
3270	3271	3275	3276	3283	3284	3290	3291	3317	3318	3319	3320	3321
3325	3326	3340	3341	3344	3345	3348	3349	3353	3354	3355	3357	3358
3360	3361	3364	3365	3366	3367	3368	3369	3375	3376	3379	3380	3382
3383	3386	3387	3423	3458	3459	3462	3463	3466	3467	3468	3471	3472
3476	3477	3481	3482	3485	3486	3487	3488	3499	3500	3501	3508	3509
3510	3511	3512	3513	3515	3516	3523	3524	3525	3526	3530	3554	3555
3556	3557	3558	3559	3560	3565	3566	3567	3570	3571	3573	3574	3575
3579	3594	3595	3596	3597	3598	3599	3600	3602	3603	3604	3605	3606
3607	3608	3609	3610	3611	3612	3613	3614	3618	3619	3620	3639	3655
3656	3658	3659	3661	3662	3665	3666	3667	3674	3675	3684	3685	3686
3695	3696	3698	3699	3701	3702	3713	3714	3716	3717	3721	3722	3723
3724	3725	3726	3732	3733	3746	3747	3748	3753	3790	3791	3792	
910*	1037*	4246*	4262*	4267	4269							
911*	1038*	1150*	1209*	1223*	1239*	1256*	1294*	1308*	1324*	1341*	1376*	1390*
1406*	1423*	1454*	1468*	1484*	1501*	1538*	1602*	1640*	1819*	1852*	1892*	2137*
2187*	2209*	2231*	2515*	2572*	2650*	2727*	2762*	4196	4246	4263*	4269	
1#	1082	1083	1085	1090	1091	1148	1149	1158	1160	1164	1172	1199
1201	1205	1212	1214	1218	1228	1230	1234	1245	1247	1251	1264	1266
1270	1284	1286	1290	1297	1299	1303	1313	1315	1319	1330	1332	1336
1349	1351	1355	1379	1381	1385	1395	1397	1401	1412	1414	1418	1431
1433	1437	1457	1459	1463	1473	1475	1479	1490	1492	1496	1509	1511
1515	1540	1542	1551	1571	1573	1576	1577	1578	1580	1583	1584	1585
1587	1590	1591	1595	1597	1599	1605	1607	1622	1643	1645	1654	1673
1675	1679	1684	1685	1712	1713	1717	1730	1732	1739	1740	1752	1753
1757	1759	1761	1763	1764	1766	1772	1775	1776	1779	1784	1785	1803
1808	1812	1839	1840	1847	1859	1861	1865	1877	1882	1886	1912	1913
1923	1934	1936	1946	1958	1962	1965	1966	1970	1984	1985	1986	1988
1992	1993	1996	1998	2003	2004	2006	2017	2025	2026	2028	2030	2032
2033	2035	2047	2049	2051	2052	2056	2058	2068	2078	2080	2087	2106
2110	2111	2113	2116	2117	2121	2124	2125	2129	2168	2170	2182	2189
2191	2205	2213	2215	2227	2245	2247	2260	2263	2265	2273	2304	2308
2309	2311	2314	2315	2319	2322	2323	2327	2339	2341	2342	2343	2346
2363	2364	2366	2367	2369	2374	2375	2381	2384	2385	2388	2390	2393
2394	2396	2400	2403	2404	2411	2413	2421	2422	2437	2439	2443	2549
2551	2553	2555	2560	2561	2566	2568	2591	2593	2597	2631	2636	2638
2692	2694	2699	2701	2707	2708	2712	2714	2747	2749	2754	2784	2786
2790	2822	2827	2831	2853	2855	2856	2857	2860	2873	2874	2877	2892
2893	2896	2914	2916	2924	2926	2927	2933	2935	2938	2940	2942	2945
2958	2963	2968	2970	2974	2976	3006	3008	3009	3010	3013	3027	3028
3033	3048	3049	3056	3071	3073	3078	3081	3082	3106	3111	3115	3182
3183	3184	3191	3193	3198	3226	3228	3232	3258	3260	3262	3264	3270
3275	3282	3284	3290	3316	3321	3325	3340	3341	3343	3347	3349	3354
3355	3356	3358	3359	3361	3364	3366	3368	3374	3376	3379	3381	3383
3386	3424	3458	3459	3461	3463	3467	3468	3471	3475	3477	3481	3499
3500	3507	3509	3512	3523	3531	3553	3555	3556	3557	3560	3565	3566
3573	3580	3593	3595	3596	3597	3600	3601	3603	3604	3605	3608	3609
3610	3612	3613	3618	3640	3655	3656	3657	3659	3660	3662	3666	3667
3674	3685	3686	3695	3697	3700	3702	3712	3714	3715	3717	3721	3723
3725	3746	3754	3790									
1#	1083	1084	1085	1090	1091	1093	1094	1095	1096	1098	1099	1102
1103	1105	1106	1107	1108	1110	1111	1112	1113	1114	1115	1116	1117

1118	1119	1120	1121	1122	1123	1124	1125	1137	1138	1140	1141	1142
1143	1144	1145	1146	1147	1150	1151	1153	1154	1158	1159	1160	1167
1168	1169	1170	1171	1172	1173	1174	1177	1178	1179	1180	1181	1182
1199	1200	1201	1202	1203	1209	1210	1212	1213	1214	1223	1224	1225
1226	1228	1229	1230	1239	1240	1242	1243	1245	1246	1247	1256	1257
1259	1260	1264	1265	1266	1284	1285	1286	1287	1288	1294	1295	1297
1298	1299	1308	1309	1310	1311	1313	1314	1315	1324	1325	1327	1328
1330	1331	1332	1341	1342	1344	1345	1349	1350	1351	1376	1377	1379
1380	1381	1390	1391	1392	1393	1395	1396	1397	1406	1407	1409	1410
1412	1413	1414	1423	1424	1426	1427	1431	1432	1433	1454	1455	1457
1458	1459	1468	1469	1470	1471	1473	1474	1475	1484	1485	1487	1488
1490	1491	1492	1501	1502	1504	1505	1509	1510	1511	1538	1539	1540
1541	1542	1571	1572	1573	1575	1576	1578	1579	1580	1582	1583	1585
1586	1587	1589	1590	1592	1593	1602	1603	1605	1606	1607	1610	1611
1640	1641	1643	1644	1645	1673	1674	1675	1676	1677	1682	1683	1687
1688	1689	1690	1697	1698	1702	1703	1704	1705	1706	1707	1708	1709
1712	1713	1725	1726	1730	1731	1732	1734	1735	1738	1739	1744	1745
1746	1747	1748	1749	1750	1751	1752	1753	1761	1762	1763	1764	1765
1766	1770	1771	1774	1775	1777	1778	1781	1782	1783	1784	1785	1803
1804	1805	1806	1807	1809	1810	1816	1817	1819	1820	1823	1824	1829
1830	1831	1832	1833	1834	1835	1836	1839	1840	1843	1844	1852	1853
1859	1860	1861	1877	1878	1879	1880	1881	1883	1884	1890	1891	1892
1893	1898	1899	1903	1904	1905	1906	1907	1908	1909	1910	1912	1913
1916	1917	1921	1922	1931	1932	1934	1935	1936	1939	1940	1944	1945
1958	1959	1960	1961	1962	1964	1965	1967	1968	1972	1973	1974	1975
1976	1977	1982	1983	1986	1987	1988	1989	1990	1991	1992	1994	1995
1998	1999	2000	2001	2002	2004	2005	2006	2010	2011	2014	2015	2026
2027	2028	2030	2031	2032	2033	2034	2035	2039	2040	2044	2045	2051
2052	2056	2057	2058	2061	2062	2065	2066	2075	2076	2078	2079	2080
2083	2084	2106	2107	2108	2109	2110	2111	2112	2113	2115	2116	2118
2119	2123	2124	2126	2127	2132	2133	2137	2138	2144	2145	2147	2148
2149	2150	2151	2154	2155	2157	2158	2159	2160	2161	2164	2165	2168
2169	2170	2173	2174	2179	2180	2187	2188	2189	2190	2191	2195	2196
2202	2203	2209	2210	2213	2214	2215	2219	2220	2224	2225	2231	2232
2236	2237	2239	2240	2241	2242	2243	2245	2246	2247	2250	2251	2257
2258	2263	2264	2265	2268	2269	2304	2305	2306	2307	2308	2309	2310
2311	2313	2314	2316	2317	2321	2322	2324	2325	2330	2331	2332	2333
2334	2335	2336	2337	2339	2340	2341	2342	2343	2344	2345	2346	2347
2348	2350	2351	2353	2354	2356	2357	2359	2360	2361	2362	2364	2365
2366	2367	2368	2369	2371	2372	2373	2374	2377	2378	2379	2380	2384
2385	2388	2389	2390	2392	2393	2396	2397	2398	2399	2400	2402	2403
2409	2410	2416	2417	2418	2419	2421	2422	2424	2425	2426	2427	2428
2429	2430	2431	2432	2435	2436	2437	2438	2439	2499	2500	2503	2504
2506	2507	2509	2510	2511	2512	2513	2514	2515	2516	2518	2519	2520
2521	2522	2523	2524	2525	2528	2529	2538	2539	2542	2543	2544	2545
2546	2549	2550	2551	2553	2554	2555	2559	2560	2572	2573	2575	2576
2578	2579	2586	2587	2588	2589	2590	2591	2592	2593	2605	2606	2607
2608	2609	2610	2631	2632	2633	2634	2635	2643	2644	2645	2646	2647
2648	2650	2651	2652	2653	2655	2656	2658	2659	2668	2669	2673	2674
2675	2676	2677	2678	2679	2680	2682	2683	2685	2686	2687	2688	2689
2692	2693	2694	2697	2698	2699	2700	2701	2706	2707	2717	2718	2727
2728	2729	2730	2733	2734	2736	2737	2739	2740	2742	2743	2744	2745
2746	2747	2748	2749	2751	2752	2756	2757	2762	2763	2764	2765	2766
2767	2773	2774	2775	2776	2778	2779	2780	2781	2782	2784	2785	2786
2797	2798	2801	2802	2805	2806	2807	2808	2809	2810	2822	2823	2824
2825	2826	2828	2829	2834	2835	2837	2838	2847	2848	2849	2850	2853

M3INDOC-.-DVDVC-A
DVDVCA.P11MACY11 27(1006)
08-AUG-77 09:1608-AUG-77 09:20 PAGE 118
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0116

2854	2855	2856	2857	2858	2859	2860	2865	2866	2867	2868	2869	2870
2871	2872	2873	2874	2875	2876	2881	2882	2884	2885	2886	2887	2888
2889	2890	2891	2892	2893	2894	2895	2899	2900	2907	2908	2909	2910
2911	2914	2915	2916	2919	2920	2926	2927	2931	2932	2933	2934	2935
2940	2941	2942	2958	2959	2960	2961	2962	2965	2966	2968	2969	2970
2971	2972	2979	2980	2990	2991	2999	3000	3001	3002	3003	3004	3006
3007	3008	3009	3010	3011	3012	3013	3019	3020	3021	3022	3023	3024
3025	3026	3027	3028	3030	3031	3037	3038	3040	3041	3042	3043	3044
3045	3046	3047	3048	3049	3053	3054	3060	3061	3064	3065	3066	3067
3068	3071	3072	3073	3081	3082	3087	3088	3106	3107	3108	3109	3110
3112	3113	3119	3120	3121	3122	3139	3140	3142	3143	3144	3145	3148
3149	3150	3151	3154	3155	3157	3158	3160	3161	3163	3164	3169	3170
3173	3174	3176	3177	3184	3185	3186	3188	3189	3191	3192	3193	3216
3217	3218	3220	3221	3223	3224	3226	3227	3228	3230	3231	3249	3250
3251	3253	3254	3255	3258	3259	3260	3262	3263	3264	3266	3267	3268
3269	3273	3274	3279	3280	3282	3283	3284	3286	3287	3316	3317	3318
3319	3320	3322	3323	3328	3329	3331	3332	3334	3335	3337	3338	3343
3344	3345	3347	3348	3349	3351	3352	3353	3354	3356	3357	3358	3359
3360	3361	3362	3363	3374	3375	3376	3381	3382	3383	3448	3449	3450
3451	3452	3453	3461	3462	3463	3464	3465	3466	3467	3469	3470	3475
3476	3477	3479	3480	3484	3485	3486	3487	3488	3492	3493	3494	3495
3496	3497	3498	3499	3500	3507	3508	3509	3510	3511	3514	3515	3516
3524	3525	3526	3527	3545	3546	3547	3548	3549	3550	3551	3553	3554
3555	3556	3557	3558	3559	3560	3561	3562	3563	3564	3565	3566	3568
3569	3570	3571	3575	3576	3591	3592	3593	3594	3595	3596	3597	3598
3599	3600	3601	3602	3603	3604	3605	3606	3607	3608	3609	3610	3612
3613	3620	3621	3634	3635	3651	3652	3653	3654	3657	3658	3659	3660
3661	3662	3663	3664	3665	3666	3668	3669	3672	3673	3676	3677	3678
3679	3680	3681	3682	3683	3684	3685	3687	3688	3689	3690	3691	3692
3693	3694	3697	3698	3699	3700	3701	3702	3703	3704	3705	3706	3707
3708	3712	3713	3714	3715	3716	3717	3719	3720	3728	3729	3730	3731
3732	3733	3748	3749	3792	3793							
1201	1205	1212	1214	1218	1228	1230	1234	1245	1247	1251	1264	1266
1270	1284	1286	1290	1297	1299	1303	1313	1315	1319	1330	1332	1336
1349	1351	1355	1379	1381	1385	1395	1397	1401	1412	1414	1418	1431
1433	1437	1457	1459	1463	1473	1475	1479	1490	1492	1496	1509	1511
1515	1540	1542	1551	1571	1573	1575	1576	1577	1578	1580	1582	1583
1584	1585	1587	1589	1590	1591	1595	1597	1599	1605	1607	1622	1643
1645	1654	1673	1675	1679	1684	1685	1712	1713	1717	1730	1732	1738
1739	1740	1752	1753	1757	1759	1761	1763	1764	1766	1772	1774	1775
1776	1779	1781	1784	1785	1803	1808	1812	1839	1840	1847	1859	1861
1865	1877	1882	1886	1912	1913	1923	1934	1936	1946	1958	1962	1964
1965	1966	1970	1984	1985	1986	1988	1991	1992	1993	1996	1998	2004
2006	2017	2025	2026	2028	2030	2032	2033	2035	2047	2049	2051	2052
2056	2058	2068	2078	2080	2087	2106	2110	2111	2113	2115	2116	2117
2121	2123	2124	2125	2129	2168	2170	2182	2189	2191	2205	2213	2215
2227	2245	2247	2260	2263	2265	2273	2304	2308	2309	2311	2313	2314
2315	2319	2321	2322	2323	2327	2339	2341	2342	2343	2346	2363	2364
2366	2367	2369	2373	2374	2375	2381	2384	2385	2388	2390	2392	2393
2394	2396	2400	2402	2403	2404	2411	2413	2421	2422	2437	2439	2443
2549	2551	2553	2555	2559	2560	2561	2566	2568	2591	2593	2597	2631
2636	2638	2692	2694	2699	2701	2706	2707	2708	2712	2714	2747	2749
2754	2784	2786	2790	2822	2827	2831	2853	2855	2856	2857	2860	2873
2874	2877	2892	2893	2896	2914	2916	2924	2926	2927	2933	2935	2938
2940	2942	2945	2958	2963	2968	2970	2974	2976	3006	3008	3009	3010

SLSTST= 177777


```

SMTYP1      001225
SMTYP2      001231
SMTYP3      001235
SMTYP4      001241
SMXCNT      013752
SNESTL=     177777

```

SNSK0 = 000300

SNSK1 = 000110

SMSK2 = 000110

SNSK.3 = 000110

SNUL - 001154

SNWTST= 000001

SOCNT 014422

SOMODE 014424

SOVER 013736

SPASS 001202

SPASTM 001006

0110 0000 0000 0000

9680												
9760												
9790												
9820												
4259	42690											
10	10820	10900	11480	11580	11640	11720	11990	12050	12120	12180	12280	12340
12450	12510	12640	12700	12840	12900	12970	13030	13130	13190	13300	13360	13490
13550	13790	13850	13950	14010	14120	14180	14310	14370	14570	14630	14730	14790
14900	14960	15090	15150	15400	15510	15710	15750	15780	15820	15850	15890	15950
15970	15990	16050	16220	16430	16540	16730	16790	16840	17120	17170	17300	17380
17520	17570	17590	17610	17640	17720	17740	17790	17810	17840	18030	18120	18390
18470	18540	18650	18770	18860	19120	19230	19340	19460	19580	19640	19700	19840
19860	19910	19960	19980	20040	20170	20250	20300	20330	20470	20490	20710	20560
20680	20780	20870	21060	21110	21150	21210	21230	21290	21680	21820	21990	22050
22130	22270	22450	22600	22630	22730	23040	23090	23130	23190	23210	23370	23390
23630	23670	23730	23810	23840	23880	23920	23960	24020	24110	24130	24210	24370
24430	25490	25530	25590	25660	25680	25910	25970	26310	26380	26920	26990	27060
27120	27140	27470	27540	27840	27900	28220	28310	28530	28730	28770	28920	28960
29140	29240	29260	29330	29380	29400	29450	29580	29680	29740	29760	30060	30270
30330	30480	30560	30710	30780	30810	31060	31150	31820	31840	31910	31980	32260
32320	32580	32620	32700	32750	32820	32900	33160	33250	33400	33430	33470	33530
33560	33590	33640	33660	33680	33740	33790	33810	33860	34240	34580	34610	34660
34710	34750	34810	34840	34990	35070	35120	35230	35310	35530	35650	35730	35800
35930	36010	36090	36120	36180	36400	36550	36570	36600	36650	36740	36840	36950
36970	37000	37120	37150	37210	37230	37250	37460	37540	37900			
10820	10900	11480	11720	11990	12050	12120	12180	12280	12340	12450	12510	12640
12700	12840	12900	12970	13030	13130	13190	13300	13360	13490	13550	13790	13850
13950	14010	14120	14180	14310	14370	14570	14630	14730	14790	14900	14960	15090
15150	15400	15510	15710	15750	15990	16050	16220	16430	16540	16730	16790	16840
17810	17840	18030	18120	18390	18470	18590	18650	18770	18860	19120	19230	19340
19460	19580	19640	19700	19840	19980	20040	20170	20250	20510	20560	20680	20780
20870	21060	21230	21290	21680	21820	21890	22050	22130	22270	22450	22600	22780
22730	23040	23210	23270	23390	24210	24370	24430	25490	25680	25910	25970	26310
26380	26920	27140	27470	27540								

D10

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 122
DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0120

STAGNU= 000250

1082	1083	1084	1085	1148	1149	1159	1160	1200	1201	1213	1214
1229	1230	1246	1247	1265	1266	1285	1286	1298	1299	1314	1315
1332	1350	1351	1380	1381	1396	1397	1413	1414	1432	1433	1458
1474	1475	1491	1492	1510	1511	1541	1542	1572	1573	1575	1577
1580	1582	1584	1586	1587	1589	1591	1606	1607	1644	1645	1674
1684	1685	1712	1713	1731	1732	1738	1740	1752	1753	1762	1763
1766	1774	1776	1804	1806	1807	1808	1839	1840	1860	1861	1878
1881	1882	1912	1913	1935	1936	1959	1961	1962	1964	1966	1984
1987	1988	1991	1993	1999	2002	2003	2005	2006	2025	2026	2027
2031	2032	2034	2035	2057	2058	2079	2080	2107	2109	2110	2112
2115	2117	2123	2125	2169	2170	2190	2191	2214	2215	2246	2247
2265	2305	2307	2308	2310	2311	2313	2315	2321	2323	2340	2341
2345	2346	2363	2364	2365	2366	2368	2369	2373	2375	2389	2390
2394	2397	2399	2400	2402	2404	2438	2439	2550	2551	2554	2555
2561	2592	2593	2632	2634	2635	2636	2693	2694	2700	2701	2706
2748	2749	2795	2786	2823	2825	2826	2827	2854	2855	2856	2859
2873	2874	2892	2893	2915	2916	2934	2935	2941	2942	2959	2961
2963	2969	2970	3007	3008	3009	3012	3013	3027	3028	3048	3049
3073	3107	3109	3110	3111	3182	3183	3192	3193	3227	3228	3259
3263	3264	3263	3284	3317	3319	3320	3321	3340	3341	3348	3349
3355	3357	3358	3360	3361	3375	3376	3382	3383	3424	3458	3459
3463	3466	3468	3476	3477	3508	3509	3531	3554	3555	3556	3559
3580	3594	3595	3596	3599	3600	3602	3603	3604	3607	3608	3640
3656	3658	3659	3661	3662	3665	3667	3684	3686	3701	3702	3713
3716	3717	3754									
1090	1091	1093	1094	1095	1096	1102	1103	1105	1106	1107	1108
1111	1112	1114	1115	1117	1118	1120	1121	1123	1124	1125	1137
1140	1141	1142	1143	1144	1145	1146	1147	1150	1151	1153	1154
1167	1168	1169	1171	1172	1173	1177	1178	1180	1181	1182	1202
1205	1209	1210	1218	1223	1224	1225	1226	1234	1239	1240	1242
1251	1256	1257	1259	1260	1270	1287	1288	1290	1294	1295	1303
1309	1310	1311	1319	1324	1325	1327	1328	1336	1341	1342	1344
1355	1376	1377	1385	1390	1391	1392	1393	1401	1406	1407	1409
1418	1427	1424	1426	1427	1437	1454	1455	1463	1468	1469	1470
1479	1484	1485	1487	1488	1496	1501	1502	1504	1505	1515	1538
1551	1575	1576	1582	1583	1584	1590	1592	1593	1595	1597	1599
1603	1610	1611	1622	1640	1641	1654	1676	1677	1679	1682	1683
1688	1689	1690	1697	1698	1717	1725	1726	1734	1735	1738	1739
1759	1770	1771	1772	1774	1775	1777	1778	1779	1781	1782	1784
1809	1810	1812	1816	1817	1819	1820	1823	1824	1843	1844	1847
1853	1865	1883	1884	1886	1890	1891	1892	1893	1898	1899	1916
1921	1922	1923	1931	1932	1939	1940	1944	1945	1946	1964	1965
1968	1970	1972	1973	1974	1975	1976	1977	1982	1983	1989	1990
1992	1994	1995	1996	1998	2001	2010	2011	2014	2015	2017	2039
2044	2045	2047	2049	2051	2052	2061	2062	2065	2066	2068	2075
2083	2084	2087	2115	2116	2118	2119	2121	2123	2124	2126	2127
2132	2133	2137	2138	2144	2145	2154	2155	2164	2165	2173	2174
2180	2182	2187	2188	2195	2196	2202	2203	2205	2209	2210	2219
2224	2225	2227	2231	2232	2236	2237	2250	2251	2257	2258	2260
2269	2273	2313	2314	2316	2317	2319	2321	2322	2324	2325	2327
2331	2332	2333	2334	2335	2336	2337	2343	2346	2347	2348	2350
2353	2354	2356	2357	2359	2360	2361	2362	2371	2372	2373	2374
2378	2379	2380	2381	2384	2385	2392	2393	2402	2403	2409	2410
2413	2416	2417	2418	2419	2421	2422	2424	2428	2429	2430	2431
2435	2436	2443	2499	2500	2503	2504	2506	2507	2509	2510	2511
2513	2514	2515	2516	2528	2529	2538	2539	2559	2560	2566	2568

STEMF = 000300

E10

MAINDEC-11-DVDVC-A MACY11 27('006) 08-AUG-77 09:20 PAGE 123
 DVDVC.A.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

SEG 0121

	2572	2575	2576	2578	2579	2597	2605	2606	2608	2609	2610	2638	2643
	2644	2645	2646	2647	2648	2650	2651	2652	2653	2655	2656	2658	2659
	2668	2669	2672	2693	2697	2698	2706	2707	2712	2714	2717	2718	2727
	2728	2729	2730	2733	2734	2736	2737	2739	2740	2751	2752	2754	2756
	2757	2762	2763	2764	2765	2766	2767	2773	2774	2775	2776	2790	2797
	2798	2801	2802	2805	2806	2808	2809	2810	2828	2829	2831	2834	2835
	2837	2838	2857	2860	2875	2876	2877	2881	2882	2894	2895	2896	2899
	2900	2907	2909	2910	2911	2919	2920	2924	2926	2927	2931	2932	2938
	2945	2965	2966	2971	2972	2974	2976	2979	2980	2990	2991	3003	3004
	3010	3013	3030	3031	3033	3037	3038	3053	3054	3056	3060	3061	3064
	3066	3067	3068	3078	3081	3082	3087	3088	3112	3113	3115	3139	3140
	3142	3143	3144	3145	3148	3149	3150	3151	3154	3155	3157	3158	3160
	3161	3163	3164	3169	3170	3173	3174	3176	3177	3184	3185	3188	3189
	3198	3216	3218	3220	3221	3223	3224	3230	3231	3232	3249	3251	3253
	3255	3266	3267	3268	3269	3270	3273	3274	3275	3279	3280	3286	3287
	3290	3322	3323	3325	3328	3329	3331	3332	3334	3335	3337	3338	3343
	3344	3351	3352	3353	3354	3362	3363	3364	3366	3368	3379	3386	3448
	3449	3450	3451	3452	3453	3464	3465	3466	3467	3469	3470	3471	3479
	3480	3481	3484	3485	3487	3497	3498	3499	3500	3512	3523	3545	3546
	3547	3551	3557	3560	3561	3562	3563	3564	3565	3566	3568	3569	3573
	3591	3592	3597	3600	3605	3608	3609	3610	3612	3613	3618	3634	3635
	3651	3652	3653	3654	3663	3664	3665	3666	3672	3673	3674	3676	3677
	3678	3679	3680	3681	3682	3683	3684	3685	3687	3688	3689	3690	3691
	3692	3693	3694	3695	3697	3698	3703	3704	3705	3706	3707	3708	3719
	3720	3721	3723	3725	3728	3729	3730	3731	3746	3790			
STESTN 001200	949	1135	1196	1281	1367	1451	1532	1565	1634	1670	1800	1874	1956
STIMES 001160	2103	2296	2496	2629	2820	2956	3103	3314	3759	4261			
	935	1034	1134	1195	1202	1280	1287	1366	1450	1531	1564	1592	1633
	1669	1676	1799	1809	1873	1883	1955	1967	2014	2044	2065	2102	2118
	2126	2179	2202	2224	2257	2295	2316	2324	2495	2628	2819	2828	2955
	2965	2971	3030	3053	3102	3112	3313	3322	3397	3805	4249	4256	4259
	4269												
STKB 001146	928	3959	3977	3987	4041	4047							
STKS 001144	927	3959	3968	3984	4008	4039	4045						
STN = 000026	1	616	1127	1134	1135	1189	1195	1196	1203	1276	1280	1281	1288
	1362	1366	1367	1444	1450	1451	1527	1531	1532	1560	1564	1565	1593
	1629	1633	1634	1663	1669	1670	1677	1787	1793	1799	1800	1810	1869
	1873	1874	1884	1949	1955	1956	1968	2015	2045	2066	2090	2098	2102
	2103	2119	2127	2180	2203	2225	2258	2276	2288	2295	2296	2317	2325
	2445	2486	2495	2496	2621	2628	2629	2814	2819	2820	2829	2951	2955
	2956	2966	2972	3031	3054	3097	3102	3103	3113	3201	3305	3313	3314
	3323	3388	3393	3397									
STPB 001152	930	3945	3956										
STPFLG 001157	934	3894	3956										
STPS 001150	929	3943	3956										
STRAP 014426	1029	4422											
STRAP2 014450	4433	4444											
STRP = 000012	4437	4446	4447	4448	4449	4450	4451	4452	4453	4454	4455	4456	
STRPQD 014462	4427	4444											
STSKD = 000243	1083	1090	1149	1172	1201	1205	1214	1218	1230	1234	1247	1251	1266
	1270	1286	1290	1299	1303	1315	1319	1332	1336	1351	1355	1381	1385
	1397	1401	1414	1418	1433	1437	1459	1463	1475	1479	1492	1496	1511
	1515	1542	1551	1573	1576	1577	1599	1607	1622	1645	1654	1675	1679
	1685	1784	1808	1812	1840	1847	1861	1865	1882	1886	1913	1923	1936
	1946	1962	1965	1966	1970	1985	1998	2006	2017	2026	2051	2058	2068
	2080	2087	2110	2124	2125	2129	2170	2182	2191	2205	2215	2227	2247

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 124
DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

S. 5X1 = 000244

STSK2 = 000245

STSK3 = 000232

STSK4 = 000122

STSTM 001004

BTSTNN 001102
BTSTNN 010012

STYVIN 012762
STYVIN 012762

STYPOS *****
STYPOS 013754

STUDS 013734
STYPE 011770

STYPEC 012202

STYPER 012250

STYPOC 014224
STYPOW 014340

STYPON 014290
STYPOS 014200

STYPOS 014200
SUNLY 001206

SUN: TH 001010

SUSAR 001220

SVECT: 001244

SVECT2 001246

EXTSTR 013506
EXTEND 000001

SYESNO= 000001

SSARGC= 000000

SSBYTE= 00G403

2260	2265	2273	2308	2322	2323	2327	2341	2343	2346	2422	2439	2443
2551	2568	2593	2597	2636	2638	2694	2714	2749	2754	2786	2790	2827
2831	2855	2857	2860	2927	2935	2938	2942	2945	2963	2976	3008	3010
3013	3082	3111	3115	3183	3184	3193	3198	3228	3232	3260	3275	3284
3290	3321	3325	3341	3343	3349	3354	3355	3368	3376	3379	3383	3386
3459	3499	3509	3512	3555	3557	3560	3566	3575	3597	3600	3613	3656
3697	3702	3725										
1085	1090	1160	1164	1580	1583	1584	1597	1685	1781	1784	1988	1992
1993	1996	2028	2051	2113	2116	2117	2121	2311	2314	2315	2319	2346
2421	2555	2560	2561	2566	2701	2707	2708	2712	2860	2926	2970	2974
3013	3081	3264	3270	3358	3366	3459	3484	3499	3560	3565	3600	3612
3659	3685	3686	3695	3714	3723							
1587	1590	1591	1595	1713	1717	1732	1739	1740	1759	1763	1775	1776
1779	2032	2049	2364	2384	2390	2393	2394	2413	2874	2877	2893	2896
2916	2924	3028	3033	3049	3056	3073	3078	3361	3364	3463	3467	3468
3471	3477	3481	3603	3605	3608	3610	3662	3666	3667	3674	3717	3721
1753	1757	1766	1772	2035	2047	2366	2384	2400	2403	2404	2411	3608
3609												
2369	2374	2375	2381									
894												
907	3804	4170	4206	4211	4238	4260	4261	4266	4270			
4066	4067	4084	4088									
4450												
4282	4449											
3894	4126	4437	4445									
3924	3931	3938	3943	3944	4010							
3949	3951	3954										
4367	4446											
4366	4369	4448										
4362	4447											
952												
896												
959	1199	1284	1578	1585	1803	1877	1960	2106	2111	2304	2306	2424
2631	2822	2958	2960	3106	3316	3318	3356	3547				
984	3682											
985												
4225												
1113	1114	1116	1117	1119	1120	1122	1123	1167	1168	1170	1171	1179
1180	1235	1226	1242	1243	1259	1260	1310	1311	1327	1328	1344	1345
1392	1393	1409	1410	1426	1427	1470	1471	1487	1488	1504	1505	1610
1611	1816	1817	1843	1844	1890	1891	1916	1917	1921	1922	1939	1940
1944	1945	1972	1973	1994	1995	2010	2011	2039	2040	2061	2062	2083
2084	2132	2133	2173	2174								

MAINDEC-11-DVDVC-A
DVDVCA.P11MACY11 27(1006)
08-AUG-77 09:1608-AUG-77 09:20 PAGE 125
CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0123

SSOST = 000067
SSFLAG = 000001

SSFROM = 000000

SSGET4 = 000000
SSLOC = 011226SSLOCN = 000000
SSRETU = 000000
SSRTN1 = 000246
SSRTN2 = 000247

1395	1412	1431	1457	1473	1490	1509	1540	1571	1578	1585	1605	1643
1673	1730	1761	1764	1803	1805	1859	1877	1879	1934	1958	1960	1986
2004	2026	2030	2033	2056	2078	2106	2108	2111	2168	2189	2213	2245
2263	2304	2306	2309	2364	2367	2389	2396	2398	2437	2549	2553	2591
2631	2633	2692	2699	2747	2784	2822	2824	2914	2933	2940	2958	2960
2968	3071	3106	3108	3191	3226	3258	3262	3282	3316	3318	3347	3356
3359	3374	3381	3461	3475	3507	3657	3660	3700	3712	3715		
2425	3548											
1082	1083	1085	1158	1160	1164	1199	1201	1205	1212	1214	1218	1228
1230	1234	1245	1247	1251	1264	1266	1270	1284	1286	1290	1297	1299
1303	1313	1315	1319	1330	1332	1336	1349	1351	1355	1379	1381	1385
1395	1397	1401	1412	1414	1418	1431	1433	1437	1457	1459	1463	1473
1475	1479	1490	1492	1496	1509	1511	1515	1540	1542	1551	1571	1573
1578	1580	1585	1587	1595	1597	1599	1605	1607	1622	1643	1645	1654
1673	1675	1679	1712	1717	1730	1732	1752	1757	1759	1761	1763	1764
1766	1772	1779	1803	1805	1808	1812	1839	1847	1859	1861	1865	1877
1879	1882	1886	1912	1923	1934	1936	1946	1958	1960	1962	1970	1986
1988	1996	2004	2006	2017	2025	2026	2028	2030	2032	2033	2035	2047
2049	2056	2058	2068	2078	2080	2087	2106	2108	2110	2111	2113	2121
2129	2168	2170	2182	2189	2191	2205	2213	2215	2227	2245	2247	2260
2263	2265	2273	2304	2306	2308	2309	2311	2319	2327	2363	2364	2366
2367	2369	2381	2388	2390	2396	2398	2400	2411	2413	2437	2439	2443
2549	2551	2553	2555	2565	2568	2591	2593	2597	2631	2633	2636	2638
2692	2694	2699	2701	2712	2714	2747	2749	2754	2784	2786	2790	2822
2824	2827	2831	2873	2877	2892	2896	2914	2916	2924	2933	2935	2938
2940	2942	2945	2958	2960	2963	2968	2970	2974	2976	3027	3033	3048
3056	3071	3073	3078	3106	3108	3111	3115	3191	3193	3198	3226	3228
3232	3258	3260	3262	3264	3270	3275	3282	3284	3290	3316	3318	3321
3325	3347	3349	3356	3358	3359	3361	3364	3366	3368	3374	3376	3379
3381	3383	3386	3461	3463	3471	3475	3477	3481	3507	3509	3512	3657
3659	3660	3662	3674	3695	3700	3702	3712	3714	3715	3717	3721	3723
3725												
1098	1702	1744	1829	1903	2147	2157	2239	2518	2542	2586	2673	2685
2742	2778	2847	2865	2884	2999	3019	3040	3119	3492	3668		
3820												
1084	1085	1159	1160	1173	1174	1200	1201	1213	1214	1229	1230	1246
1247	1265	1266	1285	1286	1298	1299	1314	1315	1331	1332	1350	1351
1380	1381	1396	1397	1413	1414	1432	1433	1458	1459	1474	1475	1491
1492	1510	1511	1541	1542	1572	1573	1579	1580	1586	1587	1606	1607
1644	1645	1674	1675	1712	1713	1731	1732	1752	1753	1762	1763	1765
1766	1806	1807	1839	1840	1860	1861	1880	1881	1912	1913	1935	1936
1959	1960	1961	1962	1987	1988	2001	2002	2005	2006	2027	2028	2031
2032	2034	2035	2057	2058	2079	2080	2107	2108	2109	2110	2112	2113
2169	2170	2190	2191	2214	2215	2246	2247	2264	2265	2305	2306	2307
2308	2310	2311	2365	2366	2368	2369	2389	2390	2397	2398	2399	2400
2438	2439	2550	2551	2554	2555	2592	2593	2634	2635	2693	2694	2700
2701	2748	2749	2785	2786	2825	2826	2873	2874	2892	2893	2915	2916
2934	2935	2941	2942	2961	2962	2969	2970	3027	3028	3048	3049	3072
3073	3109	3110	3185	3186	3192	3193	3227	3228	3259	3260	3263	3264
3283	3284	3319	3320	3344	3345	3348	3349	3357	3358	3360	3361	3375
3376	3382	3383	3462	3463	3476	3477	3508	3509	3658	3659	3661	3662
3698	3699	3701	3702	3713	3714	3716	3717					
3424	3526	3531	3575	3580	3620	3640	3748	3754	3792			
3424	3510	3514	3524	3531	3574	3580	3619	3640	3747	3754	3791	
3424	3510	3523	3531	3570	3573	3580	3618	3640	3732	3746	3754	3790
3424	3515	3525	3531	3574	3580	3619	3640	3747	3754	3791		

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 126
DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

```

SSSRC = 000027
SSIO  = 000000

```

2425#	3548#											
1098#	1099	1702#	1708	1744#	1750	1829#	1835	1903#	1909	2147#	2150	2157#
2160	2239#	2242	2518#	2524	2542#	2545	2586#	2589	2673#	2679	2685#	2688
2742#	2745	2778#	2781	2847#	2848	2849	2865#	2871	2884#	2890	2999#	3000
3001	3019#	3025	3040#	3046	3119#	3120	3121	3492#	3495	3668#	3669	

\$OFFILL	014423
\$1	001712
\$10	002552
\$100	004740
\$101	004750
\$102	004762
\$103	005070
\$104	005124
\$105	005162
\$106	005240
\$107	005260
\$11	002602
\$110	005410
\$111	005376
\$112	005406
\$113	005420
\$114	005454
\$115	005452
\$116	005626
\$117	005512
\$12	003640
\$120	005546
\$121	005534
\$122	005544
\$123	005560
\$124	005610
\$125	005602
\$126	005610
\$127	005674
\$13	002660
\$130	006154
\$131	006152
\$132	006154
\$133	006232
\$134	006320
\$135	006324
\$136	006514
\$137	006512
\$14	002706
\$140	006514
\$141	006606
\$142	006704
\$143	007004
\$144	007014
\$145	007060
\$146	007056
\$147	007222
\$15	002734
\$150	007126
\$151	007172
\$152	007220

[illegible]

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 127
 DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0125

\$153	007242	2934	2938#
\$154	007254	2941	2945#
\$155	007312	2959	2962#
\$156	007342	2961	2976#
\$157	007342	2969	2974#
\$16	002764	1350	1355#
\$160	007414	3007	3010#
\$161	007412	30 3#	3081
\$162	007560	3012	3082#
\$163	007466	3027	3033#
\$164	007536	3048	3056#
\$165	007556	3072	3078#
\$166	007624	3107	3110#
\$167	007634	3109	3115#
\$17	003034	1380	1385#
\$170	007742	3182#	3185
\$171	007766	3192	3198#
\$172	010032	3227	3232#
\$173	010110	3259	3275#
\$174	010104	3263	3270#
\$175	010126	3283	3290#
\$176	010172	3317	3320#
\$177	010202	3319	3325#
\$2	002122	1084	1091#
\$20	003062	1396	1401#
\$200	010230	3340#	3344
\$201	010256	3348	3354#
\$202	010304	3353	3368#
\$203	010304	3357	3366#
\$204	010304	3360	3364#
\$205	010320	3375	3379#
\$206	010332	3382	3386#
\$207	010634	3510	3523#
\$21	003110	1413	1418#
\$210	010636	3515	3525#
\$211	010504	3458#	3499
\$212	010610	3485	3487
\$213	010524	3462	3467#
\$214	010532	3466	3471#
\$215	010550	3476	3481#
\$216	010622	3508	3512#
\$217	010736	3570	3573#
\$22	003140	1432	1437#
\$220	010736	3574#	
\$221	010702	3554	3557#
\$222	010676	3555#	3565
\$223	010726	3559	3566#
\$224	011016	3618#	
\$225	011016	3619#	
\$226	010764	3594	3597#
\$227	010760	3595#	3612
\$23	003176	1458	1463#
\$230	011010	3599	3613#
\$231	010776	3602	3605#
\$232	010774	3603#	3609
\$233	011006	3607	3610#

3500#

J10

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 128
 DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0126

\$234	011272	3732	3746#
\$235	011272	3747#	
\$236	011044	3655#	3698
\$237	011130	3658	3685#
\$24	003224	1474	1479#
\$240	011070	3661	3666#
\$241	011076	3665	3674#
\$242	011150	3684	3695#
\$243	011236	3701	3725#
\$244	011236	3713	3723#
\$245	011236	3716	3721#
\$246	011462	3790#	
\$247	011462	3791#	
\$25	003252	1491	1496#
\$26	003302	1510	1515#
\$27	003342	1541	1551#
\$3	002320	1148#	1173
\$30	003372	1572	1576#
\$31	003426	1575	1599#
\$32	003404	1579	1583#
\$33	003426	1582	1597#
\$34	003416	1586	1590#
\$35	003426	1589	1595#
\$36	003456	1606	1622#
\$37	003516	1644	1654#
\$4	002344	1159	1164#
\$40	003554	1674	1679#
\$40CAT= ***** U		4180	4222
\$41	003562	1684#	1784
\$42	003772	1782	1785#
\$43	003636	1712	1717#
\$44	003664	1731	1739#
\$45	003720	1738	1759#
\$46	003720	1752	1757#
\$47	003752	1762	1775#
\$5	002456	1200	1205#
\$50	003750	1765	1772#
\$51	003760	1774	1779#
\$52	004040	1804	1807#
\$53	004050	1806	1812#
\$54	004132	1839	1847#
\$55	004154	1860	1865#
\$56	004212	1878	1881#
\$57	004222	1880	1886#
\$6	002476	1213	1218#
\$60	004312	1912	1923#
\$61	004344	1935	1946#
\$62	004404	1959	1961
\$63	004414	1964	1970#
\$64	004440	1984#	2001
\$65	004460	1987	1992#
\$66	004464	1991	1996#
\$67	004504	1999	2002#
\$7	002524	1229	1234#
\$70	004534	2005	2017#
\$71	004534	2025#	2051

1965#

K10

MAINDEC-11-DVDVC-A MACY11 27(1006) 08-AUG-77 09:20 PAGE 129
 DVDVCA.P11 08-AUG-77 09:16 CROSS REFERENCE TABLE -- USER SYMBOLS

SEQ 0127

\$72	004606	2027	2052#																	
\$73	004604	2031	2049#																	
\$74	004604	2034	2047#																	
\$75	004636	2057	2068#																	
\$76	004662	2079	2087#																	
\$77	004752	2107	2109	2124#																
.	= 014506	856#	860#	869	870#	872#	874#	875#	881	882#	884#	886#	904#	941						
		1014#	1022	1037	1038	1084	1085	1086#	1088#	1159	1160	1173	1174	1200						
		1201	1213	1214	1229	1230	1246	1247	1265	1266	1285	1286	1298	1299						
		1314	1315	1331	1332	1350	1351	1380	1381	1396	1397	1413	1414	1432						
		1433	1458	1459	1474	1475	1491	1492	1510	1511	1541	1542	1572	1573						
		1579	1580	1586	1587	1606	1607	1644	1645	1674	1675	1712	1713	1731						
		1732	1752	1753	1762	1763	1765	1766	1806	1807	1839	1840	1860	1861						
		1880	1881	1912	1913	1935	1936	1959	1960	1961	1962	1987	1988	2001						
		2002	2005	2006	2027	2028	2031	2032	2034	2035	2037	2058	2079	2080						
		2107	2108	2109	2110	2112	2113	2169	2170	2190	2191	2214	2215	2246						
		2247	2264	2265	2305	2306	2307	2308	2310	2311	2365	2366	2368	2369						
		2389	2390	2397	2398	2399	2400	2438	2439	2478#	2550	2551	2554	2555						
		2592	2593	2634	2635	2693	2694	2700	2701	2748	2749	2785	2786	2825						
		2826	2873	2874	2892	2893	2915	2916	2934	2935	2941	2942	2961	2962						
		2969	2970	3027	3028	3048	3049	3072	3073	3109	3110	3185	3196	3192						
		3193	3227	3228	3259	3260	3263	3264	3283	3284	3319	3320	3344	3345						
		3348	3349	3357	3358	3360	3361	3375	3376	3382	3383	3401#	3462	3463						
		3476	3477	3508	3509	3658	3659	3661	3662	3698	3699	3701	3702	3713						
		3714	3716	3717	3743#	3758#	3773#	3779#	3828	3832	3948	3872	3956	3959						
		4088#	4089	4095	4147#	4206	4269	4270	4336#											
		4099	4102																	
		881#	886																	

.SASTA= ***** U
 .SX = 001000

. ABS. 014506 000

ERRORS DETECTED: 0
 DEFAULT GLOBALS GENERATED: 0

DVDVCA, DVDVCA=SPMAC.SML DVDVCA.P11
 RUN-TIME: 89 84 4 SECONDS
 RUN-TIME RATIO: 1407/177=7.9
 CORE USED: 34K (68 PAGES)

L10